2014 ASHRAE/IBPSA-USA
Building Simulation Conference
Atlanta, GA
September 10-12, 2014

BUILDING SIMULATION MODELERS – ARE WE BIG DATA READY?

Jibonananda Sanyal and Joshua New
[sanyalj, newjr]@ornl.gov
Oak Ridge National Laboratory

ABSTRACT

Recent advances in computing and sensor technologies have pushed the amount of data we collect or generate to limits previously unheard of. This paper focuses on the big data challenges that building modeling experts may face in data collected from a large array of sensors, or generated from running a large number of building energy/performance simulations. A case study is presented that highlights the technical difficulties that were encountered and overcome in order to run 3.5 million EnergyPlus simulations on supercomputers and generating over 200 TBs of simulation output. While this is an extreme case, it led to the development of technologies and insights that will be beneficial to modelers in the immediate future. The paper discusses the different data management technologies and data transfer choices in order to contrast the advantages and disadvantages of employing each. The paper concludes with an example and elaboration of the tipping point where it becomes more expensive to store the output than re-running a set of simulations for a suficiently large prametric ensemble.

INTRODUCTION

“Big data” is now a common buzz term. Many institutions and corporations are preparing themselves for a massive deluge of anticipated data. A survey of 1,217 companies in nine countries revealed that 53% of the organizations had big data initiatives [Tata 2012] and were expecting the data insights to drive growth. Another survey revealed that only 4% of the organizations considered themselves good at using the data well to derive insights. In fact, there has been a recent explosion in the number of database technologies simply for storing the data [Aslett 2014]. Few organizations are prepared to adapt existing systems to integrate advantages of these quickly-evolving technologies. Over 2.5 exabytes (2.5 quintillion = $2.5x10^{18}$ bytes) are created each day [IBM 2012]. Furthermore, the rate is increasing so quickly that 90% of the data in the world today has been generated in the last 2 years [SINTEF 2013]. In such an environment, the capability of predictive analytics on big data is considered critical in many application domains. Fields such as marketing, finance, security, and social networks have been leading the research and developing state-of-the-art techniques in addressing the challenges that big data poses. The building sciences domain has largely been just slightly touched by the data deluge, but, with newer technologies being implemented into sensors and controls, building simulation engines, automated calibration, demand response, and building-to-grid integration, the discipline is set to undergo a sea change in how data is generated and used.

As energy-harvesting, peel-and-stick, system-on-a-chip sensors continue to decrease in price while increasing in the amount of sensing and on-board processing, the amount of data available from relatively few sensors could increase in the coming years. There are also efforts underway by major companies to manufacture cheap, building-level Non-Intrusive Load Monitoring (NILM) devices which use the current and voltage fluctuations from the breaker box to register the location and electrical draw from multiple electrical devices throughout the building. Advanced signature recognition algorithms allow one contactless physical device [Patel 2010] infer dozens of channels of data as part of the transition to a smarter electrical grid [Froehlich 2011]. In addition to the rollout of traditional smart meters, advanced system-on-a-chip sensors, NILMs, and inferential sensing are three technologies recently being deployed in buildings that piggyback on one another to allow a non-linear increase in the amount of data available on building performance.

As more data becomes available, it is anticipated that there will be a concomittant increase in software-informed building operation decisions to leverage the available data. This trend has been observed in other application domains, but faces additional challenges since buildings are largely one-off constructions in the field rather than built to a specific engineering standard. A Navigant Research report projects an increase of Building Automation Systems (BAS) from 2014 to 2021 with data sizes in North America expected to increase from 17 to 52 petabytes per year from 2015 to 2021 [Navigant 2014]. Specific equipment types use control algorithms or Fault Detection and Diagnostics (FDD) from sensor data to improve equipment performance or maintenance. Unfortunately, optimization of any specific piece of equipment can be at the expense of building-wide optimization or operated in a way that causes the building to expend energy in fighting itself (e.g. two HVAC units with one cooling and one heating). As data generation, communication, and processing decrease in cost, high-fidelity, simulation-based alternatives could help in optimization of operation and grid-interaction for commercial as well as residential buildings.

This paper focuses on a few diverse aspects involved with the management of sensor and simulation data to motivate the case for big data and its analysis in the building sciences community.

BIG DATA IN THE BUILDING SCIENCES

How big is big data in the building sciences? There are many ways to address this question. Often, the ovarall size of the data is used, but the size that constitutes big data is often defined as relative to the capabilities of current tools in a given domain. The more important question is “what is the purpose of the data?” Given a specific use case from such a question, any point where the management of the data poses a non-trivial challenge is considered a big data scenario.

Sub-minute resolution data from dozens of channels is becoming increasingly common and is expected to increase with the prevalence of non-intrusive load monitoring. Experts are running larger building simulation experiments and are faced with an increasingly complex data set to analyze and derive meaningful insight.

Scalability, analysis requirements, and adaptability data storage mechanisms is becoming very important. Additionally, unique behavior of different tools make it non-trivial for larger workflows. Practical experience becomes increasingly critical in selecting cost-effective strategies for big data management.

As data becomes large in size, network performance issues arise. Practical issues involving lag, bandwidth, and methods for transferring and synchronizing logical portions of the data become important.

The cornerstone of big data is its use for analytics; data is useless unless actionable information can be meaningfully derived from it. In the rest of this paper, we discuss challenges and opportunities of applications involving big data, sensor data from user facilities, and data generated from simulations.

BIG DATA APPLICATIONS

Simulation data handling and analysis are important in many practical uses in the building community including efforts such as DEnCity [Roth 2012] to build a national simulation database, effect of proposed policy changes on the building stock [Taylor 2013], prioritization of energy-saving technologies for investment [Farese 2013], optimized retrofits [DOE 2014], uncertainty quantification for energy savings likelihoods [Zhang 2013], and many other ongoing uses. Future uses for large-scale simulation studies could allow simulation engine approximation for dramatic speedup with allowable tradeoffs in accuracy, enhanced inverse simulation for any traditional/forward-simulation tool, surrogate model creation and parallelization for computational bottlenecks of existing simulation tools, and enhanced testing of simulation tools as development progresses.

There are many forward-looking applications for big data from both sensors and simulations. Automated calibration efforts rely on simulation of building variants in comparison with measured sensor data. There are already emerging examples of sensor/simulation hybrid techniques such as sensor-based Building Energy Modeling (sBEM) [Edwards 2011] which use direct sensor measurements from buildings in combination with machine learning techniques to answer questions or make predictions about building performance. Provenance is growing as an independent field for tracking and exploring the log and lineage of how data changes over time; it only recently began being used in buildings [Sanyal 2014, Castello 2014]. As energy dashboards, augmented reality, and other methods for making invisible energy use visible to building operators become more popular, there will likely be an increased demand on data handling, analysis, and intuitive/actionable visualization.

DATA FROM SENSORS

The Flexible Research Platformss at the Oak Ridge National Laboratory are currently collecting over 1,000 channels of data, most of which are sampled at 30-second resolution. These channels include temperature, relative humidity, and heat flux for the building envelope, flowmeters and wattnodes for energy consumption throughout appliances, and weather data. With these channels stored as 8-byte double-precision floating points, this amounts to over 1 billion data points per year, or 8 gigabytes/year. While this is significantly less than the 10s of terabytes to petabytes considered “big data” in supercomputing or similar fields, it is still more than enough to break most building-related applications. The “big” in “big data” is often defined relative to the maturity of the technology currently available to process the data such as database management tools and traditional data applications. As such, we propose that gigabytes constitutes big data for a single building’s measured properties since most simulation, building model, model calibration, or M&V applications cannot currently use this amount of data.

A major aspect of sensor data collection is effectively managing faults. Sensors will drift and will require periodic calibration. Sensors will fail and require replacement. Fault tolerance and mechanisms to automaticaly detect and correct such errors is an important requirement. In the facility described previously, a software system has been built that periodically calculates the standard-deviations for each channel in the collected sensor data. These values are used to generate a statistical range that is between 1 to 6 standard-deviations from the mean. By the three-sigma rule of statistics, a range of 3 standard deviations about the mean covers 99.7% of the range of data values possible, assuming a normal distribution. A script runs periodically checking the latest data values against the range detecting potential outliers which may indicate a fault, and sends out an email alert. Until a full year of data is collected, the ranges must be periodically recalculated to account for changing seasons.

Upon detection of a fault, it is possible to correct for the values. The team has developed tools that check and detect missing or bad values and provides the user with a battery of filtering [Castello 2012], statistical [Castello 2013], and machine learning [Smith 2013] algorithms that attempts to intelligently infer the missing or corrupt values. The accuracy of these algorithms depend on a variety of factors including the time range of the missing data.

DATA FROM SIMULATIONS

Simulation is a powerful tool to determine possible outcomes from a mathematical model for various input conditions. Even though simulation can be a very powerful tool, as with all computer models, it is still necessarily an approximation of reality. Various uncertainties also exist in the description of the input and in the analysis of output. Modelers use an ensemble of simulations with different combinations of input parameters to capture some of the uncertainties inherent in the modeling process. The mass availability of cheap computing power allows building modelers to run increasingly larger ensembles in relatively shorter time periods. The recent ease for utilization of cloud computing adds significant capabilities but complicates the simulation workflow and data communication methods, strengthening the case for pro-active big data readiness.

Simulation Input

The generation of inputs for an ensemble of simulations can become a daunting task if the number of simulations is large. There is a wide body of literature on the generation of an adequate number of samples that sufficiently sample and are representative of the range of inputs [Winer 1962]. The design of experiments is a well established field in statistcs. A full exploration of the field is beyond the scope of this paper, however, we present a few design paradigms that were relevant for the case study presented later in the paper:

a. Random sampling: Random samples are selected from each input’s range. This strategy does not assume any underlying distributions in the inputs and in theory, retains an equal probability of picking an point anywhere in the range. This method is scalable since it requires minimal computation to pick a sampling point.

b. Uniform sampling: Using a uniform sampling strategy asures equally spaced input samples.

c. Markov Order Sampling: Often used when the input space is large, this sampling varies a subset of *k* inputs (from a set of *n* inputs) between their minimum and maximum values at a time while holding all other variables at their default value. This sampling creates a maximal sensitivity analysis for combinations of *k* inputs, but a computationally prohibitive larger O(*n* choose *k*) number of samples as *k* increases.

d. Latin square and higher orders [Stein 1987]: This design provides a customizable sample size while attempting to evenly sample the multidimensional space. This method retains desirable statistical features but can be a challenge to compute for very large numbers of variables.

It is very important to choose the parameters, ranges, distributions, and sampling strategy appropriately for a given problem that needs to be answered. The statistical analysis of outputs is always dependant on the experimental design and relative independence of parameters. Non-independent parameters lead to interaction effects which canbecome complicated for higher order interactions.

A toolkit named DAKOTA [Giunta 2003] alleviates many of these challenges by providing algorithms for design, estimating uncertainty, and performing system analysis, and powers the Parametric Analysis Tool in OpenStudio with capabilities to run simulations on Amazon's cloud computing platform.

Simulation Output

Simulation output is almost always much larger in size than the input and consists of various output variables. It is usually not necessary to save the output from simulations for an extended period of time, however, the output must be stored for the time that is required to perform the analysis. Some data analysis algorithms parallelize well and can work on chunks of the output while others may require the entire ensemble in memory to derive meaningful conclusions.

For example, in EnergyPlus, the output is an *.eso file which can be processed with another program (readvars) to output various summaries of the data. As an ensemble becomes larger, the post-processing overhead can become a significant fraction. Conventional wisdom is to save the raw output for a period of time so that any type of summary can be generated if required in the future.

Post-processing is used to transform or draw summaries from raw simulation output. With a large ensemble, the raw output may take less disk, however, the computational requirement to regenerate all the summaries can be expensive. Instead, there may be a benefit in saving just the processed output. The key is to evaluate the trade-off of between recomputing and the storage and retreival overhead for pre-computed values.

BIG DATA MANAGEMENT

In this section, we cover many aspects of big data management that include existing or new techniques to handle obvious challenges as well as some that are not so apparent.

Generic Considerations

Perhaps the biggest motivator of big data strategies is the analysis requirement and the challenges posed in time-efficiency and resource-efficiency. The most effective strategy is almost always driven by the analysis requirement, however, some key approaches have helped the authors immensely.

First, one must get away from the notion of working with a single unit of data, be it a small collection of files or a single database. Large data tends to not be highly structured and tends to entropy, partly because of having to work in chunks. Big data storage could span multiple machines.

Second, a key strategy is balancing storage to computational requirements. This may be particularly challenging because one must access all non-obvious overheads. This may include number of files expected to be generated and how they will be physically arranged in storage to optimize for data access patterns.

Another overhead is the determination of the physical location on disk from the logical schema. If this mapping involves repeatedly accessing a small subsystem or a single module, it could easilty become the bottle neck in the system.

When using multiple-cores, determining the exploitable level of parallelism is important. All parallelly decomposed problems have a serial component. The overall speedup and efficiency is determined by the serial fraction given by Ahmdal's law.

Third, it helps to determine the typical types of analysis performed and design the system with a moderate amount of extra capacity. There is likely to be a few analysis scenarios that will run very slowly and designing the system for such cases will drive up the cost significantly.

Fourth, always design for errors to happen and use bottom-up scalability techniques to recover from an error when possible. Sometimes, an error can propagate across the system and the means of catching such occurences in a meaningful way can greatly improve the resiliency of the data-store.

Practical Data Transfer Methods

Big data movement is very expensive. The authors have experience where it took just 68 minutes to generate and write 45 TB of data to disk but took 10 days to move to a more permanent storage location! It is highly desirable to keep big data where it needs to be and move it minimally. Since big data tends to be in pieces, logical partions of the data can almost always be treated as smaller units of information and can be moved or rearranged when required. It is very helpful to have an overall schema that is flexible and allows such changes.

There are several tools that allow efficient movement of large data across networks. These include the standard File Transfer Protocol (*ftp*) or Secure Shell (*ssh*) based protocols (such as *scp* and *rsync* [Tridgell 1996]) some of which open multiple network connections to exploit parallelism in the transfer as well as data compression on the fly. More efficient tools such as *bbcp* [Hanushevsky 2001] or *GridFTP* [Allcock 2005] are particularly designed for very large data transfers. Other considerations in data movement include the available bandwidth and network lag in movement.

Database Storage Solutions

Sensor data as well as building simulation output have been traditionally stored in comma separated value files with data retreival and subsequent analysis speeds slower than database engines or binary data formats.

Database technologies are often discussed in terms of Atomicity, Consistency, Isolation, and Durability (ACID) compliance [Gray 1981]. Atomicity means a transaction is all or nothing. Consistency means any transaction leaves the database in a valid state. Isolation ensures that concurrent transactions give the same results as if the transactions were executed serially. Durability requires that a committed transaction remains so regardless of crashes or other errors. While all these properties are desirable, the scale of big data presents significant logistical challenges and costs to retain these properties. As a result, it is common for many big data storage solutions to bend or break compliance with some or all of these properties.

Traditional databases store data as rows of logical records. To save space, most database engines allow row compression, which adds a decompression overhead in retrieval. Several columnar databases, such as the Infobright database engine, use a column based storage format instead of storing rows of data. These engines promise better compression since each column is of the same the data type and good compression can be achieved. This is usually true for most data types except floats/doubles. Building data are typically floating point numbers which the authors have found to compress comparably to the row based engines.

The authors used 15-minute EnergyPlus output consisting 35,040 records and comprised of 96 variables in comma separated value files for testing compression. These files are about 35 MB in size which compress to about 7-8 MB indicating a 20-22% compression rate. Two hundred output CSV files were inserted into a row-compression enabled MySQL database resulting in 7 million records. The observed average compression was 10.27 MB. The database was further compacted to a read-only version which brought down the avaerage down to 6.8 MB.

Traditional databases offers some advantages in storing channels of time series data. It is easier to calculate summaries of channels but the data insertion itself can become an expensive process. For example, in MySQL, using the default InnoDB engine, inserts take increasingly longer as the data size grows. Part of the reason is the increased overhead in maintaining indices and foreign keys. In contrast, the MyISAM engine is always much faster than InnoDB and achieves the performance by not enforcing foreign keys and therefore, not being fully ACID compliant. Table 1 provides an illustrative comparison of the two engines.

In addition to ACID compliance, structured query language (SQL) ALTER TABLE commands make a copy of the entire table first before making any changes to the table structure which can be quite expensive for large tables. Using table partitioning in the schema design helps to speed up the ALTER TABLE command as well as the overall performance. The authors experienced a runtime of 8 hours and 10 minutes on a 386 GB table consisiting of 1,041,893,313 records stored in 12 partitions on a typical quad-core desktop computer with 4GB RAM. The same billion row table takes about a week on the ALTER TABLE command when unpartitioned. Although logical partitioning helps, it is still better to minimize any schema changes to large tables.

Table 1: Comparison of MyISAM and InnoDB engines.

MyISAM	InnoDB
No ACID	ACID compliant
No foreign keys	Allows foreign keys
Fast bulk insert – 0.71 s average	Slower – 2.3 s average
Better compression – 10.27 MB average – 6.003 MB read-only	Poorer compression – 15.4 MB
2^{32} maximum rows	
Error recovery from logs	Rebuilds unflushed indexes
Table-level locks	Row-level locks

Hadoop data storage removes the ACID compliance and leaves the schema description up to the user, which is why these are often called the schema-less engines. Hadoop emerged from Hadoop Distributed File System (HDFS) as a resilient form of data store where key-value pair associations are used to create meaningful representation of the data. Although Hadoop is being used increasingly for various numerical applications, its most versatile use is still in text and information mining purposes. Hadoop also offers an eco-system of tools for various analysis tasks. Mahaout is one such application that exposes machine learning algorithms for use.

In the context of EnergyPlus output, an additional translation layer is necessary that converts the date-time expressions to either separate fields or forces to the date-time fields of the data store. In particular, EnergyPlus outputs the 24th hour in the 24:00:00 format which must be rolled over to 00:00:00 on the next day to insert into a database. This requires line by line processing of the output file and thus the efficiency of bulk import functionality of databases is lost.

Access control and security

The sensor data collected is often sensitive and access restrictions must be placed on its use. Gatekeeping for large data, especially across multiple machines, is challenging. Multiple machines occupy more physical space with implications on physical security.

Backups

A less obvious challenge with big data are backups. It may becomes cost prohibitive to backup all the data. Unlike small units of data which can be copied or synchronized to another machine, big data across multiple machines requires elaborate backup plans. Tape is still a cost-effective, long-term backup mechanism for both full and incremental backups.

Table 2: Runtime and disk write time of EnergyPlus simulations for variations of DOE's commercial reference buildings on the *Titan* supercomputer.

Num of Processors	Time (mm:ss)	Data Size	Number of E+ simulations
16	18:14	5GB	64
32	18:19	11GB	128
64	18:34	22GB	256
128	18:22	44GB	512
256	20:30	88GB	1,024
512	20:43	176GB	2,048
1,024	21:03	351GB	4,096
2,048	21:11	703GB	8,192
4,096	20:00	1.4TB	16,384
8,192	26:14	2.8TB	32,768
16,384	26:11	5.6TB	65,536
65,536	44:52	23TB	262,144
131,072	68:08	45TB	524,288

Provenance

Sensor data and simulation data are manipulated during user experiments/analysis. Data undergoes creation, trnasformation, modification, and even deletion. It participates with various other units of information to create more information. Often, it is highly desirable to know the lineage, or provenance, of the data.

Mechanisms to track and trace the provenance of data become essential as data size grows. The team have created a software sytem that transparently allows the tracking of the use of sensor data in various user experiments [Zachary 2014, Castello 2014].

Workflow tools

Working with big data is almost always a multi-step process and involves the management of shifting bottlenecks. It is critical to design automated workflow tools to help in working with big data. Often these are scripts that automate large batch processes. Knowledge of parsing and scripting helps in these automation tasks.

CASE STUDY: RUNNING A LARGE PARAMETRIC ENSEMBLE

A large parametric experiment was conducted for calibration purposes that ran about 3.5 million EnergyPlus simulations generating over 200 TB of data on different computing resources. A total of four types of residential and commercial buildings were simulated:

a. Residential: Two heavily instrumented residential building having more than 150 sensor channels each were sampled using Uniform sampling, Markov Orders 1 and 2, and Latin Hypercube strategies and totaled about 500,000 individual simulations.

b. Commercial: ~1 million simulations each of medium office, warehouse, and stand-alone retail reference buildings of three vintages across 16 ASHRAE climate zones. A single job for a subset of these 3 million simulations is shown in Table 2.

Several supercomputers were used to run such a large number of simulations. Systems include Nautilus, a 1024 core shared memory supercomputer, Frost, a 2048 core cluster, and Titan, which is a 299,008 core distributed memory supercomputer.

In this case study, 96 EnergyPlus outputs were chosen in the residential and commercial buildings which closely corresponded either to sensor data from the real residential building or to sensor data we believe to be most likely available in commercial buildings. This data was collected at 15-minute timesteps for each simulation and typically resulted in 10-90MB/simulation.

A central challenge in running EnergyPlus on supercomputers was to optimize the input and output so that the overall system performance would not degrade. This was achieved through four strategies: grouping and packing up (via tar) the input *.idf files to minimize number of concurrent reads, streamlining and customizing the EnergyPlus workflow, running the simulations from a memory mounted file system local to each supercomputer node (via tmpfs [Snyder 1990] and

RAMDisk [Flouris 1999]), and packing up all output (via gzip) on each node to one compressed file [Sanyal 2014]. Use of memory mounted file systems was a critical breakthrough in alleviating the performance limitations. In such a system, a path is used for file reading and writing, but is actually operating from random access memory (RAM) rather than disk; this alone lead to a performance improvement of over 20x for file I/O.

The conventional approach in scaling up code to run in parallel on supercomputers is to double the number of processors and either observe improvement in runtime (strong scaling), or to double the problem size also and observe any change in execution time (weak scaling). Table 2 illustrates the runtimes and the number of simulations executed in our weak scaling scenario.

The last row in the table illustrates that it took 68 minutes to run 524,288 simulations and write 45 TB to disk. Each processor in the ensemble ran 4 EnergyPlus simulations back to back which executes in under 20 minutes. With all processors running in tandem, all 64 simulations on each node complete in 20 minutes. This means that 48 minutes was spent in reading and writing to storage! It may be argued that any further analysis on the output will require a comparable read time. Since a bulk of the time spent is in writing to disk, we could potentially re-run the simulation and analyze the data while still in main memory (*in-situ*) and would require only a small fraction of the 48 extra minutes to write just the relevant analysis results to disk.

This tipping point is different for different simulation I/O and the computer system employed, but it has been demonstrated that it can be cheaper to re-run the analysis than store the data. This was demonstrated on a supercomputer, but may be equally applicable when running simulation data on a laptop and outputting large amounts of high-fidelity data to disk.

Re-running the simulation, however, does bring us back to the consideration of optimizing for what we want to analyze and the computational requirements for the analysis. Running large simulations with *in-situ* processing might be less expensive than storing and re-reading all the output, but can quickly add up if we have to repeat the simulations a few times. Carefully designing the analysis can mitigate such scenarios.

CONCLUSION

The paper presented various big data management challenges that the authors faced when running large EnergyPlus ensemble simulations and storing the output. Practical considerations for effectively managing large amounts of sensor data and simulation output, such as considerations of data compression, database technologies, data movement, and analysis driven hardware considerations were presented along with observed quantitative metrics for comparison between systems.

It is hoped that the practical experience presented in the paper will be beneficial to the building sciences community as cheap computing power and the availability of fine-resolution multi-channel sensor data becomes commonplace.

ACKNOWLEDGMENTS

This work was funded by field work proposal CEBT105 under the Department of Energy Building Technology Activity Number BT0201000. We would like to thank Amir Roth for his support and review of this project. This research used resources of the OLCF at the Oak Ridge National Laboratory, which is managed by UT-Battelle, LLC, for the U.S. Dept. of Energy under contract DE-AC05-00OR22725. The United States Government retains and the publisher, by accepting the article for publication, acknowledges that the United States Government retains a non-exclusive, paid-up, irrevocable, world-wide license to publish or reproduce the published form of this manuscript, or allow others to do so, for United States Government purposes.

REFERENCES

Allcock, William, Bresnahan, John, Kettimuthu, Rajkumar, Link, Michael, Dumitrescu, Catalin, Raicu, Ioan and Foster, Ian. (2005). “The Globus Striped GridFTP Framework and Server.” In Proceedings of the 2005 ACM/IEEE conference on Supercomputing (SC '05). Washington, DC, USA.

Aslett, Matthew, 451 Research. (2014). “Database Landscape Map.”

Castello, Charles C. and New, Joshua R. (2012). "Autonomous Correction of Sensor Data Applied to Building Technologies Utilizing Statistical Processing Methods." In Proceedings of 2nd Energy Informatics Conf., Atlanta, GA, Oct. 2012.

Castello, Charles C., New, Joshua R., and Smith, Matt K. (2013). "Autonomous Correction of Sensor Data Applied to Building Technologies Using Filtering Methods." In Proceedings of the IEEE Global Conference on Signal and Information Processing (GlobalSIP), Austin, TX, December 3-5, 2013.

Castello, Charles C., Sanyal, Jibonananda, Rossiter, Jeffrey S., Hensley, Zachary P., and New, Joshua R. (2014). "Sensor Data Management, Validation, Correction, and Provenance for Building Technologies." Proceedings of the ASHRAE Annual Conference and ASHRAE Transactions 2014, Seattle, WA, June 28-July 2, 2014.

Department of Energy (2014). "Building Energy Software Tools Directory's Whole Building Analysis: Retrofit Analysis."

Edwards, Richard E., New, Joshua R., and Parker, Lynne E. (2011). "Sensor-based Building Energy Modeling." ORNL internal report ORNL/TM-2011/328, September 2011, 79 pages.

Flouris, Michail D. and Markatos, Evangelos P.. "The network RamDisk: Using remote memory on heterogeneous NOWs." Cluster Computing 2, no. 4 (1999): 281-293.

Froehlich, Jon, Larson, Eric, Gupta, Sidhant, Cohn, Gabe, Reynolds, Matthew S. and Patel, Shwetak N. (2011). "Disaggregated End-Use Energy Sensing for the Smart Grid." In Proceedings of the IEEE Pervasive Computing, Special Issue on Smart Energy Systems, March 2011.

Giunta, Anthony A., Wojtkiewicz, Steven F., and Eldred, Michael S. "Overview of Modern Design of Experiments Methods for Computational Simulations." In American Institute of Aeronautics and Astronautics, 2003.

Gray, Jim. (1981). "The transaction concepts: Virtues and limitations." In Proceedings of the International Conference on Very Large Data Bases, pages 144-154, 1981.

Hanushevsky, Andrew, Trunov, Artem, and Cottrell Les. (2001) "Peer-to-Peer Computing for Secure High Performance Data Copying." Proceedings of the 2001 Int. Conf. on Computing in High Energy and Nuclear Physics (CHEP 2001), Beijng.

IBM (2013). "What is big data?" Available: http://www-01.ibm.com/software/data/bigdata/ what-is-big data.html

Patel, Shwetak N., Gupta, Sidhant, and Reynolds, Matthew S. (2010). "The Design and Evaluation of an End-User-Deployable, Whole House, Contactless Power Consumption Sensor." In Proceedings of CHI2010: Domestic Life, Atlanta, GA, April 10-15, 2010.

Farese, Philip, Gelman, Rachel, and Hendron, Robert. (2013). "A Tool to Prioritize Energy Efficiency Investments." Technical Report 54799, National Renewable Energy Laboratory (NREL).

Navigant Research (2014). "Intelligent Buildings and Big Data."

Roth, Amir, Brook, Martha, Hale, Elaine T., Ball, Brian L., and Long, Nicholas (2012). In Proceedings of the ACEEE Summer Study on Energy Efficiency in Buildings, 2012.

SINTEF (2013). "Big Data, for better or worse: 90% of world's data generated over last two years." ScienceDaily. ScienceDaily, May 22, 2013.

Sanyal, Jibonananda, New, Joshua R., Edwards, Richard E., and Parker, Lynne E. (2014). "Calibrating Building Energy Models Using Supercomputer Trained Machine Learning Agents." Journal Concurrency and Computation: Practice and Experience, March, 2014

Smith, Matt K., Castello, Charles C., and New, Joshua R. (2013). "Machine Learning Techniques Applied to Sensor Data Correction in Building Technologies." In Proceedings of the IEEE 12th International Conference on Machine Learning and Applications (ICMLA13), Miami, FL, December 4-7, 2013.

Snyder, Peter. (1990). "tmpfs: A virtual memory file system." In Proceedings of the Autumn 1990 EUUG Conference, pp. 241-248. 1990.

Stein, Michael. 1987. "Large sample properties of simulations using latin hypercube sampling." Technometrics 29, 2 (May 1987), 143-151.

TATA Consultancy Services (2012). "Majority of the Companies have Big Data Initiatives." Available:http://sites.tcs.com/big-data-study/big-data-study-key-findings/.

Taylor, R., Casey, P., Hendricken, L., Otto, K., Sisson, W., Gurian, P., Wen, J. (2013). "The Simulation of Long Term Trends in Building Energy Consumption Due to the Impact of Market-Based Policies to Encourage Adoption of Energy Conservation Measures." Proceedings of the 11th REHVA World Congress and the 8th International Conference on Indoor Air Quality, Ventilation and Energy Conservation in Buildings *CLIMA 2013*, Prague, Czech Republic, June 16-19, 2013.

Tridgell, Andrew and Mackerras, Paul (1996). "The rsync algorithm". Technical Report TR-CS-96-05, The Australian National University, June 1996.

Winer, B. J. (1962). Statistical principles in experimental design.

Zachary Hensley, Jibonananda Sanyal, and Joshua New. 2014. Provenance in sensor data management. Commun. ACM 57, 2 (February 2014), 55-62.

Zhang, Yuna, O'Neil, Zheng, Wagner, Timothy, and Augenbroe, Godfried. (2013). "An inverse model with uncertainty quantification to estimate the energy performance of an office building." In Proceedings of 13th International Building Performance Simulation Association Conference, 2013.