INL/EXT-21-65476

Light Water Reactor Sustainability Program

Release a Public Version of HERON (HERON 2.0) with Improved Algorithms for the Treatment of Energy Storage

December 2021

U.S. Department of Energy

Office of Nuclear Energy

INL/EXT-21-65476

Release a Public Version of HERON (HERON 2.0) with Improved Algorithms for the Treatment of Energy Storage

Binghui Li, Paul W. Talbot, Dylan McDowell, Jason K. Hansen

December 2021

Prepared for the
U.S. Department of Energy
Office of Nuclear Energy

EXECUTIVE SUMMARY

Integrated energy systems (IESs) are essential for decarbonizing electricity and industrial sectors and fully exploiting these systems requires sophisticated planning, scheduling, and dispatching tools to maximize their socio-economic benefits. The Holistic Energy Resource Optimization Network (HERON) is a generic software plugin for the Risk Analysis Virtual Environment (RAVEN) to perform stochastic technoeconomic analysis of IES with economic drivers. This report summarizes the updates made to HERON 2.0. Particularly, we demonstrate one of the added features, Function-based Control Mechanics, by comparing a price-based dispatch strategy with the original perfect foresight baselines. Our case studies successfully demonstrate that the model is capable to incorporate artificial control algorithm into the dispatch of IES components. In addition, our results from the price-based strategy indicate that the control strategies of IES components are key to the economic and temporal performance of our model; therefore, more sophisticated dispatch strategies are required to improve the model performance.

ACKNOWLEDGEMENTS

This research made use of resources of the High Performance Computing Center at Idaho National Laboratory, which is supported by the Office of Nuclear Energy of the U.S. Department of Energy and the Nuclear Science User Facilities under Contract No. DE-AC07-05ID14517.

CONTENTS

FIGURES

TABLES

ACRONYMS

AI	artificial intelligence
API	application programming interfaces
CDF	cumulative distribution functions
HERON	Holistic Energy Resource Optimization Network
IES	integrated energy systems
INL	Idaho National Laboratory
NPV	net present value
NYISO	New York Independent System Operator
PF	perfect foresight
RAVEN	Risk Analysis Virtual Environment
ROM	reduced-order models
RTE	round-trip efficiency

RELEASE A PUBLIC VERSION OF HERON (HERON 2.0) WITH IMPROVED ALGORITHMS FOR THE TREATMENT OF ENERGY STORAGE

1. INTRODUCTION

Integrated energy systems (IES) are essential for decarbonizing electricity and industrial sectors [1-3]. Innovation in design and utilization of these systems is required now in order to meet the Biden administration's aggressive decarbonization goals [4]. These systems may consist of one or more nuclear reactors, renewable energy sources, and energy storage systems that are tailored in their design and operation to meet specific objectives, such as providing flexible electricity and providing heat for manufacturing processes [5,6]. The complex network structure of the IES, however, requires comprehensive examination of their cost-effectiveness, reliability, and resilience to extreme events before large-scale deployment. Although studies to date have been conducted to optimize the operation of electrical power systems [7-9], IES is a multi-carrier energy system that usually includes other forms of end-use demand, e.g., heat and cooling. Fully exploiting these systems requires sophisticated planning, scheduling, and dispatching tools to maximize their socio-economic benefits [10].

The Holistic Energy Resource Organization Network (HERON) is a generic software plugin for the Risk Analysis Virtual Environment (RAVEN) developed by Idaho National Laboratory (INL)[11] to perform stochastic technoeconomic analysis of IES with economic drivers [12]. HERON is developed to drive optimization via economic drivers such as system cost minimization, profitability, and net present value (NPV) maximization. It can be used to construct workflows solving complex resource allocation problems that involve multiple forms of energy carriers. In addition, HERON leverages the synthetic history training and generation tools, sampling workflows, code Application Programming Interfaces (API), and optimization schemes. At each time point in a dispatch optimization, HERON attempts to calculate the most optimal usage of each component in the system.

Since its last stable release in 2020, several additions have been made to HERON, ranging from user-facing features to optimizations and new features. We summarize these improvements and their impact on HERON users in this section. The following is a short list of the improvements:

- Validator granted access to metadata: Allows Validators access to run metadata, including macro index, micro index, cluster information, etc.
- Debug mode for Example Dispatch: Adding the "debug" node to the Case node in a HERON XML input allows a custom, simplified run that only performs a select set of dispatches and produces plots of dispatch decisions. This allows users to carefully review some examples of dispatch optimization to ensure the system is working as expected before performing full-scale analyses. See, for example, "HERON/tests/integration_tests/mechanics/debug_mode/heron_input.xml"
- Activity-type ValuedParam: Allows users to use the "activity" of a component as the source of values for CashFlow entries. This is particularly useful for the CashFlow Driver, where often the amount of resources consumed or produced by a component leads directly to a CashFlow such as variable operations and maintenance costs or sales profits. See, for example, "HERON/tests/integration_tests/mechanics/pyomo_options/heron_input.xml"
- ROM-type Source: Allows Reduced-Order Models (ROMs) trained by RAVEN to be used as fixed-source data types, such as pricing models or demand generators. See, for example, "HERON/tests/integration_tests/mechanics/ROM_source/heron_input.xml"
- Manual Compilation on Windows: Using a batch file, the HERON manual can now be compiled from the LaTeX source on Windows machines.

- Automatic optimization path plotting: After running an "opt" mode HERON case, generic figures plotting the optimization path taken by the optimization variables is produced and saved in the working directory. These figures are a series of plots sharing an x-axis ("iterations"), each of which show accepted, rejected, and rerun optimization points, with successive accepted points linked together. The objective (e.g., NPV) is also plotted as a function of optimizer iteration.
- Function-based Control Mechanics: Allows a user to specify control strategies for storage units. By default, HERON still uses a perfect foresight governance strategy; this offers an alternative for control via custom Python strategy, machine learning or artificial intelligence algorithms, etc. For example, see "HERON/tests/integration_tests/mechanics/storage_func/heron_input.xml"
- Round-Trip Efficiency (RTE): Allows storage units to include a "loss" factor for round-trip resource acquisition and release. HERON simulates this loss as occurring in equal parts as the resource is absorbed and as it is released, multiplying by the square root of the RTE at each stage to apply loss. See, for example, "HERON/tests/integration_tests/workflows/storage/heron_input.xml"

The remainder of this document focuses on the Function-based Control Mechanics as the primary improvement in this release of HERON. The goal of using HERON in this demonstration is to optimize the dispatch strategy of the IES components and subsystems under volatile market electricity prices with the simultaneous constraint that energy balance and conversion efficiencies must be satisfied. We will demonstrate the new Function-based Control Mechanics by comparing the default HERON strategy with perfect foresight with a user-defined price-based control strategy. Note that the price-based control strategy accounts for uncertainties associated with imperfect foresight into the future, which can be integrated with real-world operation of IES.

2. STUDY CASES

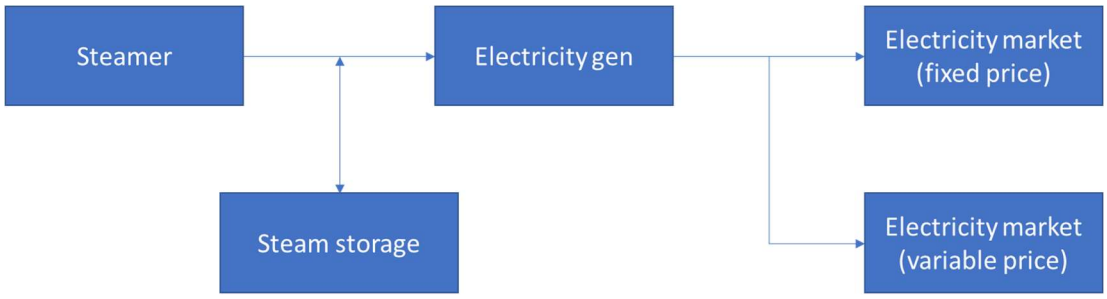

Figure 1. Schematic overview of the studied system.

The example system in this study is shown in Figure 1. The system consists of a steam generator, a steam storage device, an electricity generator that takes in steam and outputs electricity. Table 1 lists techno-economic parameters of the constituent components. Note that the capacity of the steamer is varied over $[1, 100]$ in the outer HERON loop to determine the optimal value. The generated electricity can be sold to electricity markets at either a fixed price or a variable price. We include the electricity market with variable prices to demonstrate the new Function-based Control Mechanics, which allow a user to apply a customized control algorithm to the IES components. In this study, we will design a price-based dispatch strategy based on the variable electricity price and apply it to the steam storage system. The modeled time horizon spans three years from 2015 to 2017, and we generate two samples (identified by 0 and 1) based on historical prices from New York Independent System Operator (NYISO) . The price profiles are presented in Figure 2.

Table 1. Techno-economic parameters of components of the studied IES.

Symbol	Notes	Value	Unit
x^S	Capacity, steamer	[1,100]	MW
x^E	Capacity, electricity generator	500	MW
η^E	Efficiency, electricity generator	0.5	–
x^{SS}	Capacity, steam storage	100	MWh
x^{MF}	Capacity, electricity market (fixed price)	2	MW
x^{MV}	Capacity, electricity market (variable price)	200	MW
r	Discount rate	0.08	–
–	Inflation	0	–
–	Tax	0	–

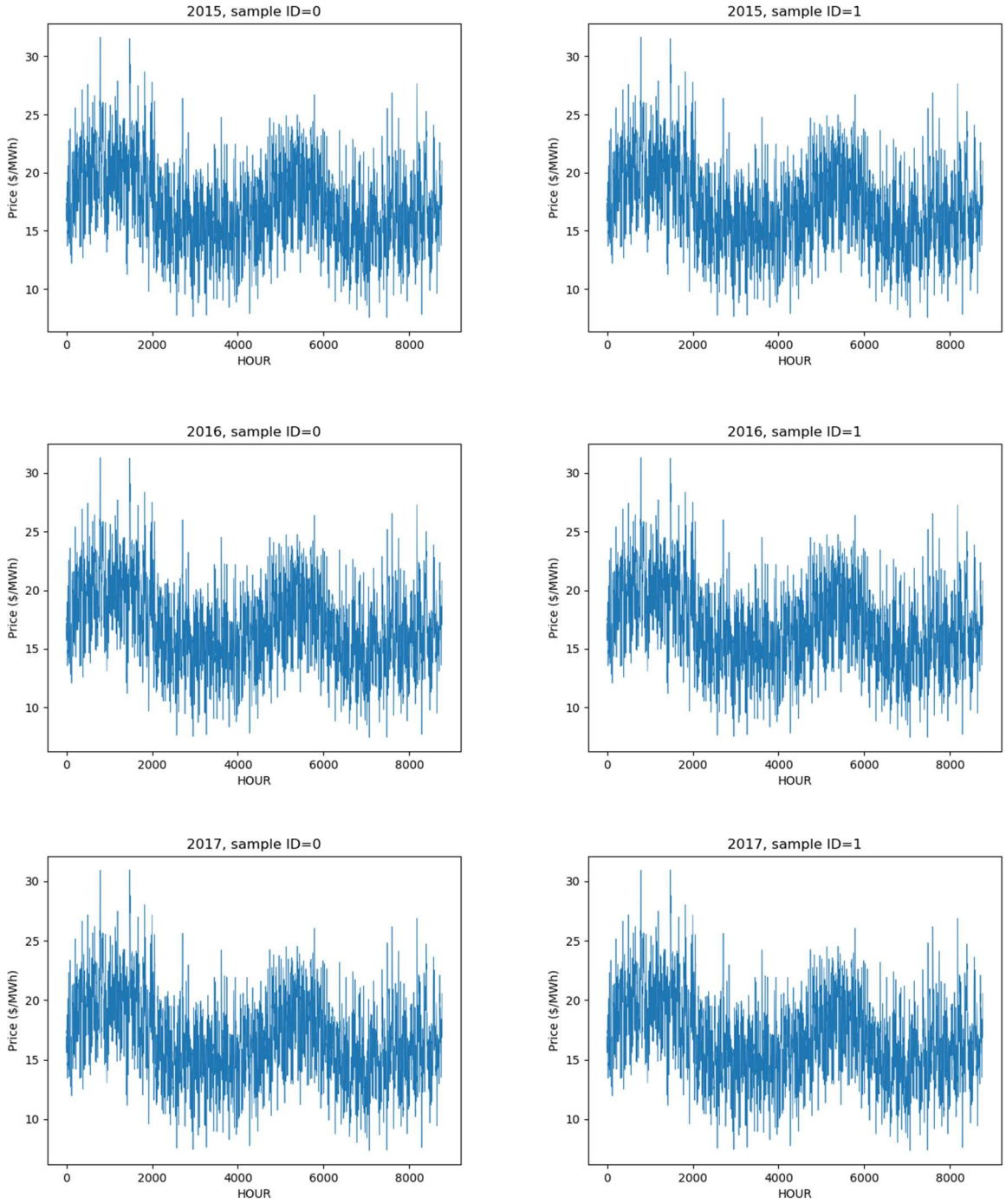

Figure 2. Electricity prices of the variable electricity market.

3. DISPATCH OF STORAGE DEVICES

3.1 Baseline: Perfect Foresight

The baseline case determines the dispatch schedules of (in this demonstration) the steam storage system by maximizing the total revenue of electricity sales and by assuming perfect foresight (PF) of electricity prices into the future, which means the operator knows the exact future electricity prices in advance. Note that the dispatch optimization aims to motivate the correct choice of technologies in HERON analyses; thus, assuming perfect foresight, while not realistic in real world, this serves to correctly motivate technology deployment in the outer optimization of HERON. Ideally, the optimization model should determine the dispatch schedule based on the electricity prices of the entire year–i.e., 8760 hours–to guarantee global optimality; however, considering 8760 hours would lead to an increased complexity of the optimization problem and drastically increase the model solving time. Therefore, we traditionally consider a reduced dispatch horizon of 24 hours and solve the reduced dispatch model 365 times. Note that the 365 dispatch models are solved sequentially to ensure the continuity of energy levels in the steam storage system. All cases in this study are run on a compute node with 12 cores and 128 GB of memory from INL's High Performance Computing platform. The time consumption and net present values (NPV) of the runs with both 8760-h and 24-h dispatch window are summarized in Table 2. By applying the 24-h dispatch window, the total revenue is reduced by only 0.05%, but the time consumption is reduced by over 90%.

Table 2. Summary of NPV and time consumption from both the 8760-h PF and 24-h PF.

Dispatch window (h)	NPV (ID: 1) ($)	Percentage	Time (s)	NPV (ID: 2) ($)	Percentage	Time (s)
8760	12,581,762.04	100.00%	875.1	12,660,397.67	100.00%	876.0
24	12,575,441.52	99.95%	71.1	12,654,514.54	99.95%	68.3

We further examine the relationship between electricity prices (π) and net changes of storage level (dE) in Figure 3. Note that $dE_t = E_t - E_{t-1}$, where E_t is the storage level at the end of time t. Across all years, higher electricity prices tend to result in negative dE, which indicates dispatching steam to generate electricity, whereas lower electricity prices are often associated with positive dE, i.e., storing steam. This observation suggests that the model is more likely to sell electricity to make profits under greater electricity prices and also is more likely to fill the steam storage system to avoid losses when electricity prices are low.

A closer look at Figure 3 suggests that price thresholds might be used to guide the dispatch of the steam storage system. For example, when the Sample ID = 0, $dE \leq 0$ when $\pi \geq 27$ $/MWh across all three years, implying that the model will stop charging when electricity price is greater than 27 $/MWh; on the other hand, $dE \geq 0$ when $\pi \leq 8$ $/MWh, indicating that the model will stop discharging when electricity price is lower. Similarly, the charging and discharging price thresholds when the Sample ID = 1 are approximately 28 and 7 $/MWh, respectively. When the electricity prices are in between the charging and discharging price thresholds, the model could charge, discharge, or do nothing. This observation implies that we can design a price-based strategy to operate the steam storage system.

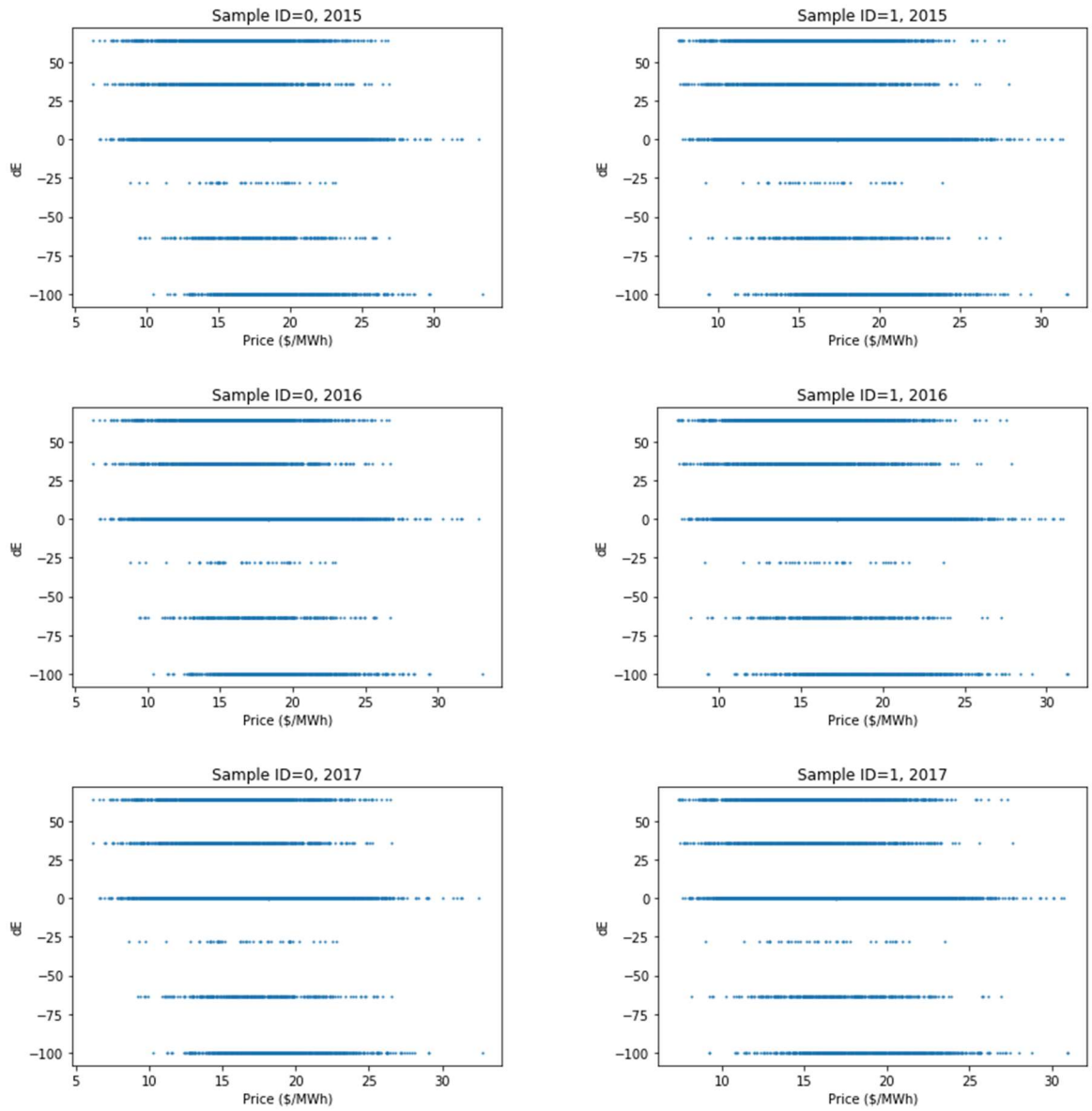

Figure 3. Scatter plots showing relationship between electricity prices (π) and net changes of storage level (dE) in the steam storage system. The dispatch horizon is 8760 hours. Note that dE > 0 indicates storing steam and dE < 0 indicates dispatching steam.

3.2 Price-based Dispatch

Although we can achieve global optimality by assuming perfect foresight, in real world, the dispatch schedules must be determined in the absence of future prices. The lack of perfect foresight introduces uncertainties in the model and often negatively impacts the profitability. To simulate real-world dispatch of energy storage systems, we adopt a simple price-based dispatch strategy. This strategy determines the dispatch schedule based on price thresholds: When the market electricity price is greater than a price threshold, steam is dispatched from the storage device to the electricity generator, whereas if the market electricity price is smaller than a price threshold (this threshold can be different from the previous threshold), steam is stored into the storage device. Figure 4 shows the flowchart of the price-based dispatch with an upper (π^{up}) and a lower price (π^{dn}) threshold.

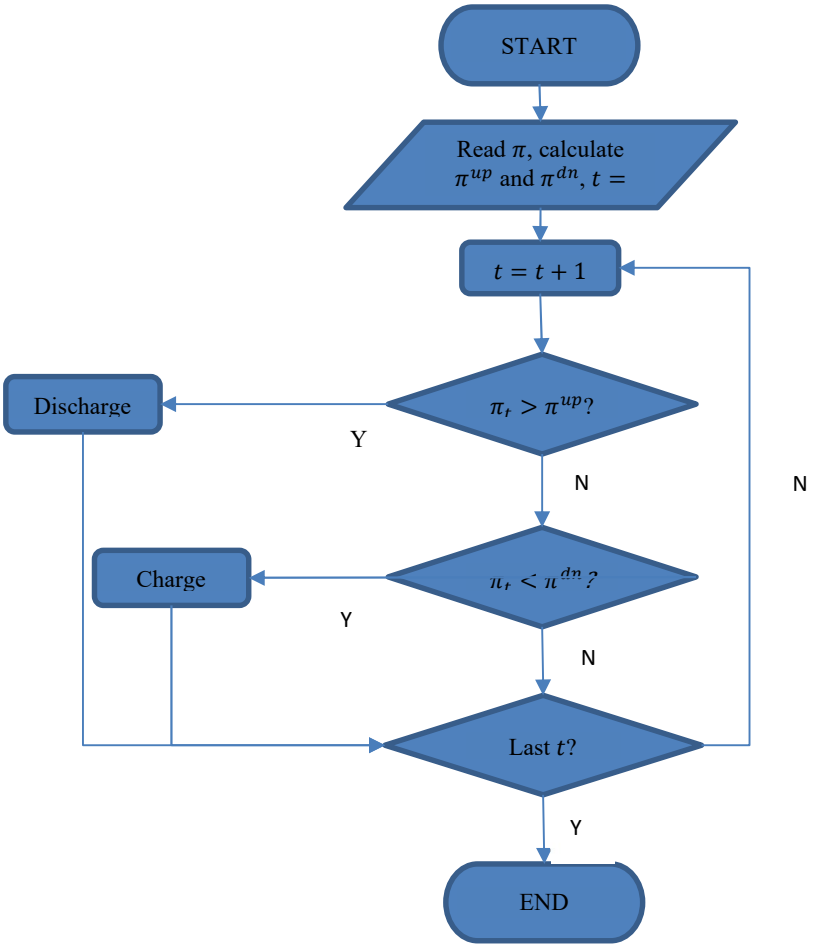

Figure 4. Flowchart of the price-based dispatch for the steam energy storage system. Note that π represents the electricity price time series, t denotes time, π^{up} and π^{dn}represent the upper and lower price thresholds, respectively.

Finding adequate price thresholds are key to the performance of the price-based dispatch method. As Figure 3 suggests, there are clear price thresholds of charging and discharging in the baseline cases; however, without knowing future electricity prices in advance, it is challenging to select the optimal price thresholds. Therefore, we examine the cumulative distribution functions (CDF) of electricity prices and select a set of percentiles as the price thresholds. Each time, we select a pair of upper and lower percentiles ($\alpha^{up} > \alpha^{dn}$) to derive the discharging (or upper, π^{up}) and charging (or lower, π^{dn}) price thresholds using the following equation:

$$\pi^{\alpha} = F^{-1}(\alpha) \quad (1)$$

where function F^{-1} is the inverse of the CDF, which can be derived from history data. Figure 5 shows that the CDFs of all years, and Table 3 summarizes the selected percentiles and corresponding electricity prices used in this study.

Table 3. Selected percentiles and corresponding prices ($/MWh).

	Sample ID = 0			Sample ID = 1		
Percentile	2015	2016	2017	2015	2016	2017
5th	11.28	11.15	11.02	11.56	11.44	11.33
25th	14.61	14.46	14.30	14.77	14.62	14.47
50th	16.98	16.80	16.63	17.05	16.87	16.69
75th	19.49	19.30	19.10	19.61	19.41	19.21
95th	23.12	22.93	22.74	23.11	22.89	22.69

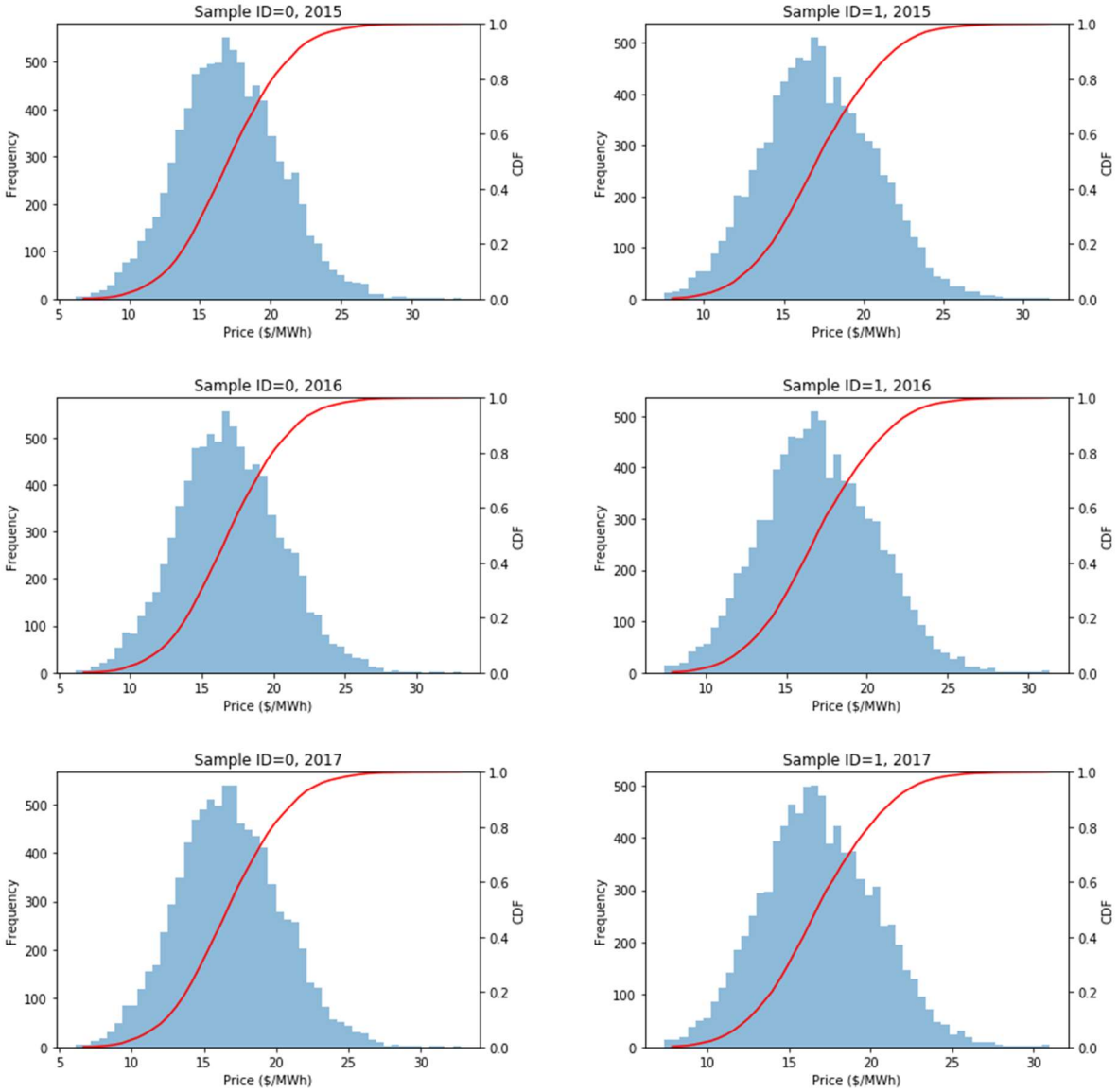

Figure 5. Frequency distributions and CDFs of electricity prices π of all years in this study.

After the charging/discharging status is determined, the next step is to determine dE, i.e., the change of storage level, which represents the amount of steam that will be charged/discharged. We first consider a simple method, where dE is only determined by the capacities of all upstream and downstream components, i.e., the steamer and electric generator, and remaining capacity in the storage system. Where dE^+ denotes the amount of steam charge and dE^-denotes discharge, they can be expressed as follows:

$$dE^+ = \min(x^S \cdot \Delta t, x^{SS} - E_t) \tag{2}$$

$$dE^- = -\min\big((x^E - x^S) \cdot \Delta t, E_t\big) \tag{3}$$

where Δt denotes the size of a time slice, E_t denotes the storage level, x^E, x^S, and x^{SS} represents the capacities of electric generator, steamer, and steam storage, respectively. Note that the capacities of x^E and x^S are in MW, and the unit of x^{SS} is MWh. Given the above definition, dE can be expressed as a two-segment piece-wise linear function of the electricity price π_t:

$$dE = \begin{cases} dE^-, \forall \pi_t > \pi^{up} \\ dE^+, \forall \pi_t < \pi^{dn} \end{cases} \tag{4}$$

The above equation implies that $dE = 0$ when the price $\pi_t \in (\pi^{dn}, \pi^{up})$, which means that the storage level remains constant when the price is in the middle. We further introduce a three-state dispatch strategy, where dE is linearly correlated with the percentile of π_t when the price is between π^{dn} and π^{up}; therefore, dE can be expressed by a three-segment piece-wise linear function of π_t:

$$dE = \begin{cases} dE^-, \forall \pi_t > \pi^{up} \\ \frac{\alpha_t - \alpha^{dn}}{\alpha^{up} - \alpha^{dn}} \cdot dE^- + \frac{\alpha^{up} - \alpha_t}{\alpha^{up} - \alpha^{dn}} \cdot dE^+, \forall \pi_t \in (\pi^{dn}, \pi^{up}) \\ dE^+, \forall \pi_t < \pi^{dn} \end{cases} \tag{5}$$

where the price percentile α_t is given by the price CDF: $\alpha_t = F(\pi_t)$. The graph of this function is shown in Figure 6.

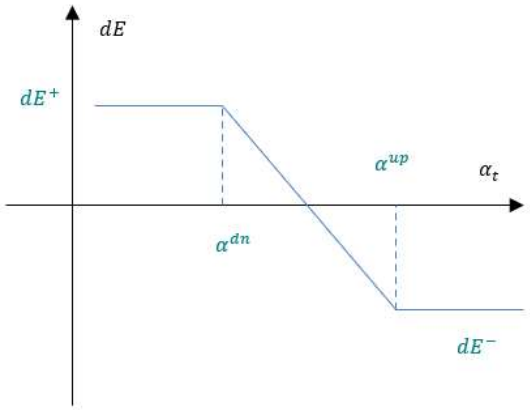

Figure 6. The graph of the three-state dispatch function.

By applying the price-based dispatch strategy, the results are presented in Figure 7 and Figure 8. By expressing total revenues as percentages of the revenue from the 8760-h PF baseline, Figure 7 compares the total revenues against the values in Table 2. Not surprisingly, the price-based dispatch strategy results in smaller total revenues than the baseline. This is because without perfect foresight, the price-based dispatch is an approximation to the baseline, where the price-based decision-making process can be viewed as additional constraints to the original optimization problem. Since adding additional constraints tightens the feasible region of an optimization problem, the price-based dispatch will only result in less revenues than the baseline.

A comparison of total revenues across different pairs of α^{up} and α^{dn} shows that the total revenue is higher when $\alpha^{up} \in [50\%, 75\%]$ and $\alpha^{dn} \in [25\%, 50\%]$. In addition, it also suggests that using a 24-h dispatch window typically results in greater revenues than the 8760-h dispatch window when the same dispatch strategy is used. For example, when the three-state dispatch strategy is used, the maximum total revenues with a 24-h dispatch window are 97.41% (ID = 0) and 97.43% (ID = 0), greater than 97.36% (ID = 0) and 97.39% (ID = 1) when an 8760-h dispatch window is used. Similarly, when the two-stage dispatch strategy is used, the maximum total revenue with a 24-h dispatch window is 98.58% (ID = 0), greater than 97.74% with an 8760-h window. Note that in the other sample (ID = 1), using a 24-h dispatch window results in smaller total revenue (97.76%) than using an 8760-h window (97.83%). This suggests that more samples are required to compare the performance of using different dispatch windows.

8760-h window, two-state dispatch, ID = 0

α^dn \ α^up	5	25	50	75	95
5	97.33%	97.33%	97.33%	97.15%	
25		97.74%	97.72%	97.40%	
50			97.72%	97.40%	
75				97.40%	
95					

8760-h window, two-state dispatch, ID = 1

α^dn \ α^up	5	25	50	75	95
5	97.36%	97.36%	97.44%	97.17%	
25		97.78%	97.83%	97.37%	
50			97.83%	97.37%	
75				97.37%	
95					

24-h window, two-state dispatch, ID = 0

α^dn \ α^up	5	25	50	75	95
5	96.95%	96.95%	96.95%	96.95%	
25		97.73%	98.58%	98.26%	
50			98.12%	97.79%	
75				97.04%	
95					

24-h window, two-state dispatch, ID = 1

α^dn \ α^up	5	25	50	75	95
5	96.96%	96.96%	96.96%	96.96%	
25		96.96%	96.96%	96.96%	
50			97.76%	97.76%	
75				97.04%	
95					

8760-h window, three-state dispatch, ID = 0

α^dn \ α^up	5	25	50	75	95
5	97.20%	97.20%	97.31%	97.28%	
25		97.36%	97.35%	97.30%	
50			97.35%	97.30%	
75				97.30%	
95					

8760-h window, three-state dispatch, ID = 1

α^dn \ α^up	5	25	50	75	95
5	97.22%	97.22%	97.34%	97.31%	
25		97.39%	97.38%	97.33%	
50			97.38%	97.33%	
75				97.33%	
95					

24-h window, three-state dispatch, ID = 0

α^dn \ α^up	5	25	50	75	95
5	97.21%	97.21%	97.35%	97.33%	
25		97.39%	97.40%	97.36%	
50			97.41%	97.34%	
75				97.27%	
95					

24-h window, three-state dispatch, ID = 1

α^dn \ α^up	5	25	50	75	95
5	97.22%	97.22%	97.37%	97.35%	
25		97.42%	97.42%	97.38%	
50			97.43%	97.36%	
75				97.28%	
95					

Figure 7. Objective functions (total revenues) when the price-based dispatch strategies are applied, expressed as percentages of the baseline with an 8760-h dispatch window.

The time performance of the price-based strategies is shown in Figure 8. Similar to the PF baseline, using an 8760-h dispatch window results in significantly longer solving time than using a 24-h dispatch window. In addition, the price-based strategies require additional implementation time and double the time consumption in the PF baseline. With an 8760-h dispatch window, it takes 1,720 and 1,740 seconds to implement the 2-state and 3-state price-based strategies, respectively. The 24-h dispatch window reduces the time consumption by over 90%, but still requires around 120 seconds to complete, which almost doubles the 24-h PF baseline.

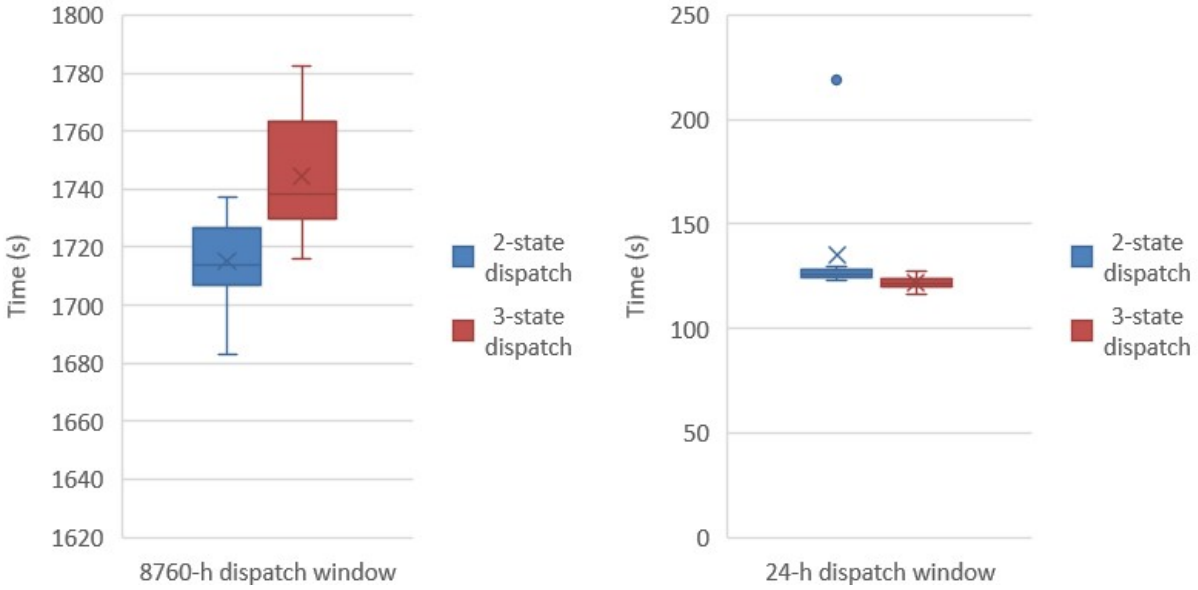

Figure 8. Time consumption of the price-based dispatch cases.

4. CONCLUSIONS

This report summarizes the updates made to HERON 2.0. Particularly, we have demonstrated Function-based Control Mechanics by comparing a price-based dispatch strategy with the perfect foresight baselines. Our case studies draw the following conclusions.

First, the solution time can be reduced substantially by using a reduced dispatch window size. By reducing the dispatch window from 8760 hours to 24 hours, the solution time has been reduced by up to 90%, and the total revenue is reduced by less than 1%. This suggests that we can use 24-h dispatch window without compromising the economic performance.

In addition, by examining the relationship between the electricity prices and the dispatch decisions in the PF baselines, we show that price thresholds can be used to guide the dispatch of the steam storage system. Furthermore, we design a simple price-based dispatch strategy and use different percentiles of prices as the thresholds. Our results indicate that total revenues from the price-based dispatch strategies are smaller than the PF baselines and the time consumptions are longer.

The price-based strategy indicates that the model is capable of incorporating artificial control algorithm into the dispatch of IES. Our results from the simple price-based strategy imply that the control strategies of IES components are key to the economic and temporal performance of our model; therefore, more sophisticated dispatch strategies are required to improve the model performance. Future work will include development of data-driven methods, such as using machine learning or reinforcement learning. These methods will depend on training artificial intelligence (AI) on intensive data sets of decisions with perfect foresight into future uncertainties.

5. REFERENCES

[1] Arent DJ, Bragg-Sitton SM, Miller DC, Tarka TJ, Engel-Cox JA, Boardman RD, et al. Multi-input, multi-output hybrid energy systems. Joule 2021;5:47–58.

[2] McMillan C, Boardman R, McKellar M, Sabharwall P, Ruth M, Bragg-Sitton S. Generation and use of thermal energy in the US Industrial sector and opportunities to reduce its carbon emissions. 2016.

[3] Bragg-Sitton SM, Boardman R, Rabiti C, O'Brien J. Reimagining future energy systems: Overview of the US program to maximize energy utilization via integrated nuclear-renewable energy systems. Int J Energy Res 2020;44:8156–69.

[4] Office of Federal Register. Executive Order on Tackling the Climate Crisis at Home and Abroad. Fed Regist 2021;86.

[5] Ruth M, Cutler D, Flores-Espino F, Stark G. The economic potential of nuclear-renewable hybrid energy systems producing hydrogen. 2017.

[6] Ruth MF, Zinaman OR, Antkowiak M, Boardman RD, Cherry RS, Bazilian MD. Nuclear-renewable hybrid energy systems: Opportunities, interconnections, and needs. Energy Convers Manag 2014;78:684–94.

[7] Wood AJ, Wollenberg BF, Sheblé GB. Power generation, operation, and control. John Wiley & Sons; 2013.

[8] Papavasiliou A, Oren SS, Rountree B. Applying High Performance Computing to Transmission-Constrained Stochastic Unit Commitment for Renewable Energy Integration. IEEE Trans Power Syst 2015;30:1109–20. https://doi.org/10.1109/TPWRS.2014.2341354.

[9] Abujarad SY, Mustafa MW, Jamian JJ. Recent approaches of unit commitment in the presence of intermittent renewable energy resources: A review. Renew Sustain Energy Rev 2017;70:215–23.

[10] O'Malley MJ, Anwar MB, Heinen S, Kober T, McCalley J, McPherson M, et al. Multicarrier Energy Systems: Shaping Our Energy Future. Proc IEEE 2020;108:1437–56. https://doi.org/10.1109/JPROC.2020.2992251.

[11] Rabiti C, Alfonsi A, Cogliati J, Mandelli D, Kinoshita R, Sen S, et al. RAVEN user manual. 2017.

[12] Talbot PW, McDowell DJ, Richards JD, Cogliati JJ, Alfonsi A, Rabiti C, et al. Evaluation of Hybrid FPOG Applications in Regulated and Deregulated Markets Using HERON. 2020.