Sustainable Building Design: A Challenge at the Intersection of Machine Learning and Design Optimization

Siamak Safarzadegan Gilan and **Bistra Dilkina**
School of Computational Science and Engineering, College of Computing
Georgia Institute of Technology, Atlanta, GA
email:{safarzadegan, bdilkina6}@gatech.edu

Abstract

Residential and commercial buildings are responsible for about 40% of primary energy consumption in the United States, hence improving their energy efficiency could have important sustainability benefits. The design of a building has tremendous effect on its energy profile, and recently there has been an increased interest in developing optimization methods that support the design of high performance buildings. Previous approaches are either based on simulation optimization or on training an accurate predictive model that is queried during the optimization. We propose a method that more tightly integrates the machine learning and optimization components, by employing active learning during optimization. In particular, we use a Gaussian Process (GP) model for the prediction and active learning and multi-objective genetic algorithm NSGA-II for the optimization. We develop a comprehensive and publicly available benchmark for building design optimization. We evaluate 5 machine learning approaches on our dataset, and show that the GP model is competitive, in addition to being well-suited for the active learning setting. We compare our optimization approach against the 2-stage approach and simulation optimization. Our results show that our approach produces solutions at the Pareto frontier compared to the other two approaches, while using only a fraction of the simulations and time.

Introduction

Residential and commercial buildings are responsible for about 40% of primary energy consumption in the United States, and reducing this consumption will play an important role in taking practical steps toward a sustainable society and the reduction in usage of the corresponding non-renewable resources (Pérez-Lombard, Ortiz, and Pout 2008; Yu et al. 2010). Enhancement of the energy efficiency of buildings is certainly a key task contributing to an immediate reduction of energy consumption and carbon emissions (Zhang et al. 2013). In particular, the design of the building has a major impact on its energy footprint. There are several factors affecting the building energy efficiency, such as floor plan design, building orientation, construction materials, daylight and solar control measures, and activity-related parameters. These have to be considered during the conceptual stage of a project, since any attempts to improve energy efficiency in the later stages could be more costly or impossible.

Several software packages have been developed to accurately simulate building energy consumption (e.g. EnergyPlus, DOE-2, and Green Building Studio) and to support decision makers at the design phase to produce more energy efficient structures. However, in reality, simulation tools are mainly used to validate the performances of the final design of a building rather than exploring multiple design possibilities (Flager and Haymaker 2009). To achieve optimal energy performance and reduce manual efforts of exploring design options, several optimization techniques, such as Genetic Algorithms (Caldas 2008; Lin and Gerber 2014; Wang, Zmeureanu, and Rivard 2005; Tuhus-Dubrow and Krarti 2010; Magnier and Haghighat 2010), have been proposed. Most computational tools for building design optimization support workflows, which directly combine genetic algorithms with accurate building simulation software (Caldas 2008; Lin and Gerber 2014; Tuhus-Dubrow and Krarti 2010; Wang, Zmeureanu, and Rivard 2005). The major disadvantage of such approaches is that simulations are computationally expensive. One way to address this is to replace the expensive simulations with surrogate fast methods for estimating the energy consumption of the proposed design. A couple of recent papers have investigated the use of genetic algorithms coupled with a predictive model of energy consumption. The predictive model is pre-trained on simulated data, and is used to quickly evaluate candidate solutions without directly interfacing the simulation during search. (Magnier and Haghighat 2010) developed a feed-forward neural network model to energy consumption of a building, and use it in the context of an NSGA-II (Deb et al. 2002) genetic algorithm to optimize with respect to energy consumption and comfort objectives. (Asadi et al. 2014) followed a similar approach based on a neural network and a genetic algorithm to optimize decisions about retrofitting a building. Although these approaches already improve upon a simulation-optimization strategy, the use of neural networks as the predictive model demands large enough training data for accurate prediction over many design variables.

Exploiting data-driven models to predict the energy consumption of buildings from their design variables has drawn

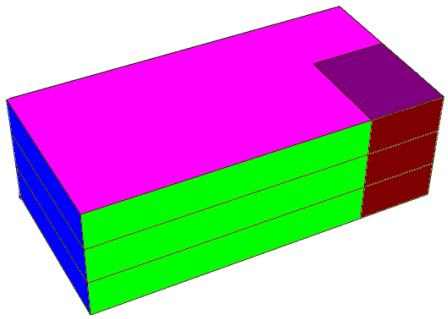

Figure 1: Schematic view of the building geometry of the benchmark model

Table 1: Parameter list and values for the developed benchmark building model

Parameter	Values
Orientation	{0,45,90,135,180,225,270,315}
Heating Set Point	{21,22,23}
Cooling Set Point	{23,24,25}
Building Fabric: Floors	{BF1,BF2,BF3,BF4,BF5}
Building Fabric: Roof	{BF1,BF2,BF3,BF4,BF5}
Building Fabric: Walls*	{BF1,BF2,BF3,BF4,BF5}
Glazing Ratio: Walls*	{25,50,75}
Glazing Coating: Walls*	{Non-Reflective, Reflective}

*Assigned to the four exterior walls of the model independently.

the attention of many researchers in recent years. Many variations of supervised regression approaches have been utilized for this purpose (for a survey see (Zhao and Magouls 2012)): neural networks (Neto and Fiorelli 2008; Magnier and Haghighat 2010; Yang, Rivard, and Zmeureanu 2005; Zemella et al. 2011); support vector machines (SVM) (Dong, Cao, and Lee 2005; Li et al. 2009), decision trees(Yu et al. 2010; Tso and Yau 2007), random forests (Tsanas and Xifara 2012), Gaussian Processes (Zhang et al. 2013; Kolter and Ferreira 2011), and others.

We propose a method that more tightly integrates the machine learning and optimization components, by employing active learning during optimization. To achieve computation savings, we train our predictive model on a small set of simulated samples. However during the optimization, additional configurations are simulated and added to the training data. In particular, we use a Gaussian Process model for the prediction, and multi-objective genetic algorithm NSGA-II for the optimization of building energy and cost. We develop a comprehensive and publicly available benchmark for building design optimization, representing a realistic commercial building with many design parameters along with corresponding energy consumption and cost. We evaluate 5 machine learning approaches that have been proposed for building energy prediction on our dataset, and show that the Gaussian Process model is a competitive predictive approach to building energy consumption, in addition to being well-suited for the active learning setting. We compare our active learning optimization approach against the previously proposed 2-stage approach of (passive) learning and optimization, as well as against a baseline where each configuration explored during search is simulated. Our results show that our approach produces solution at the Pareto frontier, compared to the solutions obtained with the other two approaches, while using only a fraction of the simulations.

Dataset Description: Building Design Model

We aim to create a benchmark building model with considerable number of decision variables to match closely near real-case design scenarios. Our benchmark is based on an open-plan side-lit building plan (OD), shown in Figure 1, developed as one of four UK office buildings archetypal models analyzed in (Korolija et al. 2013). The major drawback of the original model is that many of the design parameters were applied uniformly to the whole building. In practice, designers often assign different building materials, window types, etc to different building parts in order to meet project requirements. In our benchmark, different parameters were assigned to each of the four individual exterior faces. The parameters and their associated values are presented in Table 1. All the other parameters in the model has been fixed to default values. For the meanings and the details of values, please refer to the description in (Korolija et al. 2013). Our benchmark model has a total of 2,916,000,000 possible designs, forming a very large search space.

We considered two performance measures for a design: energy and cost. Energy consumption was simulated using the EnergyPlus toolbox (Crawley et al. 2001). In addition to energy, the cost for material and labor associated with each design option is also an important factor to be considered during the design development cycles. The unit costs for selected parameters were identified based on a widely used construction cost database (http://www.rsmeansonline.com/). While cost details were not available for some of the materials, approximated unit costs were obtained by interpolation. To evaluate different predictive models for energy consumption in our benchmark, we generate two datasets of size 5000 (train) and 1000 (test) samples using Latin hypercube sampling (LHS) (Iman 2008). To this aim, we used the jEPlus parametric study toolbox (Zhang 2012) for accelerating the simulation process and generating the input files for EnergyPlus. Further details, the models, scripts and precompiled datasets are available from http://www.cc.gatech.edu/~bdilkina/#Projects.

Energy Consumption Prediction

We first use our benchmark dataset to evaluate several models for predicting energy consumption using the building design parameters as input features: ensemble methods of random forest (RF) (Breiman 2001) and least squares boosting (LSBoost) (Hastie et al. 2009); neural network (NN) (Hastie et al. 2009); support vector regression (SVR) (Vapnik, Golowich, and Smola 1997); and Gaussian process (GP) (Rasmussen 2004). The details of the methods and their parameters are listed in Table 2. Moreover, in the experiments, we used MATLAB's toolboxes for NN, RF, and LSBoost,

Table 2: ML approaches and their parameters

Model	Learning Parameters
NN	Multilayer feed-forward structure with one or two hidden layers of {10,20,30,40,50} neurons, Levenberg-Marquardt back propagation algorithm, hyperbolic tangent sigmoid function and linear transfer function for hidden and outer layers.
RF	Number of trees of {10,20,30,40,50}.
LSBoost	Weak learner of tree, Number of trees of {10,20,30,40,50}.
SVR	Regularization $C \in [2^{-5}, 2^{5}]$, RBF kernel with $\gamma \in [2^{-15}, 2^{3}]$, $\epsilon \in [0.01, 1]$.
GP	Covariance function: Matern kernel with $\nu = \{3/2, 5/2\}$ and automatic relevance determination, Gaussian likelihood, exact inference.

Table 3: Accuracy of Energy Prediction

Model	Training		Testing	
	RMSE	R^2	RMSE	R^2
RF	5809	0.9731	5615	0.9718
LSBoost	4257	0.9841	4364	0.9822
NN	1405	0.9982	1306	0.9984
SVR	5791	0.9704	5746	0.9690
GP	1406	0.9983	1331	0.9984

the LIBSVM toolbox (Chang and Lin 2011) for SVR, and the GPML toolbox (Rasmussen and Nickisch 2014) for GP. The performance was evaluated based on root mean square error (RMSE) and correlation coefficient (R^2).

Input variables were normalized to the range $[0, 1]$ for better training performance. The model selection was done with 10-fold cross validation on the 5000 samples, and performance was measured on the additional 1000 samples, reported in Table 3. The Neural Network performs slightly better than GP on RMSE, and they both have the best correlation coefficient. While GP is competitive on performance with big enough training set, it also has a fast learning curve as can be seen in Fig. 2. The learning curve is obtained by training the GP model based on different number of samples and then measuring its performance on 1000 independent testing samples.

Building Design Optimization Coupled with Active Learning

In order to facilitate fast building design optimization that does not require many simulations, we propose to couple the optimization with active learning. First, the predictive model is pre-trained with a small number of simulated samples. Then, during the optimization, new configurations are evaluated using the predictive model, and a small fraction of them are selected, their energy consumption is evaluated using the simulation model (labeled), and they are added to

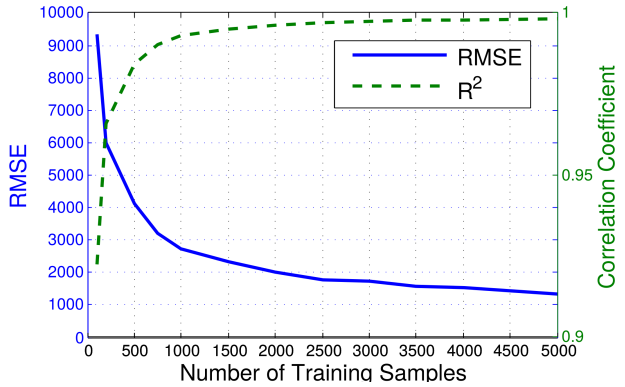

Figure 2: GP learning curve: Number of training samples vs. Accuracy (best in color)

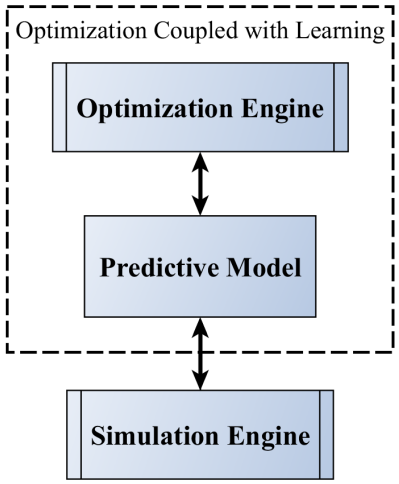

Figure 3: Schematic illustration of optimization coupled with learning.

the training dataset. The schematic process of optimization coupled with learning is depicted in Figure 3.

In order to select active learning queries in an informed way, we will use a Gaussian Process model as our predictive approach.

Active learning with Gaussian Processes

Gaussian process (GP), as a stochastic process, provides a powerful tool for probabilistic inference on a infinite-dimensional space of functions, $\{f(x)\}$, where any finite collection of random variables have a joint multi-variate Gaussian distribution, and the random variables in GP represent the value of $f(x)$ at x. Generally, GP is classified as a non-parametric supervised learning method defining a prior on the space of functions, which could be written as $f(x) \sim \mathcal{GP}(m(x), k(x, x'))$, and is defined by its mean $m(x)$ and covariance function $k(x, x')$. Common covariance functions are linear, squared exponential, γ-exponential, and Matérn with automatic relevance determination (ARD). Matérn covariance functions with ARD (Abramowitz, Stegun, and oth-

ers 1965), where the degree of Matérn covariance function is $d = \{3, 5\}$, simplifies to:

$$k_d(x, x') = g_d(\sqrt{d}r)exp(-\sqrt{d}r) \quad (1)$$

where $g_3(z) = 1 + z$, $g_5(z) = 1 + z + z^2/3$, and r defines the distance between the two points as follows: $r = \sqrt{(x - x')' * P * (x, x')}$, where P is a diagonal matrix with ARD parameters $\{\ell_1, \ldots, \ell_D\}$ and D is the data input dimension.

The main concept behind active learning is to query points which are the most informative to the learning algorithm. (Seo et al. 2000) studied a method, proposed by (Cohn 1994), for active learning with GPs based on the minimization of the generalization error. The goal is to choose the query $\hat{x}$ that when added minimizes the overall variance of the estimator. The variance on a reference data point ξ after adding a query candidate $\hat{x}$ to the training dataset is defined by $\sigma^2_{\hat{y}(\xi)}(\hat{x})$. Therefore, the change in variance is:

$$\begin{aligned} \triangle\sigma^2_{\hat{y}(\xi)}(\hat{x}, X_{tr}) &= \sigma^2_{\hat{y}(\xi)} - \sigma^2_{\hat{y}(\xi)}(\hat{x}) \\ &= \frac{(KC^{-1}m - k(\hat{x}, \xi))^2}{(k(\tilde{x}, \tilde{x}) - m^T C^{-1} m)} \end{aligned} \quad (2)$$

In the above equation, C is the covariance matrix of the training dataset $X_{tr} = \{x_i, i = 1, ..., n\}$, $K = [k(x_1, \xi), \ldots, k(x_n, \xi)]$ is the vector of covariances between the training data and a reference data point ξ. The overall impact of adding $\hat{x}$ is measured as the average $\triangle\sigma^2_{\hat{y}(\xi)}(\hat{x}, X_{tr})$ over a set of reference points, as a surrogate for the expectation of the overall variance. High value means high reduction in variance, and so such candidate points should be preferred for active labeling.

Algorithm 1: Optimization coupled with active learning using NSGA-II and GP.

input : $\{X_{tr}, Y_{tr}\} = \{(x_i, y_i), i = 1, ..., n\}$: samples used for pre-training GP
n_p = population size
n_{mg} = maximum number of generations
δ = threshold for active query selection

Initialize P_0 population of size n_p
$X_{tr} \leftarrow X_{tr} \bigcup$ `ActiveQuery` (P_0, X_{tr})
Retrain GP using X_{tr}
Evaluate P_0 using GP
for $p \leftarrow 1$ **to** n_{mg} **do**
  Generate new population P_p of size n_p
  $X_{tr} \leftarrow X_{tr} \bigcup$ `ActiveQuery` (P_p, X_{tr})
  Retrain GP using X_{tr}
  Evaluate P_p using GP
  Combine P_p and P_{p-1}
  Extract new population using fast non-dominated sorting
end for

Function Q = `ActiveQuery` (P, X_{tr})**:**
  $Q \leftarrow \emptyset$
  for $\hat{x} \in P$ **do**
    $R \leftarrow P - \{\hat{x}\}$
    for $\xi \in R$ **do**
      Compute $\triangle\sigma^2_{\hat{y}(\xi)}(\hat{x}, X_{tr})$
    end for
    if $\mathbb{AVG}_{\xi \in R}[\triangle\sigma^2_{\hat{y}(\xi)}(\hat{x}, X_{tr})] > \delta$ **then**
      $Q \leftarrow Q \bigcup \hat{x}$
    end if
  end for

Optimization Coupled with Learning

We utilized the non-dominated sorting genetic algorithm (NSGA-II) (Deb et al. 2002) for the multi-objective optimization. The algorithm proceeds in generations, where a new population of candidate configurations is produced at each step. We evaluate the effect of adding each candidate to the training dataset by Eq. 2 using the rest of the new candidate solutions as reference points. If a candidate solution's average variance reduction exceeds a specified threshold δ, then this configuration is simulated and added to the training set. At each population, no more than 50% of all candidates are selected for active learning. After adding more than 25 new points to the training dataset, we re-train the GP. After updating the GP model, the new candidate solutions are evaluated for energy using the predictive model. Finally, the population is updated. The pseudocode of optimization coupled with learning is given in Algorithm 1.

Results

We conducted a comparative study between simulation-based optimization (Opt-Sim), optimization using predictive models (Opt-PL), and our method, active learning for optimization (Opt-AL). After studying the learning curve for GP, we set the training size for Opt-PL to 2000 as a good compromise between quality and time. For optimization using NSGA-II, the maximum number of generations was 40 and the population size was set to 50, which implies that for Opt-Sim there could be upto 2000 simulations during the optimization procedure. Having in mind that the goal of this paper is to propose a faster method for building energy optimization, for Opt-AL we start with just 500 training samples, we limit the total number of additional queries to 1000 (at most half of each generation), and set the threshold δ for active learning to $0.2 * 10^4$. The Pareto curve of Energy vs Cost of the final non-dominated solutions obtained using all three approaches is shown in Figure 4. The number of simulations and run-times have been summarized in the Table 4. Our results indicate that our approach Opt-AL has produced a very competitive Pareto curve in comparison with the other two approaches (in terms of ground truth energy and cost), while taking less than one third of their running time. It can be also observed that the discrepancy between the predicted energy and the simulated (ground truth) energy for the Pareto solutions is much smaller for Opt-AL than for Opt-PL. Evaluating the Pareto solutions obtained by Opt-PL trained only on the 500 initial samples used in Opt-AL, we obtain inferior solutions and much greater discrepancy be-

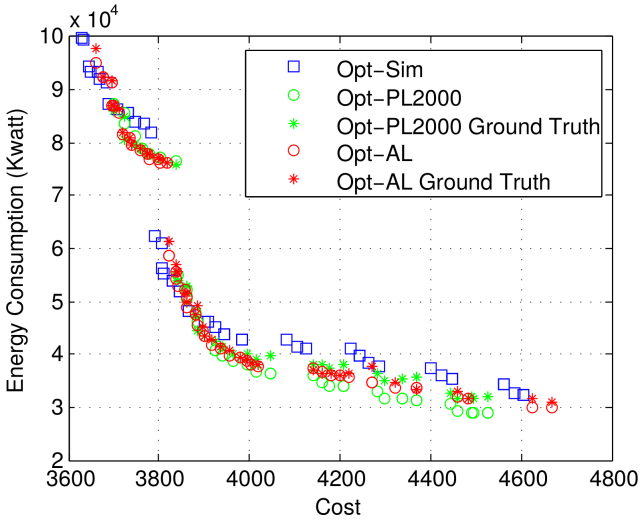

Figure 4: The Pareto front between cost and energy obtained by Opt-Sim, Opt-PL and Opt-AL. Best seen in color.

Table 4: Number of simulations and run-times for Opt-Sim, Opt-PL and Opt-AL.

Approach	# of Simulations			Time
	Before Opt.	During Opt.	Total	(Min)
Opt-Sim	-	2000	2000	4310
Opt-PL	2000	-	2000	4325
Opt-AL	500	320	820	1283

tween predictions and ground truth (results omitted). Hence, the active learning component, which simulated an additional 320 configurations during the optimization, is critical to the performance of Opt-AL.

Conclusion

The results presented in this preliminary work indicate that combining active learning and optimization approaches is a promising direction for achieving scalable building design optimization. The use of genetic algorithms like NSGA-II allows us to optimize multiple objectives, such as energy and cost. Future extensions to this work will explore how to automatically tune the parameters that guide the rate of active learning. We will explore how the size of the starting training set impacts the final performance. Also we would like to investigate whether other predictive models in addition to GPs could be easily employed under our optimization + active learning framework. We would like to extend our benchmark with other building performance measures, such as environmental impact and comfort.

Our approach falls in the class of methods that address multi-objective optimization problems, in which evaluating the objective functions is unaffordably expensive. Recently, (Zuluaga et al. 2013) proposed the Pareto active learning (PAL) method, which gradually classifies all points in the search space as Pareto optimal, non-Pareto optimal, and unclassified based on their evaluations and associated uncertainty bounds using a GP trained on a small subset of the search space. However, the measure used to choose the next sample to add to the training set needs to evaluate a covariance matrix over all points in the search space, limiting the applicability of this method to small search spaces. In contrast, our method uses the current population as the reference samples in the active learning selection, and hence can potentially scale up to larger search spaces.

Acknowledgments

The authors would like to thank Mahdi Roozbahani for technical assistance and Kyungki Kim for collaborating in the development of the building design model.

References

Abramowitz, M.; Stegun, I. A.; et al. 1965. *Handbook of mathematical functions*, volume 1046. Dover New York.

Asadi, E.; Silva, M. G. d.; Antunes, C. H.; Dias, L.; and Glicksman, L. 2014. Multi-objective optimization for building retrofit: A model using genetic algorithm and artificial neural network and an application. *Energy and Buildings* 81:444–456.

Breiman, L. 2001. Random forests. *Machine learning* 45(1):5–32.

Caldas, L. 2008. Generation of energy-efficient architecture solutions applying GENE_ARCH: An evolution-based generative design system. *Advanced Engineering Informatics* 22(1):59–70.

Chang, C.-C., and Lin, C.-J. 2011. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology* 2:27:1–27:27. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm.

Cohn, D. A. 1994. Neural network exploration using optimal experiment design. In *Advances in Neural Information Processing Systems (NIPS)*, volume 6, 679–686. Morgan Kaufmann.

Crawley, D. B.; Lawrie, L. K.; Winkelmann, F. C.; Buhl, W. F.; Huang, Y. J.; Pedersen, C. O.; Strand, R. K.; Liesen, R. J.; Fisher, D. E.; Witte, M. J.; et al. 2001. Energyplus: creating a new-generation building energy simulation program. *Energy and buildings* 33(4):319–331.

Deb, K.; Pratap, A.; Agarwal, S.; and Meyarivan, T. 2002. A fast and elitist multiobjective genetic algorithm: Nsga-ii. *Evolutionary Computation, IEEE Transactions on* 6(2):182–197.

Dong, B.; Cao, C.; and Lee, S. E. 2005. Applying support vector machines to predict building energy consumption in tropical region. *Energy and Buildings* 37(5):545–553.

Flager, F., and Haymaker, J. 2009. A comparison of multidisciplinary design, analysis and optimization processes in the building construction and aerospace.

Hastie, T.; Tibshirani, R.; Friedman, J.; Hastie, T.; Friedman, J.; and Tibshirani, R. 2009. *The elements of statistical learning*. Springer, 2 edition.

Iman, R. L. 2008. *Latin hypercube sampling*. Wiley Online Library.

Kolter, J. Z., and Ferreira, J. 2011. A large-scale study on predicting and contextualizing building energy usage. In *AAAI Conference on Artificial Intelligence*.

Korolija, I.; Marjanovic-Halburd, L.; Zhang, Y.; and Hanby, V. I. 2013. Uk office buildings archetypal model as methodological approach in development of regression models for predicting building energy consumption from heating and cooling demands. *Energy and Buildings* 60:152–162.

Li, Q.; Meng, Q.; Cai, J.; Yoshino, H.; and Mochida, A. 2009. Applying support vector machine to predict hourly cooling load in the building. *Applied Energy* 86(10):2249–2256.

Lin, S.-H. E., and Gerber, D. J. 2014. Designing-in performance: A framework for evolutionary energy performance feedback in early stage design. *Automation in Construction* 38:59–73.

Magnier, L., and Haghighat, F. 2010. Multiobjective optimization of building design using trnsys simulations, genetic algorithm, and artificial neural network. *Building and Environment* 45(3):739–746.

Neto, A. H., and Fiorelli, F. A. S. 2008. Comparison between detailed model simulation and artificial neural network for forecasting building energy consumption. *Energy and Buildings* 40(12):2169–2176.

Pérez-Lombard, L.; Ortiz, J.; and Pout, C. 2008. A review on buildings energy consumption information. *Energy and buildings* 40(3):394–398.

Rasmussen, C. E., and Nickisch, H. 2014. The gpml toolbox version 3.4. http://www.gaussianprocess.org/gpml/code.

Rasmussen, C. E. 2004. Gaussian processes in machine learning. In *Advanced Lectures on Machine Learning*. Springer. 63–71.

Seo, S.; Wallat, M.; Graepel, T.; and Obermayer, K. 2000. Gaussian process regression: Active data selection and test point rejection. In *DAGM-Symposium*, 27–34. Springer-Verlag.

Tsanas, A., and Xifara, A. 2012. Accurate quantitative estimation of energy performance of residential buildings using statistical machine learning tools. *Energy and Buildings* 49:560–567.

Tso, G. K., and Yau, K. K. 2007. Predicting electricity energy consumption: A comparison of regression analysis, decision tree and neural networks. *Energy* 32(9):1761–1768.

Tuhus-Dubrow, D., and Krarti, M. 2010. Genetic-algorithm based approach to optimize building envelope design for residential buildings. *Building and Environment* 45(7):1574–1581.

Vapnik, V.; Golowich, S. E.; and Smola, A. 1997. Support vector method for function approximation, regression estimation, and signal processing. *Advances in neural information processing systems* 281–287.

Wang, W.; Zmeureanu, R.; and Rivard, H. 2005. Applying multi-objective genetic algorithms in green building design optimization. *Building and environment* 40(11):1512–1525.

Yang, J.; Rivard, H.; and Zmeureanu, R. 2005. On-line building energy prediction using adaptive artificial neural networks. *Energy and Buildings* 37(12):1250 – 1259.

Yu, Z.; Haghighat, F.; Fung, B. C.; and Yoshino, H. 2010. A decision tree method for building energy demand modeling. *Energy and Buildings* 42(10):1637 – 1646.

Zemella, G.; De March, D.; Borrotti, M.; and Poli, I. 2011. Optimised design of energy efficient building façades via evolutionary neural networks. *Energy and Buildings* 43(12):3297–3302.

Zhang, Y.; ONeil, Z.; Wagner, T.; and Augenbroe, G. 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*.

Zhang, Y. 2012. Use jeplus as an efficient building design optimisation tool. In *CIBSE ASHRAE Technical Symposium, London*, 18–19.

Zhao, H.-X., and Magouls, F. 2012. A review on the prediction of building energy consumption. *Renewable and Sustainable Energy Reviews* 16(6):3586 – 3592.

Zuluaga, M.; Sergent, G.; Krause, A.; and Püschel, M. 2013. Active learning for multi-objective optimization. In *Proceedings of the 30th International Conference on Machine Learning*, 462–470.