Issa, O., Silva-Lopez, R., Baker, J. W., and Burton, H. V. (2023). "Machine-learning-based optimization framework to support recovery-based design." *Earthquake Engineering & Structural Dynamics*, 52(11), 3256–3280. https://doi.org/10.1002/eqe.3860

RESEARCH ARTICLE

Machine-learning-based optimization framework to support recovery-based design

Omar Issa*[1] | Rodrigo Silva-Lopez[1] | Jack W. Baker[1] | Henry V. Burton[2]

[1]Department of Civil & Environmental Engineering, Stanford University, California, USA
[2]Department of Civil & Environmental Engineering, University of California, Los Angeles, California, USA

Correspondence
*Omar Issa, Email: oissa@stanford.edu

Summary

Recovery-based design links building-level engineering and broader community resilience objectives. However, the relationship between above-code engineering improvements and recovery performance is highly nonlinear and varies on a building- and site-specific basis, presenting a challenge to both individual owners and code developers. In addition, downtime simulations are computationally expensive and hinder exploration of the full design space. In this paper, we present an optimization framework to identify optimal above-code design improvements to achieve building-specific recovery objectives. We supplement the optimization with a workflow to develop surrogate models that (i) rapidly estimate recovery performance under a range of user-defined improvements, and (ii) enable complex and informative optimization techniques that can be repeated for different stakeholder priorities. We explore the implementation of the framework using a case study office building, with a 50th percentile baseline functional recovery time of 155 days at the 475-year ground-motion return period. To optimally achieve a target recovery time of 21 days, we find that nonstructural component enhancements are required, and that increasing structural strength (through increase of the importance factor) can be detrimental. However, for less ambitious target recovery times, we find that the use of larger importance factors eliminates the need for nonstructural component improvements. Such results demonstrate that the relative efficacy of a given recovery-based design strategy will depend strongly on the design criteria set by the user.

KEYWORDS:
recovery-based design, earthquake resilience, optimization, surrogate modeling, FEMA P-58, functional recovery

1 | INTRODUCTION

A study by the Federal Emergency Management Agency (FEMA)[1] found that 20-40% of modern code-conforming buildings would be unfit for re-occupancy following a major earthquake (taking months or years to repair) and 15-20% would be rendered irreparable. Functional recovery has been proposed as a performance objective to address the missing link between current design provisions and community resilience goals in the United States[2,3]. Buildings designed for functional recovery are expected to recover their basic, tenant-specific functions in target time T_{target} for defined seismic hazard levels. In a 2021 report to Congress[4], a committee of experts recommended the development of a national, consensus-based policy that would determine target times

and hazard levels for various classes of buildings. This shift, coupled with the emergence of enabling tools, has since led to numerous early adoptions of recovery-based design by individual owners[5,6,7,8].

While these are critical steps toward widespread implementation, and eventually, resilience-based engineering provisions, little research has been done to (i) inform the selection of recovery-based design criteria for particular occupancy types and (ii) systematically identify effective engineering strategies to achieve them. Generalizing (ii) is challenging since post-earthquake recovery performance depends on the building geometry, structural system, and site-specific conditions. Furthermore, the relationship between engineering decisions and recovery performance is highly nonlinear. Increasing structural strength, for example, may reduce damage to displacement-sensitive components, but amplify floor accelerations, and hence, increase damage to acceleration-sensitive components. Optimization algorithms have the potential to assist in determining effective design strategies. Such studies could explore how solutions (which satisfy design criteria such as T_{target}) can be configured to achieve stakeholder priorities such as minimizing cost or number of above-code modifications. Similarly, optimization studies could quantify the implications of the recovery-based policy objectives across a wide variety of building archetypes.

Here the recovery performance of individual buildings is evaluated using performance-based earthquake engineering[9] (PBEE) procedures, typically performed in two stages. First, a damage and loss assessment is performed using the FEMA P-58 methodology[10], which provides realizations of component- and building-level repair times. Second, to perform more refined simulations of recovery sequences than FEMA P-58 provides, the damage and loss assessment results are post-processed using a downtime assessment methodology [e.g., 11,12,13,14]. This is done to produce more realistic repair schedules, and impose appropriate impeding factors, such as inspection, financing, design, and contractor mobilization[15,14]. In this paper, we use the term "PBEE-based" to refer to performance assessment methodologies that use the FEMA P-58 method as well as a refined downtime assessment methodology. This combination of analyses was used in the example buildings mentioned above[5,6,7,8] and is quickly becoming the preferred benchmark for recovery-based design.

Several studies have optimized building designs within the context of PBEE. Rejas et al.[16] and Saadat et al.[17] performed multi-objective, risk-based design optimization using the original PBEE methodology. Both cases considered structural enhancements (in the form of section sizing) as the primary form of design enhancement. Steneker et al.[18,19] expanded the scope of enhancement to include both structural and nonstructural upgrades. One common obstacle across these studies is the high computational cost associated with evaluating building performance through PBEE-based methods, which imposes issues on scalability. Burton et. al[20] demonstrated that surrogate models are effective at reducing this issue. Their study implemented a computationally efficient multi-objective optimization, considering the influence of rocking steel braced frame parameters on up-front cost, economic losses, and environmental impacts. None of these optimization studies, however, leveraged the advanced downtime assessment procedures that build on FEMA P-58.

The main objective of our research, therefore, is to develop a recovery-based optimization framework that is high-fidelity and computationally efficient. It is high-fidelity in that it leverages downtime simulations to generate realistic repair sequences that quantify component- and system-level influences on damage and recovery time. In addition, the framework accommodates structural, nonstructural, recovery planning, and utility mitigation actions that may reduce downtime, as opposed to constraining improvements in only one or two categories. To achieve computational efficiency, we employ machine learning methods to estimate recovery times under parameterized building improvements. The resulting framework is flexible enough to enable optimization for early-stage conceptual design by individual owners, while remaining sufficiently efficient to perform batch optimization for the purposes of researching recovery-based policy decisions.

In the first half of this paper, we present the proposed framework (Section 2), along with the procedure to develop surrogate models for recovery performance evaluation (Section 3). These two sections describe the inputs and modeling considerations needed to perform an optimization using surrogate models. In the second half of the paper, we apply these procedures to the early-stage conceptual design of a new code-conforming office building in Oakland, California (Section 4). As a part of this case study, we perform the optimization for a single design criterion, then repeat across a continuum of target recovery times (through batch optimization) to demonstrate the utility of surrogate modeling. Finally, we present results and limitations, followed by conclusions of the paper (Section 5).

2 | AN OPTIMIZATION FRAMEWORK TO SUPPORT RECOVERY-BASED DESIGN

In this section, we describe the overall framework within the context of early-stage conceptual design of new buildings. Each stage of the framework is assigned a corresponding subsection that provides a generalized overview and important considerations. We will discuss additional details relevant to implementation during the case study (Section 4). The framework is illustrated in Figure 1, and can be split into two main parts:

I **Recovery modeling and scoping of design modifications (Stages 1, 2, and 3)**: these stages are used to prepare all necessary recovery modeling inputs for the building of interest, set the design criteria, and select the scope and design space of above-code enhancements. The gap between code-delivered performance and the target recovery time is quantified.

II **Optimization of design modifications (Stages 4 and 5)**: these stages set the optimization objectives and use them to guide the elimination of the gap identified in Part I.

To reduce the computational barrier associated with performing the optimization at scale, we propose machine learning methods to develop surrogate models. The resulting models are used in place of simulation-based procedures during Part II.

The five stages of the framework are as follows: (1) A site-specific design is performed in accordance with current code objectives, followed by a baseline recovery performance assessment. The goal of the baseline assessment is to quantify the recovery performance achieved when no above-code design modifications are made. Recovery performance is evaluated by first applying a probabilistic damage and loss assessment to obtain component- level damage and repair time metrics. These results are then used as inputs to a downtime assessment, which uses repair sequence logic to estimate the functional recovery time, considering various impeding factors. (2) Recovery-based design criteria are set to define the target recovery time, hazard level of interest, and acceptability threshold. The selected target time is used to calculate the recovery gap: the difference between the baseline functional recovery and the target functional recovery time. (3) The recovery-based design strategy and scope of design modifications are selected with the objective of closing the recovery gap. In some cases, the selected modifications will be unable to satisfy the design criteria set in (2), and will prevent optimization convergence. For this reason, (4) a feasibility check is performed to ensure that the selected strategy is capable of achieving the target time under the design criteria. If not, (3) is repeated as necessary. Finally, (5) optimization is performed to isolate solutions within the design space that satisfy design criteria. Optimality is defined using an objective function that reflects stakeholder-specific priorities. These steps are defined in further detail in the following subsections.

2.1 | Code-based design and baseline performance assessment

The first step of the proposed framework is to determine baseline recovery metrics that reflect as-is building performance with no improvement. A baseline recovery performance assessment is conducted using PBEE-based approaches, which require three user inputs: (i) characterization of earthquake demands (e.g., ensemble of ground motions that reflect the site-specific hazard), (ii) an analytical structural model that captures structural response when subject to ground shaking, and (iii) a building performance model, which summarizes building assets and their post-damage consequences (Figure 2). We describe each of these inputs in greater detail below.

- **Characterization of earthquake demands**: earthquake demands are characterized by a suite of site-specific ground motions at the intensity level of interest.
- **Structural model**: an analytical model captures structural response when subjected to the selected ground motions. Engineering demands parameters (EDPs), namely, story drift ratio (SDR) and peak floor acceleration (PFA), are recorded at each level of the building by performing nonlinear response history analysis (NLRHA). EDPs are used to map demands – generated by the ground motion hazard – to component-level damage.
- **Performance model**: the performance model summarizes assets at risk to earthquake damage. In addition to building attributes, the performance model should contain a component ensemble, with asset-specific locations, quantities, vulnerability functions, and consequence functions. Ensembles typically contain a combination of structural components, nonstructural components, and contents.

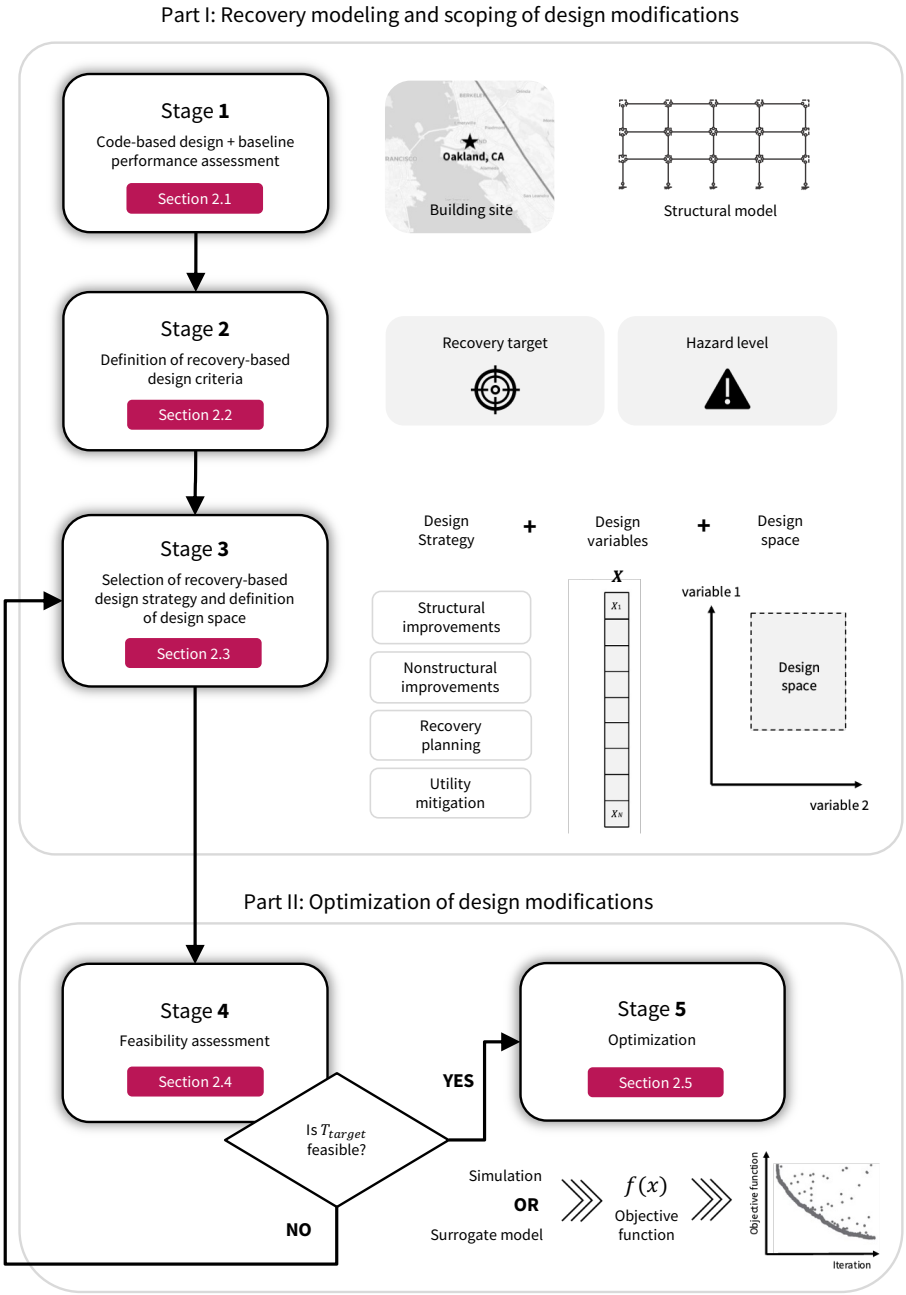


FIGURE 1 Major steps in the proposed framework, from code-based design to recovery-based optimization.

Each of the user-provided inputs is propagated through a downtime assessment. We illustrate a typical simulation in Figure 2. The final output from this step is a set of building recovery realizations, in terms of functional recovery time. Summary statistics from this set (e.g., 50th and 90th percentile) are used to report the results of the baseline assessment.

2.2 | Define recovery-based design criteria

In the second stage of the framework, the recovery-based design criteria are defined. These criteria serve as the target performance objective of the design, and during optimization, are used to assess the validity of a given solution. Here, the objective

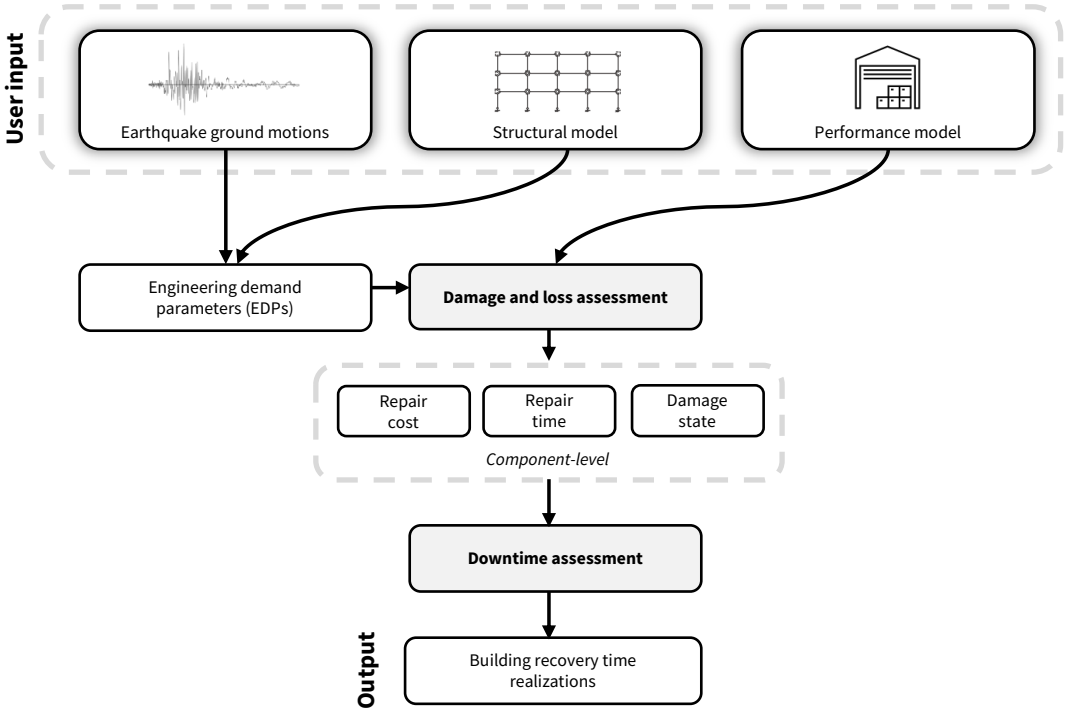


FIGURE 2 Overview of a typical downtime simulation used throughout the framework. User inputs are propagated through a damage and loss assessment, which generates component-level damage and repair metrics. These results are then used as inputs to the downtime assessment.

of recovery-based design is to limit the probability of exceeding a particular target time, subject to a seismic intensity level of interest[3]:

$$P(T_{FR} > T_{target} | D_{FR}) < Y\% \tag{1}$$

where: T_{FR} = functional recovery time
T_{target} = target functional recovery time
Y = threshold probability of exceedance
D_{FR} = selected hazard level for recovery-based design

The design criteria in Equation 1 should be selected to reflect tenant, owner, or community goals when recovery-based design is being performed on a voluntary basis. Within the context of national policy, on the other hand, these will likely be decided by consensus.

Note that the selection of Y will determine the p-percentile functional recovery time, $T_{FR,p}$, that is compared directly with T_{target}. For example, if Y is set to 10% such that $P(T_{FR} > T_{target} | D_{FR}) < 10\%$, then the $T_{FR,p}$ would be set to the 90th percentile functional recovery time when comparing against the target time. During the optimization step, all solutions that yield $T_{FR,p} > T_{target}$ will be deemed non-compliant with the design criteria. A key output from this stage is the recovery gap, defined as the difference between the baseline $T_{FR,p}$ and T_{target}. The objective of the framework is to identify optimal design solutions that eliminate this gap.

2.3 | Selection of recovery-based design strategy and definition of design space

The third stage is dedicated to defining a recovery-based design strategy, along with the corresponding variables and design space. We define a design strategy as the set of potential building enhancements. Enhancements include, but are not limited

to, structural improvements, nonstructural improvements, recovery planning, and utility mitigation[3]. When numerous viable recovery-based design strategies are under consideration, the framework can be used to rapidly compare optimality in each case.

Structural improvements could include modifying the importance factor I_e, modifying drift limits, and choice of the lateral-force-resisting system. Nonstructural component improvements could include modifying, hardening, relocating, or eliminating components. Recovery planning encompasses actions that accelerate re-occupancy and recovery, such repair services on retainer, pre-defined permit applications, or business continuity plans. Utility mitigation actions include backup and substitution of one or more building utilities to minimize lifeline-induced downtime.

For each enhancement considered in the analysis, a design variable (and a corresponding design space) must be defined. Each enhancement involves a single design variable that is defined on a discrete or continuous scale. Design variables are organized in an array $\mathbf{X}$ that summarizes the entire design strategy, such that:

$$\mathbf{X} = \begin{bmatrix} \mathbf{X_S} \\ \mathbf{X_{NS}} \\ \mathbf{X_{Rec}} \\ \mathbf{X_{Util}} \end{bmatrix} \tag{2}$$

where: $\mathbf{X_S}$ = array of structural modification variables
$\mathbf{X_{NS}}$ = array of nonstructural modification variables
$\mathbf{X_{Rec}}$ = array of recovery planning variables
$\mathbf{X_{Util}}$ = array of utility mitigation variables

Next, enhancement resolution – whether each variable will be imposed on a global basis (across the entire building) or on a localized basis (a specific floor, tenant unit, etc.) – is selected, and will further influence the number of variables in the problem. For a 3-story building, for example, modeling suspended ceiling improvements on a floor-level basis for a would require three separate variables. Finally, for each variable, upper and lower bounds are defined. Ideally, these bounds should be based on empirical/laboratory data that reflect the physical extents of possible improvement. The design variables and design space are later used in Stage 5, as constraints during the optimization.

2.4 | Feasibility assessment

At Stage 3, it is not yet known whether the selected recovery-based design strategy is able to eliminate the recovery gap identified in Stage 2. Therefore, in Stage 4, a feasibility check is performed to ensure that at least one solution in the design space meets the target time. If no solution exists, Stage 3 will be repeated as necessary, as illustrated in Figure 1.

The check is performed by assessing $T_{FR,p}$ under the influence of maximal enhancement using the proposed strategy. All design variables in $\mathbf{X}$ should be considered simultaneously. Here, maximal enhancement refers to the recovery-based design strategy defined in Stage 3, evaluated at values that yield maximal $T_{FR,p}$ improvement. For example, if the recovery-based design strategy consists solely of nonstructural component hardening, one âĂIJstress testâĂİ could be to assess $T_{FR,p}$ under the influence of maximum nonstructural component capacity improvements for all components in the building. For enhancements where the maximal improvement is not always clear, (e.g., in the case of lateral system selection) more extensive checks may be required to determine which improvements yield the greatest reduction in downtime. The assessment is performed by first recording the (i) p-percentile functional recovery time achieved during the baseline assessment, and (ii) the p-percentile functional recovery time achieved under maximal enhancement.

If (ii) is lower than T_{target}, then some combination of enhancements will meet the design criteria set in Stage 2, and the optimization will converge. If (ii) is higher than T_{target}, then it is impossible to achieve the target time despite maximal enhancement, and the optimization will not converge. In this case, Stage 3 of the framework should be repeated. Once the feasibility assessment demonstrates that a viable solution exists, the optimization procedure in next stage will locate an optimal solution within the defined design space.

2.5 | Optimization

In the final stage of the framework, an optimal suite of improvements for the building will be isolated using the strategy and design space defined in Stage 3. To perform the optimization, an objective function $f(\mathbf{X})$ must be defined, and should consider stakeholder-specific objectives, which could include number of modifications, upfront cost of enhancement, construction days added, etc. By doing so, the above-code enhancements that originate from this analysis will effectively deliver the performance objective in Stage 2 within the context of owner and tenant priorities. It is important to note, however, that the pre-requisite data is required when considering certain stakeholder priorities (e.g., upfront cost).

The objective function $f(\mathbf{X})$ is evaluated under the parameterized design improvements in $\mathbf{X}$, which should be formulated to impose (i) a constraint on the T_{target}, and (ii) constraints on the design variables using the upper and lower bounds defined in Stage 3 . Other design criteria (e.g., Y, D_{FR}) are satisfied implicitly during each evaluation of the objective function, since $f(\mathbf{X})$ involves the evaluation of the p-percentile functional recovery time, which is computed at the D_{FR} hazard level. The main output of the framework is an updated design array $\mathbf{X}^*$ that minimizes the objective function (Equation 3) subject to constraints.

$$\mathbf{X}^* = \arg\min(f(\mathbf{X})) \tag{3}$$

with the constraints that: 1) $T_{FR,p} \leq T_{target}$, and 2) the design strategy is in the design space.

In general, the evaluation of the objective function can be divided into three steps (Figure 3):

1. **Select design variables:** Improvements defined in Stage 3 are organized in an array $\mathbf{X}$ of design variables, used to guide the necessary modifications to the user inputs for the subsequent downtime assessment.
2. **Calculate functional recovery time:** The downtime analysis is run using updated inputs, generating an ensemble of recovery realizations. The p-percentile functional recovery time, $T_{FR,p}$ is extracted and used in the evaluation of the objective function.
3. **Evaluate objective function:** If $T_{FR,p} \leq T_{target}$, $\mathbf{X}$ is considered a viable solution, and the value of the objective function is recorded. Otherwise, $\mathbf{X}$ is discarded, as it did not meet the design criteria set in Stage 2. The optimization is complete when a solution yielding the optimal $f(\mathbf{X})$ value is found.

From Figure 3a, we observe that evaluating the objective function using simulation-based approaches is computationally expensive, since each evaluation involves an end-to-end downtime assessment. Furthermore, the downtime assessment may need to be performed using a high number of realizations to ensure that the value of $T_{FR,p}$ is stable across runs, particularly when impeding factors are considered in the analysis. In our experience, we find that a downtime assessment using 5000 to 100,000 FEMA P-58 realizations can take between 30s to 500s to run (our experiments are run locally, using a computer with 64GB RAM and Apple M1 Max processor). It is necessary to reduce this computational expense since it hinders the framework's ability to be deployed at scale. For example, a user of this methodology may be interested in performing the optimization for multiple target times, hazard levels, and building sites – expensive objective function evaluations would impose a significant computational barrier on this process.

To address this challenge, the framework employs surrogate models to directly map the design variables $\mathbf{X}$ to the p-percentile functional recovery time, $T_{FR,p}$ (Figure 3b). There are a number of decisions that need to be made when designing a surrogate model to replace a traditional downtime assessment. We discuss these details in the next section.

The final consideration made during the optimization stage is the selection of the optimization algorithm. If simulation-based methods (Figure 3a) are employed to evaluate the objective function, then gradient-free algorithms, such as those within population-based methods, are most suitable. Population methods use a collection of points to explore the design space (as opposed to a functional form), do not require gradient information (which is not available when using simulation-based approaches), and are slow but generally easy to parallelize[21]. Examples of population-based methods include genetic algorithms, differential evolution, and hybrid methods. It is also noteworthy that population-based methods are stochastic in nature, so multiple optimization runs can yield slightly different results, depending on the size of the population used. For this reason, extra care must be taken to ensure the algorithm result is acceptably stable.

When a surrogate-based approach is employed (Figure 3b), function gradients can typically be computed, in which case both gradient-based and gradient-free optimization algorithms are possible. Gradient-based methods tend to converge to a solution faster, though can struggle in high-dimensional space and care needs to be taken to avoid converging to local minima. Unlike population-based approaches, many gradient-based approaches are not stochastic, and therefore converge to stable

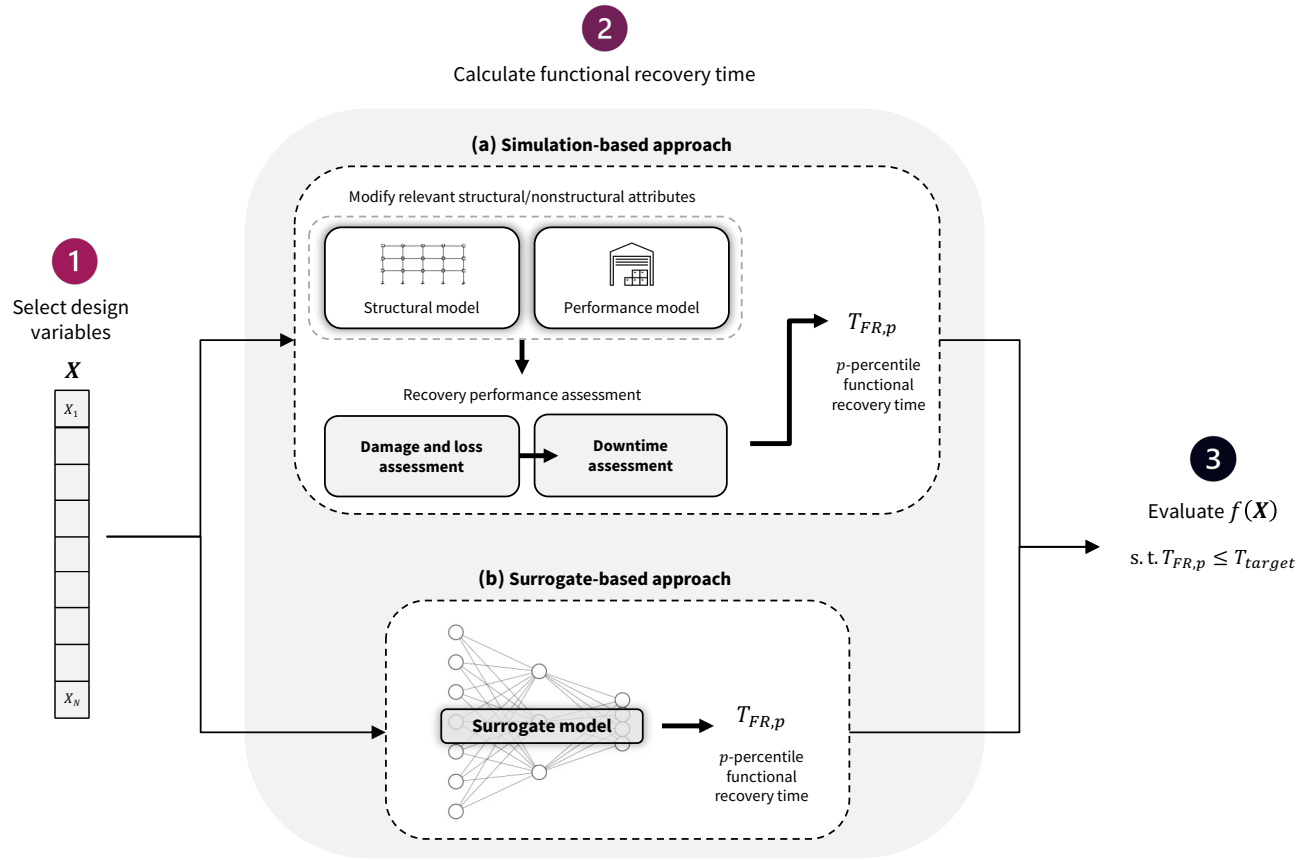


FIGURE 3 General process to evaluate the objective function $f(\mathbf{X})$ using (a) a traditional simulation-based approach and (b) a surrogate-model-based approach.

optima under identical algorithm settings. Examples of gradient-based approaches include gradient descent, RMSprop[22], and Adam[23]. Whether a gradient-based or gradient-free algorithm is selected, the optimization should yield consistent results from one analysis to the next, and be sufficiently powerful to locate the global optimum, likely in high-dimensional space.

3 | DEVELOPING SURROGATE MODELS FOR RECOVERY TIME PERFORMANCE ASSESSMENT

We propose the use of machine learning surrogate models to rapidly and accurately estimate functional recovery times, enabling the scaled implementation of complex and informative optimization techniques. We summarize the process of developing these surrogate models in this section, following the workflow illustrated in Figure 4. In some applications where numerous structural enhancements are being considered, a second, auxiliary model may be required to account for changes made to the structural response. In this section, we cover general strategies that can be applied in either case.

3.1 | Define I-O and surrogate model architecture

In the first step of the workflow, the architecture of the surrogate model is defined. The input-output (I-O) of the model must be developed for compatibility with the optimization problem defined in Stages 1 through 5 (i.e., the final surrogate model should be able to directly replace the simulation-based approach illustrated in Figure 3a). First, we discuss outputs, followed by user-defined inputs.

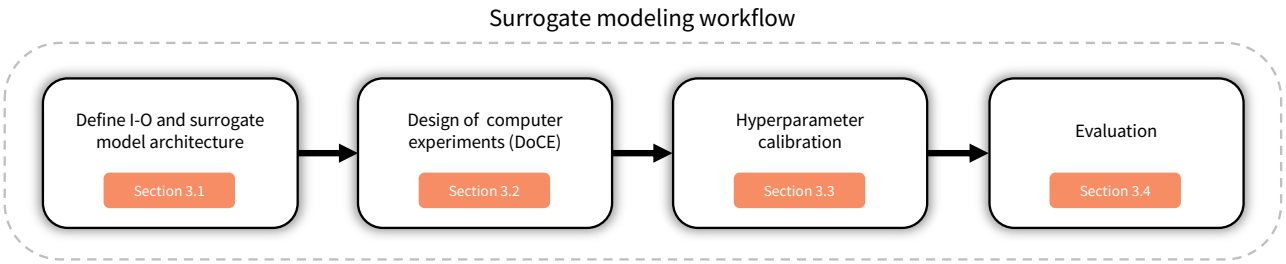


FIGURE 4 The proposed workflow used to develop surrogate models for compatibility with the optimization framework.

The output of the developed surrogate model, $T_{FR,p}$, is a scalar value used to evaluate the recovery target constraint (i.e., to check whether a given design solution satisfies T_{target} set in Stage 2). While this output represents the recovery time at a particular percentile, it is based on simulation data that spans the entire ensemble of results.

The user-defined input features play a significant role in both the development of the surrogate model (e.g., through training and testing protocols) and optimization (e.g., through the evaluation of the objective function). Overall, the input feature vector is developed in part by handpicking design variables from $\mathbf{X}$ (defined in Stage 3). Due to structural-nonstructural interaction, it is important to consider the trade-off between simplicity of the resulting surrogate model (e.g., mixing structural and nonstructural variables) and generality (e.g., a surrogate model's ability to be re-used under different circumstances).

It is possible to generalize a surrogate model to predict the functional recovery time at a range of structural response intensities. One way of achieving this is by extending the input feature vector to include engineering demand parameters (EDPs). In doing so, the surrogate model can be used to predict functional recovery time under both nonstructural improvements and a range of structural response conditions. These conditions include varying seismic hazard levels, site locations, and structural modifications.

When EDPs are used as input features, explicit variables that represent structural enhancements (e.g., I_e, drift limits) should not appear in the same surrogate model. Any structural enhancement variables, in these cases, should either (i) appear as variables to an auxiliary model that predicts a representative EDP profile or (ii) are accounted for implicitly through the use of EDPs (e.g., mapping drift limit changes to EDPs).

Case (i) is appropriate in situations where more than one structural enhancement is being considered at a time, or the design space is large and continuous. These structural design variables would be used as inputs to an auxiliary surrogate model that is used to predict EDPs (as opposed to recovery time). The resulting EDPs of this auxiliary model could then be plugged into the main surrogate model that is parameterized by EDP inputs and any nonstructural component enhancements. Case (ii) is more appropriate when a single structural design variable is being considered, and the design space is relatively small and discrete (e.g., $I_e \in \{1.00, 1.50, 2.00\}$). In this case, it is more appropriate to simulate an EDP profile associated with each option independently. The resulting EDPs could then be plugged into the main surrogate model that is parameterized by EDP inputs and any nonstructural component enhancements. No auxiliary model is needed.

If generality is not desired – and EDPs are not used as inputs to the surrogate model – then it is possible to include all structural and nonstructural modifications in the same input feature vector. While this is a simpler implementation, an implication of this decision is that the surrogate model would lose its ability to predict changes in recovery time at various seismic intensity levels and sites since, training data would be required to use the building's response generated by the structural design enhancements under a single, site-specific intensity level.

3.2 | Design of computer experiments (DoCE)

After the surrogate model architecture has been defined, the next step is to develop data for model training. This is typically done by creating a sampling protocol, or design of computer experiments (DoCE), which aim to capture as much information as possible about the relationships between the inputs and outputs[24] defined in Section 3.1. Typical DoCE techniques include one-at-a-time (OAT) analyses, space filling designs, random designs, and Latin hypercube sampling[25]. In general, the choice of DoCE should be based on the number of variables in the input feature vector, size of the design space, and shape of the design space.

Once the training examples are assembled, they are randomly placed into three subsets: a training dataset, a validation dataset, and a testing dataset. While the training and testing datasets are used for the purposes of model learning and model evaluation, respectively, the validation set is dedicated to hyperparameter calibration, which is discussed in the next section.

3.3 | Hyperparameter calibration

Developing a surrogate model is an iterative process, with numerous modeling decisions such as those described in the previous two sections. The third step involves the selection of hyperparameters, which control the learning process of a surrogate model, and hence, its ability to make accurate predictions. These parameters are selected after performing hyperparameter calibration (also referred to as tuning), which evaluates how well the surrogate model performs across a variety of hyperparameter sets, and locates optimal values. This is done using the validation set, and should be re-performed after making significant changes to the modeling decisions in Sections 3.1 and 3.2. Common hyperparameter calibration methods include grid search, random search, and gradient-based optimization.

3.4 | Evaluation

Using the selected architecture and hyperparameters, the surrogate model is trained (using the training set) and evaluated for accuracy (using the testing set). An evaluation metric must first be selected, such as the coefficient of determination (R^2), root mean squared error (RMSE), or mean absolute error (MAE). At a minimum, the surrogate model accuracy should be evaluated across all examples in the test set.

It is also recommended that the user evaluates the accuracy of relevant test set sub-intervals to ensure accuracy is reasonable at particular output values of interest (e.g., accuracy for $30 < T_{FR,p} < 60$ days). Here, "reasonable" implies that the predicted value of $T_{FR,p}$ is within a tolerable absolute mean error. Depending on the target time, this error (measured in days) may become significant in comparison, which may diminish the utility of any downstream optimization output (e.g., if the optimization output is actually providing designs that deliver one week of downtime when immediate functional recovery (<1 day) is desired).

These insights are valuable when the original dataset is imbalanced, and performance in one region does not necessarily guarantee performance in another. If the evaluation metrics yield unsatisfactory results in one region or across the test set, the DoCE can be revisited to better capture relationships between the inputs and outputs. In many cases, targeted sampling and/or additional data can help to improve overall performance. We discuss the relationship between DoCE data needs and model performance during the case study (Section 4.2).

4 | ILLUSTRATIVE CASE STUDY

In this section, we describe the application of the framework to the early-stage conceptual design of a new, 3-story steel moment resisting frame (SMRF) office building in Oakland, CA (Figure 5). Each level consists of 10,000 square feet of commercial space, occupied by a single tenant (i.e., there are 3 tenants total across the building). As a proof-of-concept, the primary objective of this application is to identify optimal structural and nonstructural improvements needed to achieve a 21-day 50th percentile functional recovery time, at the 475-year seismic hazard level. In the process, we will explore how the optimal design changes across different target recovery times. We facilitate all optimization analyses in this case study through the use of a calibrated surrogate model.

In Section 4.1, we describe the implementation of each step of the framework to achieve the aforementioned objectives. In Section 4.2, we describe the surrogate model architecture, calibration, and predictive performance. Finally, in Section 4.3, we perform the optimization, present the results, and highlight the limitations.

4.1 | Application of framework

Next, we apply the optimization framework to determine structural and nonstructural improvements for the 3-story SMRF office building, performed in accordance with Section 2. Considered improvements for this case study are nonstructural component strengthening and modifications to the importance factor, I_e. Structures designed with an importance factor $I_e > 1.00$ typically exhibit higher strength and experience reduced drifts at each story level. It is important to note, however, that reductions in story

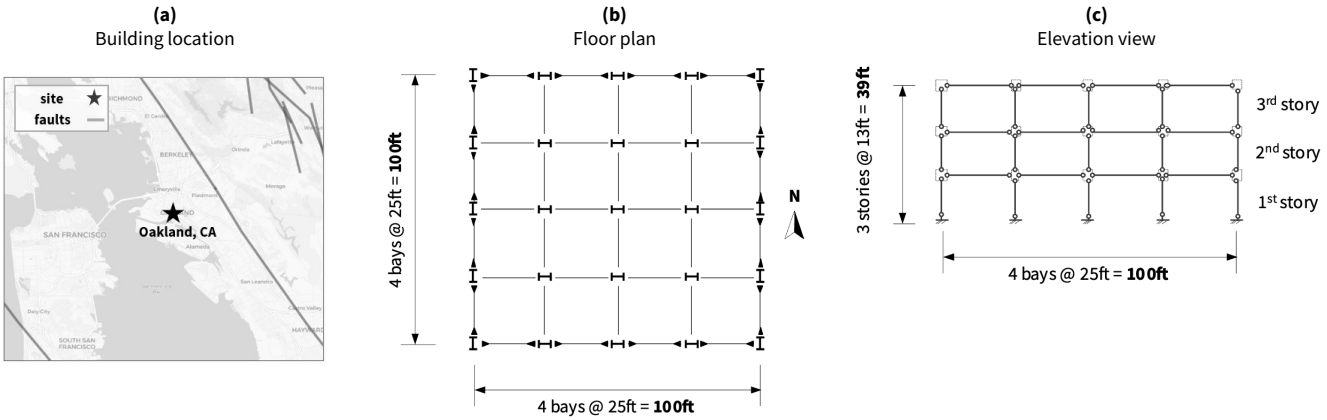


FIGURE 5 The location, floor plan, and elevation view of the three story office building used in the case study.

drift may lead to amplifications in floor acceleration, and have negative implications on nonstructural component performance. Optimization is performed using a calibrated surrogate model that rapidly estimates $T_{FR,p}$ under these enhancements. We cover the calibration of the surrogate model in Section 4.2. In the next five subsections, we describe each stage of the framework, starting with the original design and corresponding baseline performance assessment.

4.1.1 | Code-based design and baseline performance assessment

We begin this application by quantifying the recovery performance achieved through prescriptive requirements at the Oakland site. We design the building in accordance with ASCE 7-16[26] using the equivalent lateral force procedure (ELF). Section sizes for both the gravity and lateral system are determined using Auto-SDA[27], an end-to-end Python platform that automates the seismic design and analysis of SMRFs. The ELF parameters used in the design are in Table 1.

TABLE 1 ASCE-7 ELF design parameters for the case study building.

ASCE-7 ELF design parameter	Value
S_s	1.559 g
S_1	0.614 g
Importance factor I_e	1.00
Risk Category	II
Structural System	SMRF
Site Class	D
R	8
C_d	5.5
T_1	0.847s

To perform the baseline assessment (Section 2.1), we prepare each of the user-provided inputs highlighted in Figure 2 – the ground motion ensemble, structural model, and performance model. Earthquake hazard at the Oakland site is characterized by a suite of 40 selected and scaled ground motions[28] matched to the uniform hazard spectrum (UHS) for the 475-year return period. The analytical structural model is generated using Auto-SDA. Using the parameters and geometry in Table 1 and Figure 5, a 2-D frame model for the X- and Y-direction is generated in OpenSees[29], characterized by code-conforming section sizes as well as detailing for beams, columns, and connections.

Finally, we assemble a performance model consisting of 41 components, which specifies quantities, locations, and key attributes for each structural and nonstructural component in the building. Component quantities are determined in accordance with the FEMA P-58 Normative Quantities[30] when available, and prescribed using best judgment otherwise. Component fragility functions are parameterized by a median capacity, θ, and dispersion β in accordance with the FEMA P-58 fragility database. In cases where no default values of θ and β are available, a value is assigned based on FEMA P-58 guidance. A list of the components in the performance model is shown in Table 2.

Following the workflow in Figure 2 for the seismic hazard level considered, we perform a downtime assessment for the code-conforming building using the prepared inputs. The damage and loss assessment is performed using NHERI-SimCenter's PELICUN program for FEMA P-58 assessment[31]. Component-level damage results for each realization are post-processed using the Cook et al.[11] method to generate a realistic repair schedule and simulated impeding factors. The final output from the downtime assessment is an ensemble of functional recovery times. We present the 50th percentile results in Table 3 for an analysis consisting of 10000 realizations. We consider impeding factors in the analyses, to reflect real-world conditions and aid in the identification of enhancements that reduce their effects.

4.1.2 | Define recovery-based design criteria

After benchmarking the at-code recovery performance, we move on to define the recovery-based design criteria that serve as the target for the modified design. Design criteria D_{FR}, T_{target}, and Y, are defined in accordance with the performance objective in Equation 1. As mentioned in the introduction of Section 4, the building is being designed for recovery voluntarily, and hence, the selection of each value is flexible. As a result, our decisions mainly serve as a proof-of-concept, and are subject to refinement based on stakeholder needs. We describe these selections in detail below.

- **Hazard level**: D_{FR} is set to the 475-year return period.
- **Target functional recovery time**: T_{target} is set to 21 days, including impeding factors. From a policy-based perspective, this value of T_{target} would place the building in the "weeks to months" functional recovery category[4], which, at the hazard level selected, is typically associated with critical business enterprises. This target time is relatively ambitious, and we consider alternative values of T_{target} at the end of this section.
- **Threshold probability of exceedance**: $Y\%$ is set to the 50th percentile. This decision allows us to compare the median functional recovery time directly with T_{target} (i.e., the p-percentile recovery time is the 50th percentile recovery time).

Based on these decisions, the difference between the baseline 50th percentile functional recovery time and the target time is 155 - 21 = 134 days at the selected hazard level.

4.1.3 | Selection of recovery-based design strategy and definition of design space

After setting the design criteria and computing the recovery gap, we now define the recovery-based design strategy that will be used to eliminate this gap. For the case study building, our selected set of enhancements include both structural and nonstructural improvements to the original design. Recovery planning and utility mitigation options are not considered. We discuss each of the considered design enhancements in more detail here, along with their placement in $\mathbf{X}$, as defined in Equation 2.

For this case study demonstration, we limit the scope of structural improvements to modifying the importance factor I_e, which is used to increase seismic design forces and, in effect, reduce inelastic demands. As a result, the structural improvement array $\mathbf{X_S}$ contains I_e as its sole variable. We set the design space for this enhancement to $I_e \in \{1.00, 1.50, 2.00\}$. We also note that in this example, the structural system has been pre-selected, to simplify the presentation of results. In many cases, the selection of the structural system is an important recovery-based design modification, and could be considered in the same way that the I_e choices have been here.

We assume that each nonstructural component can be directly strengthened (i.e., achieve higher median capacity θ across all damage states). We modify each nonstructural component fragility function by applying a strengthening factor x_i, which we assume varies continuously from $x_i = 1.00$ (no change) to $x_i = 3.00$ (θ is tripled across all damage states). The fragility functions of each nonstructural component are modified by multiplying the original median capacity (as determined by the FEMA P-58 fragility database) by x_i (Equation 4). We organize all nonstructural enhancements x_i in the array $\mathbf{X_{NS}}$ (Equation

TABLE 2 Summary of FEMA P-58 performance model used in the case study.

FEMA P-58 component ID	System	Name	Unit	Location
B1031.001	Structural	Bolted shear tab gravity connections	ea	all
B1031.011a	Structural	Steel Column Base Plates	ea	1
B1031.021a	Structural	Welded column splices	ea	2
B1035.001	Structural	Post-Northridge RBS connection: single beam side	ea	all
B1035.011	Structural	Post-Northridge RBS connection: double beam side	ea	all
B2011.201a	Exterior	Precast Concrete Panels - in plane deformation	ft	all
B2011.201b	Exterior	Precast Concrete Panels - out of plane deformation	ft	all
B2022.002	Exterior	Glass Curtain Walls	ft^2	all
C1011.001b	Interior	Partial-height Gypusm with metal studs	ft	all
C3011.001b	Interior	Partial-height Gypusm + wallpaper	ft	all
C2011.001b	Stairs	Stairs	ea	all
C3027.002	Interior	Raised access flooring	ft^2	all
C3032.003a	Interior	Suspended Ceiling I	ft^2	all
C3032.003b	Interior	Suspended Ceiling II	ft^2	all
C3032.003c	Interior	Suspended Ceiling III	ft^2	all
C3032.003d	Interior	Suspended Ceiling IV	ft^2	all
C3034.002	Interior	Independent Pendant Lighting	ea	all
D1014.011	Elevators	Traction Elevator	ea	1
D2021.013a	Plumbing	Potable Water Piping Branches	ft	all
D2021.013b	Plumbing	Potable Water Piping Branches Braces	ft	all
D2021.023a	Plumbing	Potable Water Piping Mains	ft	all
D2021.023b	Plumbing	Potable Water Piping Mains Braces	ft	all
D2031.023a	Plumbing	Sanitary Waste Piping	ft	all
D2031.023b	Plumbing	Sanitary Waste Piping Braces	ft	all
D2051.023a	HVAC	HVAC Cooling Piping	ft	all
D2051.023b	HVAC	HVAC Cooling Piping Braces	ft	all
D3031.013c	HVAC	Chiller	ea	R
D3031.022c	HVAC	Cooling Tower	ea	R
D3041.001c	HVAC	HVAC In-line fans	ea	all
D3041.012c	HVAC	HVAC Duct Mains	ft	all
D3041.011c	HVAC	HVAC Duct Branches	ft	all
D3041.032c	HVAC	HVAC Duct Drops	ea	all
D3041.103c	HVAC	HVAC Fan	ea	R
D3052.013c	HVAC	HVAC Air Handling Unit	ea	R
D3067.012c	HVAC	HVAC Control Panel	ea	1
D4011.023a	Fire	Fire sprinkler Piping	ft	all
D4011.053a	Fire	Fire sprinkler Drops	ea	all
D5011.013c	Electrical	Transformer	ea	1
D5012.013d	Electrical	Motor Control Center	ea	1
D5012.023c	Electrical	Low Voltage Switchgear	ea	1
D5012.033c	Electrical	Distribution Channel	ea	all

5). Improvements to nonstructural components are applied on a global basis (i.e., components across all locations in the building are hardened equally.)

$$\theta'_{DS,j} = x_i \times \theta_{DS,j} \tag{4}$$

TABLE 3 50th percentile functional recovery time for the code-conforming office building, reported with and without impeding factors.

	50th percentile functional recovery time
repairs only	35 days
repairs + impeding factors	155 days

where: $\theta'_{DS,j}$ = updated median capacity for damage state j, post-strengthening
x_i = strengthening factor applied to all damage states of component i
$\theta_{DS,j}$ = original median capacity for damage state j

and

$$\mathbf{X_{NS}} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{bmatrix} \tag{5}$$

where N is the number of nonstructural component types.

The design space of each x_i in this example is meant to serve as a proof of concept only. The actual design space would likely vary from component to component, and would be limited by physical limits. Furthermore, scaling each component damage state by the same x_i value may not be realistic. In total, 36 nonstructural components (of the 41 components in Table 2) will be included in the scope of this enhancement. The first five are structural components, so they will be omitted.

4.1.4 | Feasibility assessment

Next, we perform the feasibility assessment on our selected strategy, by checking that T_{target} is met (i.e., the recovery gap has been eliminated) under the maximal enhancement allowed for our selected scope in (3), and under the remaining design criteria (D_{FR}, $Y\%$) in (2). An important consideration is that increases in I_e do not necessarily map to improvements in functional recovery performance (due to the amplification of floor accelerations, which increase nonstructural damage). Therefore, we decouple structural and nonstructural enhancements by performing three separate assessments, one at each I_e. For nonstructural components, maximal enhancement is achieved by setting all x_i to 3.00 (i.e., the upper bound defined in the design space), since there is a direct relationship between increasing x_i and reducing downtime.

We perform the assessment by first recording the (i) 50th percentile functional recovery time achieved during the baseline assessment, and (ii) the 50th percentile functional recovery time achieved under maximal enhancement. We perform the assessment for all $I_e \in \{1.00, 1.50, 2.00\}$, and find that an acceptable solution exists for each case.

4.1.5 | Optimization

Following the successful feasibility assessment, we perform the optimization stage for the case study building, using the design space and design criteria set in Sections 4.1.2, 4.1.3, respectively.

We judge the efficacy of individual design solutions using an objective function $f(\mathbf{X})$ (Equation 6) that measures aggregate nonstructural component improvements made to the original building. Here, N is equal to the number of nonstructural components in the performance model (Table 2). Setting up the objective function in this form (i) allows us to identify the most important nonstructural components that are critical to the recovery process (i.e., equal weight is placed on each component) and (ii) requires no additional data regarding costs of upgrades.

$$f(\mathbf{X}) = \frac{1}{N} \sum_{x_i \in \mathbf{X_{NS}}} x_i \tag{6}$$

s.t.:

$$T_{FR,50} \leq T_{target}$$
$$I_e \in \{1.00, 1.50, 2.00\}$$

$$x_i \in [1.00, 3.00] \ \forall \ x_i \in \mathbf{X_{NS}}$$

This simplified proof-of-concept function would be modified in a real-world application to account for the costs of upgrading individual components. A cost-weighted objective function would in general produce different optimal solutions, but this function utilizes the same computational workflow and illustrates the type of insights that this approach can produce.

However, this simplified objective function cannot mix structural and nonstructural enhancements (e.g., I_e and $\mathbf{x}$) since there is no common unit of measure. For this reason, $f(\mathbf{X})$ does not include I_e in the objective function, and hence, we perform independent optimizations using each $I_e \in \{1.00, 1.50, 2.00\}$. As a result, optimality is based on nonstructural improvements, but compared across the structural improvement design space.

Next, we select the algorithm to perform all subsequent analyses. We employ real-valued genetic algorithms[32], since population-based methods perform well in high-dimensional space, and are compatible with both surrogate- and simulation-based approaches (Figure 3), supporting comparative studies.

As mentioned in Section 2.5, the direct evaluation of the objective function using simulation-based procedures (Figure 3) is computationally expensive, and therefore limits our ability to explore optima across I_e and alternative recovery time targets. For this reason, we evaluate the objective function $f(\mathbf{X})$ using a surrogate model, developed in accordance with Section 3, whereby design variables $\mathbf{X}$ are directly mapped over to $T_{FR,50}$ (Figure 3b). The input feature vector contains all nonstructural component improvements in $\mathbf{X_{NS}}$ (Equation 5), in addition to a set of EDPs that capture changes made to I_e. We describe the architecture, development, and efficacy of the surrogate model in the next section.

4.2 | Development of surrogate model used in case study

In this section, we describe the development of the surrogate model used to perform the optimization in Section 4.1, following each step of the workflow presented in Section 3. We start with an overview of the surrogate model architecture, then describe the sampling protocol used to curate the dataset needed for training and testing. In the process, we study the relationship between data needs and model performance, and discuss strategies to reduce the number of examples required for training. Finally, we present the hyperparameters used to calibrate the final model, along with evaluation metrics that assess prediction performance.

We would like to emphasize that all decisions in the subsequent sections take place prior to running the analysis in Section 4.1.5.

4.2.1 | Define I-O and surrogate model architecture

We define the input and output of the surrogate model based on the scope of enhancements for the case study building, defined in Section 4.1.3. These improvements, encoded in the design vector $\mathbf{X}$, include both structural and nonstructural modifications in the form of I_e and x_i, respectively. The objective of our surrogate model is to map the design variables in $\mathbf{X}$ directly to the 50th percentile functional recovery time to be consistent with the design criteria set in Section 4.1.2.

As highlighted in Section 3.1, the input feature vector of the surrogate model is influenced by structural-nonstructural interaction, and whether we wish to maintain model generality. In our case study, generality is a high priority since we are interested in rerunning the surrogate model for a range of structural response intensities, as opposed to training a new model for each I_e variant (i.e., modifications to I_e would influence the response of the building). Therefore, we account for structural enhancements implicitly through the use of EDPs. The final surrogate model input feature vector is a 43×1 array consisting of 36 nonstructural component enhancements (Table 2), 3 story drift ratios (SDRs), and 4 peak floor accelerations (PFAs). The seven SDR and PFA values collectively make up the EDP values for the building. For the case study building, we use these inputs to evaluate the structural response (through EDPs obtained from nonlinear analysis) for each I_e value considered at the 475-year return period.

The output of the surrogate model is the (scalar) 50th percentile functional recovery time, $T_{FR,50}$. Given the highly nonlinear relationship between these inputs and recovery performance, we use a deep neural network to form a statistical relationship between the inputs and outputs. The final surrogate model architecture is illustrated in Figure 6.

4.2.2 | Design of computer experiments (DoCE)

Next, we define the DoCE, or sampling protocol, to generate a set of training examples. The objective of the DoCE is to capture as much information between the input feature vector and $T_{FR,50}$ as possible. Our proposed method generates training examples

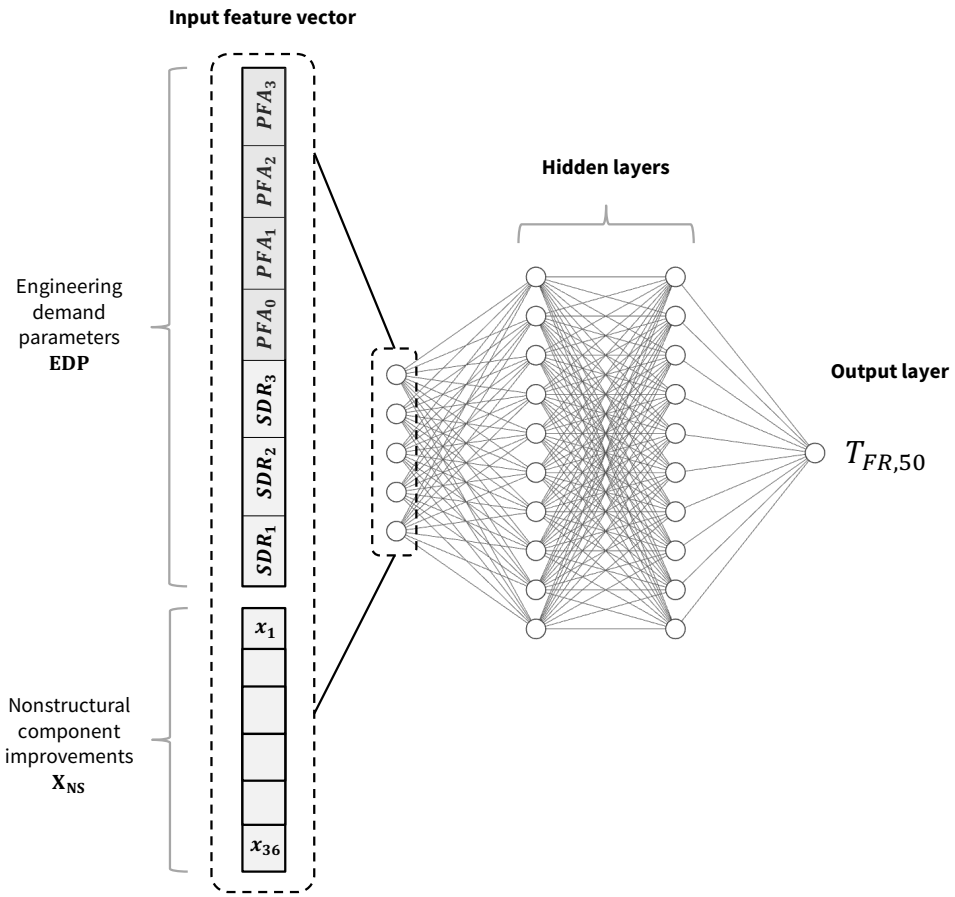


FIGURE 6 Surrogate model architecture, using a deep neural network.

using the combination of two sampling approaches: (1) a space-filling (random) approach and (2) a hand-selected approach. We handle **EDP** and $\mathbf{X_{NS}}$ differently. We describe each of these strategies in more detail.

- **Space-filling (random) sampling:** this sampling approach covers the input domain of design improvements uniformly, and with approximately constant sampling density. $\mathbf{X_{NS}}$ values are drawn from a uniform random distribution, using the upper and lower bounds defined in Section 4.1.3. **EDP** values, however, are sampled from a multivariate lognormal distribution, calibrated (using FEMA P-58 procedures) based on results from several hundred sets of EDPs generated via nonlinear dynamic analysis of the building. This is done since the domain is large, and there are known regions of greater interest (e.g., some combinations of EDPs are highly unlikely). This resulting distribution accounts for EDPs across various seismic intensity levels of interest and structural configurations (I_e). Correlations across all variables, and across the height of the building are maintained.

- **Hand-selected sampling:** here, $\mathbf{X_{NS}}$ and **EDP** are selected to teach the surrogate model about important lower-order interactions that are statistically difficult to achieve using random sampling. Since functionality in Cook et al.[11] is assessed using a fault-tree method, interactions between specific nonstructural component improvements are important to capture. In many cases, the recovery timeline of the building can only be improved under the influence of a particular combination of nonstructural improvements. The sampling protocols used as a part of this strategy are summarized in Table A1. In each case, the protocol attempts to provide information that will accelerate surrogate model learning when compared to solely using space-filling sampling.

Once the DoCE is complete, we generate each training example using a simulation-based approach (Figure 3a, steps 1-2) whereby the $\mathbf{X_{NS}}$ and **EDP** values in the input feature vector are used as inputs to a full downtime assessment (FEMA P-58, followed by Cook et al.). We leverage 75,000 realizations when generating each training example of $T_{FR,50}$ to limit the absolute

run-to-run dispersion to 1 day. Fewer realizations may be sufficient depending on the hazard level of interest, and the impeding factor distributions that are used in the downtime assessment.

Developing a training set can be a computationally expensive task on its own. However, the dataset will only be generated once, so gains in computational efficiency are unlocked for all subsequent optimization analyses, as demonstrated in Section 4.3. Furthermore, parallelizing this process across multiple CPUs (locally or via high-performance computing) can greatly reduce computational times.

Computational cost can be further reduced by carefully selecting (i) the number of training examples, and (ii) the ratio of random-to-hand selected cases in the training set. We demonstrate this by re-training the surrogate model on an increasingly large pool of training examples (ranging from 10,000 to 200,000 examples), across sets that contain smaller and larger proportions of hand-selected examples, ranging from 0 (no hand-selected, all random) to 0.5 (half hand-selected, half random). After we train each surrogate model, we evaluate predictive performance using the coefficient of determination, R^2 for: (1) the entire test set, (2) a subset of the test set where $T_{FR,50} < 30$ days. We show results from this analysis in Figure 7.

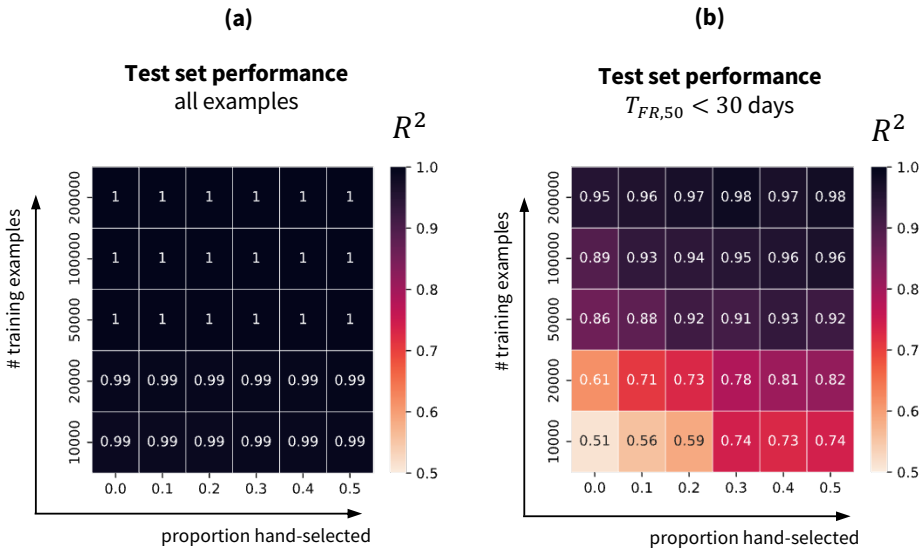


FIGURE 7 Assessing the predictive performance of the surrogate model for varying numbers of training examples and proportions of hand-selected cases. We report R^2 for (a) all examples in the test set, (b) examples where $T_{FR,50} < 30$ days.

Figure 7 shows that the performance of the surrogate model is high ($R^2 \geq 0.99$) when evaluated over the entire test set, even when data is limited. If we focus on the prediction performance for very low target times ($T_{FR,50} < 30$ days), however, significantly more training data is required to order to maintain a high coefficient of determination. This demonstrates that, depending on the design criteria (e.g., target times of interest) set in Stage 2 of the framework, more or fewer data may be needed to develop the surrogate model.

Furthermore, Figure 7 demonstrates that leveraging hand selected examples for 10% to 50% of the training set improves performance. The performance gains are most evident when the availability of training data is low, and for very low target times ($T_{FR,50} < 30$ days). At some point – beyond 200,000 examples – the benefit of including hand-selecting cases diminishes. Nonetheless, it is still recommended that some hand-selected cases are included as part of the sampling protocol. Doing so ensures that R^2 also reflects examples that test lower-order interactions between components and systems, as opposed to a purely randomized set. A trained model using only random data (no hand selected cases) requires substantially more data in order to capture lower-order effects (performance on hand-selected cases), particularly across $T_{FR,50} < 30$ (Figure 7b) examples.

Overall, we observe that maximum performance can be achieved when anywhere between 30% to 50% of the training set consists of hand-selected cases. In our final surrogate model, we leverage a one-third to two-thirds split between hand selected and random examples, respectively. Since we intend to perform the optimization for target times as low as 7 days (Figure 10),

additional data is necessary to maintain accuracy of $T_{FR,50}$ at or above $R^2 > 0.90$. In total, we use 400,000 examples to calibrate our model and randomly distribute this final set into a training, validation, and testing set using a 80-10-10 split.

4.2.3 | Hyperparameter calibration

Using the validation set developed in the previous step, we now tune the surrogate model to locate optimal hyperparameters. These hyperparameters include the number of layers, number of hidden units, and activation function. The optimal set of hyperparameters are influenced by the architecture and dataset defined in Sections 4.2.1 and 4.2.2, respectively, and can significantly improve the surrogate model's predictive performance. We perform the optimization using grid search, and tabulate the results in Table 4.

TABLE 4 Hyperparameters selected to calibrate the neural network.

Hyperparameter	Value
Number of layers	4
L2 Regularization term	0.005
Number of hidden units	150
Optimization algorithm	Adam
Activation function	ReLU

4.2.4 | Evaluation

Using the architecture, dataset, and hyperparameters defined in the previous steps, we evaluate the surrogate model for accuracy by calculating the coefficient of determination and the mean error on the test set (Figure 8). The mean error across the $T_{FR,50}$ space is calculated using a 7-day moving average.

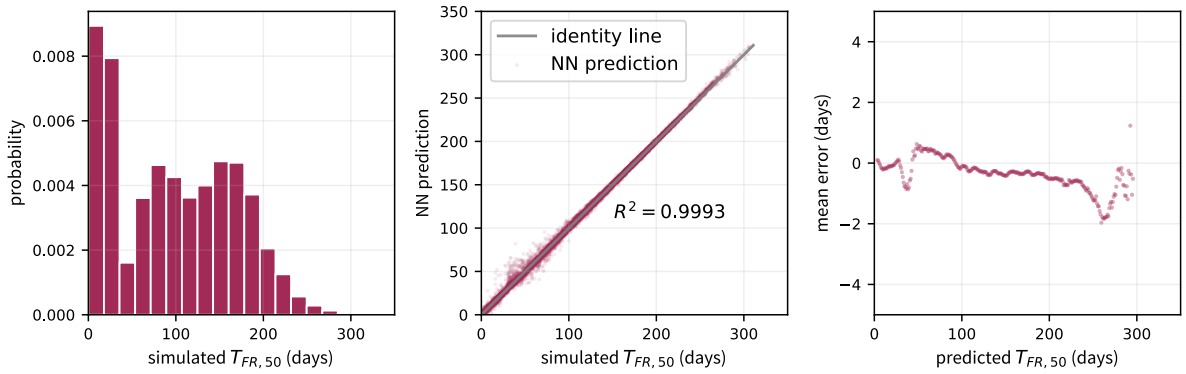


FIGURE 8 Distribution of $T_{FR,50}$ labels in the training set (left). Predictive performance of the final calibrated surrogate model, in terms of R^2 (middle) and a moving mean error, based on a seven-day window for averaging (right).

Since we use the surrogate model solely for the purposes of replacing the simulation-based downtime assessment (e.g., interpolation as opposed to extrapolation), high values of R^2 are desired. We find that, generally, $R^2 > 0.999$ across the test set. We also find that the surrogate model is able to replicate the $T_{FR,50}$ predictions with relatively low absolute error. During our experiments, the moving mean error never exceeds two days across the recovery time space. This result demonstrates that our optima in Section 4.1.5 are generated to a high degree of accuracy. We find that leveraging a surrogate model in place of traditional, simulation-based approaches to perform a single evaluation of $T_{FR,50}$ improves average computational times from 300s

to 0.0002s, respectively, on on a computer with 64GB RAM and Apple M1 Max processor. Note that this comparison does not include the time required to develop the surrogate model. Simulation-based evaluations of $T_{FR,50}$ are generated using 75,000 downtime realizations using the approach illustrated in Figure 2. Overall, this amounts to a computational time improvement of 1,500,000 times.

4.3 | Results

In this section, we present the results of the optimization for the case study building, using the setup and surrogate model defined in Sections 4.1, and 4.2, respectively. All results leverage the objective function and optimization algorithm defined in Stage 5 of the framework (Section 4.1.5). To illustrate the output from the optimization, we first present the results considering no structural improvement ($I_e = 1.00$). We then repeat this analysis across the structural improvement space to identify broader trends. We perform the optimization for a range of T_{target} values, since performance targets may vary and the efficient surrogate model enables rapid re-runs, but focus extra discussion on the 21 day target defined in Stage 2 of the framework. Finally, we discuss the results and limitations of our analysis.

4.3.1 | Optimization considering non-structural modifications

We perform the optimization using the design criteria set in Section 4.1.2, for $I_e = 1.00$ (no structural modification over the original design) across an ensemble of target times, ranging from 7 to 180 days, including impeding factors (Figure 9). For our original target time of $T_{target} = 21$ days at the 475-year return period, the recovery gap can be eliminated by modifying only 10 of the 36 nonstructural components; 7 of the 10 components involve a nonstructural enhancement of 50% or greater. The components requiring the most significant modifications are stairs (SDR-sensitive), glass curtain walls (SDR-sensitive), the air handling unit (PFA-sensitive), and precast concrete panels (SDR-sensitive).

Figure 9 also depicts how the optimum solution changes across a range of recovery time targets. Notably, the number of modified components depends on the recovery target time: only 3-4 components are necessary to achieve recovery targets between 60 and 180 days. For target times lower than 60 days, however, an increasing number of components are required to deliver the recovery performance (up to 20 in the case of $T_{target} = 7$ days). Underlying this observation is each component's relative contribution to downtime, and hence, its ability to reduce the repair schedule and impeding factors.

Relative importance can also be understood by observing the order in which certain components appear in the solution (starting from baseline recovery performance and lowering T_{target}). We observe that the first four components to appear are stairs, glass curtain walls, suspended ceiling type IV, and precast concrete panels. In addition, we find that while optimal nonstructural enhancements (in terms of x_i) increase monotonically for some components as the target time decreases (e.g., stairs, glass curtain walls), other component enhancements appear initially but then saturate (e.g., ceiling type IV, which never exceeds $x_i > 1.3$ across the spectrum of target times). This suggests that for some components, there is a strong potential to reduce downtime once strengthened, but this potential saturates as the target becomes increasingly ambitious.

4.3.2 | Optimization considering structural and non-structural modifications

We next enhance the analysis by performing the optimization for $I_e \in \{1.00, 1.50, 2.00\}$, using the design criteria set in Stage 2 (Section 4.1.2). To understand efficacy trends for structural modifications, we repeat the optimization for target times ranging from 7 to 180 days. The results and key trends are presented in Figure 10, where the objective function values are plotted across the range of target times considered. Recall that the objective function value $f(\mathbf{X})$ measures the average capacity increase across the ensemble of nonstructural components.

We observe that near the baseline performance of 155 days, $f(\mathbf{X})$ is equal to 1, indicating that no improvements are needed to achieve the target. At point (c), where $T_{target} = 120$ days, however, we find that if no structural modifications are made, the objective function rises above 1, and hence, the optimum solution requires some nonstructural improvements (Figure 10c). For higher values of I_e, the resulting objective function values remain at one (i.e., no nonstructural improvements are necessary). This observation, where higher I_e values maintain lower $f(\mathbf{X})$ values, is observed from $T_{target} = 180$ days to $T_{target} = 40$ days.

At roughly $T_{target} = 35$ days, however, we notice an inversion in this trend, whereby higher I_e values perform worse, with $I_e = 2.00$ optima requiring more nonstructural improvements on average than either $I_e = 1.50$ or $I_e = 1.00$. The optima at point (b), taken at our original T_{target} value of 21 days, provides some insight behind this shift. While the optimal solution for our design with no structural modifications (using $I_e = 1.00$) contains substantial improvements to SDR-sensitive components and

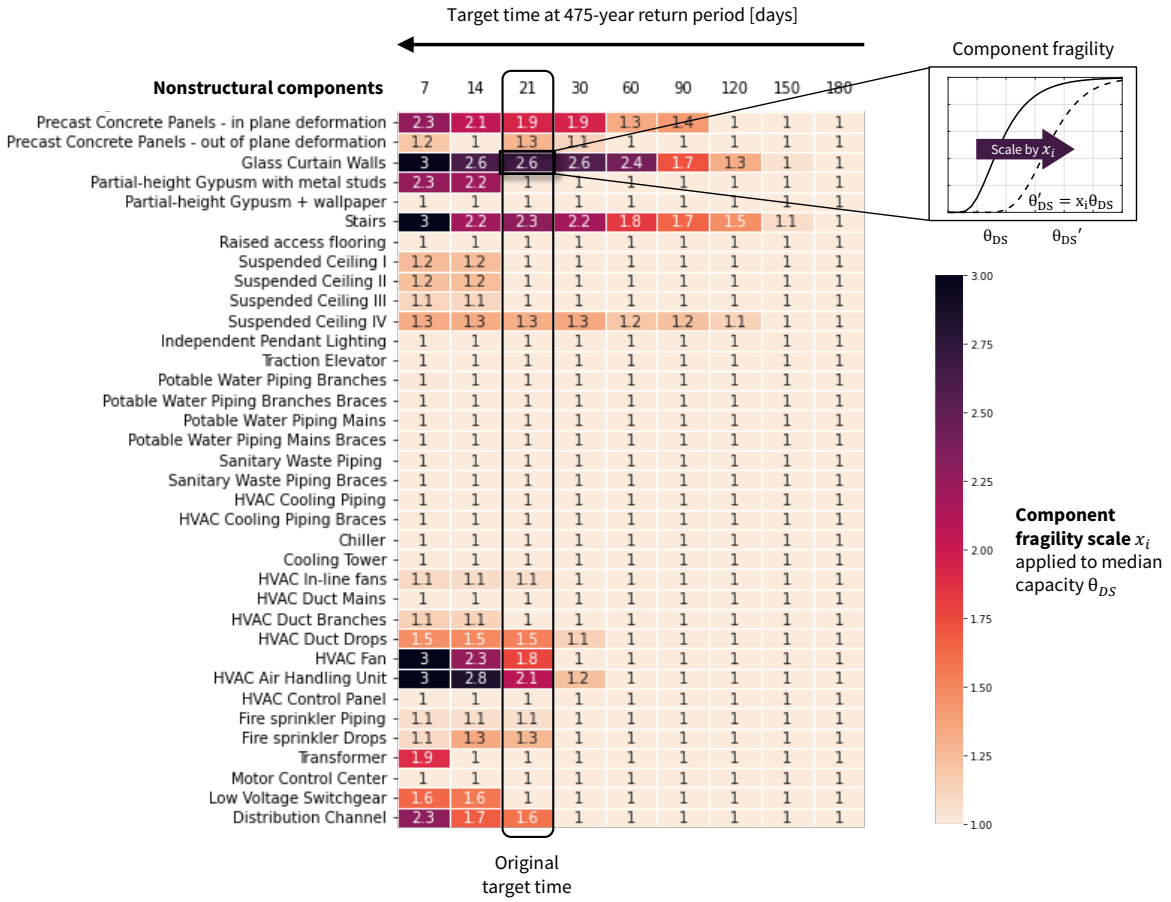


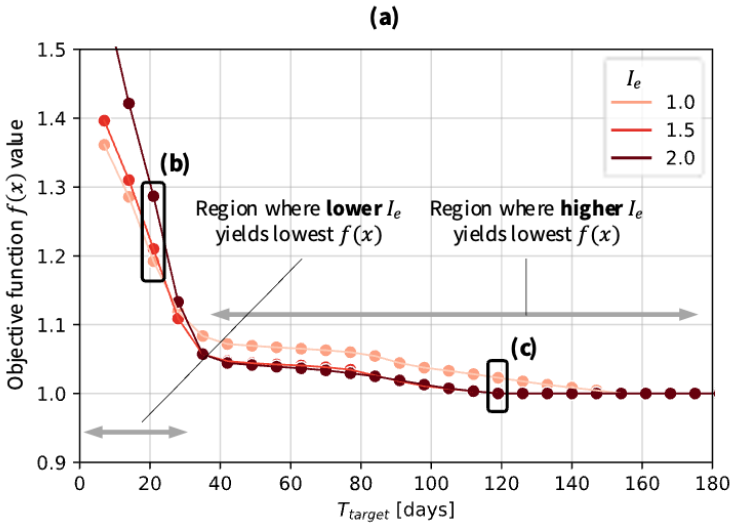
FIGURE 9 Optimal set of nonstructural component enhancements x_i for the case study building ($I_e = 1.00$). Here, T_{target} ranges from 7 days to 180 days, including impeding factors, at $D_{FR} = 475$ years and $Y = 50\%$. Each set of optima represents the best of $n = 20$ successive runs using the genetic algorithm.

a few PFA-sensitive components, the acceleration amplifications incurred by $I_e = 1.50$ and $I_e = 2.00$ necessitate more extensive strengthening to PFA-sensitive components, that outweigh the reduction in enhancement required for the SDR-sensitive components. From these results, we can deduce that at point (b), increasing I_e has a beneficial effect on SDR-sensitive components (in the form of lower x_i) until $I_e = 1.50$, at which point, the effects saturate. However, increasing I_e has a negative effect on PFA-sensitive components (in the form of higher x_i) across the entire I_e space.

We draw several conclusions from these results. First, the efficacy of structural modifications is contingent on the value of T_{target} set in Stage 2. For this case study, at higher T_{target} values, increases in I_e yield lower objective function values, and hence, require fewer nonstructural modifications. At the lowest target times, however, this observation is reversed, and lower values of I_e outperform. Second, regardless of I_e, optimal nonstructural component enhancement schemes at the lowest target times require substantial improvements to both SDR-sensitive components (e.g., stairs, curtain walls) and PFA-sensitive components (e.g., HVAC fan, duct drops, air handling unit). At higher recovery targets (> 120 days), however, using an I_e of 1.50 or greater eliminates the need to strengthen nonstructural components altogether.

4.3.3 | Limitations

Our case study serves as a proof-of concept to exercise the framework and demonstrate its feasibility, and is subject to a few limitations. First, the Equation 6 objective function (and resulting optima), is based exclusively on relative importance to recovery without consideration of the differential costs of enhancing various nonstructural components. In addition, we constrain the design space for all nonstructural components using a lower and upper bound of 1.00 (no change) and 3.00 (triple the median



(b)
T_{target} = 21 days

	I_e 1.00	I_e 1.50	I_e 2.00
Precast Concrete Panels - in plane deformation	1.9	1.6	1.4
Precast Concrete Panels - out of plane deformation	1.1	1.2	1.5
Glass Curtain Walls	2.7	2.2	1.9
Partial-height Gypusm with metal studs	1	1	1
Partial-height Gypusm + wallpaper	1	1	1
Stairs	2.2	1.8	1.8
Raised access flooring	1	1	1
Suspended Ceiling I	1	1	1
Suspended Ceiling II	1	1	1
Suspended Ceiling III	1	1	1
Suspended Ceiling IV	1.4	1.4	1.9
Independent Pendant Lighting	1	1	1
Traction Elevator	1	1	1
Potable Water Piping Branches	1	1	1
Potable Water Piping Branches Braces	1	1	1
Potable Water Piping Mains	1	1	1
Potable Water Piping Mains Braces	1	1	1
Sanitary Waste Piping	1	1.1	1.3
Sanitary Waste Piping Braces	1	1	1
HVAC Cooling Piping	1	1	1
HVAC Cooling Piping Braces	1	1	1
Chiller	1.1	1.2	1.3
Cooling Tower	1	1	1
HVAC In-line fans	1	1.1	1.4
HVAC Duct Mains	1	1	1
HVAC Duct Branches	1	1.4	1.7
HVAC Duct Drops	1.4	1.7	2.2
HVAC Fan	1.7	1.9	2.3
HVAC Air Handling Unit	2.2	2.4	2.8
HVAC Control Panel	1	1	1
Fire sprinkler Piping	1	1.1	1.4
Fire sprinkler Drops	1.1	1.2	1.5
Transformer	1	1	1
Motor Control Center	1	1	1
Low Voltage Switchgear	1.5	1.4	1.4
Distribution Channel	1.7	1.8	1.9

Component fragility scale x_i
applied to median capacity θ_{DS}

(c)
T_{target} = 120 days

	I_e 1.00	I_e 1.50	I_e 2.00
Precast Concrete Panels - in plane deformation	1	1	1
Precast Concrete Panels - out of plane deformation	1	1	1
Glass Curtain Walls	1.4	1	1
Partial-height Gypusm with metal studs	1	1	1
Partial-height Gypusm + wallpaper	1	1	1
Stairs	1.4	1	1
Raised access flooring	1	1	1
Suspended Ceiling I	1	1	1
Suspended Ceiling II	1	1	1
Suspended Ceiling III	1	1	1
Suspended Ceiling IV	1.1	1	1
Independent Pendant Lighting	1	1	1
Traction Elevator	1	1	1
Potable Water Piping Branches	1	1	1
Potable Water Piping Branches Braces	1	1	1
Potable Water Piping Mains	1	1	1
Potable Water Piping Mains Braces	1	1	1
Sanitary Waste Piping	1	1	1
Sanitary Waste Piping Braces	1	1	1
HVAC Cooling Piping	1	1	1
HVAC Cooling Piping Braces	1	1	1
Chiller	1	1	1
Cooling Tower	1	1	1
HVAC In-line fans	1	1	1
HVAC Duct Mains	1	1	1
HVAC Duct Branches	1	1	1
HVAC Duct Drops	1	1	1
HVAC Fan	1	1	1
HVAC Air Handling Unit	1	1	1
HVAC Control Panel	1	1	1
Fire sprinkler Piping	1	1	1
Fire sprinkler Drops	1	1	1
Transformer	1	1	1
Motor Control Center	1	1	1
Low Voltage Switchgear	1	1	1
Distribution Channel	1	1	1

Component fragility scale x_i
applied to median capacity θ_{DS}

FIGURE 10 Optimal objective function values (a) for the case study SMRF ($I_e = 1.00, 1.50, 2.00$). Here, T_{target} ranges from 7 days to 180 days, including impeding factors, at $D_{FR} = 475$ years and $Y = 50\%$. Optimal nonstructural enhancements x_i are shown at two points: $T_{target} = 21$ days (b) and $T_{target} = 120$ days (c). Each set of optima represents the best of $n = 20$ successive runs using the genetic algorithm.

capacity for all damage states), respectively. For real-world analyses, the upper bound will need to be modified for each component to reflect physical limits of improvement. The improvement values may also be discrete rather than continuous, reflecting discrete options for component specifications.

Furthermore, the surrogate model's input feature vector, as defined in Figure 6, contains only a single set of EDPs. This implies that any downtime estimates are conditioned on the user-provided EDP (i.e., there is uncertainty in the modeling but not the response), and it is up to the user to consider uncertainty by evaluating $T_{FR,50}$ (using the surrogate model) across a suite of EDP sets. For simplicity, we perform all optimization in this section using a single, representative set of EDPs (the median set across 40 nonlinear analyses).

It is important to note, however, that these limitations are specific to the application presented in this section, as opposed to the general methodology. Overall, the case study conveys the depth and breadth of the framework outputs that are enabled through surrogate modeling approaches. In a real-world application, additional data on design constraints and costs of enhancements can be used to modify the objective function and optimization constraints, without requiring other changes to the analysis workflow.

5 | CONCLUSIONS

This study proposed a framework to identify optimal design improvements to achieve building-specific recovery time objectives. The framework was partly enabled by a workflow to develop surrogate models that (i) rapidly estimate functional recovery performance under a range of user-defined improvements, and (ii) enable complex and informative optimization techniques for a range of stakeholder priorities.

The proposed optimization framework consists of five stages, beginning with a site-specific design in accordance with current code objectives, followed by a baseline assessment that quantifies the "at-code" recovery performance. Recovery-based design criteria are then set to define the target recovery time, hazard level, and acceptability threshold. In the process, the recovery gap, defined as the the difference between the baseline functional recovery and target functional recovery time, is computed. Next, the recovery-based design strategy and design space is defined, with the objective of achieving the target (i.e., closing the recovery gap). If the considered enhancements, under maximal enhancement, yield acceptable recovery performance, then the optimization is performed to identify the optimal solution in the design space that satisfies the user-defined design criteria.

A number of conclusions were made from the case study, which applied the framework to the early-stage conceptual design of a new low-rise office building. Leveraging a surrogate model in place of traditional, simulation-based approaches improved computational times (e.g., of computing $T_{FR,50}$ under parameterized design variables) by an order of over 1,500,000 times on a local computer. Results from repeating the optimization for different values of T_{target} revealed trends that would otherwise be difficult to extract from a single analysis. These trends include the influence of the selected design criteria (in the form of T_{target}) on the efficacy of structural modifications. In addition, we found that at the 475-year hazard level, (and in the absence of structural modifications) only four of the 36 nonstructural components required strengthening to halve the original 50th percentile functional recovery time. Such insights can support recovery-based design of individual projects, along with the development of general, prescriptive requirements as a function of downtime targets.

While most of the computational cost savings are made during the execution of the surrogate model, the case study also demonstrated that using hand-selected examples for 10% - 50% of the training set significantly reduced the required number of training examples, and hence, has the potential to reduce the computational cost of generating training data. Furthermore, the number of training examples also depended on the target time of interest – for the case study, achieving high predictive performance for low target times ($T_{FR,50} < 30$ days) required additional data.

Finally, the case study demonstrated that the proposed optimization framework is flexible and can accommodate various design modifications and design criteria. While our case study leveraged nonstructural and structural modifications through the use of x_i and I_e, respectively, these serve as a proof of concept and can be extended to consider additional engineering decisions. Since the efficacy of different engineering modifications will vary on a building- and site-specific basis, such flexibility will help enable the exploration of optimal solutions across different buildings.

ACKNOWLEDGMENTS

This work was supported by the National Science Foundation under NSF grant number CMMI-2053014, and by the Stanford Shah Family Fellowship on Catastrophic Risk. We thank Adam ZsarnÃşczay for providing valuable guidance on integrating

the PELICUN tool into the case study, Peter Lee for performing supporting analyses on gradient-based methods, and Emily Mongold for reviewing this manuscript. We thank Gregory Deierlein, Francisco Galvis, Peter Lee, Emily Mongold, Kuanshi Zhong, and Adam ZsarnÃşczay for their insightful comments and advice throughout the project.

How to cite this article: Issa O., Silva-Lopez. R, J. W. Baker, and H. V. Burton (2022), Machine-learning-based optimization framework to support recovery-based design, *Earthquake Engineering & Structural Dynamics*, *9999;99:1–9*.

APPENDIX

A CASE STUDY SURROGATE MODELLING

A.1 Hand-selected sampling protocols

TABLE A1 Sampling protocols used as part of the hand-selected DoCE strategy.

Sampling protocol name	Sampling protocol size	EDP sampling	x_i sampling
Baseline	Varies	Random*	All x_i held at 1.00 (no modification)
One component at a time	Fixed	Incrementally increased by decile**	OAT analysis: x_i incrementally increased by 10% from $x_i \in [1.00, 3.00]$. All other values held at 1.00 (no modification)
One component at a time (inverse)	Fixed	Incrementally increased by decile**	OAT analysis: x_i incrementally increased by 10% from $x_i \in [1.00, 3.00]$. All other values held at 3.00 (maximum modification)
One system at a time	Fixed	Incrementally increased by decile**	OAT analysis: all components within specific system have their x_i values incrementally increased by 10% from $x_i \in [1.00, 3.00]$. All other system components held at 1.00 (no modification)
One system at a time (inverse)	Fixed	Incrementally increased by decile**	OAT analysis: all components within specific system have their x_i values incrementally increased by 10% from $x_i \in [1.00, 3.00]$. All other system components held at 3.00 (maxmium modification)
Uniform scaling	Fixed	Incrementally increased by percentile**	All x_i simultaneously increased. All values incrementally increased by 10% from $x_i \in [1.00, 3.00]$.
Random EDP, Random within-system	Varies	Random*	Uniform random sampling of $x_i \in [1.00, 3.00]$, for components within specific system. All other system components held at 1.00 (no modification)
Random EDP, Random within-system (inverse)	Varies	Random*	Uniform random sampling of $x_i \in [1.00, 3.00]$, for components within specific system. All other system components held at 3.00 (max modification)
Random EDP, Random within-system, low-modifications	Varies	Random*	Uniform random sampling of $x_i \in [1.00, 1.50]$, for components within specific system. All other system components held at 1.00 (no modification)
Random EDP, Random within-system, low-modifications (inverse)	Varies	Random*	Uniform random sampling of $x_i \in [1.00, 1.50]$, for components within specific system. All other system components held at 3.00 (max modification)

*Set of examples randomly sampled from multivariate lognormal.

**Set of examples randomly sampled from multivariate lognormal. Sorted. Percentiles and deciles computed by ranking resulting EDP set by PFA_0.

References

1. FEMA . FEMA P-58: Seismic Performance Assessment of Buildings, Volume 5 – Expected Seismic Performance of Code-Conforming Buildings. 2018: 196.
2. EERI . Functional Recovery: A Conceptual Framework with Policy Options. 2019.
3. FEMA . FEMA P-2082: NEHRP Recommended Seismic Provisions for New Buildings and Other Structures, Volume II: Part 3 Resource Papers. 2020.
4. FEMA . FEMA P-2090: Recommended Options for Improving the Built Environment for Post-Earthquake Reoccupancy and Functional Recovery Time. 2021. doi: 10.6028/NIST.SP.1254
5. Mar D, Aher S. Affordable Resilience Using Rocking Walls and Foundation Dampers. *Proceedings of the 12th National Conference in Earthquake Engineering* 2022.

6. Stewart S, Hata O, Youssef N, Panagiotou M. Resilient Building Design. *Structure Magazine* 2022.

7. Forell/Elsesser Engineers I. UCSF Wayne and Gladys Valley Center for Vision. 2018.

8. Zimmerman RB, Herdrich B. Resilience-Based Design of a Seismically Isolated, Continued Operations Building. *Proceedings of the 12th National Conference in Earthquake Engineering* 2022.

9. Moehle J, Deierlein G. A Framework Methodology for Performance-Based Earthquake Engineering. *Proceedings of the 13th World Conference on Earthquake Engineering* 2004.

10. FEMA . FEMA P-58: Seismic Performance Assessment of Buildings, Volume 1 – Methodology, Second Edition. 2018.

11. Cook DT, Liel AB, Haselton CB, Koliou M. A framework for operationalizing the assessment of post-earthquake functional recovery of buildings. *Earthquake Spectra* 2022; 38(3): 1972–2007. doi: 10.1177/87552930221081538

12. Terzic V, Villaneuva PK, Saldana D, Yoo DY. F-Rec Framework: Novel Framework for Probabilistic Evaluation of Functional Recovery of Building Systems, PEER Report No. 2021/06. 2021. doi: 10.55461/DPBD8076

13. Molina Hutt C, Vahanvaty T, Kourehpaz P. An analytical framework to assess earthquake-induced downtime and model recovery of buildings. *Earthquake Spectra* 2022; 38(2): 1283–1320. Publisher: SAGE Publications Ltd STMdoi: 10.1177/87552930211060856

14. Almufti I, Willford M. REDi Rating System: Resilience-based Earthquake Design Initiative for the Next Generation of Buildings. 2013.

15. Comerio MC. Estimating Downtime in Loss Modeling. *Earthquake Spectra* 2006; 22(2): 349–365. doi: 10.1193/1.2191017

16. Rojas HA, Foley C, Pezeshk S. Risk-Based Seismic Design for Optimal Structural and Nonstructural System Performance. *Earthquake Spectra* 2011; 27(3): 857–880. doi: 10.1193/1.3609877

17. Saadat S, Camp CV, Pezeshk S. Seismic performance-based design optimization considering direct economic loss and direct social loss. *Engineering Structures* 2014; 76: 193–201. doi: 10.1016/j.engstruct.2014.07.008

18. Steneker P, Filiatrault A, Wiebe L, Konstantinidis D. Integrated StructuralâĂŞNonstructural Performance-Based Seismic Design and Retrofit Optimization of Buildings. *Journal of Structural Engineering* 2020; 146(8): 04020141. doi: 10.1061/(ASCE)ST.1943-541X.0002680

19. Steneker P, Wiebe L, Filiatrault A, Konstantinidis D. A framework for the rapid assessment of seismic upgrade viability using performance-based earthquake engineering. *Earthquake Spectra* 2022: 875529302110657. doi: 10.1177/87552930211065771

20. Burton HV, Lee JY, Moradi S, Dastmalchi S. *Multi-objective Performance-Based Design Optimization of a Controlled Rocking Steel Braced Frame System*: 243âĂŞ268; Singapore: Springer Singapore . 2019

21. Kochenderfer M, Wheeler T. *Algorithms for Optimization*. The MIT Press . 2019.

22. Tieleman T, Hinton G. Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. *COURSERA: Neural Networks for Machine Learning, 4, 26-30.* 2012.

23. Kingma DP, Ba J. Adam: A Method for Stochastic Optimization. *International Conference on Learning Representations 2015* 2017.

24. Burton H, Xu H, Yi Z. Design of computer experiments for developing seismic surrogate models. *Earthquake Spectra* 2022; 38(1): 384–406. doi: 10.1177/87552930211033309

25. Viana FAC, Gogu C, Goel T. Surrogate modeling: tricks that endured the test of time and some recent developments. *Structural and Multidisciplinary Optimization* 2021; 64(5): 2881–2908. doi: 10.1007/s00158-021-03001-2

26. American Society of Civil Engineers . *Minimum Design Loads and Associated Criteria for Buildings and Other Structures*. Reston, VA: American Society of Civil Engineers. 7 ed. 2017

27. Guan X, Burton H, Sabol T. Python-based computational platform to automate seismic design, nonlinear structural model construction and analysis of steel moment resisting frames. *Engineering Structures* 2020; 224: 111199. doi: 10.1016/j.engstruct.2020.111199

28. Baker JW, Lin T, Shahi SK, Jayaram N. New Ground Motion Selection Procedures and Selected Motions for the PEER Transportation Research Program. *PEER Technical Report 2011/03* 2011: 106.

29. McKenna, F , Fenves, G L , Scott, M H . Open system for earthquake engineering simulation. 2003.

30. FEMA . FEMA P-58: Seismic Performance Assessment of Buildings, Volume 2 – Implementation Guide, Second Edition. 2018.

31. ZsarnÃşczay A, Deierlein G. PELICUN âĂŞ A Computational Framework for Estimating Damage, Loss, and Community Resilience. *Proceedings of the 17th World Conference on Earthquake Engineering* 2020.

32. Deb K, Agrawal RB. Simulated binary crossover for continuous search space. *Complex Systems 9(2), 115âĂŞ148* 1995.