CS230

DL-based optimization framework to support recovery-based design of buildings

Omar Issa[*]
Stanford University
oissa@stanford.edu

Peter Lee[*]
Stanford University
jlee6925@stanford.edu

1 Introduction

A recent study by the Federal Emergency Management Agency (FEMA) NIST [2021] found that 20-40% of modern code-conforming buildings would be unfit for re-occupancy following a major earthquake (taking years to repair) and 15-20% would be rendered irreparable. In recent years, functional recovery NIST [2021] FEMA [2020] has been proposed in which buildings would be required to recover their basic, tenant-specific functions in target time T_{target} for a region-specific seismic hazard. Our aim is to understand how modern, code-conforming buildings can optimally be improved to achieve explicit post-disaster recovery time targets. Currently, this can be done using a simulation-based approach, but makes such a study computationally difficult to implement in practice.

To bypass these computational constraints, we have developed a surrogate model based on a simple 3-story welded-steel moment frame (WSMF) building, a common structure that can be found across the U.S. This NN is trained using a dataset of median recovery times from two probabilistic (Monte Carlo) simulation platforms under a range of nonstructural component improvements:

- A building damage and loss platform – FEMA-P58 Mahoney et al. [2018]
- A building recovery platform – ATC-138 Cook [2021]

The raw input to the platform is a 48x1 array containing a 41x1 array of nonstructural component improvements x_i and 7x1 array of engineering demand parameters (EDPs) EDP_i (which represent earthquake demands - e.g. peak floor acceleration and story drifts). The output of the simulation is the median recovery time in number of days. Overall, the training data represents the range of possible recovery times that can achieved under (i) different combinations of nonstructural component improvements and (ii) different seismic intensity levels for the proposed code conforming building. The NN is then used in place of the simulation platform for the purposes of optimization.

2 Dataset

Training examples for the NN were generated using a combination of random sampling of the input feature vector (i.e., randomly varying x_i and EDP_i) and hand-engineered cases that would otherwise not appear in the training set. Overall, the training set contained:

- Subset 1 ($\sim$75,000 examples): randomized engineering demands EDP_i and randomized component improvements x_i.
- Subset 2 ($\sim$7,000 examples): hand-engineered, "uniform" set; all components in the building are scaled together at increasing increments of 10%. Repeated at each percentile of earthquake demands considered for our dataset.

[*]Department of Civil and Environmental Engineering

- Subset 3 (∼6,000 examples): hand-engineered "baseline" set; no component enhancements are made (i.e., each x_i is 1.00). Repeated for random samples of EDP_i.
- Subset 4 (∼6,000 examples): hand-engineered, one-component-at-a-time (OCAT) set; components are incrementally strengthened by 10%, one at a time. Repeated at each decile of engineering demands considered for our dataset.
- Subset 5 (∼6,000 examples): hand-engineered, one-system-at-a-time (OSAT) set; one or more components belonging to a particular system (e.g., electrical) are incrementally strengthened together by 10%, one system at a time. Repeated at each percentile of engineering demands considered for our dataset.

In each case, $x_i \in [0.50\ 4.00]$. While improvements are more realistically bounded by $x_i = 1.00$ (no change to original component capacity) and $x_i = 3.00$ (3x the original component capacity), these broader bounds help increase the number of samples available at our extremes of interest. Together, the randomized and hand-engineered training examples teach the NN how to predict recovery time at a wide variety of earthquake intensities that may occur, under a combination of user-defined nonstructural component improvements.

3 NN Architecture

Our surrogate model took the form of a multi-layered regressor where the inputs are the aforementioned [48x1] vector of engineering demand parameters and non-structural component features. Initially, we generated hazard-specific models which only accepted the 41 non-structural components as input features. This required the training of discrete models for three different return periods of occurrence (72, 475, and 2475 years) and soon proved to be a bottleneck in terms of being generalizable to other levels of hazard. The EDPs were introduced to resolve this issue; these numbers represent the intensity of the demand posed on the structure under different levels of hazard, thus freeing the model from being constrained to a specific return period.

In terms of the general framework, we experimented between the KerasRegressor (which relies on a scikit-learn wrapper) and the mlpregressor in scikit-learn. Although they are theoretically relying on the same principles, the mlpregressor proved to be more cost-efficient in terms of training time. To expedite model development, we trained the respective models via mlpregressor and transferred the weights and biases to a Keras sequential model.

4 Hyperparameter Search

An iterative approach was taken to strike a balance between hyperparameter tuning and training set generation to achieve increased accuracy. We aimed to achieve 99.5% accuracy on both the train and test set and increased the number of training samples incrementally. As can be seen in Figure 1(c), this level of accuracy was generally achieved using 90,000 samples; we observed that the performance begins to slightly saturate afterwards and thus used 100,000 samples in the final iteration.

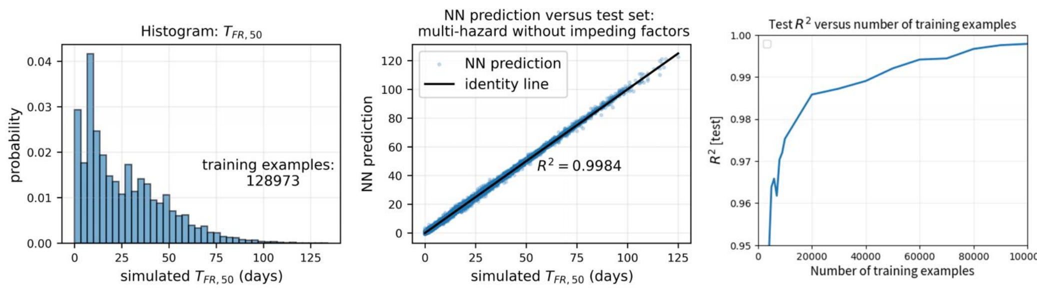
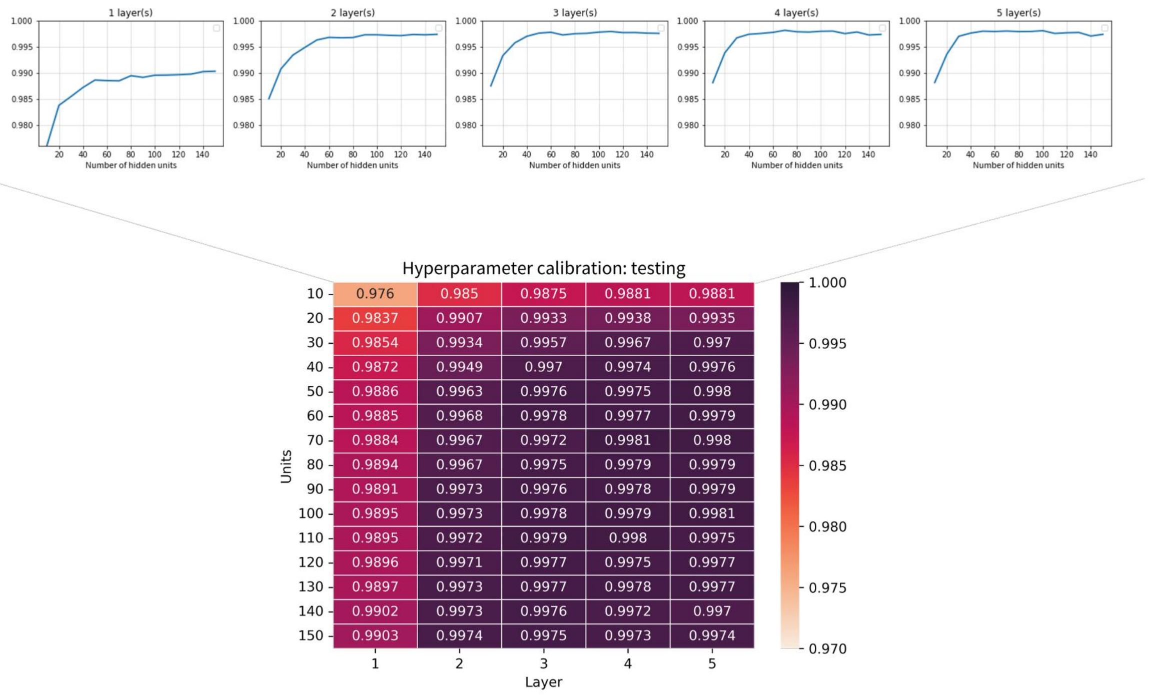


Figure 1: Model performance and dataset generation (a) Histogram of recovery time outputs of the training set (b) Test-set accuracy of the final model with 3 layers, 120 units (c) Test-set accuracy vs. training set size.

Particular attention was given to the accuracy on hand-engineered examples as the majority of these lay at the boundaries of our dataset. For each set of training sample, we tested the performance for layers ranging from 1 to 5 and hidden units ranging from 10 to 150. The accuracy for each iteration is shown in Figure 2 below. Across different sizes of training samples, we observed that the performance generally saturates around $N_{layers} = 3$ and $N_{units} = 120$, which we used in the final model.



Units	1	2	3	4	5
10	0.976	0.985	0.9875	0.9881	0.9881
20	0.9837	0.9907	0.9933	0.9938	0.9935
30	0.9854	0.9934	0.9957	0.9967	0.997
40	0.9872	0.9949	0.997	0.9974	0.9976
50	0.9886	0.9963	0.9976	0.9975	0.998
60	0.9885	0.9968	0.9978	0.9977	0.9979
70	0.9884	0.9967	0.9972	0.9981	0.998
80	0.9894	0.9967	0.9975	0.9979	0.9979
90	0.9891	0.9973	0.9976	0.9978	0.9979
100	0.9895	0.9973	0.9978	0.9979	0.9981
110	0.9895	0.9972	0.9979	0.998	0.9975
120	0.9896	0.9971	0.9977	0.9975	0.9977
130	0.9897	0.9973	0.9977	0.9978	0.9977
140	0.9902	0.9973	0.9976	0.9972	0.997
150	0.9903	0.9974	0.9975	0.9973	0.9974

Figure 2: Test set accuracy with varying number of layers and hidden units (a) Breakdown by layer (b) Color plot of test-set accuracy.

5 Optimization Approaches

5.1 Optimization objectives

Our optimization isolates the optimal set of nonstructural component enhancements that achieve two objectives: (a) meet a set target recovery time T_{target} while (b) minimizing the number of said component enhancements. The two objectives together prevent trivial solutions (i.e., a vector in which each $x_i = 3.00$) since we are not merely trying to minimize the recovery time.

5.2 Genetic algorithm

For an unguided approach, we used a genetic algorithm, a flexible, population-based method that is appropriate for use with simulation-based evaluations of the objective function. The objective function selected for use in this case was:

$$f(x) = \sum_{i=1}^{n} x_i + \rho(\widehat{T}(\mathbf{x}) - T_{target})^2 \tag{1}$$

where $\rho(T - T_{target})^2$ represents a quadratic penalty applied to solutions that do not meet the target time.

5.3 Gradient-based approach

For a guided approach, we were inspired by adversarial attacks since our optimization process involves a fixed neural network model and "perturbing" the input feature vector to meet the target time T_{target} under the constraint of perturbing as few features as possible. The fitting of an NN to the Monte-Carlo based framework has the effect of "smoothing out" an otherwise discrete optimization surface. Thus, our trained NN enables the use of a gradient-descent optimization. To satisfy the two aforementioned objectives, we utilized the following loss function.

$$L(\widehat{T}(\mathbf{x}), T_{target}) = \sqrt{(\widehat{T}(\mathbf{x}) - T_{target})^2} + \lambda \sum_{i=1}^{n} |x_i - x_{target,i}| \tag{2}$$

Since we want the enhancements to be sparse rather than spread out across a large number of components, we relied on L_1 regularization. In order to only perturb the non-structural components vector (i.e. the input of EDPs should be left constant), the gradients of the first seven input features were reset to zero before applying the perturbations. The gradient-descent was carried out based on $\mathbf{x} = \mathbf{x} - \eta \frac{\partial L(\widehat{T}(\mathbf{x}), T)}{\partial \mathbf{x}}$ via where $\eta = 0.001$ and $\lambda = 2000$ was used by default with 2000 iterations.

6 Discussion of Results

6.1 Comparison of the two approaches

The optimization results of the two approaches are shown in Appendix A (Figure 5). General trends hold under both methods, at all earthquake intensities considered. As expected, the gradient-based approach is more restrained in its selection of components for the design solution, while the genetic algorithm is slightly more flexible. Components which are deemed important in one method (e.g., stairs, glass curtain walls, and HVAC duct drops) reoccur in the other.

6.2 Spectrum of varying target and hazards

We tested our optimization framework to find the optimal set of component enhancements under two different schemes. In the first scheme, we fix the target time and increase the EDPs from the 0th to 60th decile as shown in Figure 3(a); this essentially has the effect of applying a more intense earthquake. We clearly observe the number, as well as the degree, of component enhancements increase with increasing deciles.

The second scheme is to fix the EDP, and decrease the target recovery time from 49 to 7 days as shown in Figure 3(b). Again, we observe that the number of enhancements increase to satisfy a lower target time. It is worth noting that the gradient-based approach faces a limit of achievable target time under a given penalty factor λ (e.g. given the 50th decile EDPs, the minimum target time possible is 13.05 days, rendering 7 days impossible). In these instances, λ was iteratively decreased so that the target time can be achieved at the cost of entailing more component enhancements.

6.3 Multi-hazard, multi-target matrix for decision-making

The resulting NN unlocks the ability to perform multiple optimizations across a range of earthquake hazard levels and recovery time targets. The resulting matrix (Figure 4) enables engineers in their decision-making to achieve the above-code performance targets. "Slices" of the matrix represent fixed target optima at varying earthquake hazards, or varying targets for a fixed intensity of interest. Cells contain the objective function value obtained through each optimization, or the number of components involved in the optimal design scheme. Stakeholders can use this information to (i) select appropriate target times as a matter of engineering policy, (ii) determine a reasonable "first estimate" of optimal nonstructural improvements subject to specific seismic intensities, and (iii) gauge the maximum performance gain that can be derived with fixed resources.

7 Codes

The codes for the training and optimization are available at the following repo:

https://github.com/Omarissa/PBEE-Optimization/tree/main/NN_optimization

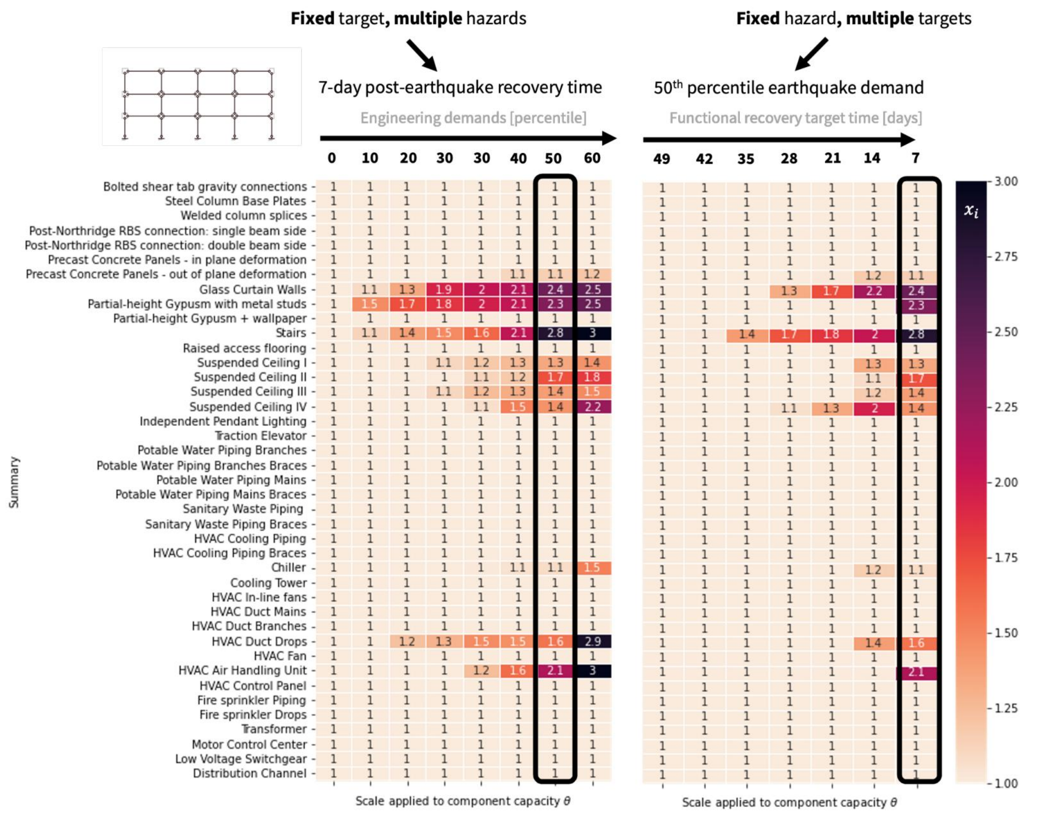
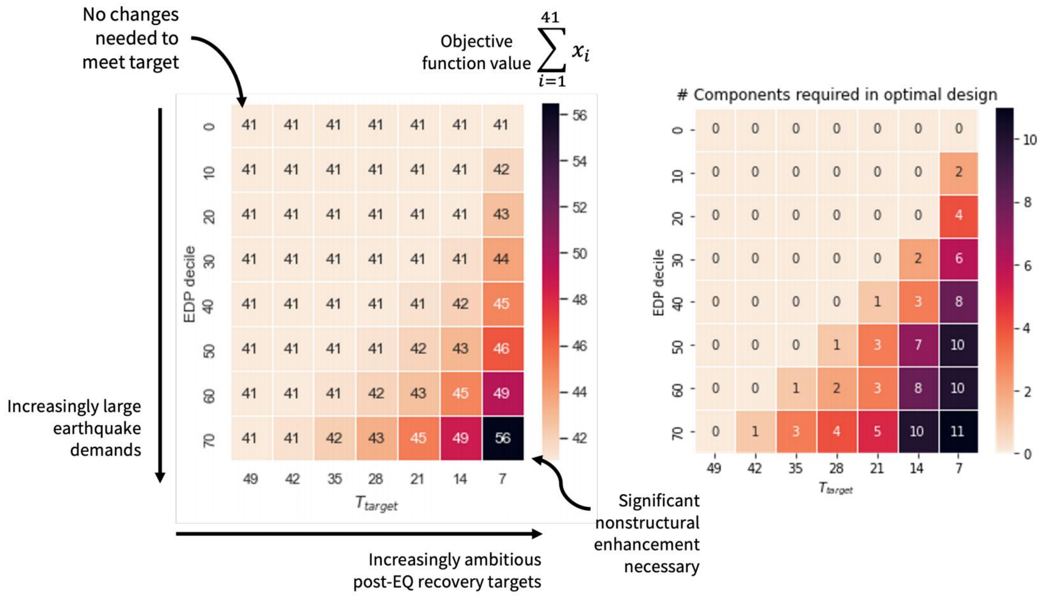


Figure 3: Optimal suite of nonstructural component improvements (x_i applied to median component capacity) for a fixed target, repeated for multiple earthquake intensities (a) and for a fixed earthquake intensity, repeated for multiple target times (b). Such results would be prohibitively expensive using a simulation-based approach.

8 Contributions

Omar was responsible for the initial ideation for the project as well as the dataset generation. Peter focused on the hyperparemeter tuning and the transfer of model from scikitlearn to tensorflow. Both members discussed the best dataset generation scheme and optimal NN architecture. In terms of optimization, Omar focused on the population-based method, while Peter implemented the gradient-based algorithm. Similarly, both collaborated on the interpretation of the optimization scheme and processing the results.



EDP decile	49	42	35	28	21	14	7
0	41	41	41	41	41	41	41
10	41	41	41	41	41	41	42
20	41	41	41	41	41	41	43
30	41	41	41	41	41	41	44
40	41	41	41	41	41	42	45
50	41	41	41	41	42	43	46
60	41	41	41	42	43	45	49
70	41	41	42	43	45	49	56

EDP decile	49	42	35	28	21	14	7
0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	2
20	0	0	0	0	0	0	4
30	0	0	0	0	0	2	6
40	0	0	0	0	1	3	8
50	0	0	0	1	3	7	10
60	0	0	1	2	3	8	10
70	0	1	3	4	5	10	11

Figure 4: The NN unlocks the ability to perform nonstructural optimization at multiple earthquake intensities and target recovery times [days]. The results can be represented in matrix format in terms of (a) raw objective function value and (b) the actual number of components in the optimal solution.

References

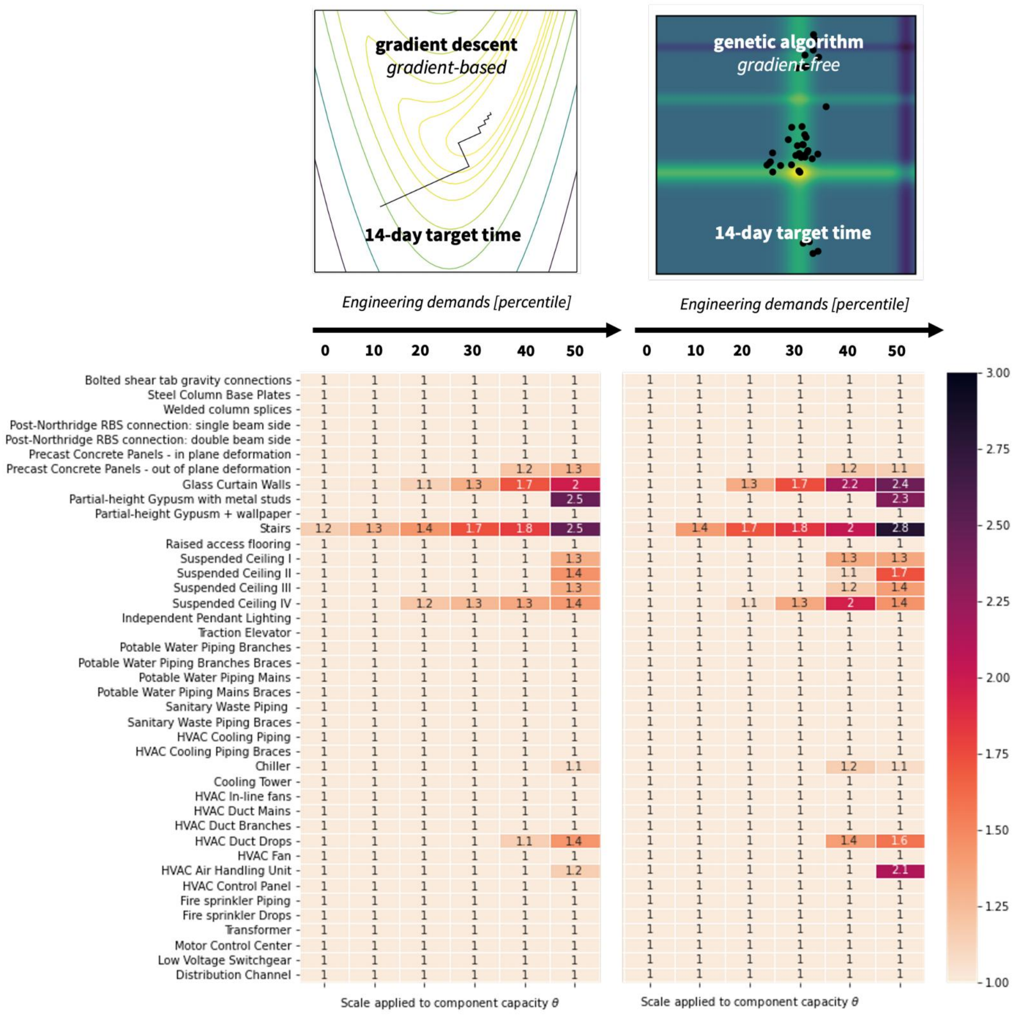
FEMA / NIST. Nist-fema special publication fema p-2090/nist sp-1254, recommended options for improving the built environment for post-earthquake reoccupancy and functional recovery time. 2021. doi: 10.6028/NIST.SP.1254. URL https://doi.org/10.6028/NIST.SP.1254.

FEMA. Fema p-2082-2 nehrp recommended seismic provisions for new buildings and other structures, 2020.

Michael Mahoney, Christopher Rojahn, Jon A Heintz, John Gillengerten, William T Holmes, John D Hooper, Stephen A Mahin, Jack P Moehle, Khalid Mosalam, Laura Samant, Steven R Winkel, Lucy Arendt, Deborah Beck, Christopher Deneff, H John Price, Jonathan C Siu, Jeffrey R Soulages, Eric Berg, Williston Warren, David R Bonneville, Dominic Campi, Vesna Terzic, Russell Larsen, and Peter Morris. Seismic performance assessment of buildings, volume 5 – expected seismic performance of code-conforming buildings, 2018. URL www.ATCouncil.org.

Dustin Trevor Cook. Advancing performance-based earthquake engineering for modern resilience objectives, 2021.

9 Appendix A



	gradient descent						genetic algorithm					
	0	10	20	30	40	50	0	10	20	30	40	50
Bolted shear tab gravity connections	1	1	1	1	1	1	1	1	1	1	1	1
Steel Column Base Plates	1	1	1	1	1	1	1	1	1	1	1	1
Welded column splices	1	1	1	1	1	1	1	1	1	1	1	1
Post-Northridge RBS connection: single beam side	1	1	1	1	1	1	1	1	1	1	1	1
Post-Northridge RBS connection: double beam side	1	1	1	1	1	1	1	1	1	1	1	1
Precast Concrete Panels - in plane deformation	1	1	1	1	1	1	1	1	1	1	1	1
Precast Concrete Panels - out of plane deformation	1	1	1	1	1.2	1.3	1	1	1	1	1.2	1.1
Glass Curtain Walls	1	1	1.1	1.3	1.7	2	1	1	1.3	1.7	2.2	2.4
Partial-height Gypusm with metal studs	1	1	1	1	1	2.5	1	1	1	1	1	2.3
Partial-height Gypusm + wallpaper	1	1	1	1	1	1	1	1	1	1	1	1
Stairs	1.2	1.3	1.4	1.7	1.8	2.5	1	1.4	1.7	1.8	2	2.8
Raised access flooring	1	1	1	1	1	1	1	1	1	1	1	1
Suspended Ceiling I	1	1	1	1	1	1.3	1	1	1	1	1.3	1.3
Suspended Ceiling II	1	1	1	1	1	1.4	1	1	1	1	1.1	1.7
Suspended Ceiling III	1	1	1	1	1	1.3	1	1	1	1	1.2	1.4
Suspended Ceiling IV	1	1	1.2	1.3	1.3	1.4	1	1	1.1	1.3	2	1.4
Independent Pendant Lighting	1	1	1	1	1	1	1	1	1	1	1	1
Traction Elevator	1	1	1	1	1	1	1	1	1	1	1	1
Potable Water Piping Branches	1	1	1	1	1	1	1	1	1	1	1	1
Potable Water Piping Branches Braces	1	1	1	1	1	1	1	1	1	1	1	1
Potable Water Piping Mains	1	1	1	1	1	1	1	1	1	1	1	1
Potable Water Piping Mains Braces	1	1	1	1	1	1	1	1	1	1	1	1
Sanitary Waste Piping	1	1	1	1	1	1	1	1	1	1	1	1
Sanitary Waste Piping Braces	1	1	1	1	1	1	1	1	1	1	1	1
HVAC Cooling Piping	1	1	1	1	1	1	1	1	1	1	1	1
HVAC Cooling Piping Braces	1	1	1	1	1	1	1	1	1	1	1	1
Chiller	1	1	1	1	1	1.1	1	1	1	1	1.2	1.1
Cooling Tower	1	1	1	1	1	1	1	1	1	1	1	1
HVAC In-line fans	1	1	1	1	1	1	1	1	1	1	1	1
HVAC Duct Mains	1	1	1	1	1	1	1	1	1	1	1	1
HVAC Duct Branches	1	1	1	1	1	1	1	1	1	1	1	1
HVAC Duct Drops	1	1	1	1	1.1	1.4	1	1	1	1	1.4	1.6
HVAC Fan	1	1	1	1	1	1	1	1	1	1	1	1
HVAC Air Handling Unit	1	1	1	1	1	1.2	1	1	1	1	1	2.1
HVAC Control Panel	1	1	1	1	1	1	1	1	1	1	1	1
Fire sprinkler Piping	1	1	1	1	1	1	1	1	1	1	1	1
Fire sprinkler Drops	1	1	1	1	1	1	1	1	1	1	1	1
Transformer	1	1	1	1	1	1	1	1	1	1	1	1
Motor Control Center	1	1	1	1	1	1	1	1	1	1	1	1
Low Voltage Switchgear	1	1	1	1	1	1	1	1	1	1	1	1
Distribution Channel	1	1	1	1	1	1	1	1	1	1	1	1

Figure 5: Comparison of optima using the two optimization methods and objective functions presented in Section 5.3: gradient descent (a) and genetic algorithm (b). The results shown are for a 14-day target time across a range of earthquake hazard levels.