INL/RPT-23-74616
Revision 0

2023 Real-Time Optimization Workflow Status Update

September 2023

Jeren Browning
Linyu Lin
Takanori Kajihara
Paul W. Talbot

Integrated Energy Systems

INL is a U.S. Department of Energy National Laboratory operated by Battelle Energy Alliance, LLC

INL/RPT-23-74616
Revision 0

2023 Real-Time Optimization Workflow Status Update

Jeren Browning
Linyu Lin
Takanori Kajihara
Paul W. Talbot

September 2023

Idaho National Laboratory
Integrated Energy Systems
Idaho Falls, Idaho 83415

http://www.ies.inl.gov

Prepared for the
U.S. Department of Energy
Office of Nuclear Energy
Under DOE Idaho Operations Office
Contract DE-AC07-05ID14517

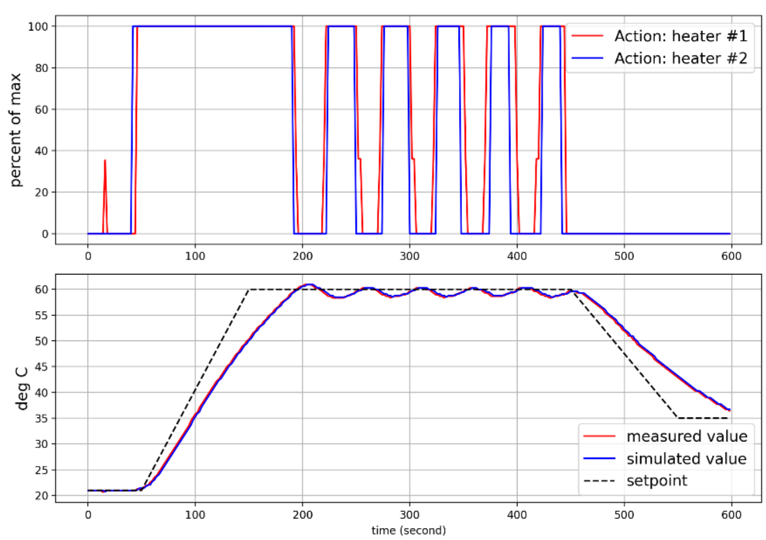
Page intentionally left blank

ABSTRACT

Nuclear integrated energy systems are composed of a diverse set of energy generation sources and exist in dynamic and competitive electricity markets. With the inclusion of thermal energy storage, nuclear power systems can store heat for future use through various processes, such as water desalination or hydrogen generation. This heat storage can be managed in such a way as to economically optimize its usage. By combining real-time price data from the day-ahead and real-time markets with predictive and intelligent models, the charging and discharging of the thermal energy storage may be determined and optimized.

This research details an approach through models and systems for the real-time optimization (RTO) of capacity allocation. Virtual models of the energy system and its physics phenomenon and component interactions provide intelligence to verification and prediction of operations. An optimization framework can use data generated from both a set of physical assets as well as the virtual models to predict future performance and create optimization and control workflows. Data warehouse technologies can be used to combine data across models, optimization workflows, market price data, and sensor data to intuitively store various types of data and provide integrations to physical control systems as well as user visualizations. Put together, these components can create a system for the RTO of nuclear integrated energy systems.

Various virtual models and bench-scale physical demonstrations have been successfully performed and verified using this system. Larger scale testbeds that include thermal energy storage systems have been identified as future opportunities. A gap analysis that details the steps necessary to reach a physical demonstration at this scale is provided, along with conclusions on the current effort and future work.



Demonstration of using Optimization of Real-time Capacity Allocation and DeepLynx for load following for RTO.

Page intentionally left blank

CONTENTS

FIGURES

TABLES

Page intentionally left blank

ACRONYMS

API	application programing interface
ASME	American Society of Mechanical Engineering
BOP	balance of plant
DIAMOND	Data Integration Aggregated Model and Ontology for Nuclear Deployment
DOE	Department of Energy
DT	digital twin
EMPC	economic model predictive control
FMU	functional mock-up unit
FORCE	Framework for Optimization of Resources and Economics
GUI	graphical user interface
HMI	human-machine interface
HERON	Holistic Energy Resource Optimization Network
IES	integrated energy systems
INL	Idaho National Laboratory
ISO	independent system operator
LP	linear programming
NPP	nuclear power plant
OASIS	Open Access Same-time Information System
ORCA	Optimization of Real-time Capacity Allocation
RAVEN	Risk Analysis Virtual ENvironment
RTO	real-time optimization
SOC	state of charge
SQA	software quality assurance
TCLab	Temperature Control Lab
TEDS	thermal energy delivery system
TES	thermal energy storage
V&V	verification and validation
VRE	variable renewable energy

Page intentionally left blank

2023 Real-Time Optimization Workflow Status Update

1. INTRODUCTION

1.1 Background

Nuclear integrated energy systems (IESs) are energy systems with higher-than-typical coupling to leverage the specific attributes of each component within the system, rather than have each focus on only producing or consuming electricity. These nuclear IESs have been investigated more in recent years with the aggressive projected adoption of nondispatchable variable renewable energy (VRE) sources such as wind and solar. Projected changes to electricity markets with high VRE penetration include conditions that drive out baseload electricity production, which is nuclear power's traditional role in energy markets. IESs offer a possibility for nuclear systems to operate with additional flexibility, providing electricity to the grid when VRE sources are not available but using reactor heat to supply alternative industrial processes, such as water desalination or hydrogen generation, when VRE production is high.

However intriguing the technical and economic promise of IESs for nuclear energy is, increasing the interoperability of these traditionally decoupled systems introduces challenges in safe operation as well as control. For instance, while nuclear power plant (NPP) operators perform dynamic maneuvers to control plant power shaping, introducing the flexible option to divert some or all of the heat produced by the reactor to a downstream process adds a significant degree of freedom that is not yet well understood. While the field of system controls is well established across many industries, topics such as model predictive control, autonomous control, and digital twins (DTs) are new to the nuclear industry, which due to safety concerns has not embraced simulation-based control to the same degree as other industries.

In this document we discuss research developments in model-based control, more specifically real-time optimization for control applications, with focus on nuclear IES applications. The Nuclear IES program, funded under the U.S. Department of Energy (DOE), has allocated resources to investigate approaches to autonomous optimal control via DTs. This report summarizes the current state of research and development within this program and gaps that exist for demonstrating autonomous control at scale for an IES.

In the rest of Section 1, we discuss the research to date on physical demonstrations with supporting virtual models, the workflow that has been established for solving these real-time optimization (RTO) problems, and how data is integrated across the various applications. Section 2 provides an overview of the key components in the RTO system, namely the Optimization of Real-time Capacity Allocation (ORCA), DeepLynx, and virtual models and controls that simulate the physical assets and processes. In Section 3, the virtual model results are presented along with a detailed gap analysis of what effort remains to achieve a full physical demonstration with this system. Conclusions are provided in Section 4 with an overview of future work in Section 5.

1.2 Preparing for Physical Demonstrations

Physical demonstrations are a critical step in the maturation of modeling and simulation software. While a virtual demonstration that consists of data and some set of models can show how the models behave given certain conditions, a physical demonstration is significantly more complex and demanding in nature. For RTO problems, various models can be created to describe the operation of a system, but applying these models to a physical system requires various integrations and validations that further the research as well as validate the models. Several systems have been selected for virtual model development and demonstration, namely a Dymola battery, a temperature microcontroller, and the thermal energy distribution system (TEDS). Working through the development and preliminary validation of these models is a necessary step towards physical demonstrations.

With a virtual model of a physical system established, various other preparation steps must be completed prior to physical demonstration. Integrating data from the physical system to the virtual model is required for generating predictions of performance by the virtual model and may require enhancements to the physical system, such as network upgrades or industrial control system and human-machine interface (HMI) development. If data will be returned to the physical system from the models or DT attached, a process must also be put in place and validated for this transfer and acceptance of DT output. For RTO, some form of near real-time price data is a necessary input to properly simulate the optimization of energy producing and storage assets in relation to the energy market. This data source must therefore be identified and connected to the DT. Finally, the control system that determines what actions to take on the physical system based off of the set of models and their results needs to be correctly integrated and established to handle the variety of operating conditions that will be tested in the physical demonstration.

In preparation for a more full-scale demonstration, like with TEDS, a bench-scale demonstration was identified and performed to validate initial concepts as part of this research. This smaller demonstration was performed with a temperature microcontroller that consisted of two heaters and two temperature sensors. The model was able to predict temperatures as the heater increased or decreased power and therefore send commands to the heaters to adjust the temperature setpoint as determined by the model.

With the microcontroller demonstration complete, this research is well positioned to solve the issues of physical system and controller integration, solve the integration of real-time price data, and perform a future physical demonstration with TEDS.

1.3 Real-Time Optimization Workflow

In deregulated energy markets, the independent system operator (ISO) establishes markets for energy and ancillary services. In the day-ahead market, power generators bid capacity and price for each hour of the next 24 hour period, then submit these bids to the ISO once per day. The ISO matches up the projected electricity demand with the capacity bids, determines the clearing price, and selects which generators will participate in the power generation for each 1 hour period. The real-time market is a spot market in which the demand not covered by the day-ahead market is met. Generators submit bids up to about 75 minutes before the start of the trading hour, and, as with the day-ahead market, the ISO matches the demand with the capacity bids, determines the clearing price, and selects which generators will participate in the real-time period, which varies per ISO from 5 to 15 minutes. The ancillary service markets include markets for frequency regulation and reserves.

Energy storage opens up the possibility for energy arbitrage, which entails purchasing or storing (charging) energy when prices are low, then selling (discharging) energy when prices are high. The typical approach to energy arbitrage and dispatch optimization is to formulate revenue maximization as a linear programming or mixed-integer programming problem. This approach has been demonstrated for the day-ahead market [1] [2], day-ahead and real-time markets [3] [4], and day-ahead and frequency regulation markets [5] [6]. The conclusion was that the real-time market offered greater arbitrage opportunities because of the increased price volatility, and energy storage enables an offset in time between power generation and power consumption.

In addition to energy arbitrage and frequency regulation services, the value of energy storage can be achieved through RTO, which solves for the maximum amount of revenue generated in the energy arbitrage and frequency regulation markets by a given energy storage system [7]. To facilitate RTO, autonomous controllers are needed to adapt to changes in electricity prices and equipment status with minimal human intervention. In the current literature, demand-side predictive controllers are a promising alternative to traditional controls for both system efficiency and operational cost. Predictive controllers require a system model to be able to predict system performance and an optimizer to minimize some performance metric, commonly a measure of cost.

Figure 1 shows the control system architecture with control systems at different time scales ranging from weeks to seconds. Specifically, RTO takes long-term decisions from asset management and plantwide scheduling, including investment strategy, operation model, and infrastructure. The hour-by-hour decision from RTO is followed by a faster control and automation layer that accounts for fast corrective actions. Such a control layer can be divided into supervisory and regulatory layer: the supervisory layer aims to coordinate multiple systems, structures, and components and track the reference trajectory and the regulatory layer aims to stabilize and avoid drift in the variables.

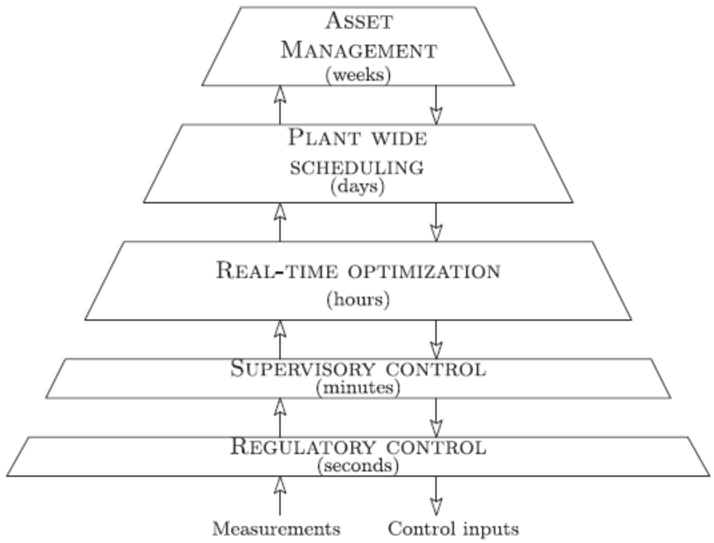


Figure 1. Hierarchical control structures for process control, where RTO provides hours information to supervisory and regulatory control. Adopted from [8].

Traditional RTO [9] focuses on a nonlinear steady-state process model to compute the optimal setpoint at steady-state operation [10] since the economic operation of the plant often occurs there, and the computed optimal decision variables simply need to be kept at constant values over a long time period. However, such static RTO faces challenges [8]:

- Cost of developing and updating model structures is high
- Model uncertainty is not accounted for
- Frequent grade changes make steady-state optimization less relevant
- Dynamic constraint violations lead to infeasibility.

It is further suggested in References [9, 11] that the steady-state wait time is a fundamental limitation of traditional RTO, regardless of the degree of rigor used. When static models are used, the model must be adapted using measurements corresponding to steady-state operation. If the process is frequently subject to disturbances or if the settling times are rather long or the process is different across different assets, this can lead to the plant being operated in transients for significant periods of time. With the inadequacy of steady-state measurements, the model is not updated frequently. As a result, with the traditional steady-state approach, the plant is operated suboptimally for long periods of time.

This report focuses on economic model predictive control (EMPC) with dynamic models. The main hypothesis to test is if the predictions of steady state from a sufficiently accurate dynamic model (and updated based on the actual process values) would settle before the actual process settles at a new steady state. The subset of measurements selected for the model update and parameterization step would be replaced with their steady-state predictions. The model update and parameterization would then adjust these predictions (and model parameters) to arrive at a consistent steady state. Meanwhile, this work uses data-driven models for directly combining experimental process data with known or approximated physics-based models.

1.4 Integration of Data

Developing a DT requires highly sophisticated software or tools for their data management. Data warehouse software is one of the essential tools to manage the data transformation and storage part of DTs. A data warehouse is a centralized repository designed to enable and support business intelligence activities, particularly analytical processing, reporting, and data mining. It offers the function of storage, aggregation, and management for the large scale of data.

In the context of DTs, the important functions of data warehouses are data aggregation, scalable and integral storage, data quality and consistency, real-time processing, and advanced analytics. DTs will generate large amounts of data from various sources, including raw sensor data, control data, use inputs, and metadata. A data warehouse can consolidate these data for visualization and analysis. As components and assets in the DT increases, the volume of data generated through the DT exponentially increases. Also, multiple different types of data in the format of structured and unstructured data can be given in the DT. The data warehouse is required to handle those large datasets, including diverse data types, and integrate them into a well-organized structure for querying historical data easily and should ascertain that the data communicated through them is clean, reliable, and consistent. In addition to batch processing, real-time data processing is imperative especially for DT applications. A data warehouse with embedded advanced analytical tools can derive additional benefits, including statistical analysis, prediction, future forecast, and pattern recognition. These functions are not mandatory to the DT application of the data warehouse, but they are significantly useful for the data warehouse acting as the DT's role for the communication, memory, and analytics supporting business intelligence.

2. METHODS

2.1 Optimization of Real-Time Capacity Allocation

ORCA is a software platform for creating and running real-time control and optimization workflows for IESs [12] [13]. The IES program aims to increase flexibility in nuclear energy generation to allow nuclear plants to thrive economically while supporting variable grid demand [14]. A key mechanism for implementing such flexibility would be to use excess heat and electricity (available at times of low net electricity demand) for hydrogen production, water desalination, thermal storage, or other industrial processes. To enable responsive generation for various industrial processes, an IES needs a control system to support dynamically incorporating multiapplication energy systems and apportion thermal or electrical energy. The demonstration of such a control system is critical for tightly coupled and thermally coupled IESs, where optimizations and controls are needed for multiple components and connection points. ORCA is developed to maximize the revenue of NPPs by efficiently operating multiapplication energy systems. The ORCA workflow is based on a receding-horizon- or EMPC-based optimization [15]. EMPC uses a dynamic model to optimize an objective function over a finite time horizon. The essence of EMPC is to optimize, over the manipulatable inputs, forecasts of process behaviors subject to equality and inequality constraints. The forecasting is performed using a process model (i.e., a predictor) over a finite time interval of length P. A generic process model f for a dynamic system is:

$$x_{k+1|t} = f\left(x_{k|t}, u_{k|t}\right), \tag{1}$$

where x stands for the state variables of a dynamic system and u represents control actions for a dynamic system. For an energy storage system, x can refer to the state of charge, internal temperatures, and flow rates, u can refer to the valves position and flow rates of charging, discharging, and balance of plant (BOP), t represents the current time, at which the EMPC is activated to find the optimal control action

$u^* = [u^*_{1|t}, \dots, u^*_{N|t}]$, and N is the total number of prediction steps ahead, while $k = 1, \dots, N$ is the index of time steps ahead of the current time t that system states are predicted. For convenience, we use subscript $k|t$ to denote k steps ahead of the real-time t, and the initial states of x when $k = 0$ ($x_{t|t}$ or x_t) are obtained from real-time measurements from the energy storage system.

Based on the system-state predictions over the prediction horizon, the optimal control action u^* can be found over a finite control sequence $u = \left[u_{1|j}, \dots, u_{N|j}\right]$ by minimizing or maximizing the summation J of stage cost function L over the entire prediction horizon, where J^* stands for the minimized or maximized summation of the stage cost function. Similar control schemes, usually known as predictive or anticipatory controls, can also be found in autonomous control for nuclear applications, including normal power transients [16] and accident scenarios [17, 18].

$$J^* = \min_U \left[\sum_{k=1}^{N} L\left(x_{k|t}, u_{k|t}\right)\right] \tag{2}$$

subject to

$$x_{k+1|t} = f\left(x_{k|t}, u_{k|t}\right),$$

$$U = \left[u_{1|t}, \dots, u_{N|t}\right] \in \mathrm{U}_i \text{ for all } i = 1, \dots, n_{c_u},$$

$$X = \left[x_{1|t}, \dots, x_{N|t}\right] \in \mathrm{X}_i \text{ for all } i = 1, \dots, n_{c_x},$$

$$x_{0|t} = x_t.$$

ORCA is solved in a receding-horizon fashion. At a given time t, EMPC receives a state measurement used for initializing the dynamic model. An optimal piecewise input trajectory for the objective function and constraints is computed over the time horizon, corresponding to times $t \in [t, t + \Delta t, t + 2\Delta t, \dots, t + N\Delta t]$, where N is the total number of time steps. The first of the optimized input trajectories (charging and discharging) is to be implemented over the next time period Δt. At the next time instance, $t + \Delta t$, the EMPC is solved again to determine the optimal input trajectories. Figure 2 shows the ORCA workflow with the virtual and physical system connected through the DeepLynx data warehouse. The optimization in ORCA is solved with the Pyomo [19] and IPOPT [20] solvers.

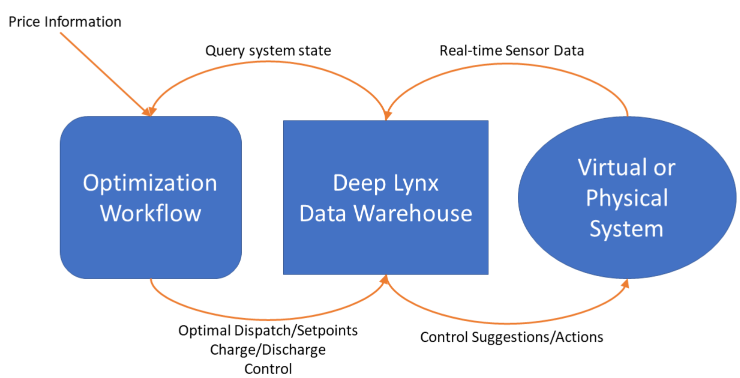


Figure 2. ORCA workflow for RTO of IES.

2.2 DeepLynx Integration

DeepLynx [21] is a unique data warehouse open-source software where users can create or edit a custom ontology for tailoring their project and is being developed at Idaho National Laboratory (INL), with version 1.0.0 being released on October 10, 2022. Since DeepLynx enables large projects to embrace digital engineering, it is an essential part of integrating DTs into these projects via integrating with various software systems or physical assets. The data communicated through DeepLynx are stored under the ontology in a graph database. The ontology represents definition of the types, properties, and interrelationships of the entities in a specific domain. The data are stored in nodes, and relationships between nodes are represented via edges in the graph. Here, a node is a point representing a fundamental unit of the graph and an edge is a line connecting two nodes. One of the unique features of DeepLynx is that time-series data are stored as table data and can be connected with nodes in the graph, which can concisely treat numerous time-series data such as sensor data. This concept realizes a well-organized graphical view to grasp the whole data structures and relationships in the system. Users can query stored data by using the nodes and relationship information, which fosters a strong performance in querying complicated relationships, flexibility in expanding a given database, and agility in changing system requirements.

DeepLynx governs its data format via custom ontology. One of the ontologies, created for the nuclear engineering field, is the Data Integration Aggregated Model and Ontology for Nuclear Deployment (DIAMOND) [22]. DIAMOND has been developed specifically for NPP and related nuclear or energy application domains. The basic components of an ontology are classes, relationship types, and relationships combining a pair of classes and a relationship type. These parameters can have multiple properties such as id, name, description, etc. as well as associated metadata. The classes and relationships are referenced when creating nodes and edges, respectively. Based on the prepared ontology, users can easily add their necessary class, relationship type, or properties in the graphical user interface (GUI). In this project, the ontology is slightly customized for the project's unique purpose based on DIAMOND.

DeepLynx is typically used with additional software applications, known as adapters, for connecting with other software or hardware. Publicly available examples of adapters are the Multiphysics Object-Oriented Simulation Environment Adapter [23], Machine Learning Adapter [24], Data Historian Adapter [25], Unidirectional Network Connector Adapter [26], MATLAB Adapter [27], and Supervisory Control Adapter [28]. This project developed an ORCA adapter for streaming time-series sensor and control data between ORCA and other entities through DeepLynx. In this ORCA adapter, users can set how to create graphs, what time-series data are stored in each node, which data are queried, and how often data are transferred through DeepLynx. This allows for even large systems to be systematically designed as a graph in DeepLynx without repetitive input creation. Fine tuning and modification of the graph can be done later in the DeepLynx GUI.

The DeepLynx ORCA adapter is being developed using the DeepLynx Python software development kit [29]. The basic flow of how to set up DeepLynx is shown in Table 1. In the latest version (1.4.0), the DeepLynx GUI can set and manipulate all operations. Hence, if the amount of data is not gigantic or repetitive communication is unnecessary, the DeepLynx GUI is the first candidate to process data. In addition to the complete function, the GUI is graphically intuitive and easy to command. The ORCA adapter manipulates DeepLynx by directly sending commands to the DeepLynx application programing interface (API) layer. ORCA adapter's ability to create or edit the ontology is limited, but other operations can be set and automated even if the volume of data is huge. For editing a large number of items in the ontology, Protégé, a free, open-source ontology editor, can be a good alternative software. The schematic data flow of the ORCA adapter is shown in Figure 3. Type mapping for nodes and edges and how to create time-series tables on the node are designed using a graph plan file in a comma-separated values format (Flow 1–4). Once the graph structure is designed, real-time data from physical assets are automatically uploaded to nodes and then queried to transfer data to ORCA (Flow 5–6). This is the same process when data is transferred from ORCA to the physical assets through DeepLynx.

Table 1. DeepLynx set up flow.

Action	Available Tool		
	DeepLynx GUI	ORCA Adapter (API layer)	Protégé
Create an ontology	Yes	No	Yes
Create a container	Yes	Yes	No
Load an ontology	Yes	Yes	No
Modify the ontology	Yes	No	Yes
Create a data source	Yes	Yes	No
Upload a file to the data source	Yes	Yes	No
Create a type mapping	Yes	Yes	No
Create a time-series data source	Yes	Yes	No
Upload a file to the time-series data source	Yes	Yes	No
Query data	Yes	Yes	No
Visualize data	Yes	No	No

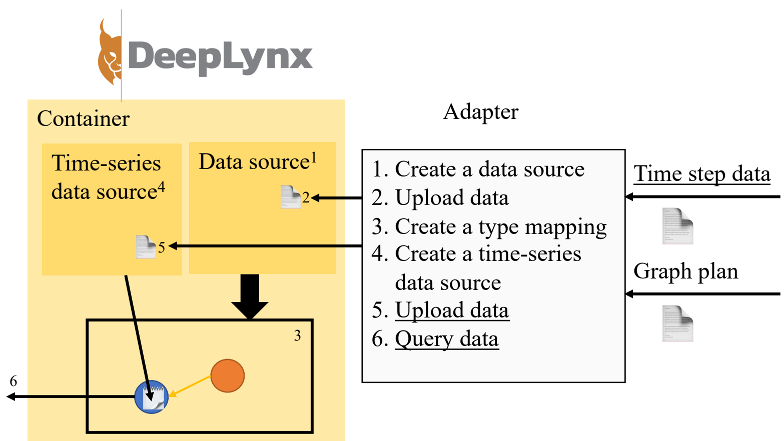


Figure 3. Data processing and storage workflow through DeepLynx.

2.3 Dynamics and Control

Focusing on a virtual RTO demonstration, this work uses the Modelica model as the virtual representation of the target energy storage system. Specifically, this work investigates two storage systems: the lithium-ion battery model [30] and TEDS. Figure 4 shows Modelica model for the virtual battery. Figure 5 shows the changes in the normalized battery state of charge (SOC), as detected via charging and discharging signals produced at random over 600,000 seconds.

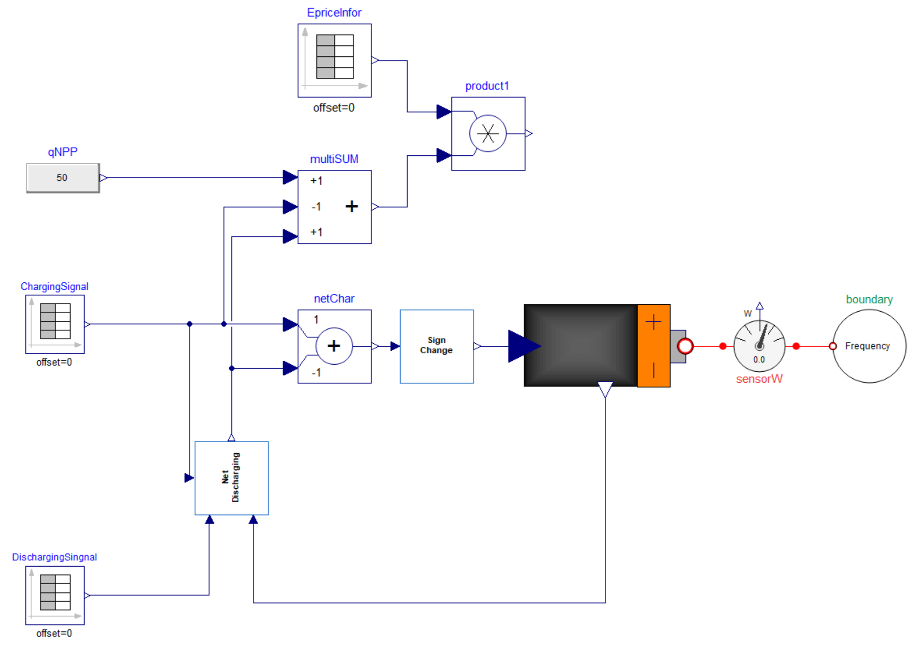


Figure 4. The Modelica model calculates the lithium-ion battery SOC over time, based on charging and discharging signals. (Modification of the original model taken from Reference [31].)

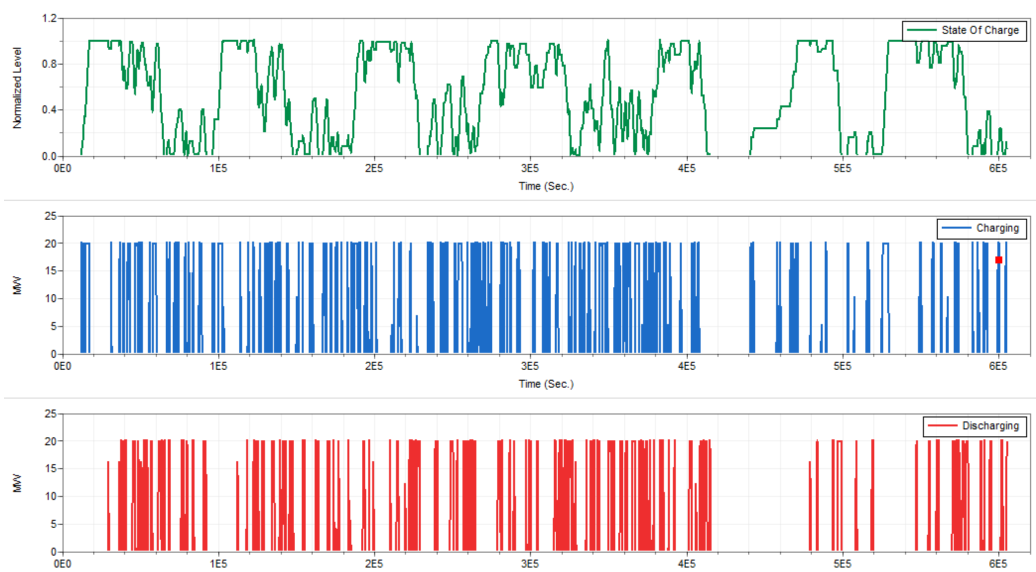


Figure 5. Lithium-ion battery SOC changes corresponding to charging and discharging signals.

Figure 6 shows the Modelica model for TEDS. In this initial deployment, the energy system is assumed to consist of a heat generator, a thermal storage unit, and a heat sink [32].

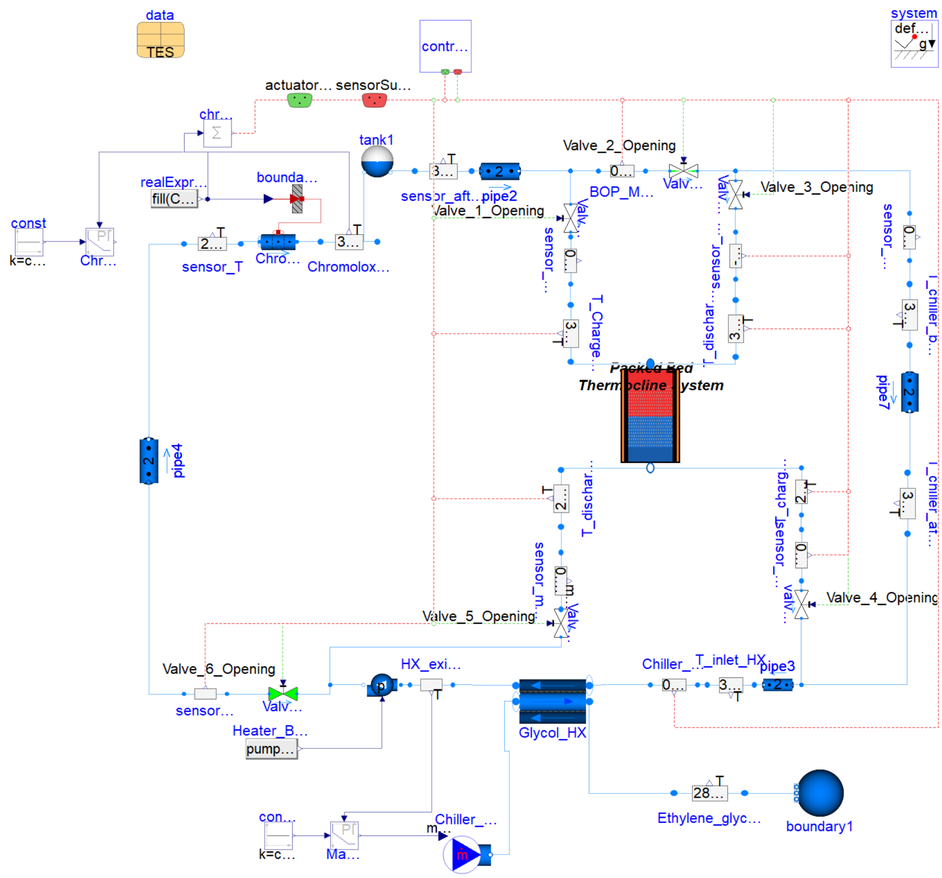


Figure 6. Modelica TEDS model

To directly respond to market demands, especially in deregulated markets, the IESs that include a thermal storage unit are expected to maximize system profits. In this scenario, it is advantageous to operate the thermal generator, particularly if it is nuclear, at full capacity and bypass excess steam into storage during times of low electricity pricing or grid demand and to discharge the thermal energy storage (TES) unit during times of high prices or demand. Meanwhile, for refueling, maintenance, or excessively low demand or pricing, the thermal generator will need to power down separately from the TES. Therefore, two separate signals for overall demand $Total_{Demand}(t)$ and thermal generator power $Heater_{Demand}(t)$ are available in the current TEDS model [33]. The thermal storage demand can then be determined from:

$$load_{TES}(t) = Total_{Demand}(t) - Heater_{Demand}(t), \tag{3}$$

If $load_{TES} < 0$, excess capacity will be sent to the thermal storage unit; if $load_{TES} > 0$, the thermal storage unit will begin to be discharged. Based on the overall demand and thermal generator power, a set of proportional integral controllers are used to control the charging flow valve, discharging flow valve, and BOP flow valve. Figure 7 shows the transients of state variables, including TEDS power rates, BOP power rates, charging, discharging, and bypass flow rates, also defined as BOP mass flow rates in Modelica model, in response to the actions of total demand and thermal generator power.

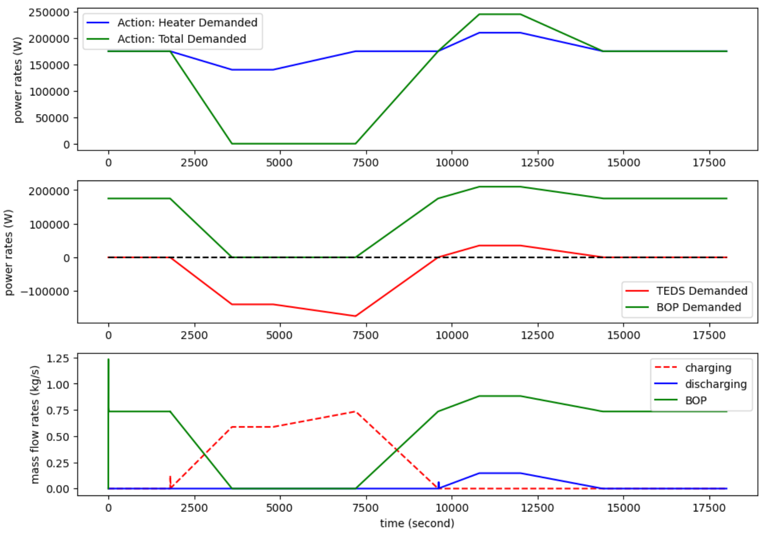


Figure 7. Transient of power rates, charging, discharging, and bypass flow rates in response to control actions of demanded heat from the heater and entire system.

In addition to the energy storage model, this work investigates and tests the real-time control capabilities of an ORCA workflow on a small temperature control lab (TCLab) device [34]. The TCLab is an application of feedback control with an Arduino, a light-emitting diode, two heaters, and two temperature sensors, as shown in Figure 8. The heater power output is adjusted to maintain a desired temperature setpoint. Thermal energy from the heater is transferred by conduction, convection, and radiation to the temperature sensor. Heat is also transferred away from the device to the surroundings. Figure 9 shows the transients of two temperature sensors in response to heater powers.

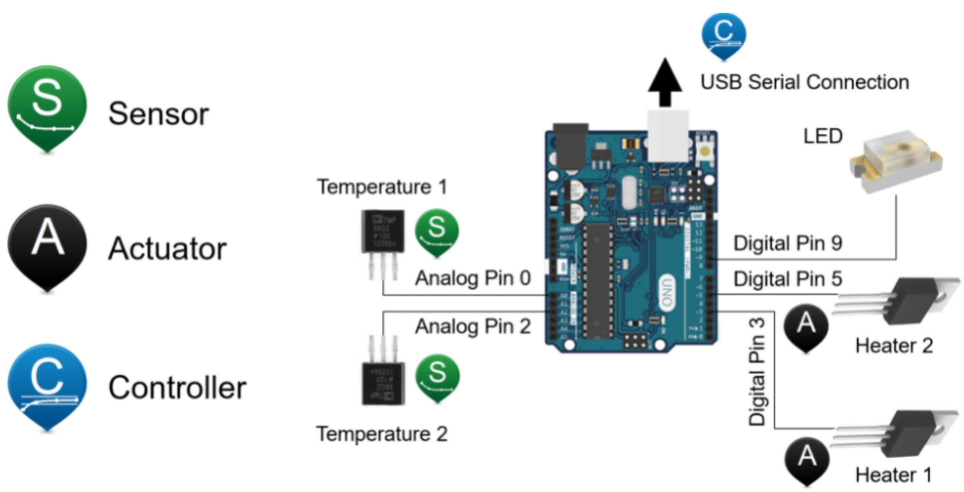


Figure 8. APMonitor TCLab kit schematic, extracted from Reference [35].

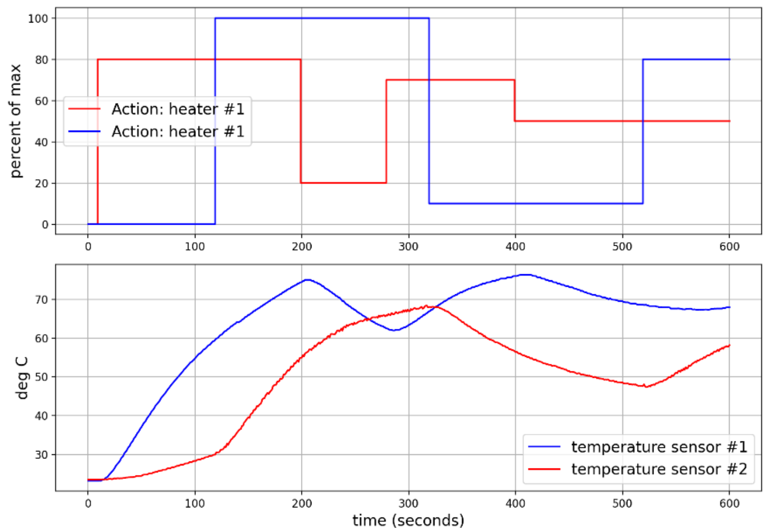


Figure 9. Transient of temperatures at two locations in response to control actions of the heater power relative to the maximum values.

Table 2 compares the setups of dispatch optimization for energy storage in IES applications against the temperature control problem in TCLab. The primary goal of TCLab is to test if the ORCA workflow can satisfy the real-time requirements given nonlinear dynamic optimization problems with fast real-time requirements.

Table 2. Comparisons between dispatch optimization for energy storage in IES applications and temperature controls for the TCLab.

	Dispatch Optimization	TCLab
Objective function	Rewards function based on the market price $P(t)$ and predicted states $X(t)$: $obj = P(t) \times X(t)$	Objective function based on the target setpoints $T_s(t)$ and predicted states $T_P(t)$: $obj = \|T_s(t) - T_P(t)\|_2$
Surrogate model	Largely linear dynamic model	Nonlinear dynamic model
Constraints	Equality and inequality constraints on state variables and control actions	
Real-time requirements	5 minutes	2 seconds
Optimization algorithms	Linear optimization	Quadratic optimization

3. RESULTS

3.1 Virtual Battery Model

For an energy storage RTO, the decision variable for the optimization problem is the input trajectory (charging and discharging) over that time horizon. The system (SOC) model serves as an additional constraint to the optimization. The IES model used for this RTO workflow reflects an NPP operating at constant capacity and an electrical storage device that can charge from the NPP and then discharge to the real-time market. The NPP directly sends electricity to either the real-time market or the electrical storage device. Meanwhile, the electrical storage device is characterized by two parameters: power rating (MW), specifying the amount of power the electrical storage device can charge or discharge, and SOC (MWh), specifying the amount of energy being stored by the electrical storage device. This work couples a battery model in Modelica [8] with an NPP that produces a constant electricity output q_{NPP}. As a result, the battery SOC S_t can be described as a function of charging q^C and discharging q^D power rates at time step t:

$$S_t = A\begin{bmatrix} q_{NPP} \\ S_{t-1} \end{bmatrix} + B\begin{bmatrix} q_t^C \\ q_t^D \end{bmatrix}, \tag{4}$$

where A and B are coefficient matrices that depend on the battery properties, including maximum power rating, maximum and minimum SOC, and conversion efficiency. Given a fluctuating electricity price P_t, the revenues can be maximized by finding the optimal charging and discharging power rates:

$$\max \sum_{t=1}^{T} P_t(q_{NPP} - q_t^C + q_t^D), \tag{5}$$

where P_t is the locational marginalized pricing in \$/MW at time step t and q_{NPP} is the constant power generation of the NPP (MW). The objective function is to search for optimal combinations of charging and discharging power rates that satisfy the following constraints:

$$0 \le S_t \le S_{max}, \tag{6}$$

$$0 \le q_t^C \le q_{max}^C, \tag{7}$$

$$0 \le q_t^D \le q_{max}^D, \tag{8}$$

$$0 \le q_t^D \le \sqrt{\gamma_{RTE}} S_t + \gamma_{RTE} q_t^C, \tag{9}$$

where S_{max} is the energy capacity (MWh) of the battery; q_{max}^C and q_{max}^D are the maximum charging and discharging power ratings (MW), respectively, based on the manufacturing specifications; Δt is the user-assigned unit time horizon; and γ_{RTE} is the conversion loss when energy is stored during the charge phase and released during the discharge phase. The present work uses Risk Analysis Virtual ENvironment (RAVEN) to calibrate coefficient matrices A and B, based on transient data from the battery model. To demonstrate the integrated RTO workflow with ORCA, a functional mock-up unit (FMU) is created and simulated in FMPy. This demonstration treats the FMU as a virtual representation of the Modelica battery model. Table 3 lists the parameters used in the demonstration.

Table 3. Parameters used in the ORCA workflow for virtual battery model.

Parameter		Value
Δt	Unit time horizon (minutes)	5
t_{window}	Prediction horizon (minutes)	300
γ_{RTE}	Round-trip efficiency during charge and discharge	0.8
q_{NPP}	Constant power generation of the NPP (MW)	50
S_{max}	Maximum energy capacity (MWh) of the battery	20
S_0	Initial SOC of the electrical storage device (MWh)	0
q_{max}^C	Maximum charging power ratings (MW)	20
q_{max}^D	Maximum discharging power ratings (MW)	20

To implement the ORCA workflow as in Figure 2, the dispatch optimization portion of the RTO workflow was set up as an liner programming (LP) problem, solved by the open-source optimization modeling language package Pyomo [36]. A sinusoidal price forecast model was used as external inputs to ORCA, with a locational marginal pricing peak or bottom every 6 hours. Figure 10 shows the transient of battery SOC in response to optimal charging and discharging rates from ORCA and a synthetic market price by a sinusoidal function. Starting from zero SOC, the battery is charged until full (20 MWh) when the market prices are low around 12:00 on May 31. Later, when the prices become high at 18:00, the battery is fully discharged. Because of the surrogate model errors, ORCA fails to realize that the battery has been fully depleted and continues the discharging action until the next lowest point in market price. Overall, ORCA can produce reasonable control actions that maximize the revenue of IES with an energy storage device.

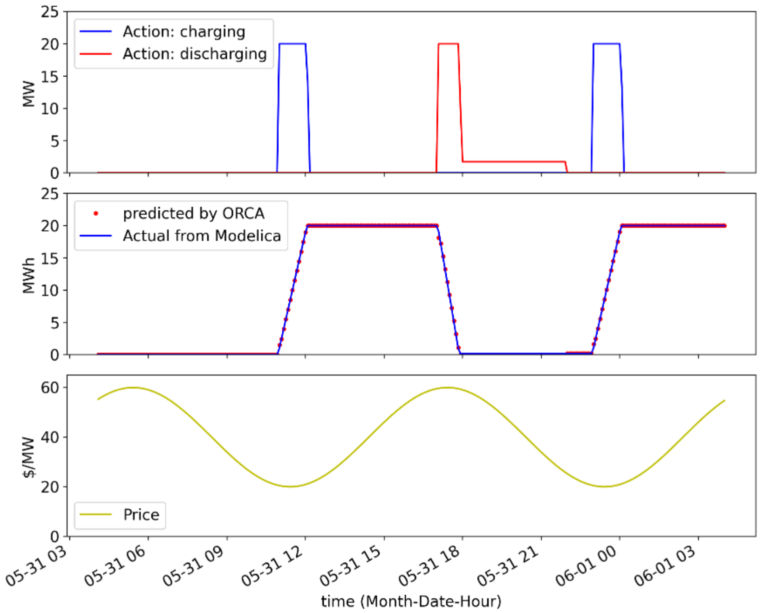


Figure 10. Transient of battery SOC in response to the market price with ORCA as an RTO. The optimal charging and discharging rates are collected from ORCA.

The data communicated between ORCA and FMU are passed through DeepLynx and stored in the node as shown in Figure 11. There are two nodes, battery and control, and these nodes are connected with an edge. The time-series data from FMU representing a virtual battery model is stored in the battery node and the data from ORCA is stored in the control node. On the left side of the figure, node information for the battery node is displayed, and it confirms that time-series data named Dymola-TS is attached to the node. Figure 12 shows how the DeepLynx GUI can deal with the time-series data transferred through DeepLynx. The DeepLynx GUI shows the record of time-series data within a focused node either in real time or from the stored database with multiple chart type options. Users can utilize the viewer for monitoring the DT or analyzing past data flows. In addition to the data trend, the viewer can show visual, statistical plots in real time. The right-side plots shown in Figure 12 are examples of a box plot and correlation matrix of each parameter stored in a node. These statistical plots will allow users to gain insights from data without tedious data processing for visualization. Examples of graph plan files (comma-separated values format) for creating a graph are shown in Table A1, Table A2, and Table A3 in Appendix A.

The demonstration of the virtual battery model successfully integrated the DeepLynx database system with the virtual battery asset and ORCA. Once users know how to create a graph in the DeepLynx and how to attach each time-series data to nodes, and then set some operational parameters using a DeepLynx ORCA adapter, the data generated from either the battery model or ORCA are stored in DeepLynx and necessary data to other applications were queried automatically during the selected period. The DeepLynx data warehouse with an ORCA adapter performed as a database, data pipeline, and visualization tool.

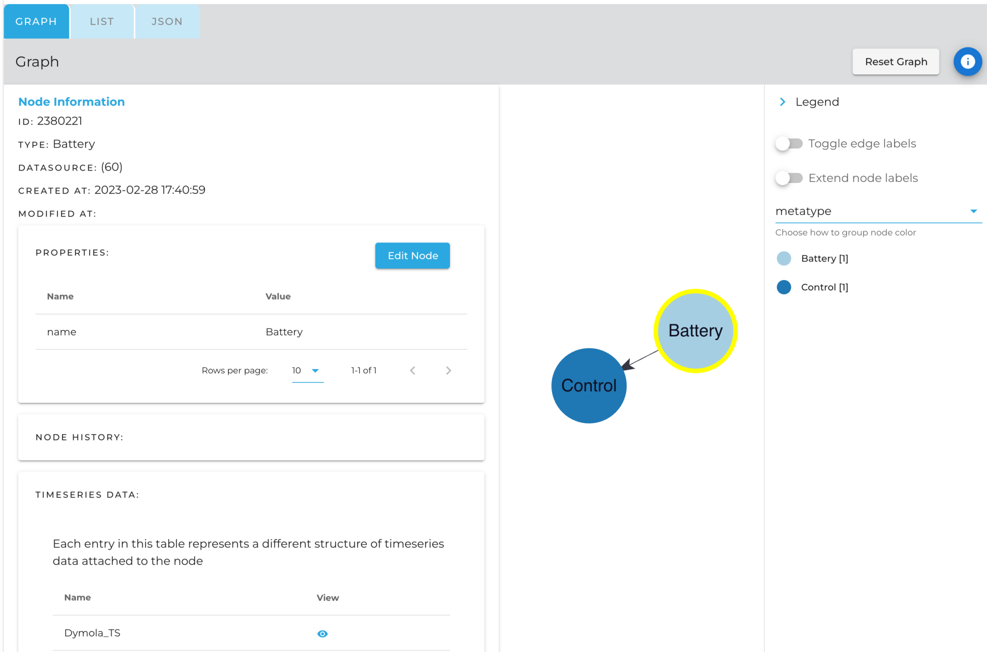


Figure 11. Data viewer in the DeepLynx GUI, showing battery and control nodes connected with an edge.

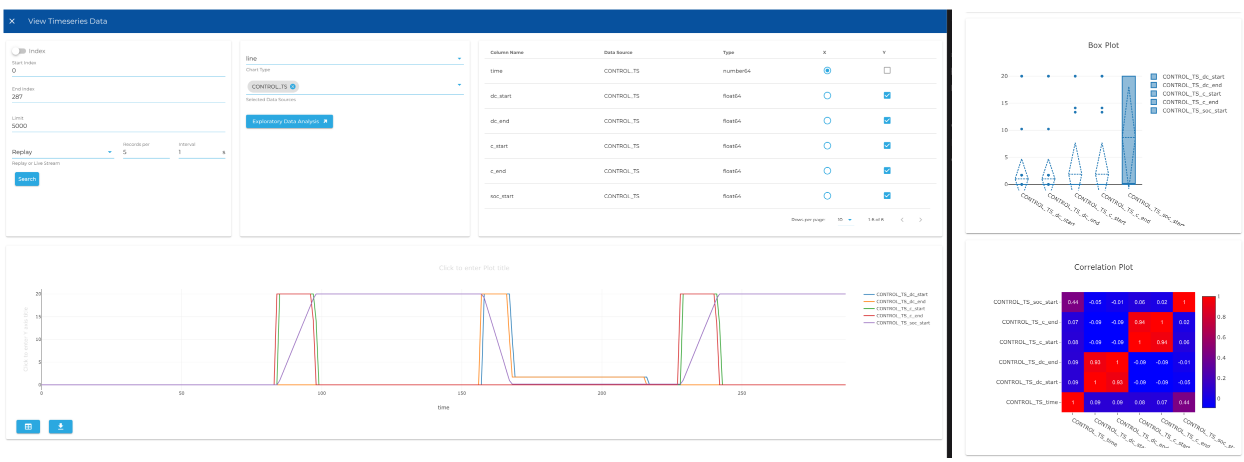


Figure 12. Time-series data viewer in the DeepLynx GUI, monitoring all time-series data from ORCA.

3.2 Temperature Controller

For the TCLab demonstration, a lumped parameter model is developed based on the physics models in Reference [37]. The dynamic input power to each heater (Q_1 and Q_2) and the temperature sensed (T_{C_1} and T_{C_2}) by each sensor is developed using energy balance equations, which account for convection, conduction, and thermal radiation:

$$\tau_H \frac{dT_{H_1}}{dt} = (T_\infty - T_{H_1}) + \beta(T_{H_2} - T_{H_1}) + \alpha_1 Q_1, \tag{10}$$

$$\tau_H \frac{dT_{H_2}}{dt} = (T_\infty - T_{H_2}) + \beta(T_{H_2} - T_{H_1}) + \alpha_2 Q_2, \tag{11}$$

$$\tau_C \frac{dT_{C_1}}{dt} = T_{H_1} - T_{C_1}, \tag{12}$$

$$\tau_c \frac{dT_{C_2}}{dt} = T_{H_2} - T_{C_2}, \tag{13}$$

where τ_H and τ_C are lumped parameters from the heater transfer law for heaters and sensors, respectively, with $\tau = mc_p/UA$; β is the lumped heat transfer coefficient between heaters; and α_1 and α_2 are factors for two heaters. These lumped parameters are calibrated using GEKKO [38] moving horizon estimation methods [39]. Figure 13 compares the simulated temperatures based on the lumped parameter model against the actual sensed temperatures given the same trajectories of control actions. The model results agree reasonably well with the actual responses, and the discrepancy is mainly because of the simplifications in surrogate models. Specifically, it is suggested in Reference [37] that a higher order radiative heat transfer model between heaters could result in better agreement.

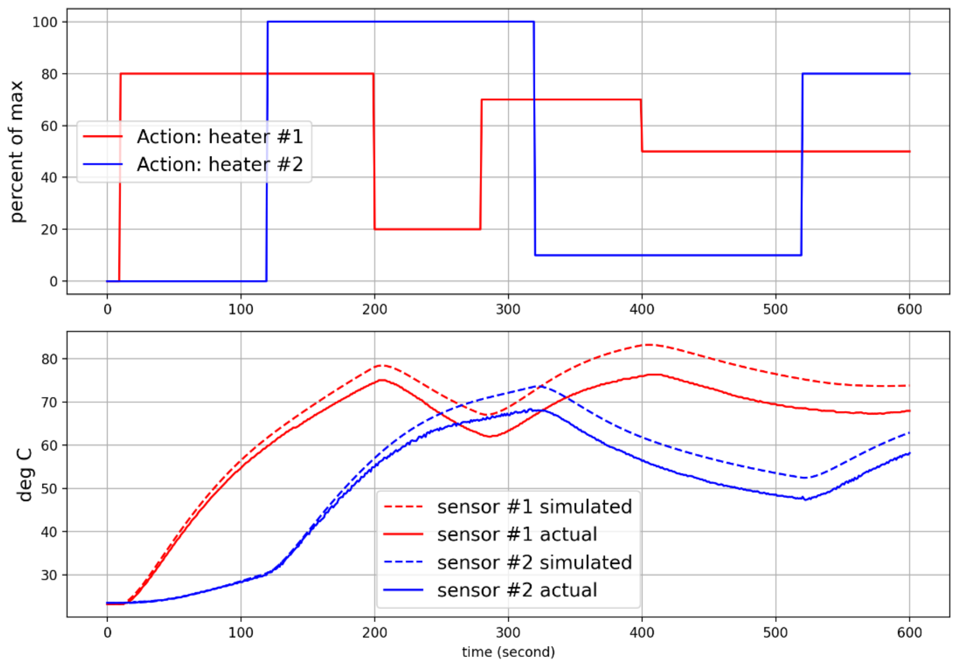


Figure 13. Comparison of actual versus simulated temperatures based on the lumped parameter models.

Table 4 summarizes the model parameters used in this TCLab demonstration. To implement the ORCA workflow as in Figure 2, Equation (10)–(13) are incorporated using DAE extension [40] in Pyomo. An objective function is added for minimizing the squared discrepancies between the sensed temperature and the setpoints by maneuvering the power rates of both heaters.

$$\min \sum_{t=1}^{T} \left\| T_{C_1}(Q_1, Q_2) - T_s \right\|_2. \tag{14}$$

Table 4. Parameters used in the ORCA workflow for TCLab.

Parameter		Value
Δt	Unit time horizon (seconds)	2
t_{window}	Prediction horizon (seconds)	10
τ_H	Lumped heat transfer coefficient for heaters	192
τ_c	Lumped heat transfer coefficient sensors	15
α_1	Factor for Heater 1	0.607
α_2	Factor for Heater 2	0.293
β	Lumped heat transfer coefficient between heaters	0.24

T_∞	Ambient temperatures	23

Figure 14 shows the transient of sensed and simulated temperatures in following the user-defined setpoints. Each step takes ~1.5 seconds to solve for the optimal control actions, which satisfies the real-time requirements of less than 2 seconds (Table 4).

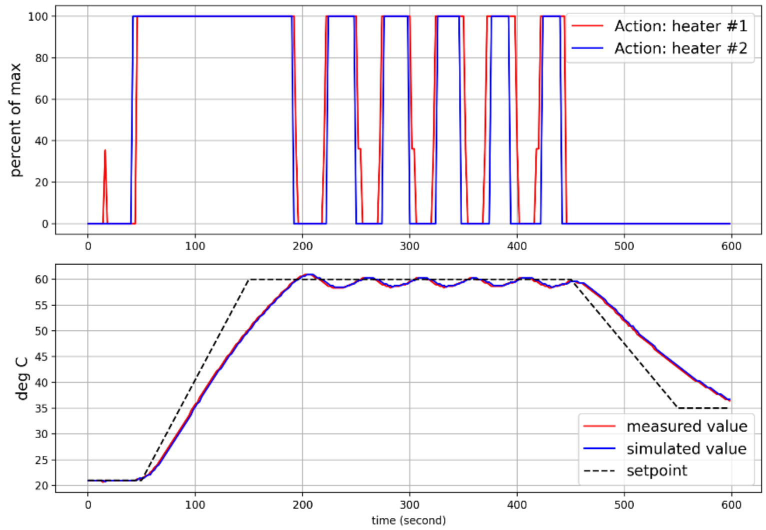


Figure 14. Transient of sensed and simulated temperatures in a setpoint following scenarios. Two heaters are controlled to adapt and maintain the sensor temperatures to user-defined setpoints.

3.3 Virtual TEDS

Based on the TEDS system analysis, two potential use cases are proposed for RTO with ORCA. The first use case focuses on the RTO layer only to ensure total demand and heater power meet the external energy demands while satisfying system specifications and constraints. The second use case includes both RTO and supervisory control to achieve autonomous operations for TEDS.

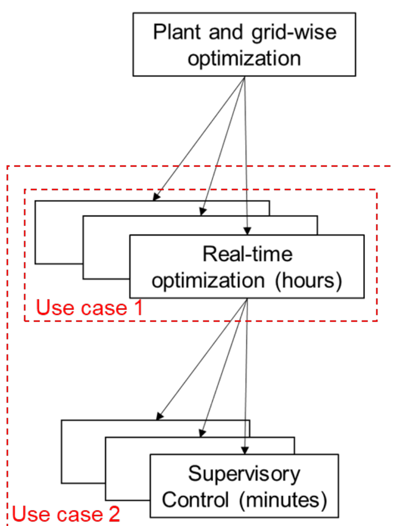


Figure 15. Proposed ORCA use cases based on the objective and control architecture.

The first use case assumes that supervisory and component control systems are available, and TEDS only decides the heater power $Q_{heater}(t)$ and TES power $Q_{TES}(t)$. As a result, the TEDS system dynamics can be expressed as:

$$Q_{TES} = Q_{total} - Q_{heater}, \tag{15}$$

$$\frac{d}{dt}T_{TES} = A \cdot T_{TES} + B \cdot Q_{TES}, \tag{16}$$

$$\dot{m}_{charge} = -\frac{Q_{TES}}{C_1} \qquad \text{if } Q_{TES} < 0, \tag{17}$$

$$\dot{m}_{discharge} = \frac{Q_{TES}}{C_2} \qquad \text{if } Q_{TES} \geq 0, \tag{18}$$

$$Q_{heater,BOP} = \min(Q_{total}, Q_{heater}), \tag{19}$$

$$\dot{m}_{BOP} = \frac{Q_{heater,BOP}}{C_3}, \tag{20}$$

where Q_{total} is the total demanded power from the TEDS and $\dot{m}_{discharge}$ is the mass flow rates through the discharging pipes. When $Q_{TES} < 0$, the TES is being charged, and $\dot{m}_{charge}$ is the mass flow rates through the charging pipes; when $Q_{TES} \geq 0$, the TES is discharging energy. In addition to TES, the heater could also contribute to the total demand through a bypass pipe, and such power rates are defined as heater BOP $Q_{heater,BOP}$. $\dot{m}_{BOP}$ measures the mass flow rates through the bypass pipe. Since the current TES has no indicator for SOC, this work suggests distributions of TES temperatures, denoted as T_{TES}, as a surrogate indicator. Constraints, including the minimum and maximum flow rates and internal temperatures, can be decided based on the TEDS specifications.

4. Gap Analysis

The gap analysis attempts to identify the remaining research tasks and requirements that must be met prior to a physical demonstration with TEDS. This physical demonstration should use models developed for RTO, incorporate price data that is regularly updated throughout the demonstration, and control a physical energy asset or system that can replicate the core functionalities of energy production and storage. TEDS has been selected due to its nature as a test bed at INL that can properly accommodate this demonstration.

4.1 Verification, Validation, and Uncertainty Quantification

In scientific computing, the fundamental aspect of verification and validation (V&V) activities is to identify sources of uncertainty and estimate the total uncertainty of modeling and simulation tools for the intended use. More specifically, verification is defined as the characterization of numerical approximation errors associated with a simulation, which includes discretization error, iterative convergence error, round-off error, and errors due to computer programming mistakes. Validation refers to the characterization of the model form uncertainty and can be generally defined as assessment of model accuracy by way of comparison of simulation results with experimental measurements [41]. For RTO applications in IES, the V&V gaps can be discussed from two perspectives: application and framework. The application focuses on specific V&V methods and gaps for evaluating the uncertainty of ORCA in TEDS. The framework focuses on formal methods for ensuring the safe and reliable use of autonomous control systems in complex systems like IESs.

From application perspectives, for a model-based RTO approach, V&V primarily refers to the quantification of the uncertainty of surrogate models and evaluation of its impacts on RTO control actions. Since the surrogate model is trained on simulation results, the first step is to determine the

Modelica model error by comparing the simulations against experimental data. Preliminary validation for the TEDS start-up process have been made in Reference [33]. Verification results, including built-in static analyzers in Dymola for consistency checking, regression tests, and spatial refinement tests, are also discussed. However, since the RTO workflow is primarily designed for system transients in response to energy demands, existing V&V results are not directly applicable, and experimental data of normal power transients are needed. Table A4 in Appendix A provides a tentative list of scenarios for the TEDS model validation based on the importance and knowledge levels of the current model to ensure the relevancy of validation results to support the confident uses of surrogate models. Depending on the scenarios, the system transients can include the internal temperatures of TES at different locations, mass and volumetric flow rates through pipes and valves, fluid temperatures measured at different locations of the system, etc.

In addition to the V&V for the RTO surrogate model, as a predictive control system, the stability of ORCA should also be investigated. For a linear control system, the system needs to satisfy bounded input and output stability, where the output of the continuous or discrete time system is bounded given bounded inputs. For nonlinear systems, numerical methods, including disturbance rejection, are needed. The Lyapunov direct method is another approach to show that the values of a cost function for an optimization problem will be continuously decreasing without the knowledge of the solution of the differential equations. The main challenge is to find the Lyapunov function for the system.

From the framework perspective, the unique challenges of RTO enabled by dynamic models and optimization are their dependence on sophisticated software control and decision, and their increasing deployment in safety-critical scenarios requires a stronger form of V&V [42]. It has also been suggested in literature that as the target applications (e.g., autonomous control) and modeling approaches (e.g., machine learning models) become more complex, there is no single V&V method that is adequate for the complexity presented by these systems [43]. As a result, to enable the safe and confident use of autonomous-enabled RTO, such as ORCA, in a complex system like an IES, formal methods are needed. Specifically, these methods are mathematically based techniques for the specification and verification of software systems to ensure the correctness of, and provide sufficient evidence for the certification of, autonomous systems. One of the potential frameworks is an integrated formal method with an integration of multiple formal and semiformal methods to capture several dimensions of a system at once for building an assurance case, which is a structured argument, supported by evidence, intended to justify that a system is acceptably safe and secure. Compared to other approaches, like theorem proving and model checking, integrated formal methods are more suitable for hybrid systems like IESs with various configurations, components, and systems [44].

4.2 Running a Physical Demonstration

In order to perform a successful physical demonstration with TEDS, several forms of integration and validation must be complete. An important integration is the physical network layer, ensuring that the RTO DT can properly receive and pass data from the HMI system connected to TEDS. Currently, the HMI is using LabVIEW to control actuators and receive data from sensors. This computer running LabVIEW is on its own isolated network. For the DT to use certain compute resources like high-performance compute or cloud environments, the TEDS local network should be connected to a wider network that can accommodate these integrations. Without this network connection, local computers could be joined directly to the TEDS network, but the DT would be unable to make use of high-performance compute or cloud environments and could not connect to remote data sources, including the acquisition of live price data, or other remote environments that could allow for superior compute capabilities or even the integration of additional test beds, assets, or data. The completion of this network connection is primarily dependent on the wider INL network owners and administrators and is expected to take 2–4 months to complete.

Another challenge for a physical demonstration is the integration of the DT and HMI. While TEDS is currently using the LabVIEW software as its HMI implementation, there is a planned switch to the

IgnitionEDGE HMI system built by Inductive Automation in early 2024. The DT integration will be different, depending on which HMI system is in place. Ideally, to minimize any wasted effort, this integration would only be built once for the physical demonstration and the demonstration would be complete either before or after the successful switch to IgnitionEDGE. Past physical demonstrations using DTs and LabVIEW have used a simple file passing back and forth between systems [45]. However, the use of APIs or similar methods should be examined. These methods of communication could eliminate the need for file storage and management and allow for more frequent, performant, and event-driven messaging between the HMI and DT. Regardless of which HMI will be used, both the HMI and DT will require some amount of development to enable this integration. For each direction of communication, the source will be required to organize and send the required data (either sensor information or suggested control actions) and the destination will need to accept, validate, and apply the data as necessary.

Regarding data integration, there is no significant gap to fill for physical demonstration; however, larger physical assets require a more complicated graph structure in a database. In that case, it is more important to organize how to design a graph and how to store each data to nodes. A previous study tested a preliminary graph for larger system such as TEDS [46] but further investigation will be needed to optimize the use of the graph database. Also, if the data warehouse scope is expanded further, graph design will take more time to create manually. In this case, natural language processing enhanced semiautomatic graph creation can be a potential resolution.

With V&V and integration completed, an experiment plan can be created that indicates the various physical and digital steps that will be taken as part of the demonstration. This plan should reference the various elements that will be involved in the demonstration as well as how they integrate and the validations that have previously occurred. This plan should be reviewed and approved by the various stakeholders, including the RTO DT research team, TEDS owners and operators, and the lab space coordinator and manager for the lab in which TEDS is located.

4.3 Connecting to Real-Time Price Data

The successful application of RTO is driven by fluctuations in the real-time energy market. This price data acts as the impetus for whether energy should be created, stored, or released (sold). To most effectively emulate the real environment in which an RTO DT would operate, real time, or as near real time as can be acquired, price data should be used as an input to the demonstration.

The California ISO Open Access Same-time Information System (OASIS) is a system that maintains real-time data related to the California ISO transmission system [47]. This data includes market prices, market results, capacity status, transmission outage, and system demand forecasts. OASIS maintains a developer site that allows users to request OASIS data in real time via an API. The connection to OASIS data via an API would require the creation of a certificate for authentication with OASIS and then the development of an application to read in market data from the OASIS source as requested by the DT or on a set interval. With real-time price data flowing in, the final step is to make sure this data is being properly used by the RTO model for price prediction and optimization steps.

4.4 Control System Vertical Integration

In addition to the integration with the physical system, as shown in Figure 1, ORCA also needs to be integrated with control systems at different time scales for optimal system performance. The Framework for Optimization of ResourCes and Economics (FORCE) tool suite, which comprises ORCA as well as other nuclear IES-related analysis tools, is being developed to integrate all the analysis tools to seamlessly leverage analysis and characterizations [48]. Specifically, two software interfaces are still needed: one for the integration between ORCA and long-term economic optimization and another for the integration with ORCA and short-term supervisory and regulatory controls. Integration is covered in Section 4.2. In addition, since ORCA relies on surrogate dynamic models based on the Modelica Dymola IES model [11], a workflow is needed to extract transient information from Modelica simulations and generate physics-based and data-driven surrogate models according to the RTO requirements. In this context, vertical integration refers to the knowledge extractions and transfers from highly dynamic and nonlinear systems to large-scale, long-term problems.

For the integration with long-term economic optimization, the objective is to accept plant and gridwise operational goals (e.g., IES configuration) and transfer them into ORCA objective functions. The Holistic Energy Resource Optimization Network (HERON) tool in FORCE is used for long-term technoeconomic viability analyses [48]. Fundamentally, HERON takes characterizations of an IES and projected market evolution as inputs and produces many examples of the optimal system dispatch as well as an analysis of optimal IES sizing. HERON also includes algebraic system optimization algorithms that may be useful to ORCA. Using FORCE to connect HERON algorithms and outputs with ORCA will allow long-term planning to transition smoothly to real-time system control. Meanwhile, ORCA's capability in dynamic simulations and optimizations could inform long-term technoeconomic optimizations with respect to the system's constraints and specifications.

The integration with Modelica focuses on the development and assessment of system dynamic models for RTO and potentially affecting the long-term technoeconomic viability. In this work, as illustrated in Section 3, the system dynamic model refers to reduced-order and surrogate models trained from simulation results from full-scale Modelica models. A typical development and assessment process for data-driven model includes five elements [49]:

- Establish requirements: the purpose is, based on the requirements of the RTO and target system, to set up purposes of the control system, transients, interfaces with other software and hardware system, modeling approaches, and primary functions.
- Develop knowledge base: the purpose is to build a database, which includes the explicit representations of the scenario space, simulation tools, reports, simulation results, and experimental data, which should be built based on established requirements.
- Develop surrogate models: the purpose is to set up the model development plan, including identifying input features, modeling approaches, and acceptance criteria, which should be informed by the established requirements and developed knowledge base.
- Assess model uncertainty and its impacts on RTO: the assessment should be separated into two tiers, a bottom-up approach that aims to evaluate each individual source of uncertainty for each surrogate model and a top-down approach that aims to analyze the overall vulnerabilities, reliability, and impact of the overall control system as a software in the IES instrumentation and control system, and should be informed by the developed knowledge base and surrogate models.
- Assess adequacy: the purpose is to evaluate the adequacy of the surrogate model and corresponding RTO applications in the target use cases based on the requirements. This approach usually relies on formal methods for determining the quality and confidence in applying the entire workflow in a consistent, transparent, and improvable manner. If the adequacy requirements cannot be satisfied, developers need to return to the appropriate elements for corrections. This iteration will continue until all adequacy standards are met.

Figure 16 shows a synthetic scheme for the vertical integration. In this example, ORCA will take system sizing information from long-term economic optimizations as the system constraints for RTO. ORCA will also collect surrogate models, including their model forms and parameters, from transient process models for constraints on system dynamics. At the same time, ORCA will give feedback limits and constraints from the perspective of system dynamics to the long-term economic optimization, focusing on the reachability and stability of proposed system sizing. ORCA will also provide information for transient modeling from an optimization perspective, focusing on the accuracy, convergence, and stability of surrogate models. Previous efforts have been focusing on streamlining knowledge base and model development, including model serialization and the use of FMUs and functional mock-up interfaces with RAVEN [50], and Modelica and external model interface in RAVEN [46]. More assessments are needed in quantifying model uncertainty and system adequacy.

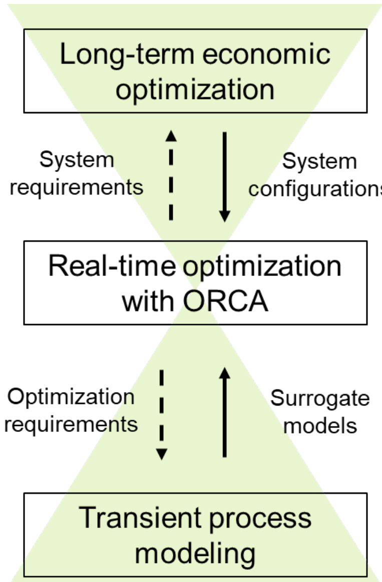


Figure 16. Schematic workflow for ORCA vertical integration with transient process models in Modelica and long-term economic optimization in HERON.

4.5 ORCA Software Quality Assurance

Software quality assurance (SQA) aims to confirm that ORCA is consistent with applicable requirements and to prevent software errors. The INL SQA assessments are scheduled at a specific frequency. As part of the INL SQA process, the goal is to assess risk factors associated with software quality, failure, performance, standards adherence, error, defects, security, and reliability. In 2010, the Nuclear Regulatory Commission issued regulatory guidelines describing methods that staff considers acceptable for complying with the provisions of Title 10, Part 50 and 52 [51]. The American Society of Mechanical Engineering (ASME) Nuclear Quality Assurance 1 certification program provides a means for an organization supplying software to have its quality assurance program recognized by ASME in conformance with the requirements of their standard. In 2011, the Nuclear Energy Institute issued the Nuclear Generation Quality Assurance Program Description. U.S. DOE also issued a quality assurance order to ensure products of DOE, including the National Nuclear Security Administration, meet or exceed customers' requirements and expectations. Currently, ORCA source code is managed on GitHub.com [13]. Performance and quality assessment techniques are still needed to evaluate and improve ORCA thoroughly with rigorous and effective corrective actions, including unit and convergence tests.

5. CONCLUSIONS

This report demonstrates ORCA for the RTO of energy storage and temperature control systems. In the virtual battery case, ORCA is coupled with a Modelica battery model for real-time energy dispatch optimization based on a synthetic sinusoidal price forecast model. ORCA can maximize the revenue, where power outputs are maximized when prices are high and minimized when price is low. The DeepLynx data warehouse was integrated with the ORCA and Modelica battery model. It stored and visualized data communicated between the two entities automatically after setting the initial condition. A temperature control problem is demonstrated by coupling ORCA with a physical TCLab module. With a user-assigned setpoint, the power of two heaters is controlled faster than real time such that the measured temperatures from the sensors align with the setpoints. The real-time requirements are satisfied by maintaining a solving time under 2 seconds. Moreover, two potential use cases on TEDS are discussed.

Based on the control architecture, ORCA can be deployed to determine the optimal heater power and TES power such that the revenue of TEDS can be maximized. Overall, ORCA shows acceptable accuracy and computational speed in solving RTO problems for IES and temperature control problems with a data integration capability realized by DeepLynx data warehouse.

Gaps are identified in deploying ORCA for a physical IES, including uncertainty quantification, integration with physical systems, connection to real-time price data, control system vertical integration, and SQA.

To support the RTO of TEDS, high-fidelity validation data from TEDS, including both experimental data and high-fidelity simulation results, are needed to set up correct constraints on system dynamics and specifications. A tentative list of scenarios is provided based on the importance and knowledge level in Appendix A. Moreover, considering the complexity of dynamic model and optimizations in RTO problems, more research is needed for the V&V framework to build a stronger assurance case for ORCA and corresponding RTO decisions. Such assurance should also include runtime assurance and monitoring, which should assimilate sensor data and development-time V&V results and activate alternative systems.

To integrate with physical systems, the primary gap is to build the interface programs for transferring information between RTO and a lower-level control system. This requires extensive collaborations with the experiment teams and system vendors in identifying their platforms and interfaces.

For real-time price data, the primary gap is to first make the integration between the data and RTO and then successfully forecast future price behavior, working these data and predictions into the RTO system.

For control system vertical integration, the primary challenges are extracting useful information from both long-term optimization schemes and system dynamics simulation results. The information extraction requires distillment and characterizations of long-term optimization results. It would also require a formalized development and assessment process for surrogate models.

Finally, ORCA SQA is needed to assess software risk factors, including managing software in a formal platform, creating unit tests, improving software capabilities, etc.

6. FUTURE WORK

Future work will be driven largely by the results of the gap analysis detailed herein. In addition to the research effort to establish and validate a comprehensive system for an IES RTO, it is critical to ensure that these efforts are best aligned with the overall IES needs and goals. As discovered frequently throughout the modeling and simulation efforts across the Nuclear IES program, the sensitivity of decision metrics to uncertain inputs frequently dwarfs the sensitivity to modeling approaches and assumptions. While early efforts in ORCA demonstrate a promising pathway from high-fidelity systems models into reduced-order modeling suitable for RTO as well as deploying this optimization for real-time semiautonomous control, the uncertainty in advanced nuclear systems as well as downstream heat users is dominant. As such, more may be gained by focusing on accurately understanding the IES that will be the target of optimal control than on building the optimization framework around those systems.

7. REFERENCES

[1] R. Byrne and C. Silva-Monroy, "Potential revenue from electrical energy storage in ERCOT: The impact of location and recent trends," *2015 IEEE Power & Energy Society General Meeting,* p. 105, 2015.

[2] R. H. Byrne, R. J. Concepcion and C. A. Silva-Monroy, "Estimating potential revenue from electrical energy storage in PJM," *IEEE Power and Energy Society General Meeting (PESGM),* pp. 1-5, 2016.

[3] T. Nguyen, R. Byrne, R. Concepcion and I. Gyuk, "Maximizing revenue from electrical energy storage in MISO energy & frequency regulation markets," *EEE Power & Energy Society General Meeting,* pp. 1-5, 2017.

[4] R. Byrne, T. Nguyen, D. Copp, R. Concepcion, B. Chalamala and I. Gyuk, "Opportunities for energy storage in CAISO: Day-ahead and real-time market arbitrage," *International Symposium on Power Electronics, Electrical Drives,* pp. 63-68, 2018.

[5] F. Wilches-Bernal, R. Concepcion and R. Byrne, "Participation of electric storage resources in the NYISO electricity and frequency regulation markets," *EEE Power & Energy Society General Meeting,* pp. 1-5, 2019.

[6] R. Concepcion, F. Wilches-Bernal and R. Byrne, "Revenue opportunities for electric storage resources in the southwest power pool integrated marketplace," *IEEE Power & Energy Society General Meeting,* pp. 1-5, 2019.

[7] S. Vejdan and S. Grijalva, "The expected revenue of energy storage from energy arbitrage service based on the statistics of realistic market data," *IEEE Texas Power and Energy Conference (TPEC),* pp. 1-5, 2018.

[8] D. Krishnamoorthy, B. Foss and S. Skogestad, "Steady-state real-time optimization using transient measurements," *Computers & Chemical Engineering,* vol. 115, pp. 34-35, 2018.

[9] M. Darby, M. Nikolaou, J. J and N. D, "RTO: An overview and assessment of current practice," *Journal of Process Control,* vol. 21, no. 6, pp. 874-884, 2011.

[10] D. Seborg, T. Edgar, D. Mellichamp and F. Doyle, Process dynamics and control, 2016.

[11] K. L. Frick, A. Alfonsi, C. Rabiti and D. M. Mikkelson, "Hybrid User Manual (INL/MIS-20-60624-Rev001)," Idaho National Laboratory, Idaho Falls, ID, USA, 2022.

[12] P. Talbot, D. Garrett, B. N. Hanna, J. Kim, T. Kajihara and L. Lin, "Optimization Of Real-time Capacity Allocation," Idaho National Laboratory, 2023.

[13] L. Lin, D. Garrett, T. Kajihara, J. Kim, J. Browning and P. Talbot, "ORCA," Github, [Online]. Available: https://github.com/idaholab/ORCA. [Accessed 23 August 2023].

[14] R. Onuschak and S. M. Bragg-Sitton, "Integrated energy systems program management plan," Idaho National Laboratory, Idaho Falls, ID, 2020.

[15] M. Ellis, H. Durand and P. Christofides, "A tutorial review of economic model predictive control methods," *Journal of Process Control,* vol. 24, no. 8, pp. 1156-1178, 2014.

[16] L. Lin, J. Oncken, V. Agarwal, C. Permann, A. Gribok, T. McJunkin, S. Eggers and R. Boring, "Development and assessment of a model predictive controller enabling anticipatory control strategies for a heat-pipe system," *Progress in Nuclear Energy,* vol. 156, p. 104527, 2023.

[17] L. Lin, P. Athe, P. Rouxelin, M. Avramova, A. Gupta, R. Youngblood, J. Lane and N. Dinh, "Development and assessment of a nearly autonomous management and control system for advanced reactors," *Annals of Nuclear Energy,* vol. 150, p. 107861, 2021.

[18] J. Oncken, L. Lin and V. Agarwal, "Adaptive Model Predictive Control for Heat-Pipe-cooled Microreactors under Normal and Heat Pipe Failure Conditions," in *13th Nuclear Plant Instrumentation, Control & Human-Machine Interface Technologies (NPIC&HMIT 2023)*, Knoxville, TN, 2023.

[19] W. E. Hart, C. D. Laird, J.-P. Watson, D. L. Woodruff, G. A. Hackebeil, B. L. Nicholson and J. D. Siirola, Pyomo - Optimization Modeling in Python, Springer, 2021.

[20] L. T. Biegler and V. M. Zavala, "Large-scale nonlinear programming using IPOPT: An integrating framework for enterprise-wide dynamic optimization," *Computers & Chemical Engineering,* pp. 575-582, 2009.

[21] J. W. Darrington, J. M. Browning and C. S. Ritter, *Deep Lynx: Digital Engineering Integration Hub,* 2020.

[22] A. A. Rashdan, J. Browning and C. Ritter, "Data Integration Aggregated Model and Ontology for Nuclear Deployment (DIAMOND): Preliminary Model and Ontology," 2019.

[23] J. M. Browning and K. N. Wilsdon, "Deep Lynx MOOSE Adapter," Idaho National Laboratory, 2021.

[24] K. N. Wilsdon, M. R. Kunz and J. M. Browning, "Deep-Lynx-ML-Adapter," Idaho National Laboratory, 2021.

[25] J. M. Browning and K. N. Wilsdon, "Deep Lynx Data Historian," Idaho National Laboratory, 2022.

[26] J. M. Browning, "Deep Lynx Unidirectional Network Connector," Idaho National Laboratory, 2022.

[27] K. N. Wilsdon, "Deep Lynx Matlab Adapter," Idaho National Laboratory, 2023.

[28] K. N. Wilsdon and J. M. Browning, "Deeplynx Supervisory Control Adapter," Idaho National Laboratory, 2023.

[29] J. M. Browning, "Deep-Lynx-Python-Package," Idaho National Laboratory, 2021.

[30] T. Kajihara, D. Garrett, J. Kim, L. Lin, J. M. Browning and P. Talbot, "Digital Twins for Optimizing the Real-Time Economy of Integrated Energy Systems," Knoxville, 2023.

[31] H. Wang, R. Ponciroli and R. Vilim, "Development of electro-chemical battery model for plug-and-play eco-system library (ANL/NSE-21/26)," Argonne National Laboratory, 2021.

[32] K. Frick, S. Bragg-Sitton and C. Rabiti, "Modeling the idaho national laboratory thermal-energy distribution system (teds) in the modelica ecosystem," *Energies,* vol. 13, no. 23, p. 6353, 2020.

[33] K. Frick, S. Bragg-Sitton and M. Garrouste, "Validation and verification methodology for INL Modelica-based TEDS models via experimental results," Idaho National Laboratory, Idaho Falls, ID, 2021.

[34] P. M. Oliveira and J. D. Hedengren, "An APMonitor Temperature Lab PID Control Experiment for Undergraduate Students," *2019 24th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA),* 2019.

[35] J. D. Hedengren, "Temperature Control Lab," Brigham Young University, 08 August 2023. [Online]. Available: http://apmonitor.com/pdc/index.php/Main/ArduinoTemperatureControl. [Accessed 16 August 2023].

[36] W. Hart, J. Watson and D. Woodruff, "Pyomo: modeling and solving mathematical programs in Python," *Math. Prog. Comp. ,* pp. 219-260, 2011.

[37] J. Park, R. A. Martin, J. D. Kelly and J. D. Hedengren, "Benchmark temperature microcontroller for process dynamics and control," *Computers & Chemical Engineering,* p. 106736, 2020.

[38] L. Beal, D. Hill, R. Martin and J. D. Hedengren, "GEKKO Optimization Suite," *Processes,* vol. 6, no. 8, 2018.

[39] L. D. R. Beal, D. C. Hill, R. A. Martin and J. D. Hedengren, "GEKKO Optimization Suite," *Process Modeling and Simulation,* vol. 6, no. 8, p. 106, 2018.

[40] B. Nicholson, J. D. Siirola, J.-P. Watson, V. M. Zavala and L. T. Biegler, "pyomo.dae: a modeling and automatic discretization framework for optimization with differential and algebraic equations," *Mathematical Programming Computation,* vol. 10, no. 2, pp. 187-223, 2-18.

[41] ASME, "Guide for verification and validation in computational solid mechanics (ASME Standard V&V 10-2006)," ASME, New York, NY, 2006.

[42] M. Luckcuck, M. Farrell, L. Dennis, C. Dixon and M. Fisher, "Formal Specification and Verification of Autonomous," *ACM Computing Surveys (CSUR),* vol. 52, no. 5, pp. 1-41, 2019.

[43] M. Clark, K. Kearns, J. Overholt, K. Gross, B. Barthelemy and C. Reed, "Air force research laboratory test and evaluation, verification and validation of autonomous systems challenge exploration," Air Force Research Lab, 2014.

[44] Air Force Research Laboratory, "Test and evaluation, validation and validation of autonomous systems challenge exploration final report," Air Force Research Laboratory, 2013.

[45] K. Wilsdon, J. Hansel, R. M. Kunz and J. Browning, "Autonomous control of heat pipes through digital twins: Application to fission batteries," *Progress in Nuclear Energy,* vol. 163, 2023.

[46] D. Garrett, T. Kajihara, J. Kim and P. Talbot, "Real-time optimization workflow status update," Idaho National Laboratory, Idaho Falls, ID, 2022.

[47] "California ISO OASIS," [Online]. Available: http://oasis.caiso.com/mrioasis/logon.do. [Accessed 30 August 2023].

[48] A. Epiney, P. Talbot and D. McDowell, "Roadmap for IES Modeling and Simulation Activities," Idaho National Laboratory, 2022.

[49] L. Lin, H. Bao and N. Dinh, "Uncertainty quantification and software risk analysis for digital twins in the nearly autonomous management and control systems: A review," *Annals of Nuclear Energy,* vol. 160, p. 108362, 2021.

[50] J. Cogliati, P. Talbot and K. Frick, "Status on the development of the infrastructure for a flexible Modelica/RAVEN framework for IES," Idaho National Laboratory, Idaho Falls, ID, 2022.

[51] U.S. NRC, "Software Quality Assurance Program and Guidelines," U.S. NRC, Washington DC, 1993.

Page intentionally left blank

Appendix A

Page intentionally left blank

Appendix A

Table A1. Graph plan file for node.

config.on_conversion_error	config.on_key_ extraction_error	type	root_array	name	keys.key	keys.value	unique_identifier_key
fail on required	fail on required	node		battery_node	name	Battery	soc
fail on required	fail on required	node		control_node	name	Control	time

Table A2. Graph plan file for edge.

initial	config.on_ conversion_error	config.on_key_ extraction_error	type	Name	root_array	origin	destination	relation
True	fail on required	fail on required	edge	battery_relates_control		Battery	Control	relates
False								
False								
False								
False								
False								
False								

direction	type	operator	key	value
origin	metatype_id	==		Battery
origin	original_id	==	soc	
origin	data_source	==		
destination	metatype_id	==		Control
destination	original_id	==	time	
destination	data_source	==		

Table A3. Graph plan file for time-series data source.

initial	adapter_Type	name	active	config.kind	config.chunk_interval	is_primary_timestamp	unique	id

True	tiemseries	BATTERY_TS	True	timeseries	100			
False						True	False	UUID
False						False	False	UUID
True	timeseries	CONTROL_TS	True	timeseries	100			
False						True	False	UUID
False						False	False	UUID
False						False	False	UUID
False						False	False	UUID
False						False	False	UUID
False						False	False	UUID

type	column_name	property_name	type2	value	operator
number64	time	time	metatype_id	Battery	==
float64	soc	soc			
number64	time	time	metatype_id	Control	==
float64	dc_start	dc_start			
float64	dc_end	dc_end			
float64	c_start	c_start			
float64	c_end	c_end			
float64	soc_start	soc_start			

Table A4. Normal power transient scenarios for TES.

ID	Scenario	Comments	Importance	Rationale	Knowledge Level	Rationale
1	Battery SOC	SOC determines the amount of available energy in TES at different modes and status.	High	Accurate inference of the SOC from measurable parameters.	Low	Current operating relies on constant setpoints at discrete modes.

				SOC directly affect optimal heat storage and extraction strategies in response to different demands.		
2	Transitions between different modes	In different modes, TEDS needs to actuate heaters, charging and discharging values while maintaining stable outputs.	High	Sustain stability and reject disturbance due to component activation/deactivation.	Low	
3	BOP valve stuck open	A typical accident scenario that affects RTO results.	High	Accurate diagnosis of system states affects RTO.	Medium	Existing operating data.
4	Couple with primary loop	The operation of primary loop heat generation device could affect the performance of TEDS.	Medium	The dynamics and constraints of primary loop device affect the charging and BOP of TEDS.	Low	Current operation relies on direct heating device.
5	Couple with electricity generation device	Replace heat sink by electricity generation device	Medium	Actual load-following and demand-driven optimizations.	Low	Current operation relies on heat sink.
6	System startup	Startup transient of TES	Low	Safely start up the system to 100% power	Medium	Existing data on startup (Mode 1).
7	System shutdown	TEDS shutdown procedures and timing could affect the entire system.	Low	Safety shut down the entire system from all operating modes	Low	The current shutdown neglects integrated system behaviors.