Coupling Visualization, Simulation, and Deep Learning for Ensemble Steering of Complex Energy Models

Brian Bush, *Member, IEEE*, Nicholas Brunhart-Lupo,
Bruce Bugbee, Venkat Krishnan, Kristin Potter, and Kenny Gruchalla, *Senior Member, IEEE*,

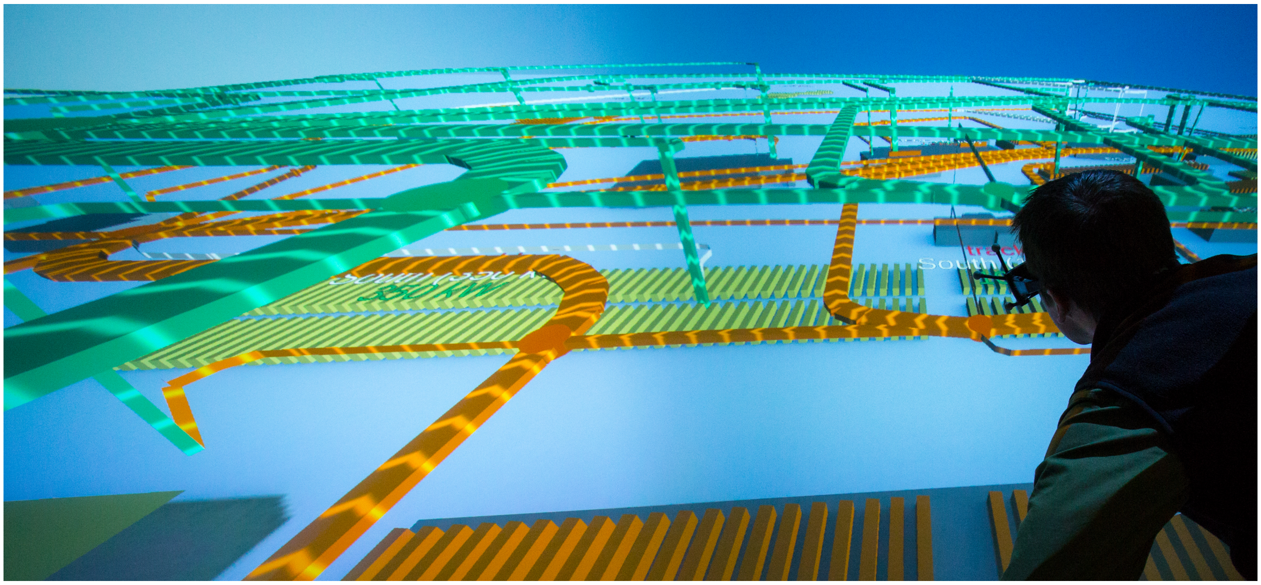

Fig. 1. Immersive visualization of an ensemble of energy simulations supports a campus renewable energy design study.

Abstract—We describe a new framework that allows users to explore and steer ensembles of energy systems simulations by coupling multiple energy models and interactive visualization through a dataflow API. Through the visual interface, users can interactively explore complex parameter spaces populated by hundreds, or thousands, of simulation runs and interactively spawn new simulations to "fill in" regions of interest in the parameter space. The computational and visualization capabilities reside within a general-purpose dataflow architecture for connecting producers of multidimensional timeseries data, such as energy simulations, with consumers of that data, whether they be visualizations, statistical analyses, or datastores. Fast computation and agile dataflow can enhance the engagement with energy simulations, allowing users to populate the parameter space in real time. However, many energy simulations are far too slow to provide an interactive response. To support interactive feedback, we are creating *reduced-form* simulations developed through machine learning techniques, which provide statistically sound estimates of the results of the full simulations at a fraction of the computational cost. These reduced-form simulations have response times on the order of seconds, suitable for real-time human-in-the-loop design and analysis. The approximation methods apply to a wide range of computational models, including supply-chain models, electric power grid simulations, and building models. Such reduced-form representations do not replace or re-implement existing simulations, but instead supplement them by enabling rapid scenario design and exploration for large ensembles of simulations. The improved understanding, facilitated by the reduced-form models, dataflow API, and visualization tools, allows researchers to better allocate computational resources to capture informative relationships within the system as well as provide a low-cost method for validating and quality-checking large-scale modeling efforts.

Index Terms—Deep learning, immersive visualization, databases, multidimensional time-series, neural networks, ensemble visualization, energy system models.

1 Introduction

Ensemble simulations, namely simulation suites using multiple models with varying input parameters and initial conditions, are a common approach to understanding highly complex natural phenomena. These simulation collections combine different models and settings to cover a range of possible outcomes and provide statistical measures indicating the similarity of individual model results. For these types of simulations, a major challenge is in determining appropriate parameter settings; often the number of parameters is quite large, some settings may fail to produce realistic results, and the cost to compute all parameter perturbations may be astronomical. Often, a pre-defined set of initial conditions and parameter settings is used, such as NOAA's Short-Range Ensemble Forecast (SREF) [14], but such an approach may not be ideal in other scenarios. More robust solutions must include methods to select regions within the parameter space that are of scientific interest and often this requires a user-in-the loop interface to guide simulations, tuning inputs within a specific range of inquiry.

To address these challenges, the National Renewable Energy Laboratory (NREL) has developed a framework for visualization-driven design, exploration, and analysis of energy simulations. The framework uses what we are terming *ensemble steering* to provide a user with

- *All authors are with the National Renewable Energy Laboratory.*
- *E-mails: {brian.bush, nicholas.brunhart-lupo, bruce.bugbee, venkat.krishnan, kristi.potter, kenny.gruchalla}@nrel.gov*

an overview of a simulation's parameter space via a visual analysis environment, and, based on the user's interplay, spawn new simulations to provide results fast enough to be interactive. In cases where the simulation response time is too slow for interactivity, we develop *reduced-form* models that approximate the full simulation model to enable interactive sessions; offline simulation of the full model may also proceed later, eventually producing more accurate results. This ensemble steering and analysis environment allows users and stakeholders to rapidly design alternative scenarios for simulation, quickly view approximate results of those simulations, and refine the design or explore the simulation results in depth. The computational and visualization capabilities reside within a dataflow architecture for connecting producers of multidimensional timeseries data with consumers of that data. The architecture is general-purpose, supporting a wide range of multivariate time-varying data producers, including measurements from real-time sensors and results from high performance computing (HPC) simulations, and supporting multiple concurrent consumers including visualizations, statistical analyses, and datastores. Consumers can request existing data records or can make a request for a non-existent record, spawning a new simulation to satisfy that request.

2 Ensemble Steering Framework

Computational steering is "the interactive control over a computational process during execution" [13], and allows a user to guide computation toward interesting aspects and react to previous results. Often this includes the ability to change or halt simulations while they are running and much of the research in simulation steering is on the interface between user and simulations [3,16,21]. Similar to the work presented here, systems such as World Lines [20] integrate visualization, simulation, and computational steering into a single framework, allowing the user to investigate alternative scenarios. A main distinction of our system is rather than steering or "nudging" [21] a simulation while being executed, we are using our framework to explore the parameter space of an ensemble via simulations running at an interactive pace, be it a full-scale simulation or an approximate model, similar to the conceptual framework proposed by Sedlmair *et al.* [17]. This approach quickly gives the user an overview of the relationship between the parameter and output spaces, allowing them to focus computational and time resources on specific areas of interest. Our system can act as a frontispiece to the full-scale simulation suite, moving the computationally intensive aspects of simulations outside of the traditional analysis workflow. By approximating results on the fly, we can maintain the low latency required for interactivity and heavy computation can be focused on scientifically relevant locations.

Our system is composed of three distinct areas of focus developed in concert, and designed to be general purpose in regards to their workflow connections, but customizable to be used within the many simulation scenarios existent at NREL. The main connector is the dataflow API, providing a highway between the visual analysis component, the reduced-form models, a datastore, and the HPC resource on which to spawn new simulations. This design provides the skeleton for ensemble steering as well as easy entry points for customization for each domain scenario.

2.1 A Dataflow API for Multidimensional Time-series

The dataflow API [1] normalizes interactions between software that serves multidimensional record-oriented data and the software that consumes such data. While developed with an eye toward time-series data, such is not a requirement. In the context of the API, multidimensional data records are defined as simple tuples consisting of real numbers, integers, and character strings. Each data value is tagged by a variable name according to a pre-defined schema, and each record is assigned a unique integer identifier. Conceptually, these records are isomorphic to rows in a relational database, JSON objects, or key-value maps. The objective of this API is to simplify and unify the interactions between records producers and consumers, with the idea being that any client designed to use this specification can speak to any server that implements the API. We have attempted to reduce the barriers of using this specification to a minimum, for both server and client implementations, so as not to preclude a micro-service model. Therefore, we primarily specify the data transport layer and messaging; storage, data structures, and other implementation details are left to the developer. However, as the API is closely related to common database models, most implementations merely need to provide a translation between backend database storage and the API.

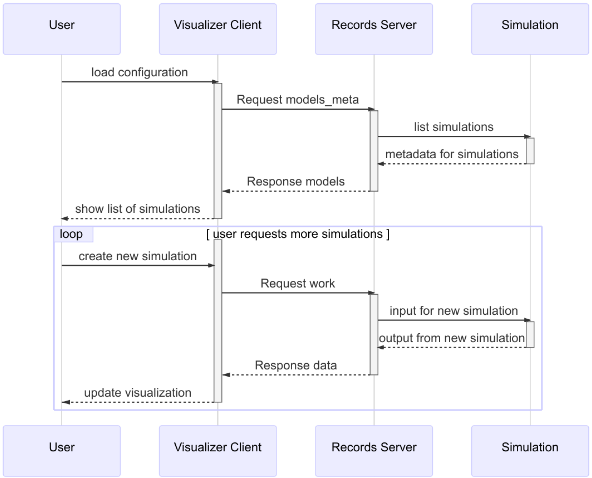

Fig. 2. Interaction diagram for discovering available simulation models, spawning new simulations, and visualizing the results.

The dataflow API is organized in a client-server model. Clients ask for available datasets (e.g., simulation results), receive extant data and any new records as they are generated, and, as needed, ask for the simulation of new data based on user input. A server may host multiple "models" (or tables, in database terms); a model may hold static unchanging data, but the design places emphasis on dynamic models, where records are being added continually, such as the case of sensor measurements being collected as new telemetry becomes available, or the generation of new simulation results. New records are then provided as a notification to clients. Following the pipeline model, dataflow API servers and clients can be chained together, creating a transformation path for records or even coupled models. Figure 2 shows a high-level view of the desired communication protocol for the simplest visualization use case. Separate server implementations of the API exist in C++ and Haskell; client implementations currently exist for C++, Haskell, JavaScript, Python, and R. Collectively, the servers support persistent backends for delimited-text files, databases (PostgreSQL, MySQL, SQLite3, and ODBC), and real-time sensor feeds (Haystack [9]).

The dataflow API also provides *bookmarking*, defined as a set of records or a query (the database analogue being an SQL view) that saves the current state of the environment. As an example, consider a researcher exploring a dataset. They may identify a set of records that represents a discovery or a query of interest, and define a tag of those records for later use. More importantly, bookmarks enable a collaborative approach to data exploration. Bookmarks can be shared across connected clients; another researcher can immediately pick up a bookmark and explore the same results, a client can continually create bookmarks of selected content so that the selection can be mirrored to another user, or clients may watch new bookmarks, and if a bookmark contains a certain tag, publish those results on a webpage.

Transport of the data is specified to take place over WebSockets [10]. WebSockets are ubiquitous and easily available to programmers of most languages. WebSockets provide a mechanism for poll-free notification and large message sizes. The format chosen for the message bodies themselves is Google Protocol Buffers [8]. Encoders and decoders for messages are automatically generated, reducing implementation effort and ensuring message correctness. In Figure 3, we show an example structure and options of our Records API system. Though we only

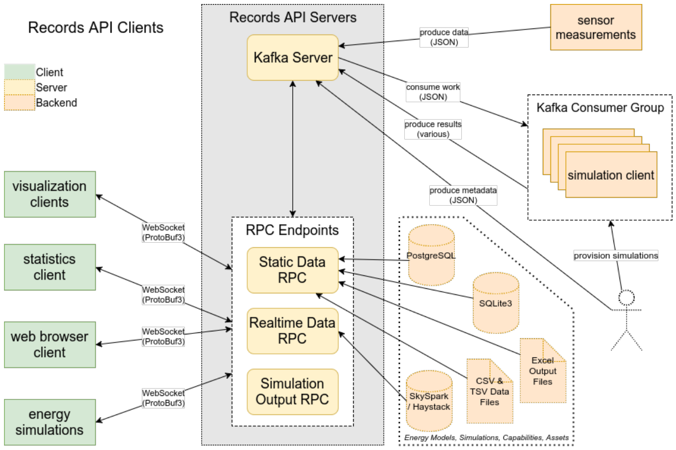

Fig. 3. Structure of an example Records API deployment. Note that the API only governs communication between remote procedure call (RPC) endpoints (servers) and the clients, as shown in the left and center columns of the diagram.

specify communication between client and server, the server itself has no restrictions on how it obtains data. Complex data harvesting systems can be completely abstracted away, providing a uniform method of data access.

In order to maximize usage by energy researchers who may not have extensive software engineering experience, this minimalist API avoids imposing burdensome metadata, structural, or implementation requirements on developers by relying on open-source technologies that are readily available for common programming languages. In particular, the API has been designed to place the smallest possible burden on services that provide data.

2.2 Deep Learning

Maximizing user engagement relies on simulations providing fast, responsive results on a sub-minute timescale; a low level of latency allows stakeholders to properly focus on exploration and inference in a truly interactive manner. Currently, only a few of NREL's large energy models are fast enough to be used seamlessly in this manner; most other important simulation suites are too computationally costly to operate in this fashion. Many of these models exhibit extensive regions of nearly linear behaviors, but those regions are punctuated by nonlinear transitions, jumps, or other critical phenomena. If the locations of the quasi-linear and the nonlinear regimes were mapped approximately, researchers could focus computation preferentially towards the nonlinearities while not sacrificing coverage of the more linear portions of parameter space. Simplified or reduced-form versions of NREL models would allow analysts to carefully plan their computer experiments with the full models, making far better use of computing resources. Therefore, we are developing *reduced-form* representations of these more computation-intensive energy models through machine learning.

Machine learning in this project has the primary aim of approximating aspects of interest of the energy simulations that are too costly to evaluate interactively. By framing this task in a standard statistical modeling framework, we have access to a plethora of methods for learning appropriate maps between relevant inputs and outputs. The choice and effectiveness of methods is highly dependent on the structure of both input and output data. For simple relationships, traditional regression methodologies such as linear regression, mixed models, gradient boosting, and random forests are competitive, particularly for one-dimensional output scenarios [11]. Longitudinal and functional analysis approaches are applicable when the stated goal is to represent some combination of input or output in functional form.

Since the primary objective of our approximate modeling is to achieve some level of predictive accuracy coupled with fast evaluation ability, we have paid particular attention to leveraging recent advances in neural networks for this project [7]. Flexibility in handling highly nonlinear relationships, advances in computational implementations, and the ability to handle multidimensional output spaces are all useful properties for the task at hand. It is important to note that while deep multilayer neural networks may take quite long to train, even with the modern GPUs available, this is a one time cost. Predictive evaluation of these networks is quite fast since it mainly requires efficient matrix multiplication and evaluation of nonlinear activation functions.

2.3 Interactive Visualization and Analysis

The challenges of rapidly developing insights from NREL's complex flagship models do not simply end with interactivity. Another substantial hurdle is abstracting features of interest from the high dimensional input/output of the models: in general, these simulation results contain lower dimensional geometric structures that have clear and insightful meanings/interpretations. Statistical and visualization techniques that identify and present such structures greatly speed the interpretation and exploration of dauntingly complex simulation results.

To facilitate the parameter space exploration and feature identification, we have developed within our framework the ability to connect a variety of data analysis environments to the dataflow API. The flexibility of our framework allows the integration of generic visualization clients such as R and Python Jupyter notebooks for quantitative analysis, web applications such as Shiny and D3 for broad deployment, and in-house tools developed for 3D immersive (i.e., head-tracked stereoscopic) environments. This multitude of visualization clients is an important aspect of our ensemble steering framework, allowing its use on different types of data and simulations scenarios. The system provides an interface to explore the multivariate ensembles as well as design new scenarios by manipulating input parameters. Thus, analysts can quickly develop and test hypotheses regarding the relationships between simulation inputs and outputs.

3 Discussion of Applications

To date, we have used our ensemble steering framework to develop customized workflows targeted at specific stakeholders exploring analytic questions using multiple energy models. We demonstrate the use of our framework on three examples, however the development of novel visualization techniques and reduced-form models is ongoing.

3.1 Renewable Energy Planning

NREL campus planners are using our ensemble steering framework to evaluate the energy impacts of a wide range of planning scenarios by combining technical, economic, and policy perspectives, and exploring the parameter space by interactively manipulating on-site power generation, electrical loads, and cost assumptions. The simulations combine techno-economic optimizations from REopt [18], whole building simulations from EnergyPlus [4], and power flow simulations from OpenDSS [5]. Figure 1 shows our immersive environment in which multiple stakeholders can gather and evaluate planning scenarios by walking inside a virtual campus, see the effects of various settings, and spawn new simulation runs. In the figure, campus buildings are shown in dark gray, and the lines are modulated by color and directional texture to show power flow variables. While in its early stages, we have already discovered opportunities for energy systems integration on our campus by bringing our site planners and leadership together in this environment, and have received requests to create similar models of other sites.

3.2 Biomass Supply-Chains

Energy analysts and stakeholders at NREL actively use in-house tools developed for the visualization of generic datasets of multidimensional timeseries to explore results of biomass supply-chain models such as the Biomass Scenario Learning Model (BSLM) [19], the Biomass Scenario Model (BSM) [15], and a waste-to-energy system simulation (WESyS). This suite of simulations uses the system dynamics methodology to model dynamic interactions within the supply chain: the models track the deployment of bioenergy given current technological development and the reaction of the investment community to those technologies.

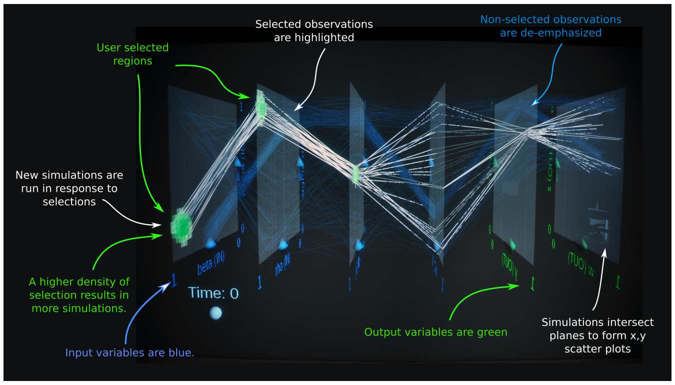
Selected observations are highlighted
Non-selected observations are de-emphasized
User selected regions
New simulations are run in response to selections
A higher density of selection results in more simulations.
Input variables are blue.
Time: 0
beta (IN)
rho (IN)
(TUO) y
(TUO) w
Output variables are green
Simulations intersect planes to form x,y scatter plots

Fig. 4. Parallel planes in an immersive virtual environment with annotations describing the visualization and user interface [2].

Immersive scatterplots and parallel planes [2] allow for the animated visualization of five to twenty dimensions of such timeseries. Figure 4 shows an immersive parallel-coordinates display of variables from the BSLM scenario. We observe that users of these visualizations can effectively explore ensembles of hundreds to tens of thousands of simulation results, interactively creating new simulations at the rate of several hundred per hour. Driving simulation studies from immersive visualization streamlines the simulation-analysis workflow.

Typically users alternate between hypothesis generation and hypothesis testing; in the hypothesis-generation phase they select, filter, and brush the existing ensemble of simulations, while in the hypothesis-testing phase they create new simulations whose input parameter sets they have tuned towards validating or falsifying the previous hypothesis. Fortunately, the round-trip time for creating new BSLM and WESyS simulations is less than ten seconds. In contrast, BSM simulations take three minutes to complete, somewhat hampering the user experience of visually responsive addition of the new ensemble results, but also motivating our development and deployment of reduced-form machine-learning models.

3.3 Electric Power System Capacity Expansion

The lack of immediate response from ensembles of simulations spawned by a visualization user is even more extreme in models like NREL's Regional Energy Deployment System (ReEDS), where each simulation in the ensemble takes five or more hours to complete. The ReEDS model is an electricity system capacity expansion model that develops scenarios of future investment and operation of generation and transmission capacity to meet U.S. electricity demand [6], representing the continental United States with a very high spatial resolution [12] and performing a system-wide least-cost optimization in two-year solve periods from 2010 out to 2050.

Initial efforts have focused on creating reduced-form predictive models for projected national capacity of a variety of resources. A dense multilayer neural network is used to map from a set of fixed category designations (demand scenario, utility-scale solar penetration scenario, etc.) to the projected capacity measurement from ReEDS. Figure 5 highlights a comparison between fully simulated and reduced-form predicted results for a small subset of estimated national wind capacities ranging from 2018 to 2050. The average percent deviation between predicted and ReEDS wind capacity was approximately 3%. Preliminary work on geothermal, coal, gas, and utility solar capacities showed similar results. We aim to expand these results by modifying the the available input data to incorporate continuous and potentially functional metrics rather than hard-coded scenarios. This would allow users to better explore new regions of the parameter space, either through mixing existing scenarios or "drawing" new input curves. Discussions with interested stakeholders in the electric power modeling domain suggest that both in-depth, immersive visualization and lightweight, web-based products using the reduced-form models would be of use. Both of these objectives fit nicely in our framework API and will continue to be developed.

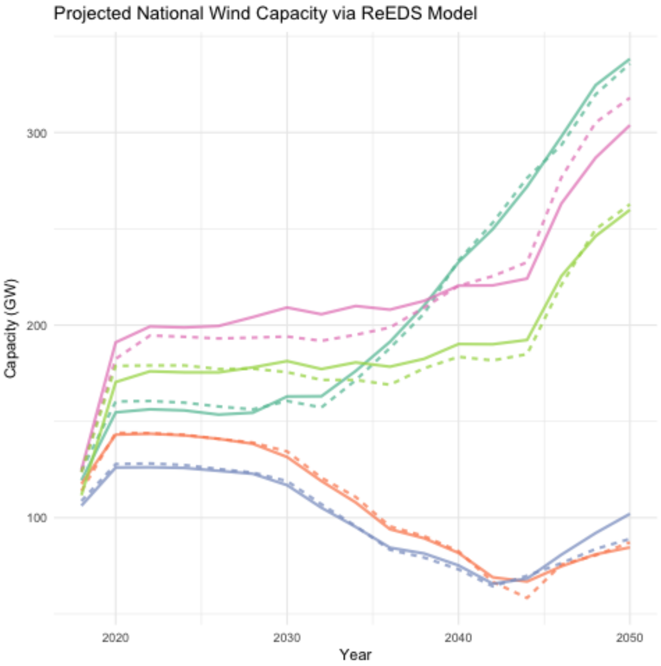

Fig. 5. Comparison of simulation and predicted results of the ReEDS model. Each color represents a sampled ReEDS scenario with solid lines corresponding to true output and dashed lines corresponding to reduced-form predictions.

4 Conclusion

Large-scale ensemble simulations are state-of-the-art in many application domains. Techniques allowing for the rapid display, understanding, and control of these simulations suites will become increasingly necessary as models escalate in complexity and computational needs swell. We believe our work here demonstrates a general application of an ensemble steering framework to energy system models. Our dataflow API allows users to explore and steer energy systems simulation ensembles by coupling multiple reduced-form energy models and interactive visualization, to provide a rich environment for engagement. Eventually, we anticipate that each of NREL's flagship energy-system simulation models will have a simplified approximation available for use in hosting stakeholder groups in design sessions and in simplifying inter-model linkages in the preliminary or envisioning phases of complex studies.

The availability of fast approximate models will greatly increase the agility of users interacting with complex models. We foresee much future work in designing approximate models in conjunction with full-scale models to facilitate stakeholder interactions, resulting in a superior user experience. The approximate models will let users home in on the regions of input parameter space of most value to the study at hand, quickly identify the extent of input parameters needing exploration, and intelligently specify the level of parameter sampling detail. Combined with customized visualization and an appropriate data workflow, this effort will collapse the time required to develop and analyze scenarios by providing previews of full model results and will likely be used in planning, quick-response, and quality assurance activities.

Acknowledgments

This work was supported by the U.S. Department of Energy. The research was performed using computational resources sponsored by the Department of Energy's Office of Energy Efficiency and Renewable Energy and located at the National Renewable Energy Laboratory. This work was initially supported by the Laboratory Directed Research and Development (LDRD) program at the National Renewable Energy Laboratory.

References

[1] N. Brunhart-Lupo, B. Bush, K. Gruchalla, and M. Rossol. Advanced Energy System Design (AESD): Technical Manual for the Records API. Technical Report TP-6A20-68924, National Renewable Energy Laboratory, October 2017.

[2] N. Brunhart-Lupo, B. W. Bush, K. Gruchalla, and S. Smith. Simulation exploration through immersive parallel planes. In *Immersive Analytics (IA), 2016 Workshop on*, pp. 19–24. IEEE, 2016.

[3] D. Coffey, C.-L. Lin, A. G. Erdman, and D. F. Keefe. Design by dragging: An interface for creative forward and inverse design with simulation ensembles. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2783–2791, December 2013. doi: 10.1109/TVCG.2013.147

[4] D. B. Crawley, C. O. Pedersen, L. K. Lawrie, and F. C. Winkelmann. Energyplus: Energy simulation program. *ASHRAE Journal*, 42:49–56, 2000.

[5] R. C. Dugan and T. E. McDermott. An open source platform for collaborating on smart grid research. In *2011 IEEE Power and Energy Society General Meeting*, pp. 1–7, July 2011. doi: 10.1109/PES.2011.6039829

[6] K. Eurek, W. Cole, D. Bielen, N. Blair, S. Cohen, B. Frew, J. Ho, V. Krishnan, T. Mai, B. Sigrin, et al. Regional energy deployment system (reeds) model documentation: Version 2016. Technical report, NREL (National Renewable Energy Laboratory (NREL), Golden, CO (United States)), 2016.

[7] I. Goodfellow, Y. Bengio, and A. Courville. *Deep learning*. MIT press, 2016.

[8] Google Developers. Protocol Buffers. https://developers.google.com/protocol-buffers/, July 2017.

[9] P. Haystack. Project Haystack. http://project-haystack.org/, 2017 July.

[10] Internet Engineering Task Force. RFC 6455 – The WebSocket Protocol. https://tools.ietf.org/html/rfc6455, 2017 July.

[11] G. James, D. Witten, T. Hastie, and R. Tibshirani. *An introduction to statistical learning*, vol. 112. Springer, 2013.

[12] V. Krishnan and W. Cole. Evaluating the value of high spatial resolution in national capacity expansion models using reeds. In *Power and Energy Society General Meeting (PESGM), 2016*, pp. 1–5. IEEE, 2016.

[13] J. D. Mulder, J. J. Van Wijk, and R. Van Liere. A survey of computational steering environments. *Future generation computer systems*, 15(1):119–129, 1999.

[14] National Centers for Environmental Protections Environmental Modeling Center. Short-range ensemble forecasting project. http://wwwt.emc.ncep.noaa.gov/mmb/SREF/SREF.html.

[15] S. Peterson, C. Peck, D. Stright, E. Newes, D. Inman, L. Vimmerstedt, S. Hsu, and B. Bush. Overview of the biomass scenario model. Technical report, National Renewable Energy Laboratory (NREL), Golden, CO., 2015.

[16] H. Ribičiċ, J. Waser, R. Gurbat, B. Sadransky, and M. E. Gröller. Sketching uncertainty into simulations. *IEEE Transactions on Visualization and Computer Graphics*, 18(12):2255–2264, December 2012. doi: 10.1109/TVCG.2012.261

[17] M. Sedlmair, C. Heinzl, S. Bruckner, H. Piringer, and T. Mller. Visual parameter space analysis: A conceptual framework. *IEEE Transactions on Visualization and Computer Graphics*, 20(12):2161–2170, Dec 2014. doi: 10.1109/TVCG.2014.2346321

[18] T. Simpkins, D. Cutler, K. Anderson, D. Olis, E. Elgqvist, M. Callahan, and A. Walker. Reopt: A platform for energy system integration and optimization. Technical Report CP-7A40-61783, 2014.

[19] L. Vimmerstedt, B. W. Bush, and S. O. Peterson. Dynamic modeling of learning in emerging energy industries: The example of advanced biofuels in the united states. In *The 33rd International Conference of the System Dynamics Society, Cambridge, Massachusetts, USA*, 2015.

[20] J. Waser, R. Fuchs, H. Ribičiċ, B. Schindler, G. Blöschl, and M. E. Gröller. World lines. *IEEE Transactions on Visualization and Computer Graphics*, 16(6):1458–1467, 2010.

[21] J. Waser, H. Ribičiċ, R. Fuchs, C. Hirsch, B. Schindler, G. Blöschl, and M. E. Gröller. Nodes on ropes: A comprehensive data and control flow for steering ensemble simulations. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):1872–1881, December 2011. doi: 10.1109/TVCG.2011.225