2018

A computational framework for infrastructure performance predictions based on data, mechanics, and machine learning

Yicheng Yang
Iowa State University

Recommended Citation

Yang, Yicheng, "A computational framework for infrastructure performance predictions based on data, mechanics, and machine learning" (2018). *Graduate Theses and Dissertations*. 16495.
https://lib.dr.iastate.edu/etd/16495

A computational framework for infrastructure performance predictions based on data, mechanics, and machine learning

by

Yicheng Yang

A thesis report submitted to the graduate faculty

in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

Major: Civil Engineering (Structural Engineering)

Program of Study Committee:
In Ho Cho, Major Professor
An Chen
Jin Tian

The student author, whose presentation of the scholarship herein was approved by the program of study committee, is solely responsible for the content of this thesis. The Graduate College will ensure this thesis is globally accessible and will not permit alterations after a degree is conferred.

Iowa State University

Ames, Iowa

2018

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Page

ACKNOWLEDGMENTS

I would like to thank my major professor, Dr. In Ho Cho, and my committee members Dr. An Chen, and Dr. Jin Tian, for their guidance and support throughout the course of this research.

In addition, I would also like to thank my friends, colleagues, the department faculty and staff for making my time at Iowa State University a wonderful experience. I want to also offer my appreciation to those who were willing to participate in my surveys and observations, without whom, this thesis would not have been possible.

ABSTRACT

Reinforced Concrete (RC) shear wall is one of the most important earthquake-resisting structures that is able to bear a horizontal shear force. The capacity curve and global stiffness reduction of reinforced shear walls are vital for understanding the properties and behaviors of the RC shear walls. Traditional approaches to obtain capacity curve are conducting experiments on shear walls or building finite element models to analyze them. However, these approaches are costly and time-consuming, especially conducting experiments. Meantime, degradation of core shear wall's flexural stiffness is vital to understand the natural frequency shift of the damaged shear walls. But it is hard to capture, often necessitating complex finite element analyses (FEAs). Therefore, this study seeks to provide efficient approaches to quickly obtain capacity curve using multi-target machine learning, and global stiffness reduction of U-shaped RC shear wall using cell network-based formulas. Importantly, both developed approaches are investigated to require only the easy-to-collect property information of shear walls. The acquirement of capacity curve and the remaining flexural stiffness of shear walls will help improve the quality of structural design.

The thesis is structured as follows. CHAPTER 1 introduces previous applications of machine learning in civil engineering domain and background of multi-target prediction model. CHAPTER 2 to CHAPTER 3 introduce the approach to predict capacity curve of shear wall using multi-target regression model. And CHAPTER 4 present computational implementation of cell network to predict remaining stiffness of shear wall. CHAPTER 3 and CHAPTER 5 separately illustrate limitation and future work of applications of multi-target prediction and cell network-based formulas.

CHAPTER 1. INTRODUCTION

1.1 Previous Machine Learning-Based Prediction of RC Structures

Reinforced Concrete (RC) shear walls, as a primary lateral load-resist system in structures, are able to resist lateral load effectively. Rectangular shear walls are the most commonly used shear walls in building designs. Better understanding of their material and properties are useful for designing. 3D finite element models are frequently built to simulate their behavior and make some predictions.

The geometry of RC shears walls is abundant (U-, L-, T-, shaped sections). These walls also play an important role on earthquake-resistant concrete buildings. For instance, U-shaped shear walls provide stiffness and resistance in both directions against earthquake energy waves. The behavior under seismic actions is more complex than that of rectangular shear walls. The behavior of shear wall under uniaxial loading and biaxial loading is not easily known, needless to say the quantity of observational and numerical studies. Once shear walls are damaged, their nonlinear behaviors are critical for designing. In the past decades, persistent efforts have been devoted to gain insights into nonlinear behaviors of rectangular RC walls [1]. However, stiffness reduction mechanisms of non-rectangular shear walls remain intractable when multi-axial loads are involved [2].

Plenty of machine learning based predictions of RC structures have been tried in civil domain. For instance, accurately estimating shear strength is a vital issue in engineering design and management. Many researchers have attempted to predict shear strength of RC deep beams. The least square support vector regression (LS-SVR) was built to predict shear strength in RC deep beams [3]. The model used the firefly algorithm (FA) instead of cross validation algorithm to optimize hyper parameters in support vector machine to boost final predictions.

FA was able to efficiently and simultaneously find both global and local optimations. The result proved that it had better perdition accuracy compared with traditional support vector machine. However, a disadvantage of the proposed model is that it works as a black box and it gives no insight to it. Most of civil engineers do not trust the accuracy of the prediction.

Also, Support Vector Regression (SVR) based modelling approach was used to predict the shear strength of reinforced and prestressed concrete deep beams [4]. To validate the result of SVM, back-propagation neural network and strut-and-tie model were developed to confirm the results. The choices of parameters of SVM had been studied though it still required a heuristic process. It indicated that SVR performed better compared with strut-and-tie model and American Concrete Institute (ACI) approach.

Besides shear strength of reinforced concrete structures, Aguilar [5] estimated in-plane shear strength of reinforced masonry (RM) using artificial neural network (ANN). The wall aspect ratio, the compressive strength of masonry, as well as details about reinforcement were taken into consideration as input variables. The ANN model was trained and tested by 285 RM walls from the literature. The back-propagation algorithm was chosen to train a multilayer network with onward connections. Regarding to the shear expression used in the algorithm, several expressions had been tested and the one from ACI Committee showed more accurate results [5]. All in all, the proposed method predicted experiment results in a more accurate and less conservative way. The predictions could be improved if adding more independencies between variables. Besides shear strength, compressive strength plays an important role on designing process.

The compressive strength of concrete is a highly nonlinear function of its ingredients. The predictions of compressive strength had been a popular issue in recent years. Six machine learning models were used to develop ensemble approach to predict high performance concrete compressive strength, including support vector machine, artificial neural network, classification and regression tree, and linear model [6]. Meantime, all the six individual models were built to compare with the predictions from ensemble learning. The result proved that ensemble approach had better performance than traditional regression models. To sum up, ensemble method significantly reduced the time required to test compressive strength in terms of traditional approaches. However, the choice of parameters required future study because paper [6] used default values.

Besides predictions of material properties in civil domain, the predictions of forces were also popular. In hope of improving the design of vertical structures, reliable predictions of horizontal forces will be very helpful. The horizontal force on vertical breakwaters has been predicted by neural network [8]. Though physical modelling was useful, due to the very complex interaction between force and structures, the derivation of reliable relations based on test was difficult to get. The NN would save time and cost to conduct the predictions efficiently. Gent [8] determined to use nine input variables, and one hidden layer with seven neutrons which were chosen by comparison test. The activation function to active output was sigmoid function. Because dataset used did not cover all possible situations, reliable intervals were developed to yield insight in the accuracy of the predictions.

Artificial Neural Network (ANN) had been widely used in civil domain because of incomplete and noisy data. A comprehensive report of presenting practical guidelines for designing ANN for engineering applications was made [9]. Three types of NN are discussed including multi-layer perceptron (MLP), radial basis network (RBF), and normalized RBF. And the paper described how to build NN in details. The discussion on selection of hidden layers, epochs and activation function was presented based on empirical experience and comparison. And an practical engineering application example was made to predict depth of the slab using MLP.

Besides NN, support vector machine (SVM) had been proved to be a powerful model on predicting horizontal force as well. The predictions of forces on vertical structures due to dynamic waves had been discussed [10]. The basic ideas underlying SVM were reviewed and potential problems had been addressed in the paper. Five kernel functions were investigated. And comparison of Root Mean Square Errors (RMSEs) of five kernel functions were made to choose suitable kernel function. The future issues on SVM had been addressed like determining best global parameters. And the relative performance of the SVM was compared with the results of previous study where NN had been applied to predict horizontal force on breakwaters [9].

Machine learning had been applied to damage identification of structural health monitoring [11]. There were four levels for damage identification including: 1. Detection of damage, 2 Position of the damage, 3. Estimate of the extent of the damage, 4. Prediction of the damage. In the paper, outlier analysis is used to detect the damages once their features selected. Outlier analysis calculates Mahalanobis Distance of targets. Then it would be judged as damaged if Mahalanobis Distance of the target is over a statistically calculated threshold. In level 3, neural network was being trained to indicate the severity of the damage. The training data came from six detectors implemented in the structure. The model was supplied with novelty indices at the input layer to predict the damage severity at the output layer.

The prediction of curve was rarely conducted in all domains, needless to say in civil domain. Predictions of soil-water characteristic curves (SWCC) using Genetic programming (GP) had been investigated [7]. Six input parameters were required and water content corresponding to suction was exported. A GP software was used to build the model in MATLAB. GP is a brand of genetic algorithm (GA) which learns the best “fit” computer programs by means of artificial evolution. However, the main difference between GP and GA is that GP creates equal length or unequal length computer programs as initialization. GA only creates equal length programs. In general, five steps were summarized for GP of SWCC:

1. Initialize a population of GPs
2. Select the program with the higher fitness from all randomly generated programs. The “fitness” is defined as differences between the obtained and expected water content.
3. Transfer the good “fit” program and replace the “loser” program.
4. Repeat step 2and 3 recursively until best “fit” program is developed which predicts water content properly.

As long as the best GP was found, plenty of water contents would be predicted. Then soil-water characteristic curve was fitted in to those points to get predicted water content curve. The proposed model based on GP was able to give a reasonable predicted curve without making any assumption or simplications.

All in all, the machine learning-based applications in civil domain focus on material properties and applied forces. At this point, one of my research topic- prediction of capacity curve of shear wall using multi-target regression model will be very useful and promising.

1.2 Predictive Clustering Trees

Multi-target regression model generates predictive cluster trees (hierarchical clustering). Before introduction of application of the model, background will be referred firstly. A decision tree is a tree shaped graph which uses a branching method to demonstrate every possible outcome of a decision from Wikipedia. It is especially used in data mining to simplify complex problems. A decision tree normally starts with a single node, which branches into all possible outcomes. Each of those outcomes will branch into other nodes which represent other possibilities. All nodes are usually represented by circles.

Multi-target regression has been implemented in the open source machine learning system CLUS which is a decision tree learner and rule learning system that works in the predictive clustering trees (PCTs) [12]. Namely, multi-target regression shares similar algorithm of PCTs. The PCTs are decision tree where the leaves do not contain classes and where each node as well as each leaf corresponds to a cluster. A cluster is a collection of points which are similar to each other compared with points out of the cluster. The similarity of points are defined as a dissimilarity function. And the partitioning of data points to clusters depends on a criterion function. And the PCTs can be constructed with a standard “top-down induction of decision trees” (TDIDT) algorithm [13]. There are two types of decision trees mostly implemented including “Top-Down” and “Bottom-Up” decision trees.. Top-Down decision tree shapes in a triangle whose root is up. The decision is taken at the top, and decisions are passed down the tree for implementation. Bottom-up decision tree shapes in a triangle with root down. Process decisions upward from the lowest levels. However, the multi-target prediction model does not generate classic regression trees or classification trees. It generates top-down hierarchical clustering to build a hierarchy of clusters. The pseudo algorithm of PCTs is shown in Table 1 [14]. The PCT function takes instances *I* as input to grow trees. An instance is a concrete occurrence of object during the running time of a program. In the paper, an instance represents a row of dataset. And function BT is the main method which searches for the best test that minimizes intra-cluster variance (and maximize inter-cluster variance) in the created cluster.

Table 1. Algorithm of PCTs

```
[1] $\boldsymbol{Function}\ PCT(Training\ instances\ I)$:
[2] $(t^* , p^*) = BT\ (I)$ ;
[3] $If\ t^* \neq none$
[4]     $for\ each\ I_k \in P^*$
[5]         $Tree_k = PCT(I_k)$ ;
[6] $\boldsymbol{return}\ node(t_k, Tree_k)$ ;
[7] $else\ if$
[8] $\boldsymbol{return}\ leaf(Prototype(I))$ ;

[9] $\boldsymbol{Function}\ BT(I)$:
[10] $p = partition\ induced\ on\ I\ by\ t$ ;
[11] $(t^*, p^*, h^*) = (None, 0.5, 0)$ ;
[12] $h = var(I) - \sum_{I_k \in p} \frac{|I_k|}{|I|} var(I_k)$ ;
[13]     $for\ each\ test$
[14]         $\boldsymbol{if}\ (\boldsymbol{h} > \boldsymbol{h}^*)$
[15]             $(t^*, p^*, h^*) = (t, p, h)$ ;
[16] $\boldsymbol{return}\ (t^*, p^*)$ ;
```

t is an attribute value test, p is partition induced on *I* by *t*. And *h* is the heuristic value of *t*. The superscript $*$ represents the best test, and its corresponding partition and heuristic value.

The function ***PCT*** in line [1] is the main function of the algorithm which grows the decision tree until stopping criteria is met. In the line [1], the algorithm takes training instances (training data) as input to train the model. The function ***BT*** is invoked in line [2] to search for the best test to partition training instances to hierarchical clusters regarding to intra-cluster variance. Before proceeding to line [3], Function ***BT*** is explained explicitly from line [9]. In line [11], the best test, and corresponding partition and heuristic value are initialized. The heuristic value is like a blind "common sense" guess of a best route out of the forest. Because the best test is unknown, the heuristic value of the best test is set as zero. The equation of heuristic value is defined at line [12]. Then the heuristic values of all possible tests to partition clusters are calculated in for loop in line [13]. The best test and partition will be chosen if its heuristic value is larger than initialized heuristic value of the best test in line [15]. As long as best test and partition are returned by function ***BT*** in line [2], invoke ***PCT*** function recursively to obtain tree and corresponding node in step *k* in line [5] and [6]. However, if best test is not found in line [7], namely, no test significantly reducing variance, then the algorithm will return a leaf and labeled as repetition of the training instances in line [8].

In order to introduce PCTs completely, it is unavoidable to mention a splitting criteria, a stopping criteria, as well as a pruning strategy.

Plenty of prevalent splitting criteria have been adopted successfully, i.e., Shannon entropy and Gain Ratio. The choice of splitting criteria is based on the purpose. The purpose of splitting clustering is to obtain clusters such that intra-cluster distance (the distance between examples belonging to different clusters) is as small as possible by splitting the nodes. For regression problems, intra-cluster distance is specified as intra-cluster variance. Given a cluster and a test which will result in partition of the cluster to decrease the variance, intra-cluster variance is defined as:

$$var = \sum_{i=1}^{N} d(x_i, \overline{x})^2 \tag{1}$$

Where $\overline{x}$ is the mean vector of the cluster, and x_i is an element in the cluster. N is the number of elements in the cluster. d stands for the Euclidean distance.

Growing trees without stopping criteria will lead to overfitting problem. Normally, a x^2-test is applied to check whether the class distribution in the sub-clusters differs significantly. Since regression problem use intra-cluster variance as heuristic for choosing the best split, then a reasonable stopping criteria is a F-test which checks whether variance decrease significantly if a test is found. Moreover, if no acceptable test is found, namely, if no test significantly reduces variance, the algorithm creates a leaf labeled as prototype of the instances and the growing is stopped.

Pruning strategy is a technique to remove parts of the tree that provide little power to identify instances. Growing clusters without reduction of its size will result in overfitting problem. An appropriate pruning strategy will decrease complexity of the model and improve accuracy of the final predictions. The strategy to prune trees is very straight-forward. If the quality of the tree is larger than the quality of the unpruned tree., then the tree is pruned. However, pruned strategy is not used in the thesis because of the small size of the database. A lot of pruning due to random influences which is not meaningful will be done if dataset is small.

1.3 Clus

In my thesis, the code of multi-target regression model is not developed by myself. It has been implemented in CLUS already. Clus, as an open source machine learning system, is a decision tree and rule learning system that works in PCT [12]. It shares similar algorithm of PCTs and please refer to section 1.2 for details. It is a Java based platform to build both classification and regression trees by choosing different parameter settings in Clus. It has been successfully applied to plenty of tasks including multi-target regression and classification, structured output learning, time series prediction and so on. In the thesis, the multiple-target regression problem is addressed and Clus displays powerful performance on the problem. Clus provides plenty of choices for parameter settings. In this section, parameter settings related to multiple-target regression are emphasized. Please refer to Clus manual for additional settings. Three files are prepared to perform the predictions. An input file with training data stored, an test data file with test data stored and a setting file which indicates all the parameter settings. In the setting file, location of training data and test data should be listed. And descriptive attributes and target attributes in dataset should be specified clearly. And other functions including ensemble method will be addressed as well. After running the model, an output file will be generated which contains predictions for target attributes specified in the setting file.

1.4 Ensemble Method

Ensemble learning has been last approach to boost prediction accuracy of the model. It generates an ensemble of classifiers whose individual decisions are combined in some way to classify new examples. Combining predictions of an ensemble is often more accurate than that of the individual classifiers (especially for independent classifiers). The general procedures for ensemble method are defined as:

Table 2. Pseudo Code of Ensemble Method

```
[1] Let $T$ denotes the original training data, $n$ denotes number of base classifiers,
    and $D$ be the test data.
[2] for $i = 1$ to $n$ do
[3]     Create training set $T_i$ from $T$
[4]     Build base classifiers $C_i$ with $T_i$
[5] end
[6] for each test record $x \in D$ do
[7]     $C_{new}(x) = vote(C_1(x), C_2(x), C_3(x), \ldots, C_n)$
[8] end
```

Create *n* sets of training data $T_1, \ldots, T_n$ by percentage defined by user in line [3]. And use sets of training data to train *n* base models $C_1, \ldots, C_n$. Then collect predictions of all the models and apply majority vote to generate final prediction in line [7]. Majority Vote is especially used in classification problems. It selects the prediction of most occurrence as the final prediction. Various of approaches have been successfully applied to construct ensemble learning. The popular ones are Bootstrap Aggregation (Bagging), Boosting, and random forests. In the thesis, random forests are implemented because PCT is a tree based classifier [15]. It shares the same general procedures with all ensemble methods shown in Table 2. The general procedures to build random forests are shown in Figure 1.

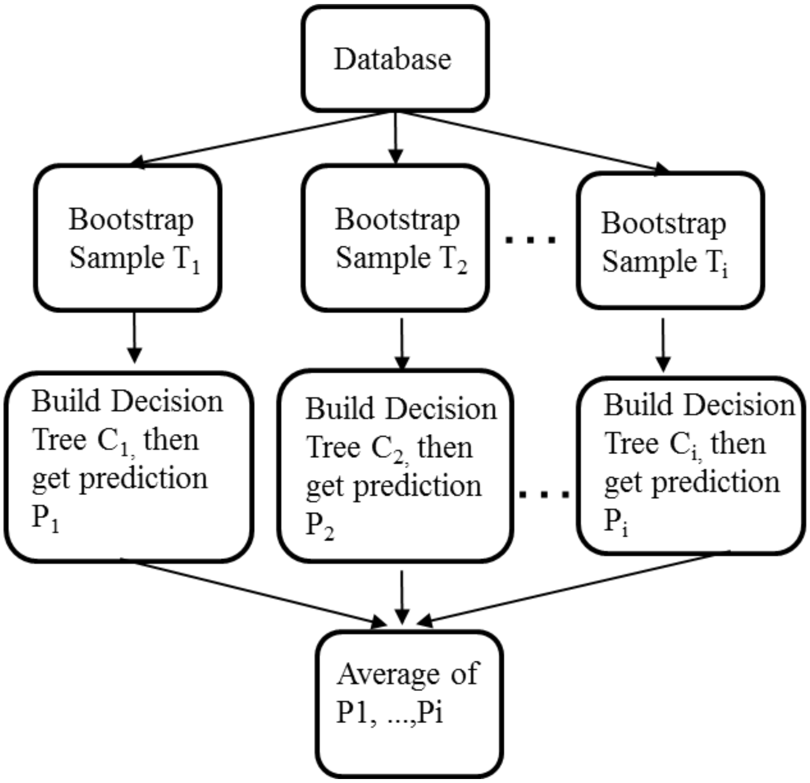


Figure 1: Flow Chart of Random Forest

1. Subsets training data T to *i* bootstrap samples $T_1, \ldots, T_i$.in Step 1.

2. Build i decision trees $C_1, \ldots, C_i$ with corresponding T_i in Step 2. At each node, variables are selected at random out of all the features and the best split on these variables are used to split the node. Each tree is growing to the largest extent possible without pruning.

3. Perform prediction with test data for each tree C_i in Step 3, and finalize predictions with majority vote if it is a classification problem in Step 4. However, in the paper final prediction will be average of P_1, P_2… P_i (P_i is prediction from decision tree C_i) because it is a regression problem.

In my thesis, random forests has been employed as ensemble learning method to cooperate with multi-target prediction model (hierarchical clustering trees). There is no doubt that random forests is not a clustering technique. It cooperates with PCTs mainly in two aspects. Firstly generate collection of PCTs by bagging of random forests. Secondly, it randomly pick attributes for function *BT* in Table 1 to find out best test to partition the cluster. Therefore, all PCTs generated by bagging will be different from each other.

1.5 Rule Ensembles for Multi-Target Regression

Rule learning is the most expressive and human readable model representation. It is a conjunction of statements, which concerns input variables. In the section, how rule ensembles been used to interpret multi-target regression model is illustrated. The algorithm to achieve rule ensembles of multi-target regression is shown in Table 3 [30]:

Table 3: Algorithm of Rule Ensembles

```
[1] $\boldsymbol{GenerateSetOfPCTs}(I)$:
[2] $\boldsymbol{return}\ T$;
[3] $ConvertPCTsToRules(T)$:
[4] $return\ R$;
[5] $OptimizeWeights(R, I)$:
[6]         $If\ (weight\ of\ r \in R\ =\ 0)$
[7]                 $remove\ r$;
[8] $\boldsymbol{return}\ (R, W)$;
```

I is training instances, and *T* is a collection of predictive clustering trees. *R* and *W* represent the collection of. rules generated from *T* and their corresponding weights. In line 1, recursively call function PCT in Table 1 to generate bagging of predictive cluster trees, then return a collection of PCTs in line 2. Moreover, such large ensembles of PCTs are impossible to interpret, therefore all the trees are transcribed to sets of rules in line 3 [31]. In line 5, find optimized weight for each of those rules R. During the process, it is trying to assign as many weights as possible to zero, in purpose of learning small and interpretable trees. All the weights are optimized by a gradient directed optimization algorithm [30]. The physical meaning of weights indicates the importance of each rule contributing to final prediction. In line 6 and 7, remove the trees if their optimal weights are zero. Finally in line 8, collection of rules whose weights are not zero, and their weights will be returned.

As long as rules and weights returned, final prediction can be calculated by following equation:

$$\hat{y} = w_0 avg + \sum_{i=1}^{M} w_i r_i(x) \tag{2}$$

Where w_0 is the baseline predication, part (*avg*) is a constant vector with the averages over all the targets. Each r_i is a vector function from *R* which gives out a constant prediction shown in Figure 2 as an example. In my thesis, there are eight target variables. That is why the rules in Figure 2 give prediction vectors of size eight. And w_i is corresponding weight of the rule. M indicates the number of rule ensembles in total.

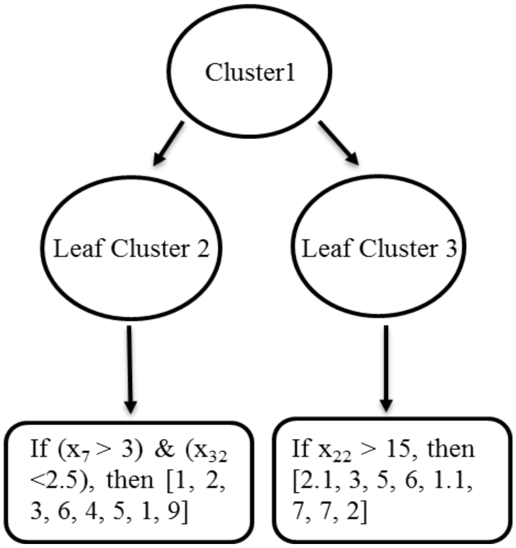


Figure 2 : Tool Example of Rule Ensembles

CHAPTER 2. PREDICTION OF CAPACITY CURVE BASED ON DATA AND MACHINE LEARNING

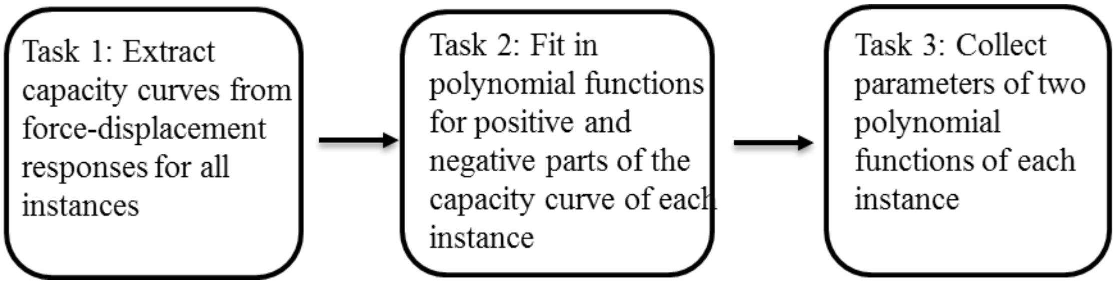


Figure 3: Data Collection Flowchart

2.1 Collection of the Capacity Curve Database

A rectangular shear wall database has been built upon simulation results of VEEL. Before proceeding toward the process of collecting data, the validation of VEEL is necessary. The validation will be provided explicitly in Chapter 4. VEEL as the FEA tool is rooted in a number of microscopic mechanisms: a multi-directional smeared crack model for nonlinear concrete, a topological information-based steel bar model (dubbed as “smart” bar by Cho, 2013) for progressive bar buckling, a random particle-based 3D interlocking model for 3D nonlinear shear, and a bar-concrete proximity-based model for general confinement effect. For more background of VEEL, the relevant papers [16] and [17] should be referred. The original database has global force-displacement responses of seven rectangular shear walls (RW1, WSH1, WSH2, WSH3, WSH4, WSH5, and WSH6) simulated by VEEL, adopted with ascending axial force ratio, yield stress, diameter of vertical reinforcement, and concrete compressive strength.

The complete procedures to build the new database are visualized in Figure 3. In order to extract skeleton curve from responses in Task 1, outermost points required to be selected. Most of outermost points are related to maximum and minimum of shear wall’s cyclic displacement.

Forty-six points in total are extracted from a force-displacement response from original database as shown in Figure 4. And a skeleton curve is fitted in regarding to these points. The choice of number of points selected is empirical. More points will definitely improve the accuracy of fitted skeleton curve.

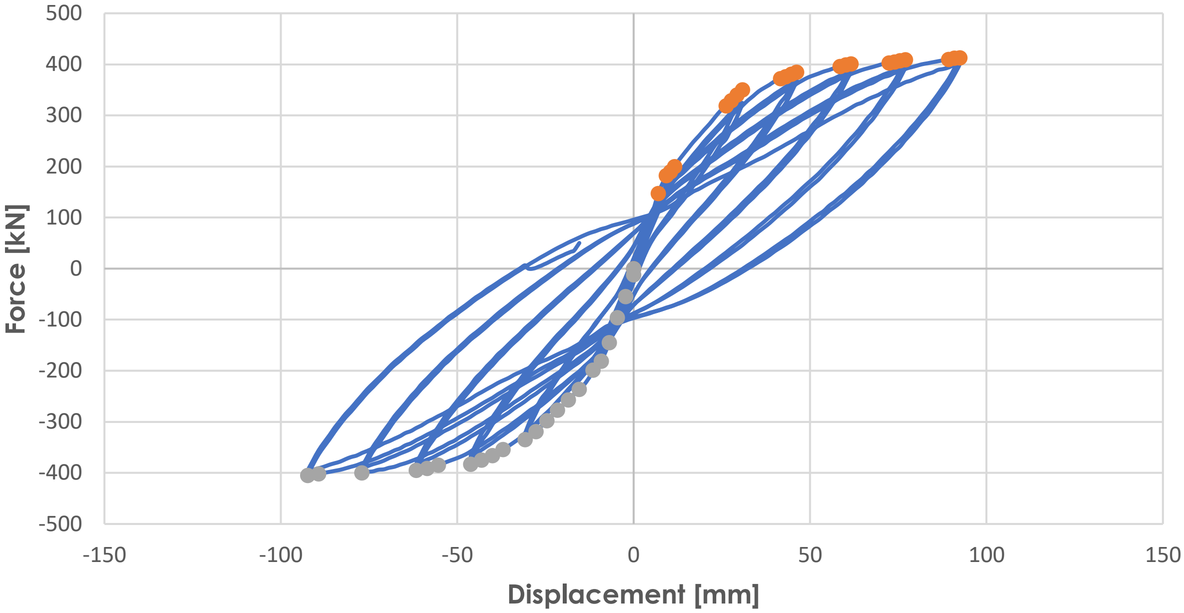


Figure 4: 46 Outermost Points

In order to fit in polynomial functions, the most popular approaches to construct a mathematical function that has the best fit to a series of data points are least squares and maximum likelihood. The methodology of the least squares is minimizing the sum of squared residuals (the difference between an observed value and the fitted value provided by a model). The fit of a model to data points is measured as residuals (denoted as $r_i \in \mathbb{R}, i = 1, \dots, n$) which is defined as

$$\boldsymbol{r_i} = \boldsymbol{y_i} - \boldsymbol{X\beta} \tag{3}$$

where adjustable parameters held in $\boldsymbol{\beta} \in \mathbb{R}^8$. Then the least squares finds the optimal parameters by minimizing the sum of squared residuals:

$$\boldsymbol{S} = \sum_{\boldsymbol{i}=\boldsymbol{1}}^{\boldsymbol{n}} \|\boldsymbol{y} - \boldsymbol{X\beta}\|^2 \tag{4}$$

As it is expected, forty-six points are enough to describe the curve explicitly. Then two polynomial functions are fitted in to describe positive part and negative part of the skeleton curve separately.

$$\boldsymbol{\beta} = \left[\boldsymbol{\beta}_p : \boldsymbol{\beta}_n\right] \tag{5}$$

$$\boldsymbol{\beta}_p \in \mathbb{R}^4, \boldsymbol{\beta}_p = \{P_1, P_2, P_3, P_4\}^T \tag{6}$$

$$\boldsymbol{\beta}_n \in \mathbb{R}^4, \boldsymbol{\beta}_n = \{N_1, N_2, N_3, N_4\}^T \tag{7}$$

To use the least squares,

$$\widehat{\boldsymbol{\beta}}_{\boldsymbol{p}} = \underset{\boldsymbol{\beta}_{\boldsymbol{p}}}{\operatorname{argmin}} \left\|\boldsymbol{y} - \mathbf{X}\boldsymbol{\beta}_p\right\|^{\mathbf{2}}, \text{for } y_i \in \mathbb{R}^+ \tag{8}$$

$$\widehat{\boldsymbol{\beta}}_{\boldsymbol{n}} = \underset{\boldsymbol{\beta}_{\boldsymbol{N}}}{\operatorname{argmin}} \|\boldsymbol{y} - \mathbf{X}\boldsymbol{\beta}_n\|^{\mathbf{2}}, \text{for } y_i \in \mathbb{R}^- \tag{9}$$

X is the model matrix of dimension $\mathbf{X} \in \mathbb{R}^{\boldsymbol{N}\times 4}$ where the ith row of **X** is given by $X_i = \{x_i^4, x_i^3, x_i^2, x_i\},\ i = 1, \dots n.$

The order of two polynomial functions is determined to be forth through the original point shown below:

$$y_i = P_1 x_i^4 + P_2 x_i^3 + P_3 x_i^2 + P_4 x_i, \qquad \text{for } y_i \in \mathbb{R}^+ \tag{10}$$

where $P_1, P_2, P_3, and\ P_4$ are parameters of polynomial function for positive part of the curve.

$$y_i = N_1 x_i^4 + N_2 x_i^3 + N_3 x_i^2 + N_4 x_i, \qquad \text{for } y_i \in \mathbb{R}^- \tag{11}$$

where $N_1, N_2, N_3, and\ N_4$ are parameters of polynomial function for negative part of the curve.

Fitting in the skeleton curve with a single polynomial function has been tested but a low accuracy, even higher order polynomial function will not help. In the thesis, the non-linear least squares is employed to estimate parameters of a non-linear function. In addition, R-squared value which is a statistic measure of how close the data are to the fitted regression line is used to evaluate the fitted polynomial function. In the database, R-squared values of most fitted functions are more than 0.99 which are accurate enough.

Eventually, eight parameters $\widehat{\boldsymbol{\beta}} = [\widehat{\boldsymbol{\beta}}_p : \widehat{\boldsymbol{\beta}}_n] = \{\hat{P}_1, \hat{P}_2, \hat{P}_3, \hat{P}_4, \hat{N}_1, \hat{N}_2, \hat{N}_3, \hat{N}_4\}^{\mathrm{T}}$ will be recorded to form new database. To sum up, 32 descriptive variables and 8 target variables are included in the finalized database. Detailed variable information is shown in APPENDIX A. The size of the new shear wall database is 40×182 (182 instances with 40 attributes).

2.2 Multi-Target Prediction of Capacity Curve

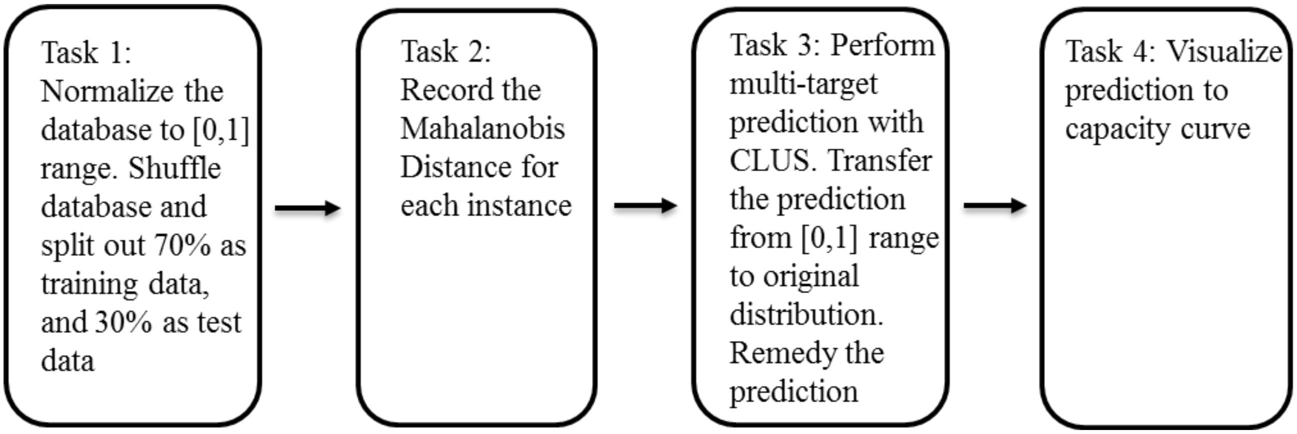


Figure 5: Multi-target Prediction Flowchart

Decision trees are widely used in the classification problems and single target regressions. However, PCTs considers a trees as a hierarchy of clusters respect to plenty of observable properties. Therefore, PCTs allows to build trees to predict multiple target attributes at the same time. In the multi-target prediction task, we are given a set of training data in the form $(descriptive\ attrabutes\ D, target\ attributes\ T)$, where D is a matrix of 32 descriptive attributes, $D = \{d_1, \dots, d_{32}\}$. And T is a matrix of 8 target attributes, $= \{\hat{P}_1, \hat{P}_{2,} \hat{P}_3, \hat{P}_4, \hat{N}_1, \hat{N}_2, \hat{N}_3, \hat{N}_4\}$. And the purpose of the task is to predict new T with a given new D. Multi-target regression has been implemented in the open source machine learning system CLUS. Please refer to Clus manual [12] for detailed implement of CLUS.

The training process of multi-target model implicitly captures the influence of descriptive attributes on target attributes. The distribution of database will play an important role on the predictions of multi-target model. Regarding the distribution of the database, numeric attributes span a long range [0.01, 2.23×10^9] which is very tough for the model to control. In order to make database easier to be controlled by the model, normalization is an unavoidable process. The normalization by standard deviation and 'Min-max normalization' are considered in the thesis. 'Min-max normalization' will rescale dataset between 0 and 1 using following equation:

$$x_{new} = \frac{x_i - x_{min}}{x_{max} - x_{min}} \tag{12}$$

where *i = 1…n (number of instances),* x_{min} and x_{max} are minimum and maximum value for an attribute respectively.

Normalization by standard deviation transforms dataset to have a mean of zero and a standard deviation of 1 with the following formula:

$$x_{new} = \frac{x - \bar{x}}{s} \tag{13}$$

where $\bar{x}$ is the data mean of the attribute, and s is the data standard deviation of the same attribute. In Task 1, "Min-max normalization" is implemented. Moreover, the x_{max} and x_{min} for each attribute will be stored for future use in Task 5.

The rectangular shear wall database is arranged by a certain order at the beginning. Data derived from the same shear wall will stay in lines. In purpose of facilitating learning process of the multi-target model, shuffling the order of dataset is necessary in Task 2. Then 70% of the dataset will be split out as training data for the model learning. The rest of them will be utilized as test data to evaluate the performance of the multi-target model. No validation set is created because of the small size.

Then, perform the multi-target prediction via Clus in Task 3. Two types of prediction results are generated including original predictions and pruned predictions. Only original predictions are considered because the pruned prediction works better for very large training data. Also, random forests are used as an ensemble learning method.

Plenty of prediction accuracy measurements have been employed in machine learning domain. The Mean Absolute Error (MAE) in percentage has been one of reliable measures of prediction accuracy. It expresses accuracy as a percentage with the formula:

$$MAE = \frac{100}{n}\sum_{t=1}^{n}\left|\frac{A_t - P_t}{A_t}\right| \tag{14}$$

where A_t is the actual value and P_t is the predicted value. And *n* is number of sample instances. In the thesis, the overall MAP of all target attributes is calculated as:

$$MAE_{new} = \frac{1}{m}\sum_{i=1}^{m} MAE_i \tag{15}$$

where *m* is number of target attributes.

Because of the size of training data, the multi-target model may not able to learn the rules behind the data deeply. And it will gives out weird predictions far from what they should be. These wrong predictions will significantly reduce the prediction accuracy of the whole test data. However, the obvious false predictions are very easy to pick out because they are ridiculously far from the range of the real data. In order to promote prediction accuracy more, the remedies about filtrating these indistinctive false prediction are came up with.

Mahalanobis distance (denoted as D_M) measures the distance between a point and a distribution (P.C. Mahalanobis, 1936). In the thesis, it refers to how many standard deviations away is from the mean of data distribution because of multi-dimensional problem. It is defined as:

$$D_M(\boldsymbol{x}) = \sqrt{(\boldsymbol{x}-\boldsymbol{\mu})^T \boldsymbol{S}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})} \tag{16}$$

where observation $\boldsymbol{x} = \{d_1, d_2, \ldots, d_{32}\}^T$, mean $\boldsymbol{\mu} = \{\mu_1, \mu_2, \ldots, \mu_{32}\}^T$ of $\boldsymbol{x}$, and $\boldsymbol{S}$ is covariance matrix. In the thesis, the calculated covariance matrix is not being non-singular. Instead, an inverse of the covariance matrix is used to calculate Mahalanobis Distance.

The core of the remedy refers to Mahalanobis distance calculated for all instances. Then a boundary defined by calculated Mahalanobis distance is determined to further filtrate indistinct predictions. The predictions with Mahalanobis distance out of boundary will be picked out in Task 6. And the remaining predictions will be reliable ones.

The radar plot of Mahalanobis Distances of 182 instances are calculated and plotted in Figure 6. Please refer to [28] for details of each wall index. In order to see how remedy strategy works regarding to Mahalanobis Distance, two relatively bad prediction (Red marks in Figure 6) instances and two good prediction (Black marks in Figure 6) instances are marked in the form of (Label, Wall Index, Mahalanobis Distance). And corresponding predicted skeleton curves of these four wall instances are shown in Figure 7. It is able to see that (a) wall index 20 has the worst performance out of four samples. And (b) wall index 4 predicts good trend of curve. However, they don't describe the outermost curve accurately. (c) Wall Index 67 and (d) 88 have satisfied performance on predictions. It is able to see that the accuracy of the prediction decreases as instance away from mean of data distribution

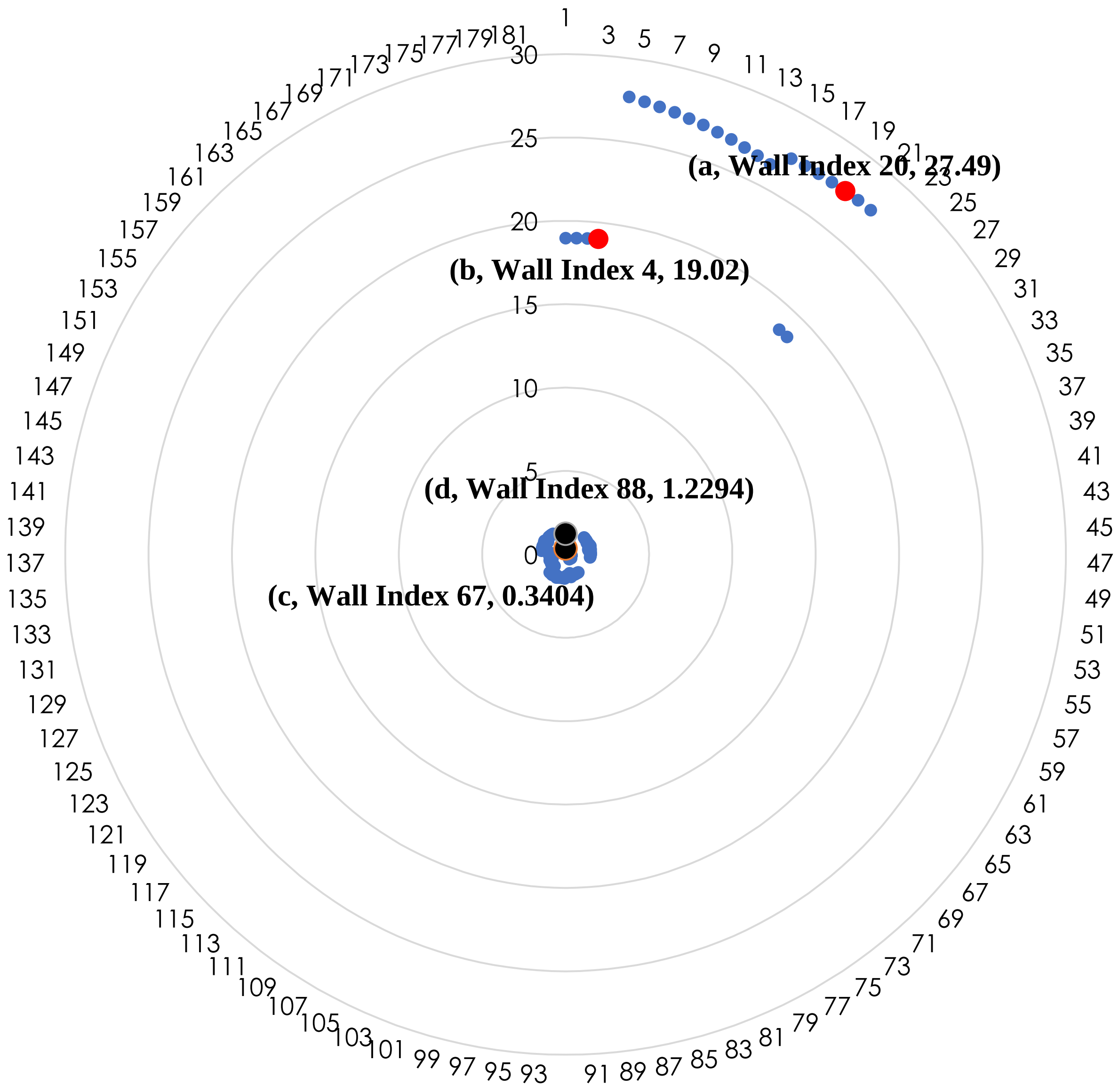


Figure 6: Mahalanobis Distances of 182 Instances

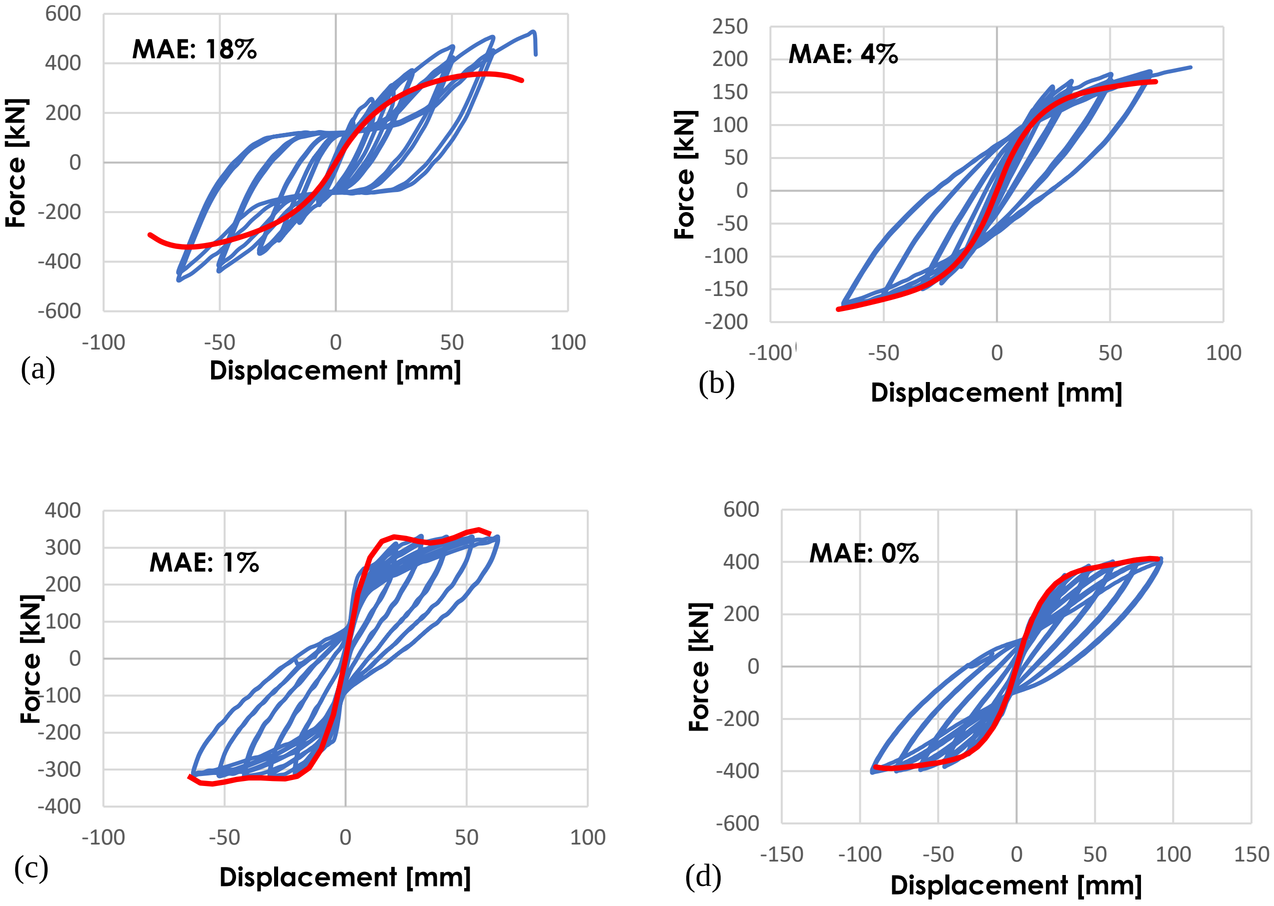


Figure 7: Wall Index: (a) Wall Index 20 (b) Wall Index 4 (c) Wall Index 67 (d) Wall Index 88

The prediction will not be utilized directly because of normalization at beginning. To visualize predictions, the predicted parameters will be transferred back to polynomial functions by equation:

$$x_{final} = x_{pred} \times (x_{max} - x_{min}) + x_{min} \tag{17}$$

where x_{final} represents final predicted parameter after transferring, x_{pred} is predicted parameter, and x_{max}, x_{min} are maximum and minimum of corresponding attributes recorded at Task 1. And final predicted force-displacement curve is plotted to compare with real force-displacement curve in Figure 7.

2.2.1 Uncertainty Estimation

Multiple-target predictions naturally include uncertainty for several reasons. To strengthen the reliability of the predictions, this section evaluates Confidence Interval of multiple-target predictions to measure uncertainty in the model. It is determined by using percentage method with bootstrapping [18].

The detailed procedure to obtain bootstrapping sample is as follows:

1. Fit a multiple-target prediction model using training data set $S_1$and obtain target response $T_1 = (P_1, P_{2,} P_3, P_4, N_1, N_2, N_3, N_4)$ of marked instance.

2. Generate a new training data set S_2 by resampling 70% of database (randomly selected with replacement).

3. Refit the regression model using training dataset S_2.

4. Repeat steps (1), (2) and (3) *i* times to generate *I* bootstrapping samples.

These *i* bootstrapping samples are used to develop *i* multi-target prediction models. In the thesis, the target instance is wall WSH3 under 590MPa shear strength. Therefore, *i* predictions of target instance will be collected in APPENDIX B. It is not intuitive to determine confidence interval only with predictions. Then integration of each prediction is generated using following equation:

$$I(x)_{pi} = \int_0^{D_{max}} (P_{1i}x + P_{2i}x + P_{3i}x + P_{4i}x)dx \tag{18}$$

$$I(x)_{Ni} = \int_0^{D_{max}} (N_{1i}x + N_{2i}x + N_{3i}x + N_{4i}x)\, dx \tag{19}$$

where $I(x)_{pi}$ donates positive integration of *x* at the *ith* iteration, and P_{1i} ,P_{2i} ,P_{3i} ,P_{4i} stands for positive parameters at *ith* iteratrion. D_{max} is the max displacement of shear wall under loading.

The cumulative distribution of bootstrap samples $\hat{G}$ [19], less than b can be expressed as:

$$\hat{G}(\mathrm{b}) = F\{\hat{I}_j^* \leq b\}, j = 1, \dots, i. \tag{20}$$

where F is frequencies of $\hat{I}_j^*$. I is target instance. An instance with a specific percentile (α) is able to be represented as:

$$\hat{y}^{*(\alpha)} = \hat{G}^{-1}(\alpha) \tag{21}$$

where $\hat{G}^{-1}$ is the inverse function of $\hat{G}$. Therefore, the 95% confidence interval is represented as:

$$\left(\hat{y}^{*(0.025)}, \hat{y}^{*(0.975)}\right) \tag{22}$$

Then the confidence interval will be judged by $|I(x)|$. In the thesis, *i* equaling 100 is adopted and 95% confidence interval is obtained . The complete detail of 100 iterations is attached in APPENDIX B. 95% confidence interval indicates probability of the predicted curves falling into the range. In the Figure 8, the regression model exhibits a narrow confidence interval of target instance, strengthening power of the model on curve prediction. In the future extension of the research, more training instances will improve the power of the model and narrow confidence interval more.

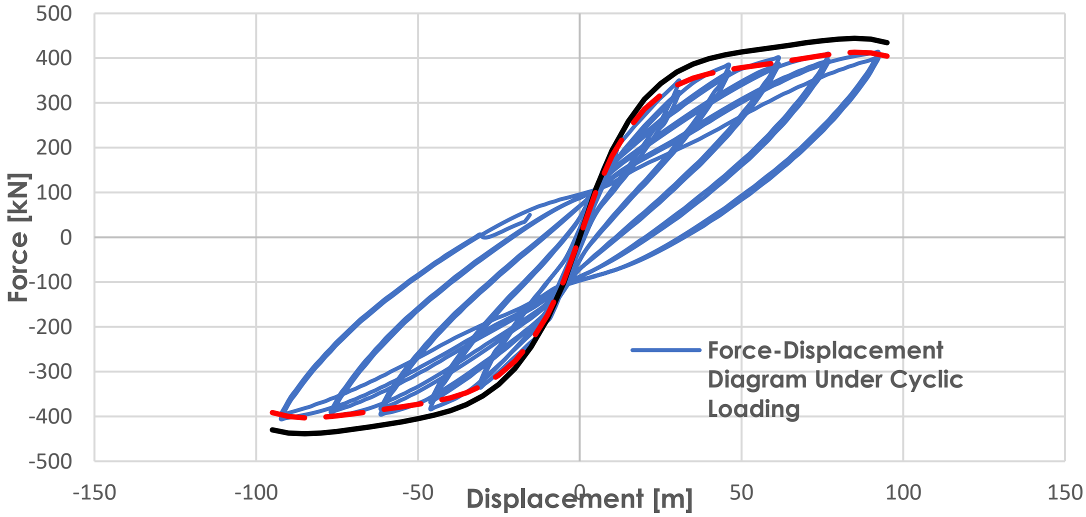


Figure 8: Confidence Interval of Wall WSH3 under 590MPa Shear Strength

CHAPTER 3. REMARKS ON MULTI-TARGET PREDICTION OF CAPACITY CURVES

As to predictions of capacity curve of force-displacement curves of rectangular shear walls, it is necessary to investigate whether database requires normalization or not. In the thesis, multi-target predictions with original database, database after normalization by standard deviation, database after Min-max normalization have been performed separately. All the initial settings of multi-target model are exactly the same for three approaches. The predictions are not remedied. And overall MAEs of predictions with three kinds of database are listed in Figure 9. Normalization of database improves performance of multi-target model a lot. Therefore, it is able to infer that distribution of database has a significant influence on performance of the model. In the thesis, Min-max normalization is chosen for database before throwing into the model.

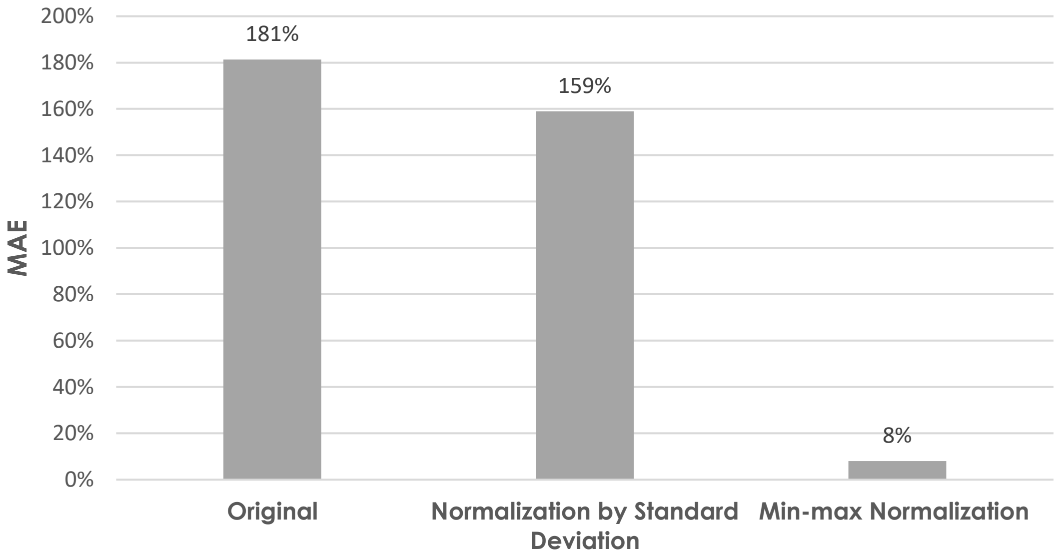


Figure 9: Data Normalization Test

After visualization, the predicted skeleton curves are intuitional to judge its performance. In Figures 10 and 11, it can be seen that both predictions have been fitted in perfectly. However, the prediction in Figure 12 is not satisfied. Even a wiggly part shows up in the curve which is not possible in practice in terms of structural engineering knowledge. The reason is that 46 points found are not distributed enough along the path, and the fitted forth-order polynomial function is very sensitive to the gap between points. Therefore, the wiggly shaped curve is fitted in for gap part between points. The solution is to try the best to find 46 distributed points along the path in data collection process for better fitted-in curves.

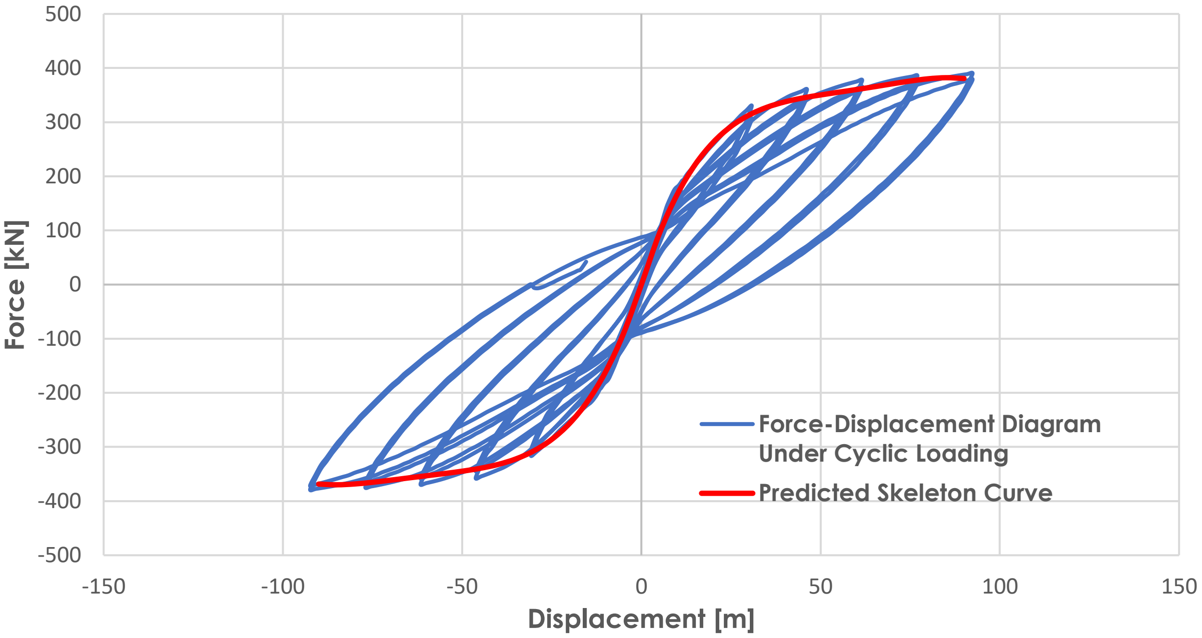


Figure 10: Predicted Skeleton Curve of Wall WSH3 with 10mm Steel Diameter

Now, it is deduced that Multi-target model has a good performance on the shear wall database after remedy strategies. However, we doubt the universality of the deduction. In the training data, similar shear wall instances with test data are involved. It is not surprising that the accuracy of prediction is good. But it is unknown that if multi-target model works well for completely new type of rectangular shear walls. Then 11 new shear wall samples are collected from Key Laboratory of Disaster Reduction in Civil Engineering of the Tongji University. And utilize the whole database as training data to predict force-displacement curve of these 11 new shear wall samples. The key point is that samples in training data are very different from new shear wall samples. Now, the multi-target model predicts the trend of force-displacement curve correctly. However, the predicted boundary of the force is very different from the real one in Figure 13. Next, 5 new shear wall samples are split out into original training data and perform prediction for the rest of 6 new shear wall instances. At this moment, training data provide more information about the new shears. As expected, the predicted curve converges to real one significantly in Figure 14. It is obvious to see that second prediction converge a lot to Figure 13 compared with first prediction.

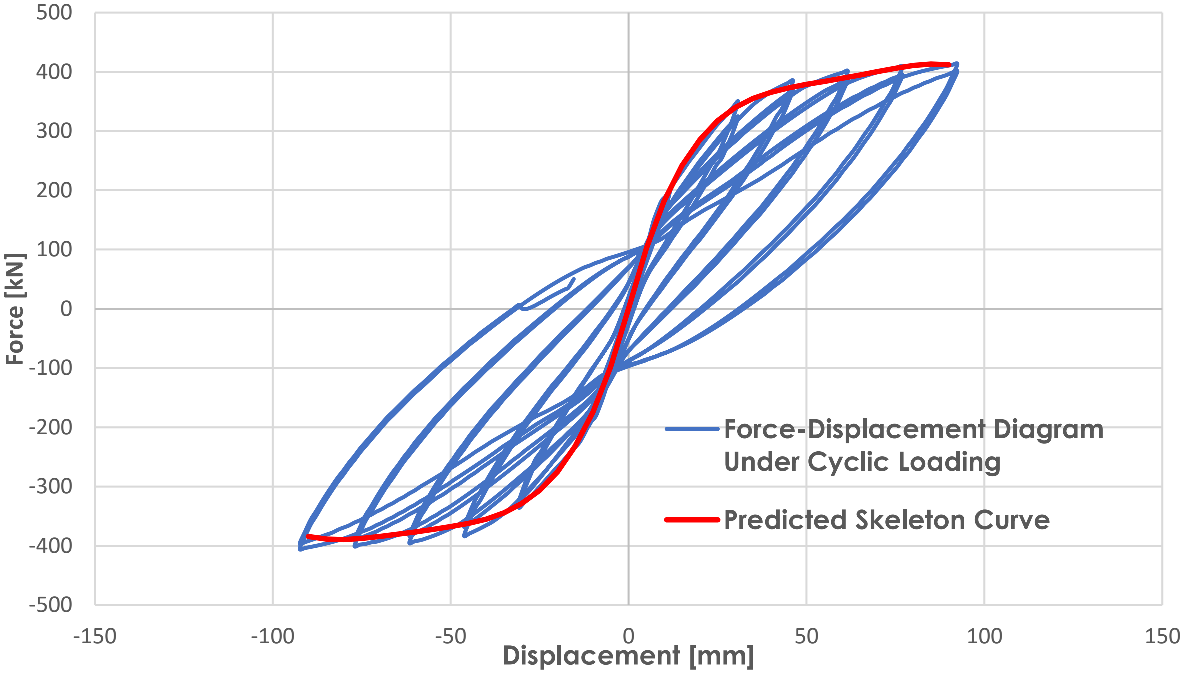


Figure 11: Predicted Skeleton Curve of Wall WSH2 under 590MPa Shear Strength

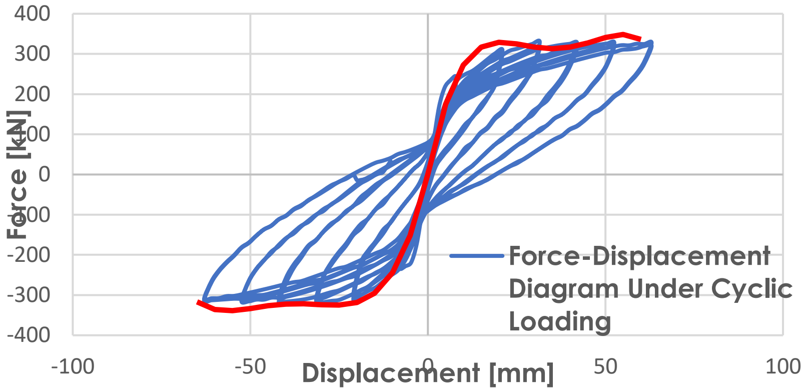


Figure 12: Predicted Skeleton Curve of Wall WSH2 under 560MPa Shear Strength

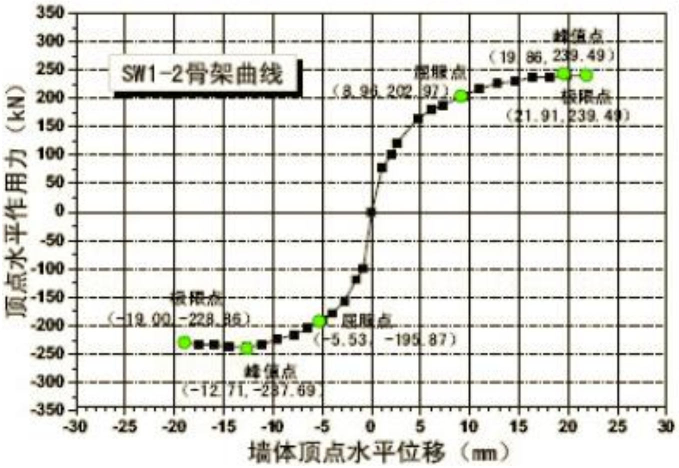


Figure 13: Skeleton Curve of Wall SW1-2 [29]

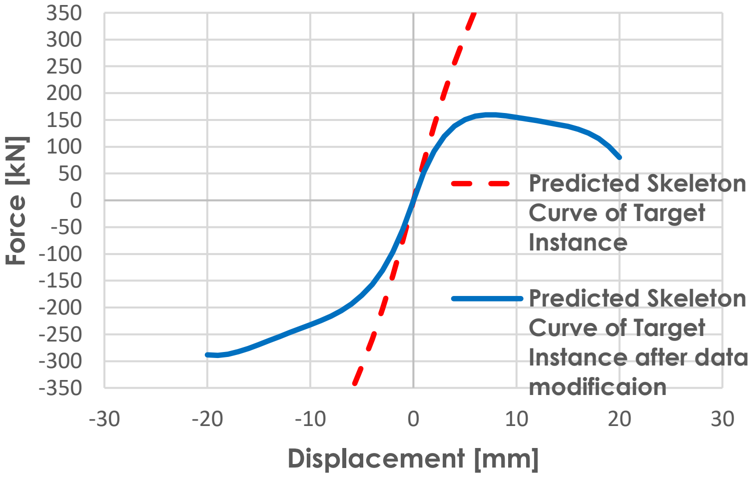


Figure 14: Predicted Capacity Curve of Wall SW1-2

CHAPTER 4. FAST-DETERMINATION OF REMAINING STIFFNESS OF U-SHAPED RC SHEAR WALL BASED ON MECHANICS

4.1 Validation of the FEA Program

Besides predictions of capacity curves of rectangular shear walls, degradation of core shear wall's flexural stiffness is vital to understand the natural frequency shift of the damaged shear walls. But it is hard to capture, often necessitating complex finite element analyses (FEAs). This study also seeks to provide an efficient tool to quickly determine the remaining flexural stiffness of U-shaped core walls after damage [20]. Before proceeding toward the full formulations, this section demonstrates the validity of VEEL as the reference FEA tool. VEEL's analytical power is rooted in a number of microscopic mechanisms: a multi-directional smeared crack model for nonlinear concrete, a topological information-based steel bar model (dubbed as "smart" bar by Cho, 2013) for progressive bar buckling, a random particle-based 3D interlocking model for 3D nonlinear shear, and a bar-concrete proximity-based model for general confinement effect. For details, one is referred to the relevant papers[16][17][21]. VEEL recently overcomes the mesh sensitivity by employing a deformation gradient-based scheme [22]. Three U-shaped wall specimens tested by [23] were modeled and simulated by VEEL on a high-performance computing cluster (Condo, 2017). As an instance, detailed information regarding a U-shaped wall is presented in Figure 15. Three walls are invariably subjected to the constant axial force, and subsequently excited by varying cyclic load patterns: i.e., Wall 1's loading is Y-directional cyclic; Wall 2, X-directional, and Wall 3, bi-directional (butterfly-shape) (see Figure 16).

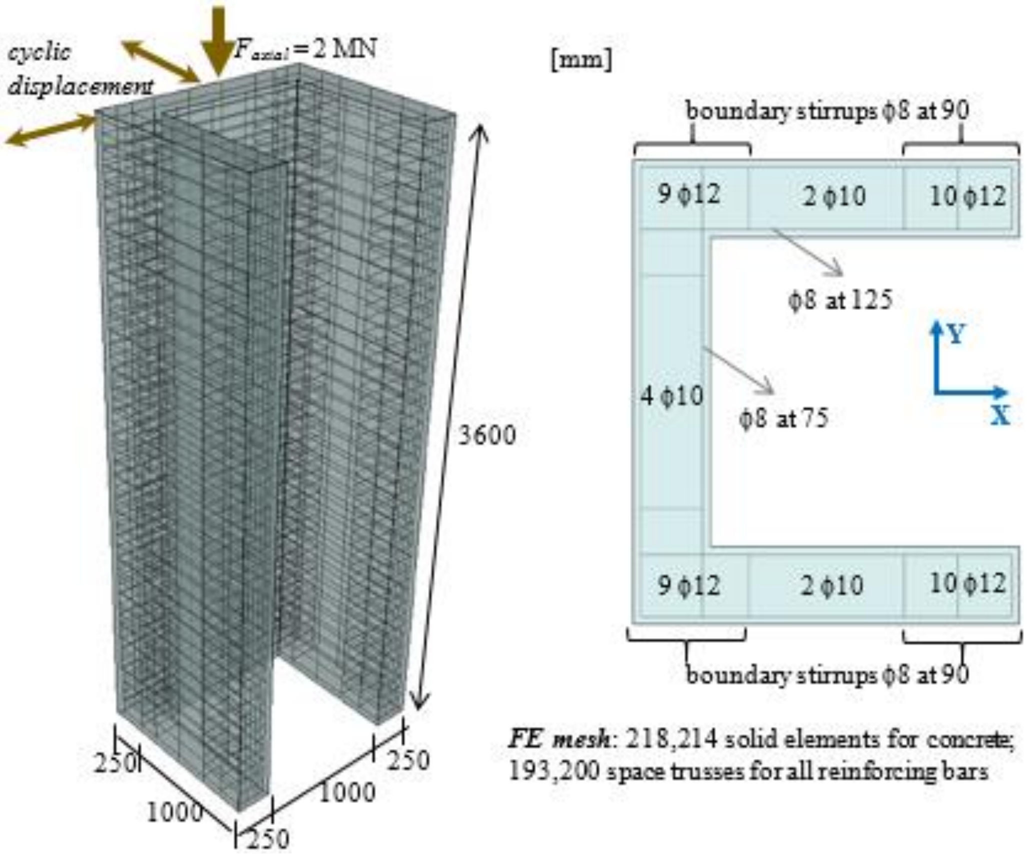


Figure 15: Multi-axial cyclic loads, reinforcements layout, overall geometry, finite element meshes and plan view. Entire reinforcements are explicitly modeled by space trusses

The entire reinforcements including all transverse bars and stirrups are explicitly modeled by using a general RC wall auto meshing preprocessor [24] (available upon request to the corresponding author). As demonstrated by [17], no specimen-dependent parameter fitting is needed for VEEL, and its microscopic mechanisms require only basic material properties such as concrete strength and steel strength: e.g., the concrete compressive strength = 23.73 MPa; the yield (ultimate) strength of primary bar = 516 MPa (615 MPa).

First, VEEL accurately reproduces the global force-displacement responses regardless of the various load patterns. Figs. 16-18 summarizes global force-displacement responses. In addition to the global responses, VEEL captures the microscopic damage phenomena. Fig. 19 compares the predicted zone of progressive bar buckling and a photo showing actual damage state. The slight difference in the post-peak regimes of Figs. 16-18 may be attributed to the perfectly bonded bar-concrete assumption of VEEL. Incorporating partial bond or bond-slip behavior into the VEEL will help the improvement, which is beyond the scope of this study. Departing from this initial validation of VEEL, this study adopts VEEL as a reference FEA program against which the new unit cell-based formulas will be compared and validated.

Importantly, VEEL can provide comprehensive details of entire U-shaped walls such as cracking, crushing, bar yielding, or buckling. Collecting such detailed information of entire U-walls from actual experiments is challenging. Therefore, this study seeks to propose a simple yet sufficiently accurate formulas that can mimic VEEL's prediction power.

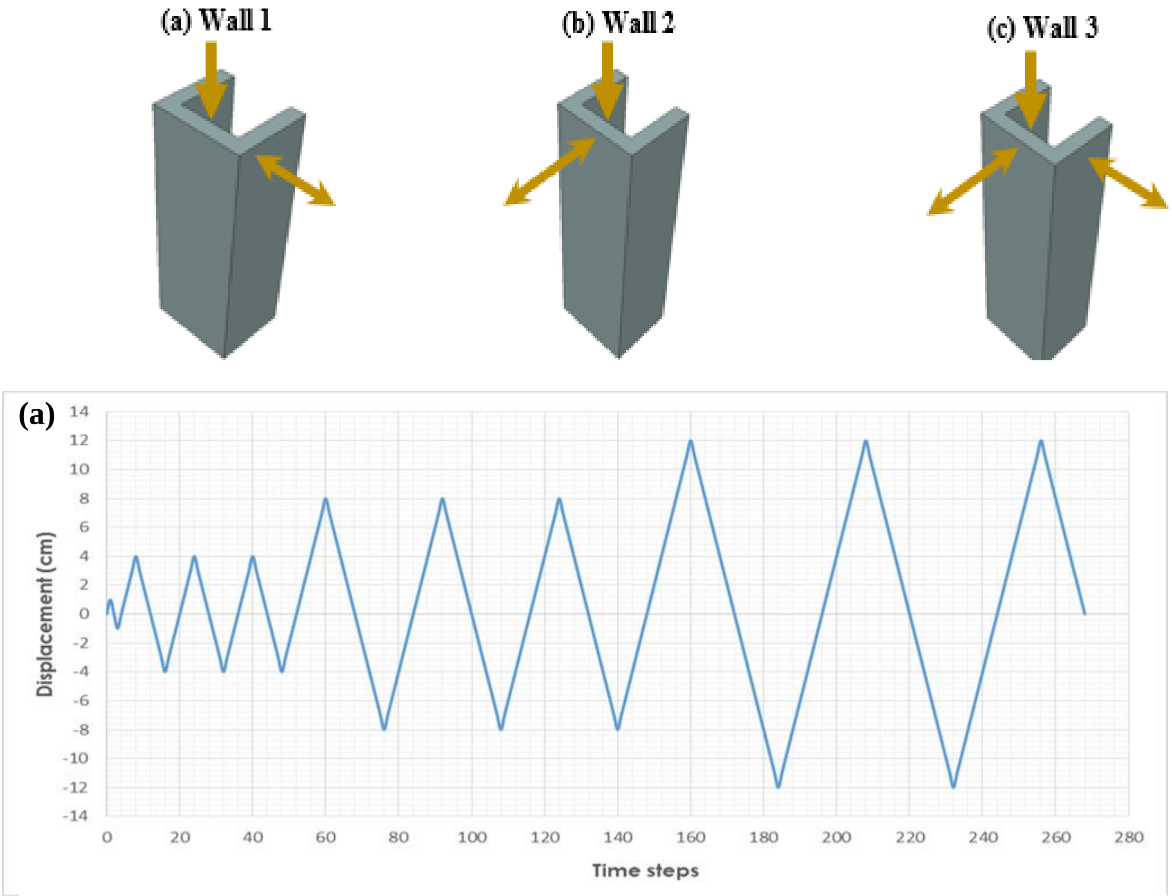


Figure 16:Cyclic loading history: (a) Wall 1 under Y-directional load; (b) Wall 2 under X load; (c) Wall 3 under bi-directional loads.

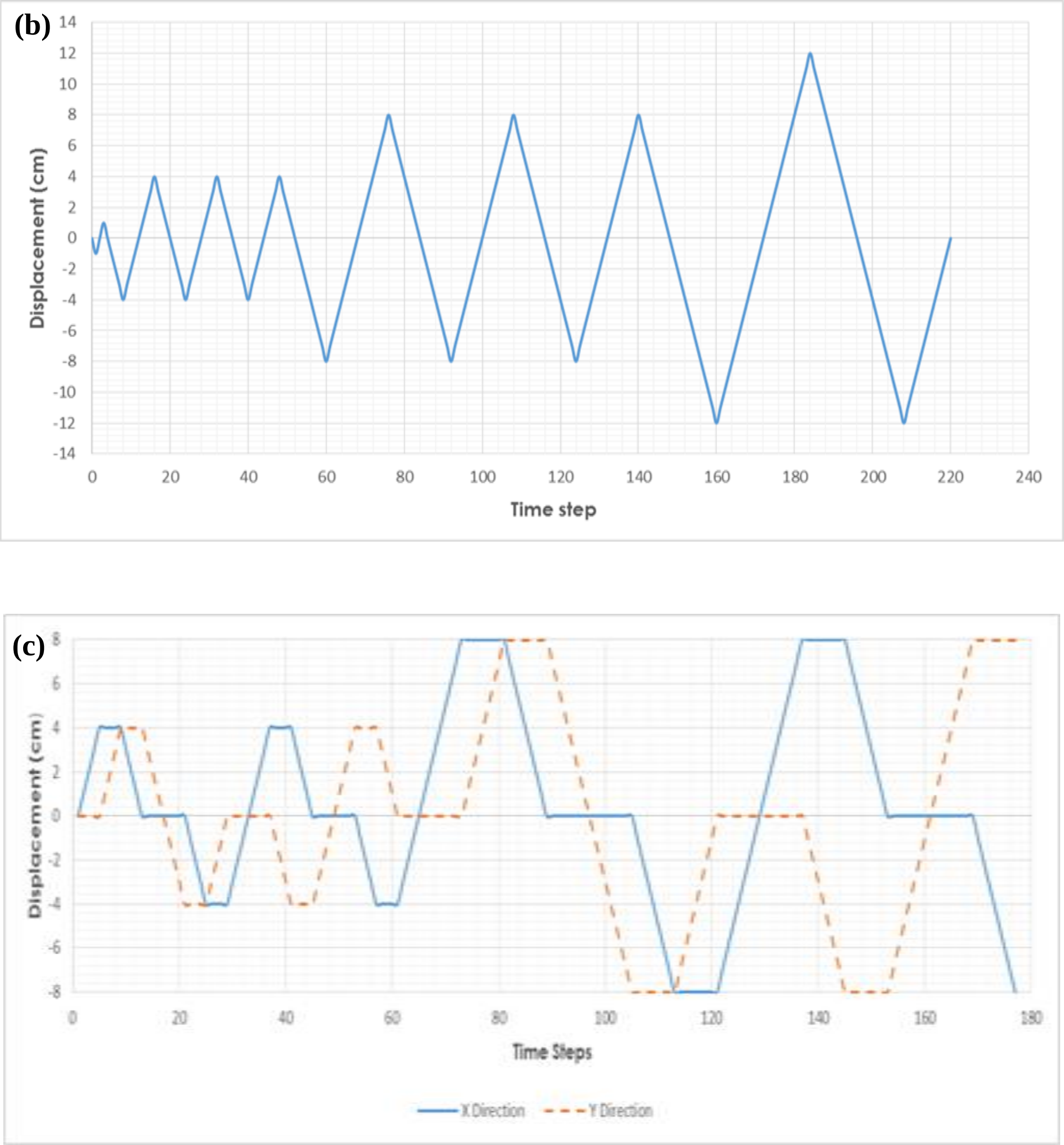


Figure 16. continued

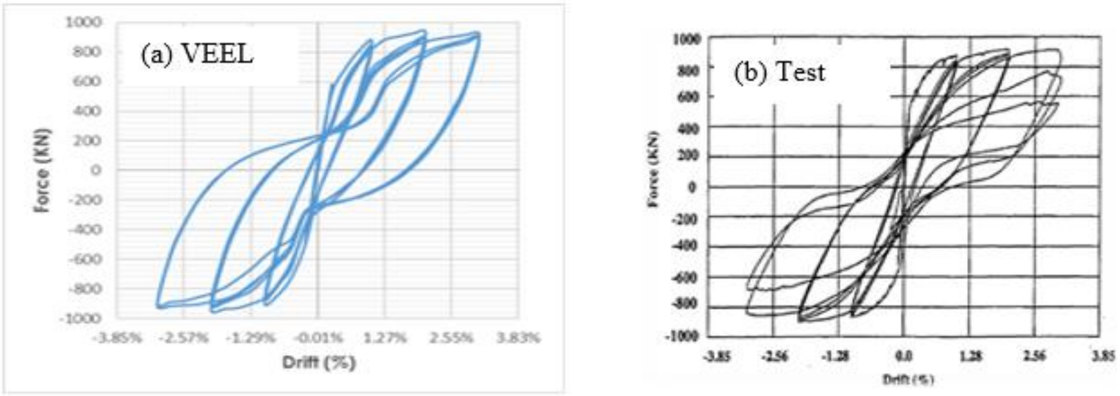


Figure 17 Force-displacement respones of Wall 1 under Y-directional cylic loading: (a) VEEL prediction; (b) real test results [25]

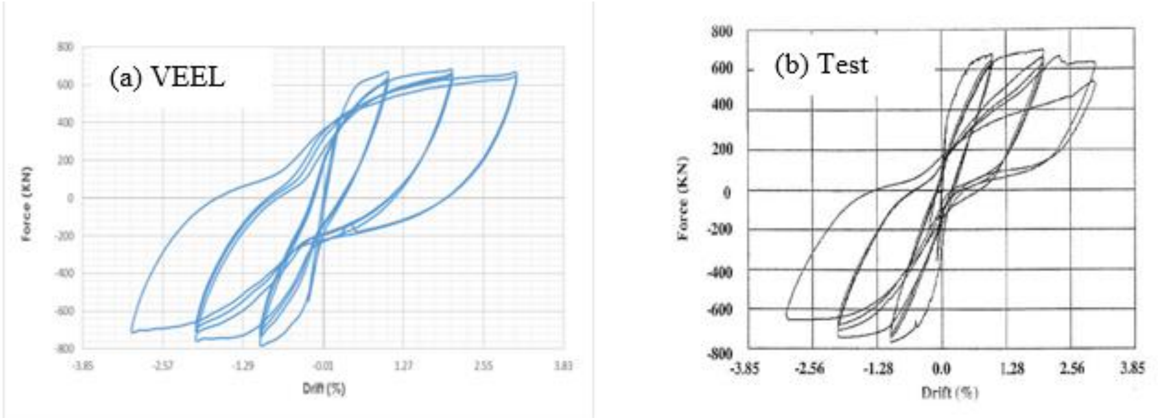
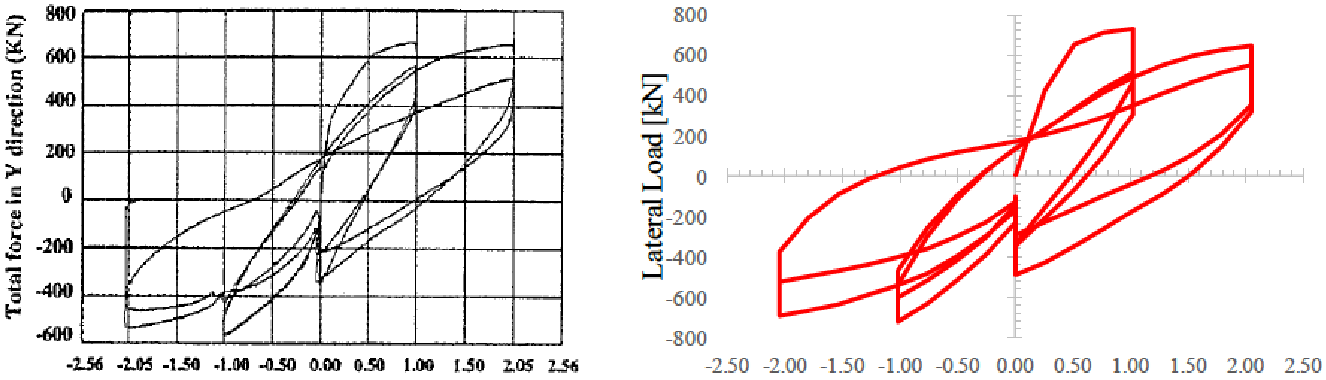
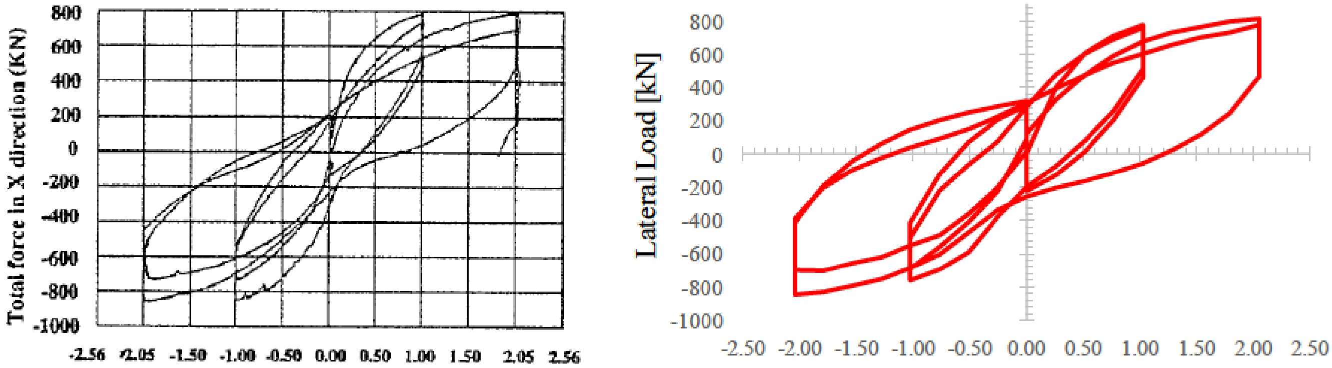


Figure 18: Force-displacement respones of Wall 2 under X-directional cylic loading : (a) VEEL prediction; (b) real test results [25]



(a) Drift in X direction (%)

(c) Drift ratio in X direction (%)



(b) Drift in Y direction (%)

(d) Drift ratio in Y direction (%)

Figure 19: Global force-displacement responses of Wall 3 under bi-directional displacements in conjunction with constant axial force: (a) and (b) are experimental results (cited from [23]); (c) and (d) are predicted responses by VEEL.

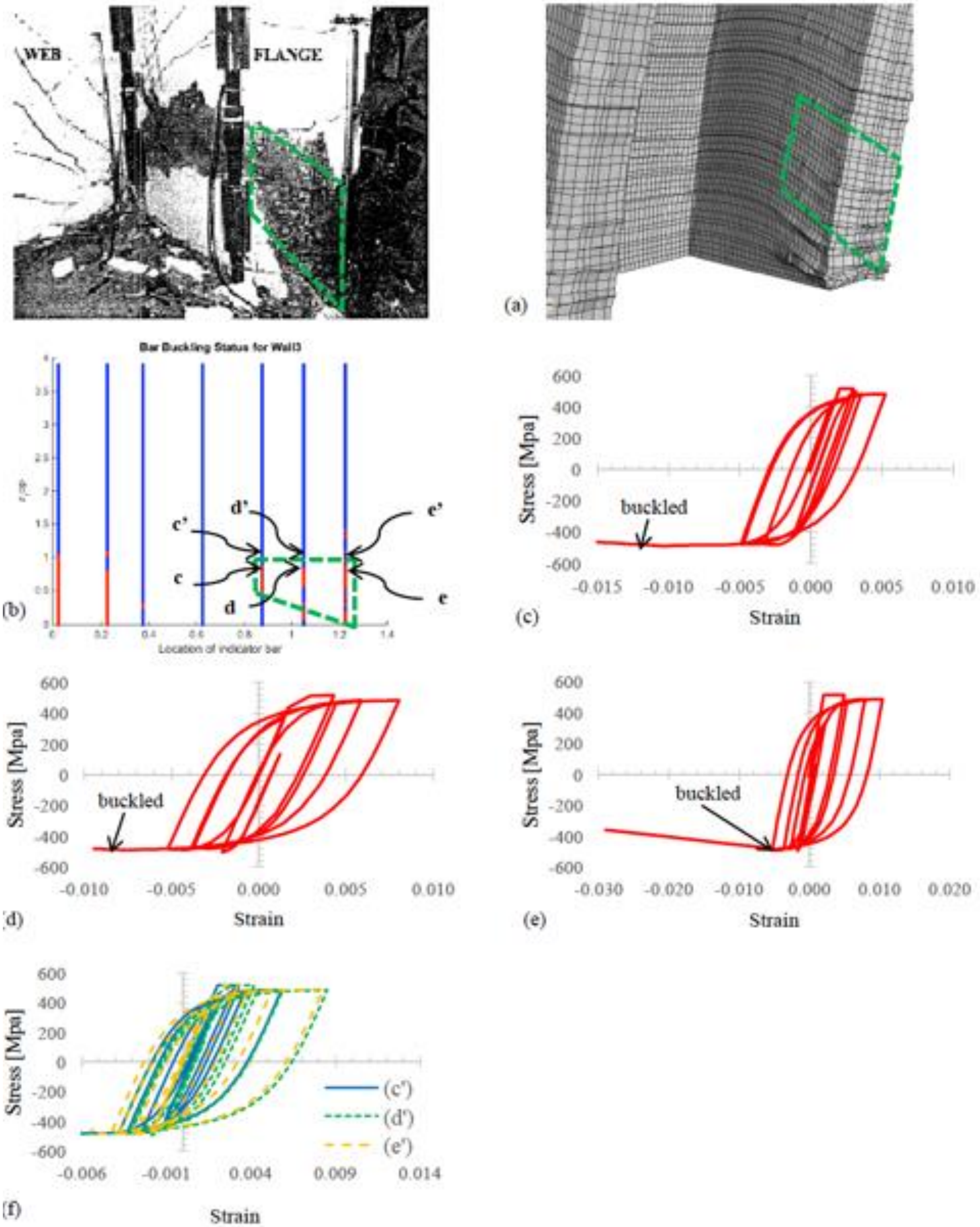


Figure 20: Predicted zone of progressive bar buckling (PBB) and concrete spalling of Wall 3: (inset photo) actual damage state (cited from [Ile and Reynouard, 2005]); (a) Deformed shape plot; (b) VEEL prediction of progressive bar buckling state at end of simulation

4.2 Quick Observation of Damage Information

From CHAPTER 4 to CAHPTER 5, the implement of cell network-based formulas to predict remaining stiffness is explicitly demonstrated. The theory of cell network has already been developed by Yemmaleni in his master thesis report [27]. However, the computational implementation and practical application, as innovations, have been developed in my thesis. Also, a corresponding paper [20] was published in 2017 to illustrate all the innovations in my thesis.

This chapter introduces several damage-related parameters, which are easily collectible on site: α = the fraction of wall area that has undergone tensile yielding; β = the fraction of wall area that has undergone compressive damage. This study suggests a small "unit cell" which is defined as a combination of two nonlinear concrete and steel springs and one compression-only gap. Each panel of U-shaped wall is replaced by a dense network of many unit cells. The areas under α and β are counted by numbers of unit cells.

- m_α = the number of horizontal unit cells in the α zone ($= L_\alpha$/unit cell size)
- h_α = the number of vertical unit cells in the α zone ($= H_\alpha$/unit cell size)
- m_β = the number of horizontal unit cells in the β zone ($= L_\beta$/unit cell size)
- h_β = the number of vertical unit cells in the β zone ($= H_\beta$/unit cell size)

L_α and H_α stand for the actual length and height of triangular α zone (Fig. 20), respectively The α zone is assumed to occur when unit cell's strain ($\varepsilon_{\text{cell}}$) exceeds 0.002 (likewise, for β zone $\varepsilon_{\text{cell}} < -0.002$). This study assumes straight boundaries of the two damage states α and β zones as illustrated in Figure 20. In view of reality, if $\varepsilon_{\text{cell}}$ becomes less than -0.003 the compressive damage is regarded as irrecoverable, and the β zone remains thereafter.

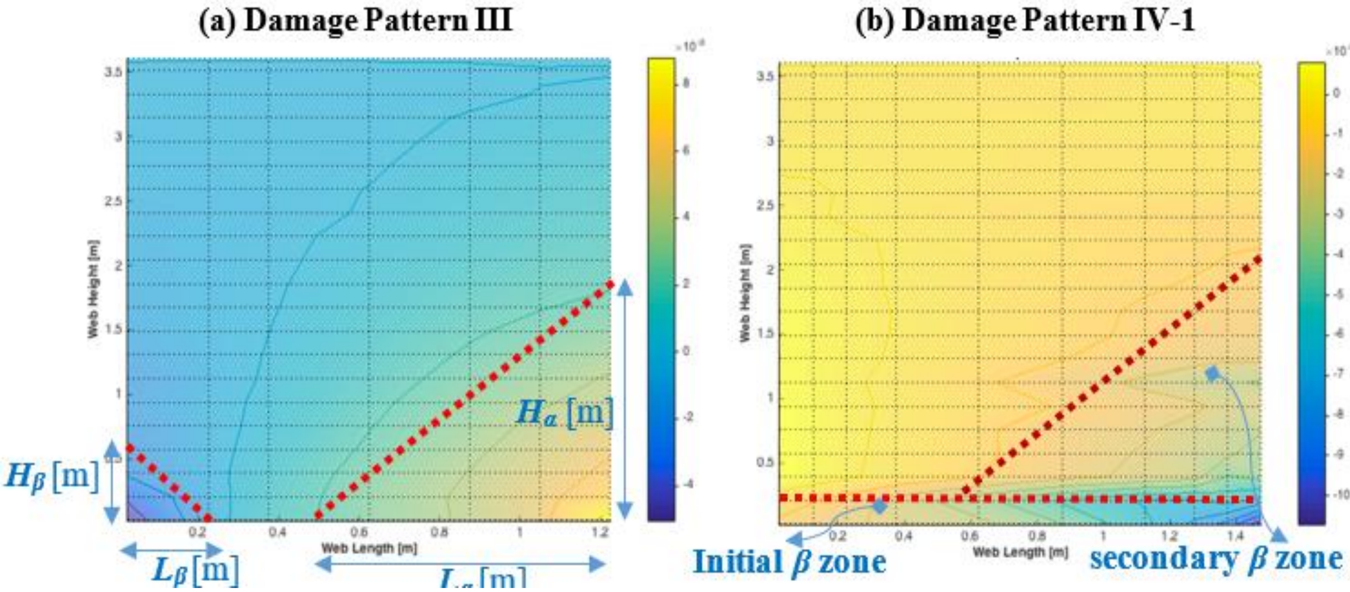


Figure 21: Example strain plot of a wall panel: (a) Damage Pattern III where compressive damage zone (β zone) and tensile yielding zone (α zone) coexist; (b) Damage Pattern IV-1 where initial horizontal β zone is followed by secondary β zone.

A wall panel is replaced with a cell network of *m* by *n* unit cells, and each unit cell comprises a nonlinear steel and concrete springs and a compression-only gap (Figure. 22). Initially, steel and concrete springs (stiffness is denoted as K_s and K_c, respectively) are connected in parallel, and the total initial stiffness is denoted as K_T

$$K_T = K_s + K_c \delta_{gap}(\varepsilon_{cell})$$

$$\delta_{gap}(\varepsilon_{cell}) = \begin{cases} 1 & \varepsilon_{cell} < 0 \\ 0 & \varepsilon_{cell} \geq 0 \end{cases} \tag{23}$$

The strain at vertical bar yielding is assumed to be 0.002. $\delta_{gap}(\varepsilon_{cell})$ corresponds to the compression-only gap. Thus, when the unit cell is under tension, $K_T = K_s$ whereas under compression $K_T = K_s + K_c$. This implies that tensile resistance of concrete is neglected for the conservative outcome. For a vertical chain of n unit cells, the corresponding effective stiffness is obtained as $\frac{K_T}{n}$. For m vertical chains in parallel (Fig. 21), the total initial axial stiffness is

$$K_o = \frac{m}{n} K_T \tag{24}$$

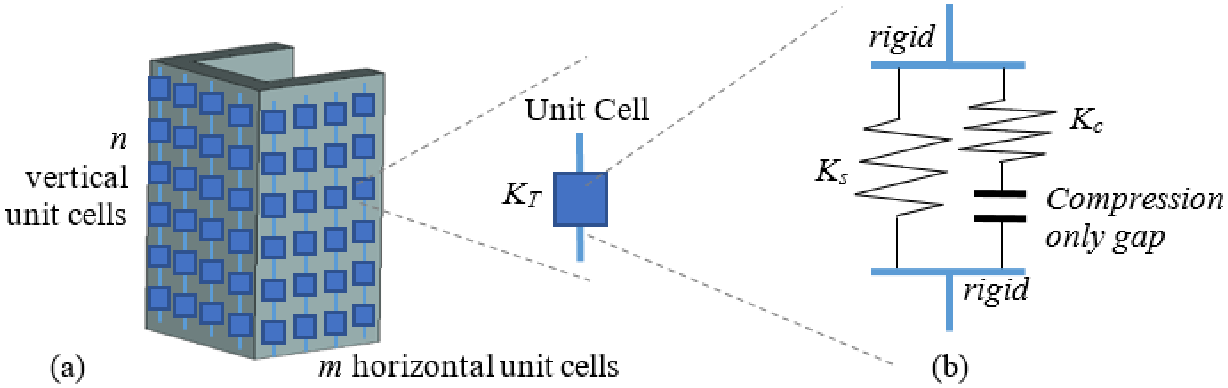


Figure 22: (a) Three panels of U-shaped wall represented by a network of m by n unit cells; (b) One unit cell consisting of steel, concrete, and compression-only gap components.

For the present work, the effects of bond-slip and tensile resistance of concrete are neglected. Two material models are required to define nonlinear concrete and steel (see Figure 22). For the steel spring (Figure 22a), a bilinear model is used. This study introduced the term b (≥0) which means the stiffness reduction factor: $b = E_1/E_0$ where E_0 is the initial stiffness of steel and E_1 is the post-yielding stiffness. Based on U-shaped walls in literature [23][25], b is calculated as 0.00156, and this value is used for all simulations. Regarding the concrete spring, this study uses a simplified backbone model. As shown in Fig. 22b, tensile stiffness is neglected, and the post-peak compressive stiffness is given by $d \times E_{c0}$ where E_{c0} is the initial concrete stiffness. According to a well-known concrete model (e.g, Kent and Park model), d may be a negative value. But, our preliminary studies show that a negative value (e.g., -0.109322 according to the Kent and Park model) may lead to numerically unstable stiffness of a wall panel from a very early loading stage. A negative wall stiffness may emerge at the post-peak regime, but too early emergence of negative stiffness may not be preferable. Hereafter, *d* is set to be 0 for numerical stability. Future extension with a realistic *d* will be straightforward owing to the separate definition of the concrete spring. Importantly, a wall panel is replaced with a dense cell network whilst a real-world wall is reinforced by longitudinal bars and transverse bars that are spatially scattered. To fill this gap, *Ks* of the stiffness of a steel spring has been reduced by multiplying the vertical steel reinforcement ratio (ρ) (e.g., ρ = 0.0056 for U-shaped walls of [23].

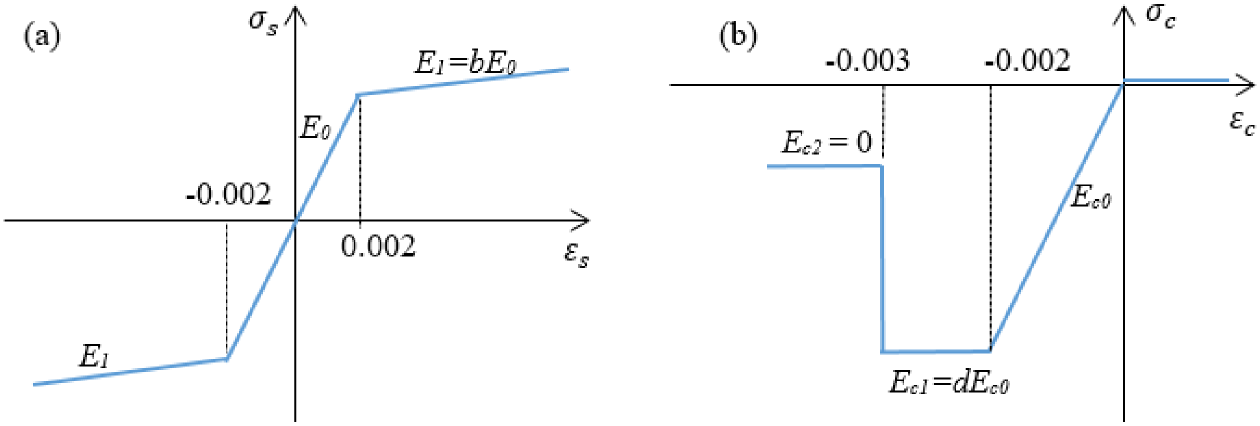


Figure 23: Simplified stiffness of Unit Cell components: (a) Ks of steel spring; (b) Kc of concrete spring.

This study introduced five damage patterns that can emerge in RC core wall panels: three damage patterns concern a panel damaged by uni-axial horizontal loads while the other two patterns deal with a damage caused by bi-directional horizontal loads. These five categories may not cover all possible damages (e.g., shear sliding failures, diagonal spalling, etc.). Still, the present work will serve as a reasonable starting point for future extensions.

Damage Pattern I: Tensile Yielding under Horizontal Uni-axial Loading

This pattern deals with a panel of which bottom portion undergoes tensile yielding (Figure 24). Damage Pattern (DP) I can occur when a panel is under uni-axial horizontal movement (e.g., dashed arrow in Fig. 23). All the unit cells within the yielding zone (denoted as α zone) will have the yielding stiffness (denoted as K_{Ty}). The unit cell's resistance comes from steel spring:

$$K_{Ty} = bK_s \tag{25}$$

In a vertical chain with *n* cells, there are $n\times\alpha$ cells with a stiffness equal to K_{Ty} whreas (*n* - $n\alpha$) cells with K_T. Hence, the stiffness for the single vertical chain of *n* cells reads

$$\frac{1}{K} = \frac{n\alpha}{K_{Ty}} + \frac{n - n\alpha}{K_T} \tag{26}$$

For *m* such chains aligned in parallel, the overal axial stiffness of the wall panel is given by

$$\textbf{Damage Pattern I}: K = \frac{m}{n}\left(\frac{K_{Ty}K_T}{\alpha K_T + (1 - \alpha)K_{Ty}}\right) \tag{27}$$

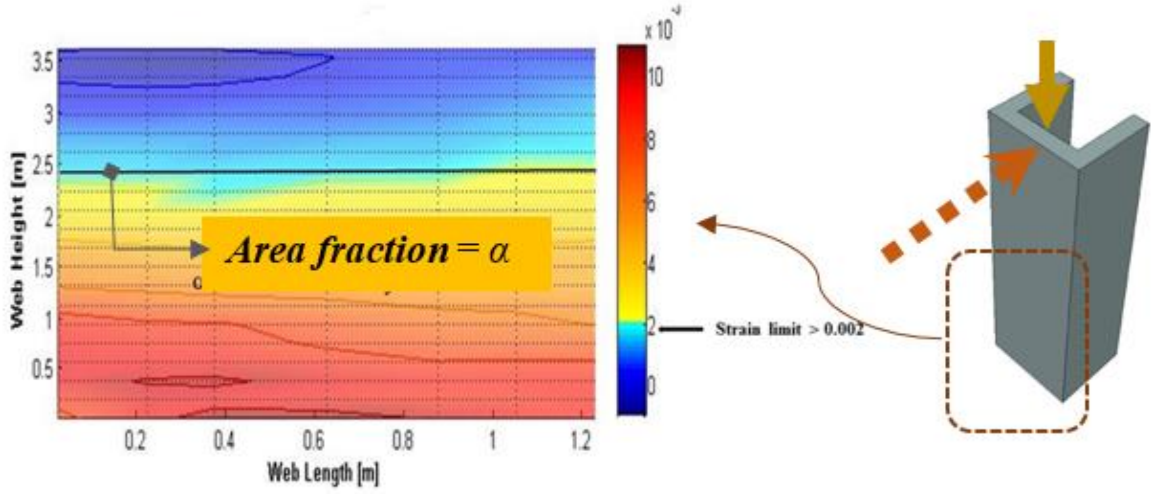


Figure 24: Damage Pattern I: Horizontal Tensile Yielding.

Damage Pattern II: Compressive Damage under Horizontal Uni-axial Loading

As shown in Figure 25, DP II can take place when a wall panel is under uni-axial horizontal movement (e.g., dashed arrow in Fig. 25). The stiffness of a unit cell that enters into post-peak compressive damage state is described by a reduce stiffness (denoted as K_{TC}). From the assumed constitutive rules of cell springs, a concrete spring's stiffness is reduced to $d \times K_C$ (in general, $|d| < 1.0$ and $d \leq 0.0$) and the steel stiffness reduces to $b \times Ks$ where $|b| < 1.0$. Therefore, the new reduced stiffness of a unit cell that falls into this DP II becomes

$$K_{Tc} = bK_s + \ dK_c \tag{28}$$

Similar to the aforementioned DP I, the overall axial stiffness of the panel is determined by

$$\textbf{Damage Pattern II}\text{:}\ K = \frac{m}{n}\left(\frac{K_{TC}K_T}{\beta\, K_T + (1 - \ \beta\,)K_{TC}}\right) \tag{29}$$

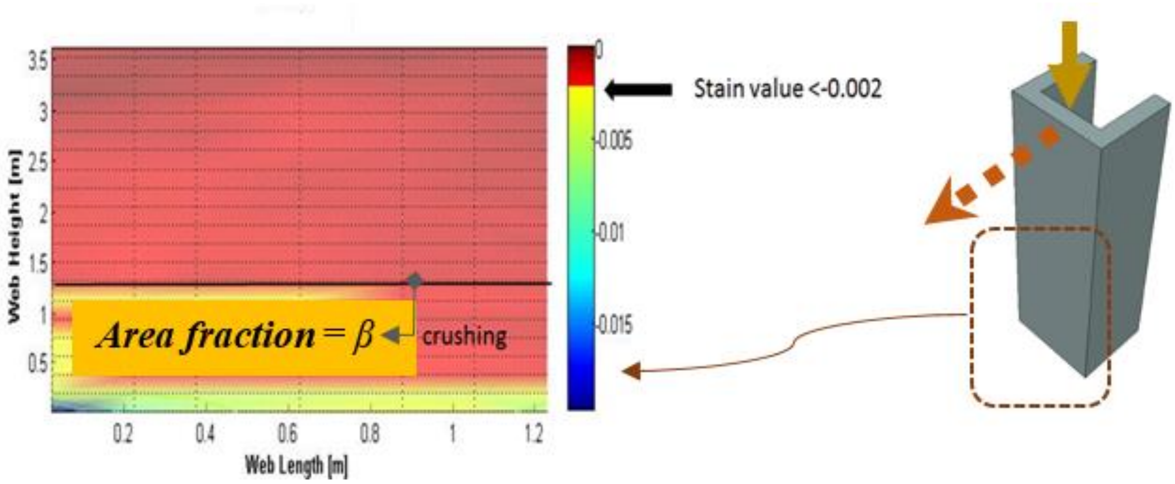


Figure 25: Damage Pattern II: Horizontal compressive damage state.

Damage Pattern III: Dual Damage States by In-Plane Uni-Axial Loads

This pattern concerns the case that a bottom corner of wall panel enters into the post-peak compressive damage regime while the other bottom corner undergoes tensile yielding. For derivation, the wall panel is divided into three segments (Figure. 26). The heights and lengths in the α and β zones can be obtained from quick observations. They are represented by a pair of (the number of vertical unit cells and horizontal unit cells) in each zone, denoted by (n_α, m_α) and (n_β, m_β), respectively (Fig. 26). The two damage zones are assumed triangular shape. The varying height of the triangular damage zones are denoted by $y(x)$ and $z(p)$. The local coordinates x and p are positive integers meaning the counts of unit cells along the horizontal axis in each damage zone. Hence, $y(x = m_\alpha) = 0$ and $z(p = m_\beta) = 0$.

All cells in panel 2 (P2) remain K_T, and the stiffness for a vertical chain is equal to K_T/n. Hence,

$$K_{P2} = \frac{m - m_\alpha - m_\beta}{n} K_T \tag{30}$$

Now, for the left segment of the wall panel, (denoted as P1 in Fig. 26), $y(x)$ means the number of vertical cells that are yielding:

$$y(x) = \frac{n_\alpha(m_\alpha - x)}{m_\alpha} \quad \text{for } x \in [0, m_\alpha] \tag{31}$$

In a vertical chain of segment P1, among n cells, $y(x)$ cells will have K_{Ty} while $(n-y(x))$ cells have K_T. Hence the stiffness for a single chain at x can be determined as $y(x)/K_{Ty} + (n - y(x))/K_T$. Therefore, the total axial stiffness of the segment P1 (denoted as K_{P1}) is given by

$$K_{P1} = \sum_{x=0}^{m_\alpha} \frac{K_T K_{Ty}}{[n - y(x)]K_{Ty} + y(x)K_T} \tag{32}$$

Similarly, one can determine the total stiffness of segment P3 (Fig. 26) as

$$K_{P3} = \sum_{p=0}^{m_\beta} \frac{K_T K_{TC}}{[n - z(p)]K_{TC} + z(p)K_T} \tag{33}$$

where $z(p)$ is the number of vertical cells at p

$$z(p) = \frac{n_\beta(m_\beta - p)}{m_\beta} \quad \text{for } p \in [0, m_\beta] \tag{34}$$

Therefore, the total axial stiffness of the wall panel is given by

$$\boldsymbol{Damage\ Pattern\ III}: K = Kp1 + Kp2 + Kp3 \tag{35}$$

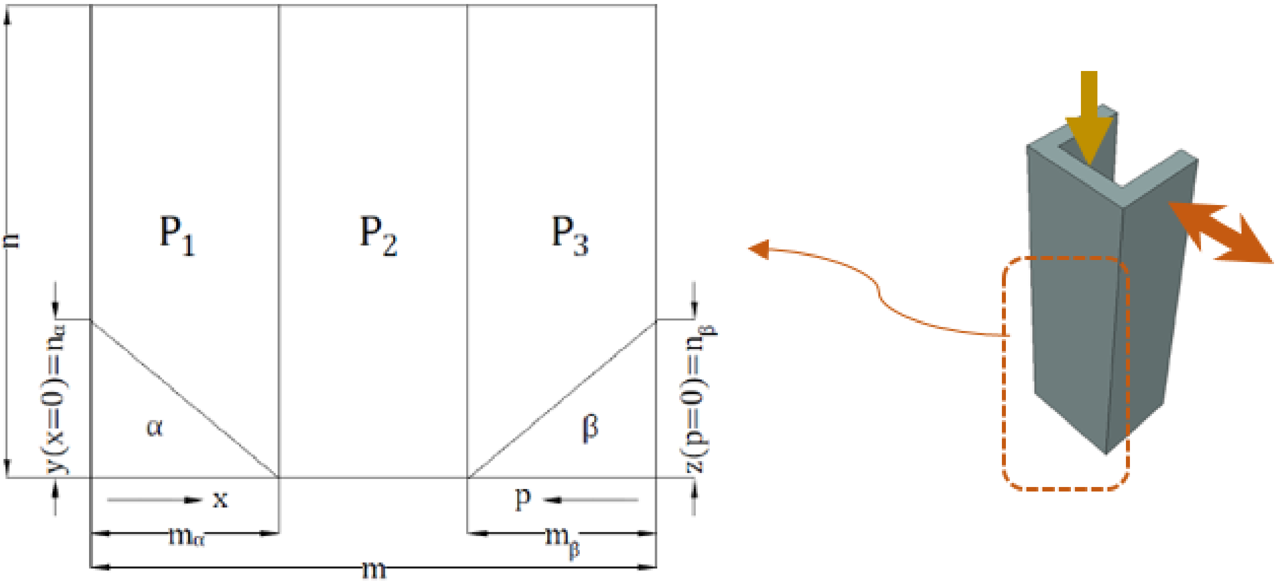


Figure 26: Damage Pattern III. Dual damage states of tensile yielding and compressive damage. (nα, mα) and (nβ, mβ) are the cell numbers associated with the observed lengths and heights.

Damage Pattern IV: Combination of Damage Patterns II and III

This pattern deals with a combined damage case, possibly caused by bi-directional loads. Initially, the wall panel has been displaced by an out-of-plane compression (Fig. 27a) and then loaded by in-plane loading in the other direction (Fig. 27b). It should be noted that Figs. 27c-e shows three possible cases on the web panel (dashed box in Fig. 27b). Particularly, when the web panel is initially loaded in the X direction (Fig. 27a), it has Damage Pattern II (denoted by horizontal zone of β_1). Afterward, due to the Y-directional loads Damage Pattern III begins to emerge (zones marked with α and β_2 in Figs. 27c-e).

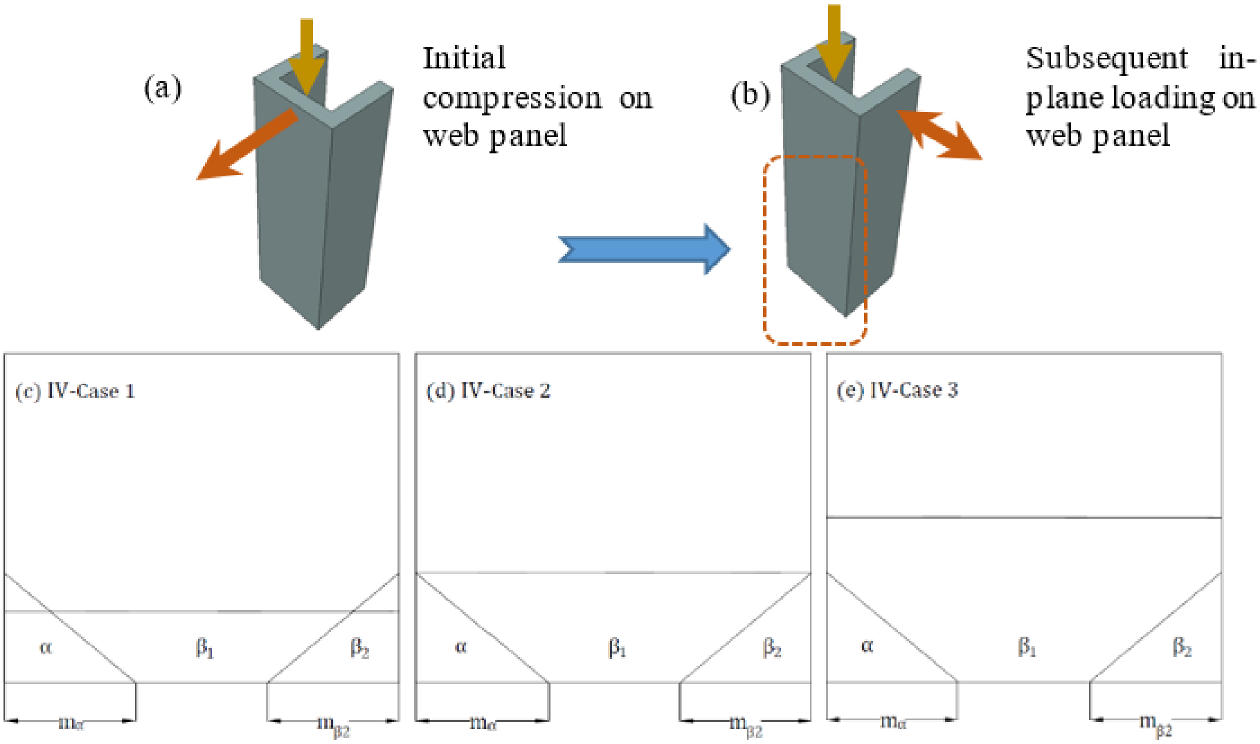


Figure 27: Thee possible cases of the Damage Pattern IV on the web panel.

Damage Pattern IV – Case 1

This case corresponds to the circumstance when $h_{\beta 1} < h_{\beta 2}$ and $h_{\beta 1} < h_{\alpha}$. Fig. 28(a) describes the vertical and horizontal local coordinates of the left segment P1 while Fig. 28(b) shows P2. Similar local coordinates are defined on right segments P4 and P5.

Segment P1: The stiffness of the segment P1 is almost identical to K_{P1} of Damage Pattern III (Eq. 29) except for the range of local coordinate x. Referring to Fig. 28(a),

$$K_{P1}^{IV1} = \sum_{x=0}^{z_1 \frac{m_\alpha}{h_\alpha}} \frac{K_T K_{Ty}}{[n-y(x)]K_{Ty}+y(x)K_T}; \tag{36}$$

$$z_1 = h_\alpha - n\beta_1 \tag{37}$$

where $z_1 \frac{m_\alpha}{h_\alpha}$ is the number (width) of horizontal cells in the segment P1; $y(x)$ is the number (height) of the vertical cells at x within P1:

$$y(x) = \frac{h_\alpha(m_\alpha - x)}{m_\alpha} \text{ for } x \in [0, z_1 \frac{m_\alpha}{h_\alpha}]. \tag{38}$$

Segment P2: the number of vertical cells within the α zone of P2, $y(x_2)$ is given by

$$y(x_2) = n\beta_1 \left(n\beta_1 \frac{m_\alpha}{h_\alpha} - x_2\right) / n\beta_1 \frac{m_\alpha}{h_\alpha} \quad \text{for } x_2 \in [0, n\beta_1 \frac{m_\alpha}{h_\alpha}], \tag{39}$$

where the number of horizontal cells, x_2 ranges from 0 to $n\beta_1 \frac{m_\alpha}{h_\alpha}$ (Fig. 27(b)). For one vertical chain of cells at x_2, $y(x_2)$ cells yielded and have a stiffness of K_{Ty} while $\left(n\beta_1 - y(x_2)\right)$ cells undergoes compressive damage and thus have K_{Tc}. The remainder in the vertical chain, i.e. $(n - n\beta_1)$ cells are elastic having K_T. Hence, the stiffness for a single vertical chain at x_2 is

$$\frac{y(x_2)}{K_{Ty}} + \frac{n\beta_1 - y(x_2)}{K_{Tc}} + \frac{n - n\beta_1}{K_T}. \tag{40}$$

By simplifying and summing up all the vertical chains in P2, the segment stiffness becomes

$$K_{P2}^{IV1} = \sum_{x_2=0}^{n\beta_1 \frac{m_\alpha}{h_\alpha}} \frac{K_T K_{Ty} K_{Tc}}{[n\beta_1 - y(x_2)]K_T K_{Ty} + y(x_2) K_T K_{Tc} + (n - n\beta_1) K_{Ty} K_{Tc}}. \tag{41}$$

Segment P3: As shown in Fig. 28(c), the segment P3 can be regarded as a Damage Pattern II, uniform compressive damage state. Hence, one can refer to Eq. (7) to derive the stiffness as

$$K_{P3}^{IV1} = \frac{m - m_\alpha - m_{\beta 2}}{n} \left(\frac{K_{Tc} K_T}{\beta_1 K_T + (1 - \beta_1) K_{Tc}}\right) \quad \text{for } m > (m_\alpha + m_{\beta 2}). \tag{42}$$

Compared to Eq. (28), a slight difference exists: i.e., $(m - m_{\alpha} - m_{\beta 2})$ is used for the number (width) of the segment P3, and β_1 is used in lieu of β. It should be noted the present study focuses on the damage state when $m > (m_{\alpha} + m_{\beta 2})$.

Segment P4: Referring to Fig. 28(c), the segment P4 was initially under compressive damage β_1 and now it is under subseqent compressive damage β_2. The segment P4 can be regarded as Damage Pattern II, the horizontal compressive damage state. Hence, referring to K_{P3} of DP II,

$$K_{P4}^{IV1} = \frac{n\beta_1 \frac{m_{\beta 2}}{h_{\beta 2}}}{n}\left(\frac{K_{Tc}K_T}{\beta_1 K_T + (1 - \beta_1)K_{Tc}}\right) \tag{43}$$

where $n\beta_1 \frac{m_{\beta 2}}{h_{\beta 2}}$ means the number (width) of the horizontal cells in the segment P4.

Segment P5: There exists analogy between segment P5 and P1 (see Fig. 28(a) and rightmost segment P5 in Fig. 28(c)), and the only difference is the compressive damange of P5 in lieu of tensile yielding. Hence, by simply replacing K_{Ty} of K_{P1}^{IV1} (Eq. 14) with K_{Tc}, P5 stiffness is

$$K_{P5}^{IV1} = \sum_{\bar{x}=0}^{z_2 \frac{m_{\beta 2}}{h_{\beta 2}}} \frac{K_T K_{Tc}}{[n - y(\bar{x})]K_{Tc} + y(\bar{x})K_T} \tag{44}$$

$$z_2 = \left(h_{\beta 2} - n\beta_1\right) \tag{45}$$

$$y(\bar{x}) = \frac{h_{\beta 2}\left(m_{\beta 2} - \bar{x}\right)}{m_{\beta 2}} \quad \text{for } \bar{x} \in \left[0, z_2 \frac{m_{\beta 2}}{h_{\beta 2}}\right] \tag{46}$$

In sum, the overall axial stiffness of the wall panel under Damage Pattern IV-Case 1 reads

$$\textbf{Damage Pattern IV} - \textbf{Case 1}: \quad K = K_{P1}^{IV1} + K_{P2}^{IV1} + K_{P3}^{IV1} + K_{P4}^{IV1} + K_{P5}^{IV1} \tag{47}$$

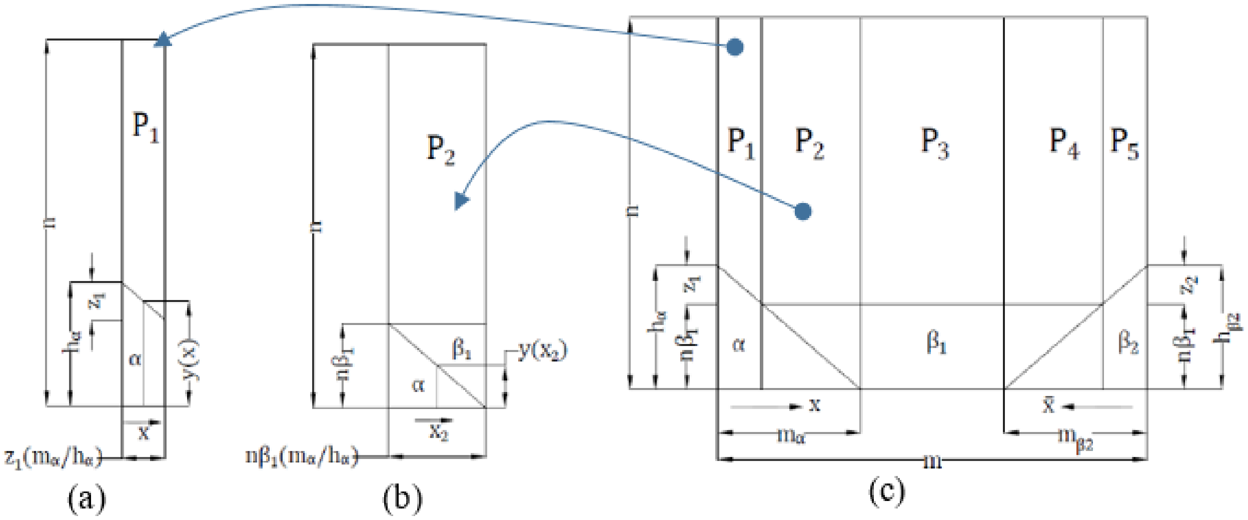


Figure 28: Damage Pattern IV - Case 1: Details and local coordinates on each part of (a) P1 and (b) P2 segment. (hα, mα) and (hβ2, mβ2) are the cell numbers associated with the observed lengths and heights.

Damage Pattern IV – Case 2

As depicted in Fig. 29, this case corresponds to a special case when the heights of two triangular damage zones at both corners are identical to that of the initial compressive damage zone.

For ease of derivation, as shown in Figure 29, the wall panel is divided into two segments.

Segment P1: Using the same derivation of the segment P2 of Damage Pattern IV-1 (Eq. 39),

$$K_{P1}^{IV2} = \sum_{x=0}^{m_\alpha} \frac{K_T K_{Ty} K_{Tc}}{[n\beta_1 - y(x)]K_T K_{Ty} + y(x) K_T K_{Tc} + (n - n\beta_1) K_{Ty} K_{Tc}} \tag{48}$$

$$(x) = \frac{h_\alpha}{m_\alpha}(m_\alpha - x)\text{for } x \in [0, m_\alpha] \tag{49}$$

Segment P2: Since β_1 and β_2 stand for the same compressive damage state, and thus lead to the same tangent stiffness under our assumption of constitutive rule, this study regards this segment P2 to be under Damage Pattern II. Hence,

$$K_{P2}^{IV2} = \frac{m-m_\alpha}{n}\left(\frac{K_{Tc}K_T}{\beta_1 K_T+(1-\beta_1)K_{Tc}}\right). \tag{50}$$

Note that the numerator $(m - m_\alpha)$ means the number (width) of the horizontal cells in segment P2. In sum, the overall axial stiffness of the wall panel under DP IV-Case 2 is

$$\textbf{Damage Pattern IV} - \textbf{Case 2}: \qquad K = K_{P1}^{IV2} + K_{P2}^{IV2} \tag{51}$$

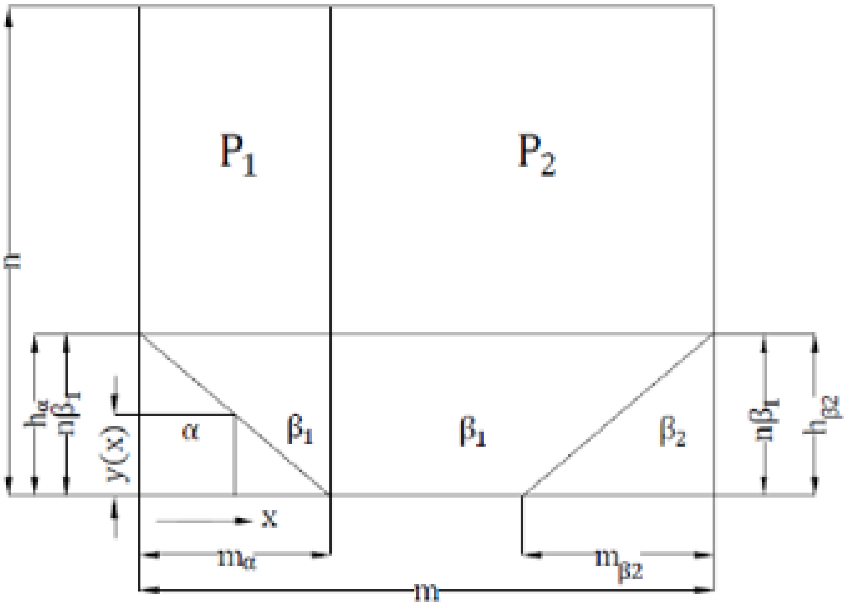


Figure 29: Special case when h β 1 = h β 2 = hα. (hα, mα) and (hβ2, mβ2) are the cell numbers from the observed lengths.

Damage Pattern IV – Case 3

This case deals with the damage state when $h_{\beta 1} > h_{\beta 2}$ and $h_{\beta 1} > h_\alpha$.

As the DP IV – Case 2, the wall panel is divied into two segments (Fig. 30).

Segment P1: Using the same approach as P2 of DP IV - Case 1,

$$K_{P1}^{IV3} = \sum_{x=0}^{m_\alpha} \frac{K_T K_{Ty} K_{Tc}}{[n\beta_1 - y(x)]K_T K_{Ty} + y(x) K_T K_{Tc} + (n - n\beta_1) K_{Ty} K_{Tc}}, \tag{52}$$

where $y(x)$, the vertical cell height of the α zone, is given by

$$y(x) = \frac{h_\alpha}{m_\alpha}(m_\alpha - x) \text{ for } x \in [0, m_\alpha]. \tag{53}$$

Segment P2: This case can be considered as a horizontal compressive damage state as DP II with horizontal width of $(m - m_\alpha)$. Thus,

$$K_{P2}^{IV3} = \frac{m - m_\alpha}{n} \times \left(\frac{K_{Tc} K_T}{\beta_1 K_T + (1 - \beta_1) K_{Tc}}\right). \tag{54}$$

In sum, the overall axial stiffness of the wall panel under DP IV-Case 3 reads

$$\textbf{Damage Pattern IV} - \textbf{Case 3}: \quad K = K_{P1}^{IV3} + K_{P2}^{IV3} \tag{55}$$

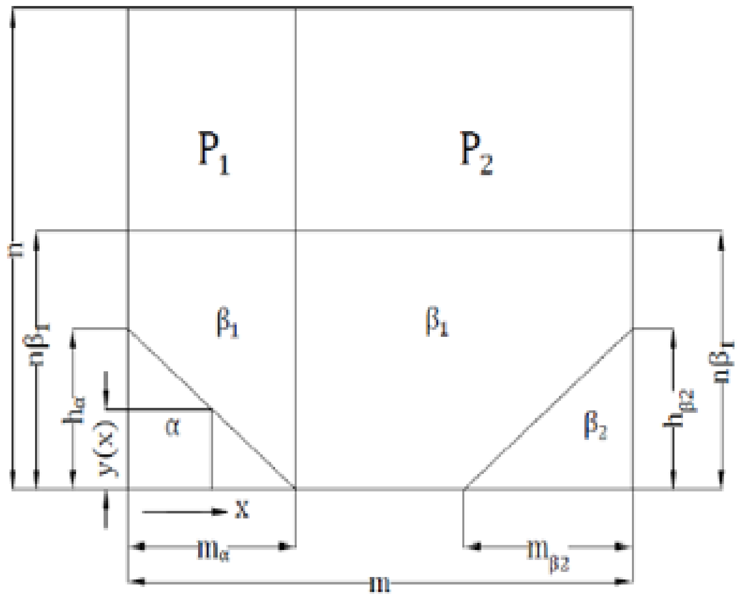


Figure 30: Damage Pattern IV – Case 3: Extensive initial compressive damage. (hα, mα) and (hβ2, mβ2) are the cell numbers associated with the observed lengths and heights

Damage Pattern V: Combination of Damage Patterns I and III

This pattern deals with a combination of DP I and DP III, possibly caused by bi-directional loads (Fig. 31). Initially, the wall panel has been displaced by an out-of-plane load (Fig. 31a) and then loaded by in-plane loading in the other direction (Fig. 31b). Figs. 17c-e show three possible sub-cases on the web panel (dashed box in Fig. 31b). Following the similar formulation of Damage Pattern IV, the total stiffness of this pattern's three cases is easily obtained. For brevity, the final formulas are summarized in APPENDIX C.

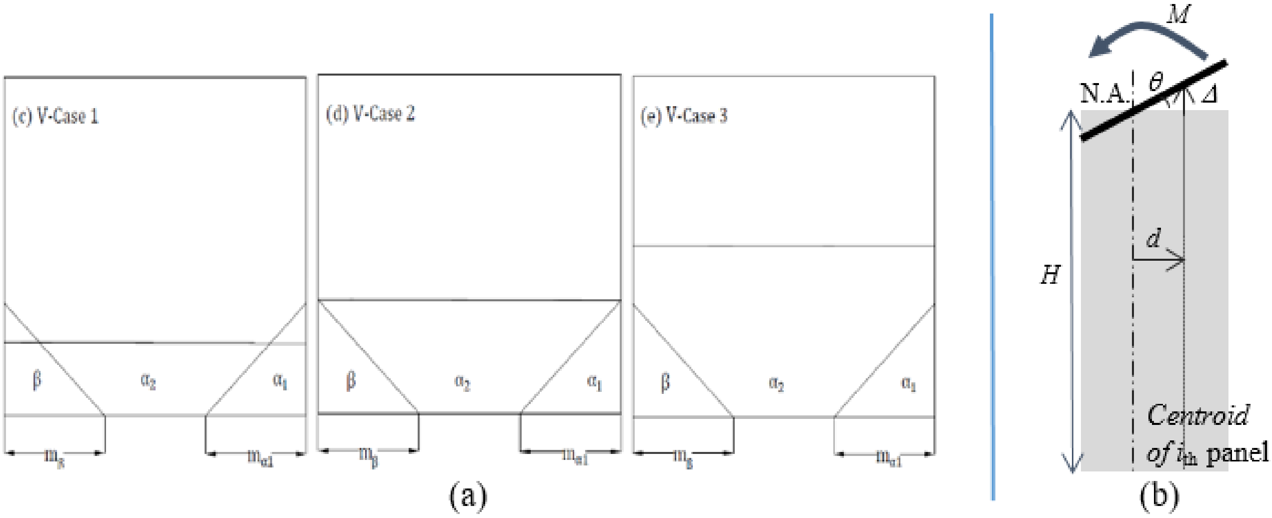


Figure 31: (a) Three possible cases of Damage Pattern V under bi-directional loads; (b) Schematic of the flexural deformation of a panel

4.3 Estimation of Remaining Stiffness Ratio

At time t, $K_{axial}(t)$ denotes the tangent axial stiffness of the entire U-shaped wall consisting of n_p panels; $K_i(t)$ denotes the i_{th} panel's tangent axial stiffness calculated by the formulas; $A_{cell,i}$ is the cross section of one unit cell (= unit cell size × the thickness of the panel). It should be noted that $A_{cell,i}$ is not the cross sectional area of a panel since in the derivation of $K_i(t)$ the panel length is already considered by means of the number of total horizontal cells

m (= wall length / unit cell size). Then, the remaining axial stiffness ratio of a U-shaped wall is

$$\text{Axial Stiffness Ratio} \equiv \frac{K_{axial}(t)}{K_{axial}(t=0)}, \tag{56}$$

$$K_{axial}(t) = \frac{1}{H}\sum_{i=1}^{n_p} K_i(t)A_{cell,i}, \tag{57}$$

where H is the wall height. Next, the ratio of the tangent flexual stiffnesses can be obtained from the Bernoulli beam assumption. As depicted in Fig. 30(b), when the infinisimal moment δM causes $\delta\theta$, the infinisimal axial deformation of the i_{th} panel is given by $d_i(t)\delta\theta$ where $d_i(t) = |\zeta_i - \zeta_{NA}(t)|$. Here, ζ_i is the distance from the N.A. to the i_{th} panel center while $\zeta_{NA}(t)$ is the distance from the datum to the N.A. Then, the moment equilibrium leads to

$$\delta M(t) = \frac{1}{H}\sum_{i=1}^{n_p} K_i(t)A_{cell,i} \times \underbrace{d_i(t)\delta\theta}_{axial\ deform.} \times d_i(t), \tag{58}$$

The datum can be any line parallel to the axis of bending, e.g., a side edge of wall system. $\zeta_{NA}(t)$ can be calculated by the sectional equilibrium as

$$\zeta_{NA}(t) = \frac{\sum_{i=1}^{n_p} K_i(t)A_{cell,i}\zeta_i}{\sum_{i=1}^{n_p} K_i(t)A_{cell,i}}. \tag{59}$$

This leads to the flexural stiffness ratio of the equivalent spring systems consisting of n_p panels,

$$\text{Flexural Stiffness Ratio} \equiv C_f \frac{K_{flexual}(t)}{K_{flexual}(t=0)}, \tag{60}$$

$$K_{flexural}(t) = \frac{1}{H}\sum_{i=1}^{n_p} K_i(t)A_{cell,i}d_i(t)^2. \tag{61}$$

Here, C_f is an empirical parameter. The following validations using three U-shaped walls suggest a range of [1.0, 2.5] within which VEEL prediction may reside. Other types of non-rectangular walls such as Box-shaped walls may call for additional investigations into the range of C_f. This parameter appears to be tied to some limitations of the cell network-based formulas.

The discrepancy may result from the Bernoulli beam assumption and the missing mechanisms (e.g., 3D nonlinear shear, interlocking mechanisms, three-dimensional damage patterns, confinement effect, etc.). Still, as shall be addressed in the validation section, the overall trend of the formulas appears promising. Future extensions and sophistication of mechanisms shall narrow the range of the empirical parameter C_f.

4.4 Sensitivity Study of Unit Cell Size

This section briefly investigates the sensitivity of the unit cell size. In general, the choice of the cell size would be arbirary, but the present study seeks to offer a reasonable recommendation for the practical use. This study considers various unit cell sizes: 1 mm, 10 mm, 50 mm, 100 mm, and 200 mm. Fig. 32 summarizes variation of remaining axial stiffness of a sample U-shaped wall under reserved, cyclic uni-directional loads (similar to Wall 1). Horizontal axis of Fig. 32 shows the peak drifts of primary displacements in the Y direction. With a rough cell size (e.g., larger than 100 mm), the calculated stiffness tends to increase substantially. However, for smaller sizes (in 10 mm and 1 mm cases), the sensitivity appears to become ignorable. Although the proposed formulas do not require expensive computational cost, less computation time is always preferrable. Also, in light of a future extension with an aggregate interlocking mechanism [16] for nonlinear shear, keeping unit cell size as

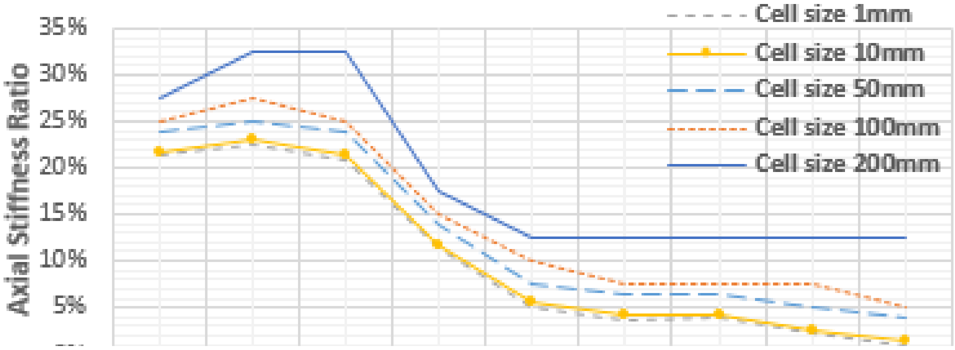


Figure 32: Variation of axial stiffness ratio of a U-shaped wall with varying unit cell size.

comparable to the real aggregate sizes (10-20 mm) appears attractable. Thus, this study recommends 10 mm, and used it throughout all validations. This implies that for a wall (length = 1.5 m, height = 3.6 m), the unit cell network requires (m = 150, n = 360) 54,000 cells.

4.5 Validation of Unit Cell-Based Formulas

For validations with various loading patterns, this study applied the cell network-based formulas to three U-shaped walls, denoted as Wall 1, Wall 3, and TUB (for detailed experimental information, see [23] for Wall 1 and Wall 3; see [25] for TUB). Fig. 33(a) depicts a 4-panel system for a simple realization of a U-shaped wall. Each panel is replaced with unit cell network with the size of 10 mm. Since the initial neutral axis (N.A.) coincides with the web center, two panels are used for the web to account for the web's contribution to the flexural stiffness about the X axis. Table 3 summarizes the key properties of unit cell network used for the validations. The three U-shaped walls differ in loading history. Loading history of Wall 1 consists of reversed cyclic displacements in the Y direction while Wall 3 is bi-directional, two-staged loads (initial X-displacement followed by the Y-displacement). TUB has the most complex "clover leaf" load patterns consisting of multiple one-directional loadings followed by diagonal displacements.

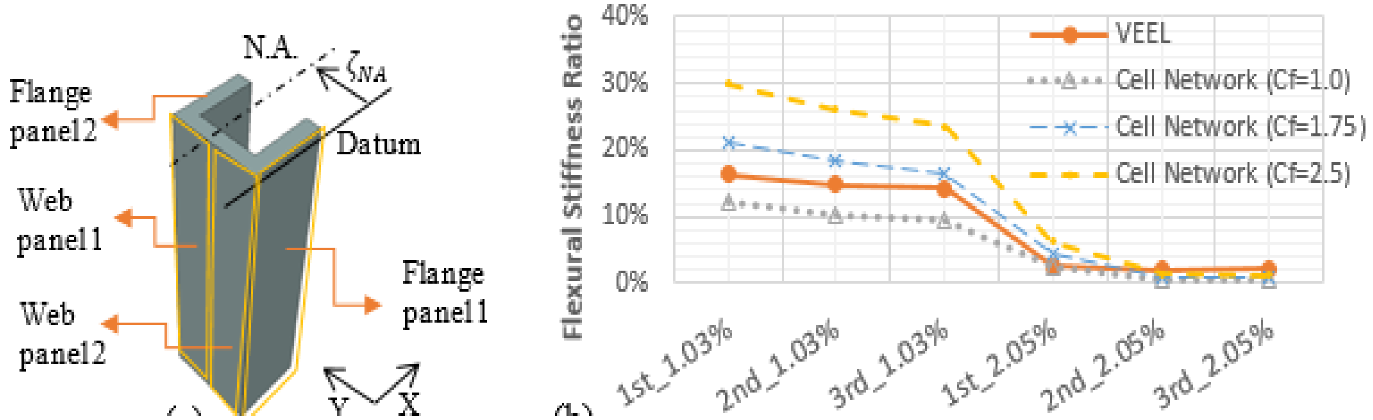
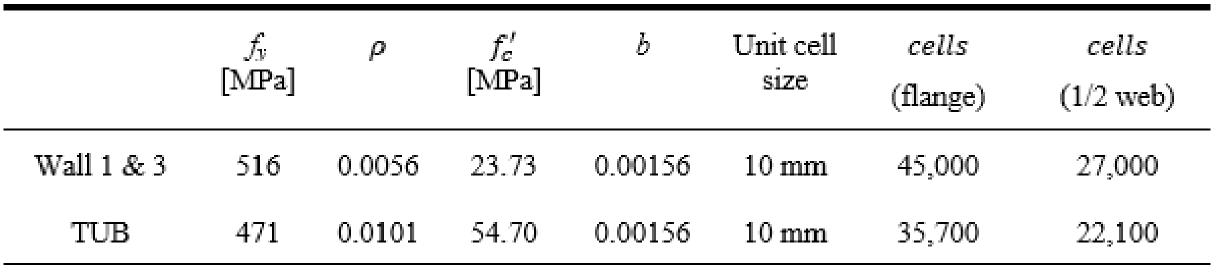


Figure 33: (a) Four-panel system of unit cell network; (b) Variation of the flexural stiffness ratios calculated by VEEL (bold line) and by the proposed formulas (dashed line).

Table 4: Unit Cell Properties for Wall 1, Wall 2, and TUB

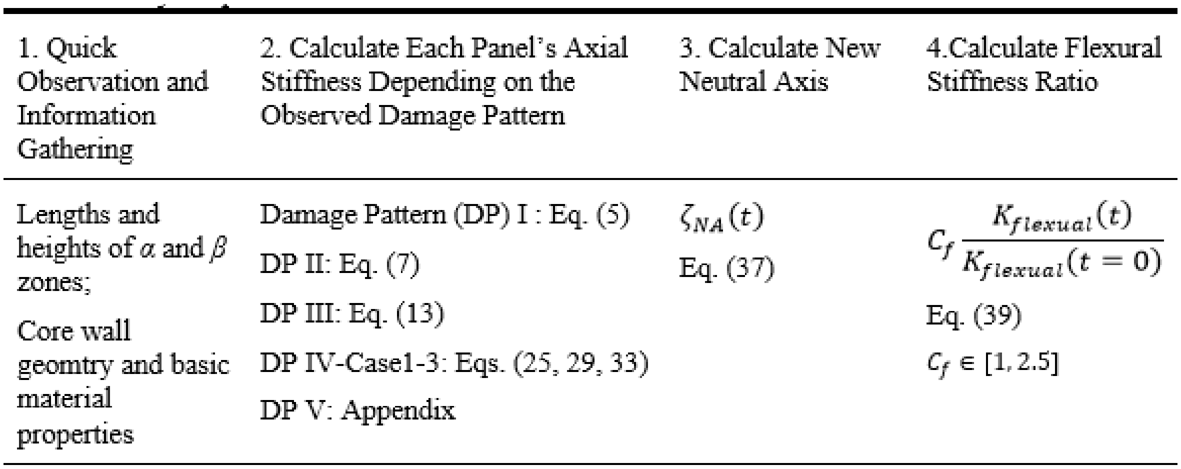


	f_y [MPa]	ρ	f_c' [MPa]	b	Unit cell size	*cells* (flange)	*cells* (1/2 web)
Wall 1 & 3	516	0.0056	23.73	0.00156	10 mm	45,000	27,000
TUB	471	0.0101	54.70	0.00156	10 mm	35,700	22,100

It should be noted that the flexural stiffness under consideration is "tangent" stiffness at specific time during the complex loading excursions. The stiffness is not a secant stiffness or is not from simplified monotonic loadings. VEEL provides comprehensive damage information (e.g., yielding, buckling, cracking, and crushing) of the entire walls at any specific load step as well as the tangent stiffness.

Wall 1 under Uni-Directional Loading. Since Wall 1 is under cyclic uni-axial displacements (parallel to web), two flanges exhibit DP I and DP II while the web panels undergo DP III. Simulation results of VEEL are used to quickly collect lengths and heights associated with the α and β zones (i.e., tensile and compressive damage zones). Then, these values are plugged into the formulas to calculate the remaining flexural stiffness ratio of U-shaped wall. Key procedures are summarized in Table 4. Despite the simplicity of the formulas, the overall trend by the formulas appears encouraging (Fig. 33).

Table 5: Key Steps for Flexural Stiffness Ratio Calculation



1. Quick Observation and Information Gathering	2. Calculate Each Panel's Axial Stiffness Depending on the Observed Damage Pattern	3. Calculate New Neutral Axis	4.Calculate Flexural Stiffness Ratio
Lengths and heights of α and β zones; Core wall geomtry and basic material properties	Damage Pattern (DP) I : Eq. (5) DP II: Eq. (7) DP III: Eq. (13) DP IV-Case1-3: Eqs. (25, 29, 33) DP V: Appendix	$\zeta_{NA}(t)$ Eq. (37)	$C_f \frac{K_{flexual}(t)}{K_{flexual}(t=0)}$ Eq. (39) $C_f \in [1, 2.5]$

Wall 3 under Bi-directional Loading. This validation focuses on the bi-directional loading conditions. Wall 3 is under two-phased bi-directional loading. By the X-directional loading, DP II emerges on the web panel (Figs. 34a and 34c) while the two flange panels undergo DP III. At the second loading phase, the loading is changed to the Y direction, and the web panel undergoes DP IV-Case 1. Figs. 34(c) and (d) show the VEEL's strain contour plots (inferred from longitudinal bar strains) of the web at the initial loading and at the last step of 40 mm (marked at Fig. 34e). The compressive damage at the web bottom is assumed to remain (since strain <-0.003), leading to DP IV-Case 1. Fig. 34(e) compares the final calculations.

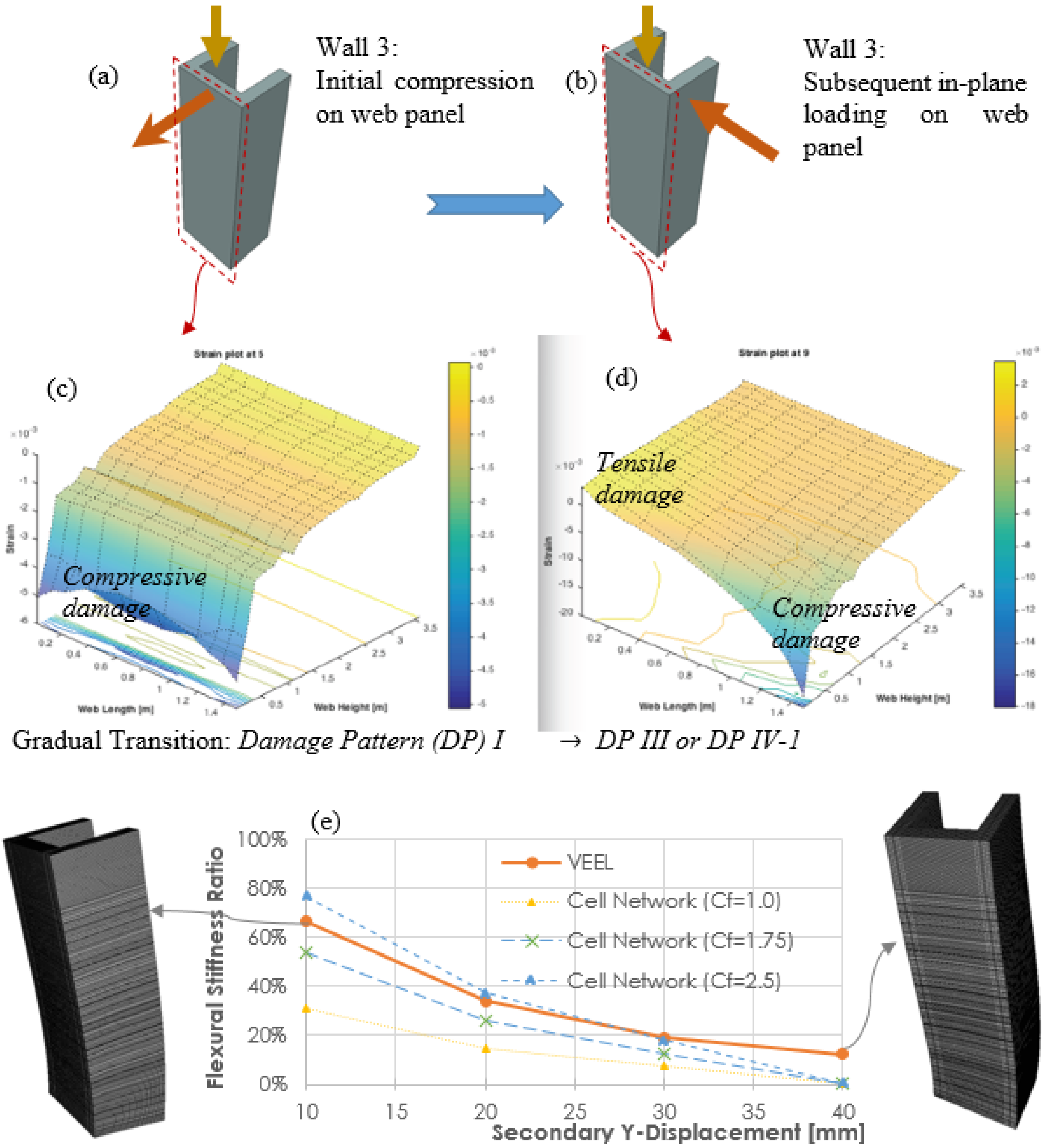


Figure 34: Wall 3: (a) Initial Loading in X; (b) Secondary Loading in Y; (c) Web panel strain contour plots at the beginneing of Y-load and (d) at 40 mm (vertical axis means strains of primary longitudinal bars); (e) Degrading flexural stiffness ratios calculated by VEEL and unit cell-based formulas. Two insets show the deformed shapes at the marked steps from VEEL (scale factor = 20).

TUB under Complex Clover-Leaf Loading Pattern. The most complex case, TUB consists of uni-directional loadings in both horizontal directions followed by diagonal loadings, which form the so-called “clover leaf” patterns (inset of Fig. 35). This validation summarizes the flexural stiffness reductions in the horizontal direction parallel to the web (i.e., East-West direction in Beyer *et al.*’s work). Fig. 35 compares the formula-based calculations against those from VEEL. Notwithstanding the clover leaf load patterns of the TUB, the formula’s quick reproduction of the stiffness reduction appears encouraging.

In the three validations, the overall flexural stiffness reduction trend appears to be reasonably reproduced by the formulas. Although the formulas appear to underestimate the remaining flexural stiffness of VEEL, the discrepancy (Fig. 33b and Fig. 34e) may be linked to some missing physical mechanisms and simplifications. Still, the proposed formulas appear to be promising in light of the cheap computational cost and the underlying simplicity.

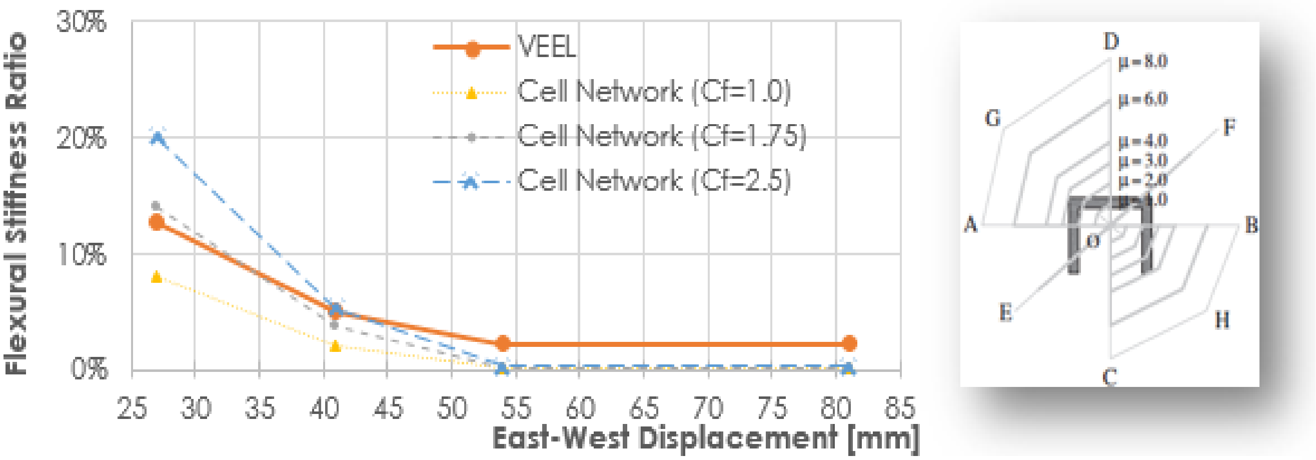


Figure 35: TUB: Degrading flexural stiffness ratios calculated by VEEL and unit cell-based formulas. Inset shows the clover-leaf loading pattern cited from [25].

CHAPTER 5. LIMITATION AND FUTURE WORK

Now the multi-target model has only been applied on curve prediction of rectangular shear wall. However, whether it can be applied to other shaped shear wall will be an interesting topic.

Also, the remedy strategies provided in the paper is not strong and reliable. A lot of empirical judgement is required to remedy prediction. A more stable and stronger remedy strategy should be suggested if the multi-target model is utilized in practice. And the choices of predictors adopted in the model will be very interesting. The effect of different combination of predictors on the prediction is not clear. Further uncertainty investigation should be addressed. Furthermore, the ends of predicted curve are chosen manually regarding to maximum and minimum displacements. Maybe two more predictors need be added including positive ending and negative ending of the curve. Finally, there is no doubt that the size of database will have significant influence on predictions. The extension of database strengthen the power of the regression model.

Meantime, how 2D cell networks for 2D behavior is also an interesting topic. To understand the impact of 3D behavior on the 2D cell network-based formulas, a well-documented rectangular wall (dubbed as RW1 by Thomsen IV and Wallace, 2004) is analyzed by the formulas and VEEL. For modeling RW1, two panels are used for the web part, and two flange panels are additionally used for the boundary elements. Fig. 36(a) confirms the reasonable accuracy of the formulas. Unlike the 3D U-shaped wall cases, $C_f = 1.0$ appears sufficient for this 2D rectangular wall. The 3D U-shaped walls require a range of C_f (i.e., [1.0, 2.5]) within which VEEL prediction may reside. The formulas appear to underestimate the flexural stiffness reductions of 3D U-shaped walls. This suggests that future study needs to incorporate 3D factors such as interaction between web and flanges, out-of-plate deformations, 3D shear deformation, and so on into the cell network-based formulas.

Confinement Effect: Increased amount of horizontal reinforcement in the boundary elements substantially influence the core concrete’s resistance. Such confinement effects may be taken into account by increasing the concrete spring strengths of the unit cells associated with the boundary elements. As an instance, the unit cells at the boundary elements of RW1 are gradually increased to study the trends. Fig. 36(b) summarizes the variation of the stiffness reduction calculated by the formulas. The ratio $f'_{c,confined}/f'_{c,unconfined}$ varies from 1.0 to 2.0. Comparison of slopes in Fig. 36(b) suggests that the larger confinement, the slower loss of flexural stiffness. This slight modification may help incorporate the confinement effect into the present formulas. However, the confinement effect is indeed 3D behavior between bars and concrete, and also a complex function of the reinforcement amount and spacing of horizontal bars, the strengths of concrete and vertical bars, and so on. Therefore, future extensions shall incorporate sophisticated derivations of confinement into the cell network-based formulas.

In addition to the aforementioned limitations, there exist other restrictions including the assumption of five Damage Patterns, no inclusion of nonlinear shear and interlocking, a small number of panels of cell network. In the future extensions, these limitations should be resolved.

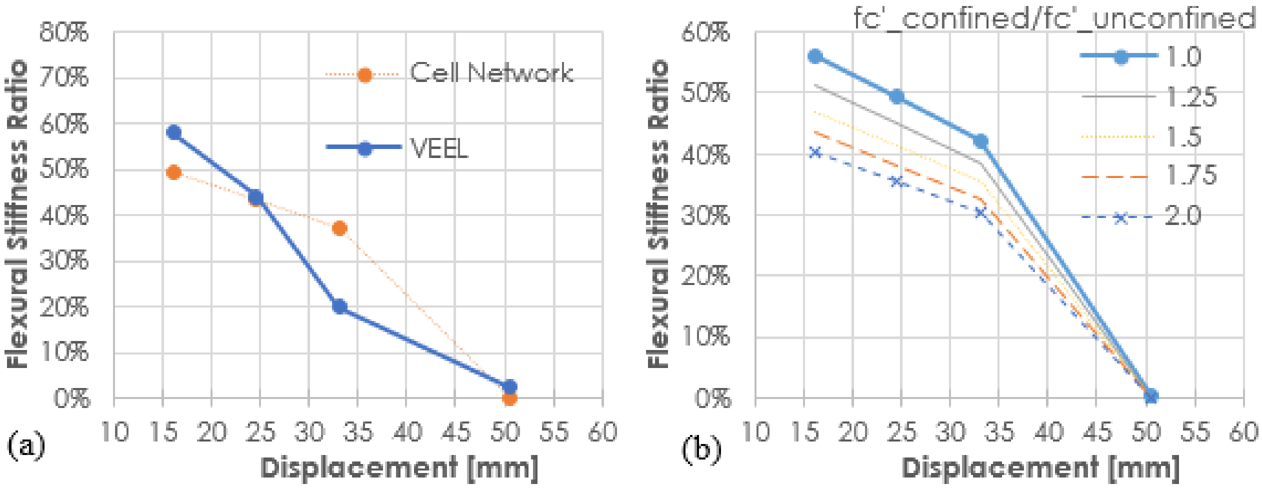


Figure 36: Rectangular Wall RW1: (a) Degrading flexural stiffness ratios calculated by VEEL and unit cell-based formulas (Cf = 1.0 used); (b) Variation of stiffness reduction trends with increasing confinement effect.

CHAPTER 6. CONCLUSIONS

Prediction of force-displace skeleton curve with multi-target prediction system CLUS is innovative and promising in structural engineering domain. The accuracy of the prediction highly depends on properties of training data. If training data contains similar shear wall information with test data, the model is able to give out very reliable prediction after remedy. However, if it performs prediction for test data which is completely new shear wall data, then the accuracy of prediction is away from what it should be. And as discussed before, the multi-target model will perform better if adding new instances samples into training data. At this moment, the shear wall database has a very small size (187 instances). In order to enhance the performance of multi-target model, more sample data of different rectangular shear wall are required.

Also, in hopes of providing an efficient and reliable tool that can help quickly estimate the remaining flexural stiffness of U-shaped core walls, this thesis developed formulas based on a novel unit cell network. Compared to the high-precision multiscale finite element analyses, the formulas' performance in quickly assessing the flexural stiffness reductions with minimal observational information appears promising. Future extensions should address more damage patterns, consider other non-rectangular core walls, and incorporate more microphysical mechanisms into the formulation. All the *Matlab* codes are available at [26].

REFERENCES

[1] Dazio, A., Beyer, K., and Bachmann, H., 2009. *Quasi-static cyclic tests and plastic hinge analysis of RC structural walls, Engineering Structures* **31**, 1556-1571.

[2] Beyer, K., Dazio, A., Priestley, M.J.N., 2008. Quasi-Static Cyclic Tests of Two U-Shaped Reinforced Concrete Walls, *Journal of Earthquake Engineering* **12**, 1023-1053.

[3] Chou, Jui-sheng, D. Ph, Ngoc-tri Ngo, Anh-duc Pham, and D. Ph. 2012. Shear Strength Prediction in Reinforced Concrete Deep Beams Using Nature-Inspired Metaheuristic Support Vector Regression, *Journal of Computing in Civil Engineering*, 30(1):1–9.

[4] Pal, Mahesh and Surinder Deswal. 2011. Support Vector Regression Based Shear Strength Modelling of Deep Beams, *Computers and Structures*, 89(13–14):1430–39.

[5] Aguilar, Vactor, Cristia Sandoval, Jose M. Adam, Julio Garzan-Roca, and Galo Valdebenito. 2016. Prediction of the Shear Strength of Reinforced Masonry Walls Using a Large Experimental Database and Artificial Neural Networks, *Structure and Infrastructure Engineering*, 12(12):1661–74.

[6] Chou, Jui Sheng and Anh Duc Pham. 2013. Enhanced Artificial Intelligence for Ensemble Approach to Predicting High Performance Concrete Compressive Strength, *Construction and Building Materials,* 49:554–63.

[7] Johari, a, G. Habibagahi, and a Ghahramani. 2006. Prediction of Soil – Water Characteristic Curve Using Genetic Programming, *Journal of Geotechnical and Geoenvironmental.* 132(5).

[8] Van Gent, Marcel R. A. and Henk F. P. van den Boogaard. 1998. Neural Network Modelling of Forces on Vertical Structures, *Coastal Engineering Proceedings,* (26):2096–2123.

[9] Rafiq, M. Y., G. Bugmann, and D. J. Easterbrook. 2001. Neural Network Design for Engineering Applications, *Computers and Structures*, 79(17):1541–52.

[10] Dibike, Yb, Slavco Velickov, and Dimitri Solomatine. 2000. Support Vector Machines: Review and Applications in Civil Engineering, *Proc. of the 2nd Joint Workshop on Application of AI in Civil Engineering,* 215–18.

[11] Worden, Keith and Graeme Manson. 2007. The Application of Machine Learning to Structural Health Monitoring, *Philosophical Transactions. Series A, Mathematical, Physical, and Engineering Sciences,* 365(1851):515–37.

[12] Struyf, Jan & Zenko, Bernar & Blockeel, Hendrik & Vens, Celine & Džeroski, Sašo. 2018. Clus: User's Manual. [https://www.researchgate.net/publication/265243854_Clus_User's_Manual]

[13] Blockeel, Hendrik, Luc De Raedt, and Jan Ramon. 2000. Top-down Induction of Clustering Trees, *Machine Learning, Proceedings of the 15th International Conference,* 55-63.

[14] Purvi, Amit, Paper, Conference. 2017. Improving the Performance of Predictive Clustering Tree Algorithm for Hierarchical Multi-Label Classification, *Proceedings of International Conference on Emerging Research in Computing, Information, Communication and Applications,* ERCICA-13

[15] Breiman, Leo. 2001. Random Forests, *Machine Learning* 45(1):5–32.

[16] Cho, In Ho. 2013. Virtual Earthquake Engineering Laboratory Capturing Nonlinear Shear, Localized Damage and Progressive Buckling of Bar, *Earthquake Spectra* 29(1):103–26.

[17] Cho, In Ho and Keith A. Porter. 2014. Multilayered Grouping Parallel Algorithm for Multiple-Level Multiscale Analyses, *International Journal for Numerical Methods in Engineering* 100(12):914–32. Retrieved (http://doi.wiley.com/10.1002/nme.4791).

[18] Efron, B. and R. J. Tibshirani. 1993. *An Introduction to the Bootstrap. Monographs on Statistics and Applied Probability, No. 57. Chapman and Hall, London, 436 P*

[19] Song, I. et al. 2018. Data-Driven Prediction of Runway Incursions with Uncertainty Quantification. 32(2).

[20] Yang, Yicheng, Sai Yemmaleni, Ikkyun Song, and In Ho Cho. 2017. Cell Network-Based. Formulas for Fast Determination of Stiffness Reduction of Non- Rectangular Core Shear Walls, *Earthquake Spectra In-Press*

[21] Cho, I., and Hall, J. F., 2014. General Confinement Model based on Nonlocal Information, *ASCE Journal of Engineering Mechanics* **140**(6). [10.1061/(ASCE)EM.1943-7889.0000724].

[22] Cho, I., 2017, Deformation Gradient-Based Remedy for Mesh Objective Three-Dimensional Interlocking Mechanism, *ASCE Journal of Engineering Mechanics* (in-press).

[23] Ile, N., and Reynouard, J. M., 2005. Behavior of U-Shaped Walls Subjected to Uniaxial and Biaxial Cyclic Lateral Loading, *Journal of Earthquake Engineering* **9**, 67-94.

[24] Cho, I., and Porter, K. A., 2013. Structure-Independent Parallel Platform for Nonlinear Analyses of General Real-Scale RC Structures under Cyclic Loading, *Journal of Structural Engineering*. 1-40.

[25] Beyer, K., Dazio, A., Priestley, M.J.N., 2008. Quasi-Static Cyclic Tests of Two U-Shaped Reinforced Concrete Walls, *Journal of Earthquake Engineering* **12**, 1023-1053.

[26] Cho, I., 2017. Research website: Data-Driven Computational Science and Engineering [https://sites.google.com/site/ichoddcse2017/home/Computational-Sci--Eng].

[27] Sai, Y., 2016. Determination of Stiffness Reduction Factor for U-shaped Reinforced Concrete Shear Walls under bi-axial loading, Thesis report.

[28] https://sites.google.com/site/ichoddcse2017/home/Computational-Sci--Eng/training-dataset-of-complex-rc-structures.

[29] SLDRCE Database on Static Tests of Structural Members and Joint Assemblies, State Key Laboratory of Disaster Reduction in Civil Engineering, Tongji University, Shanghai, China, September 2008, 169-188

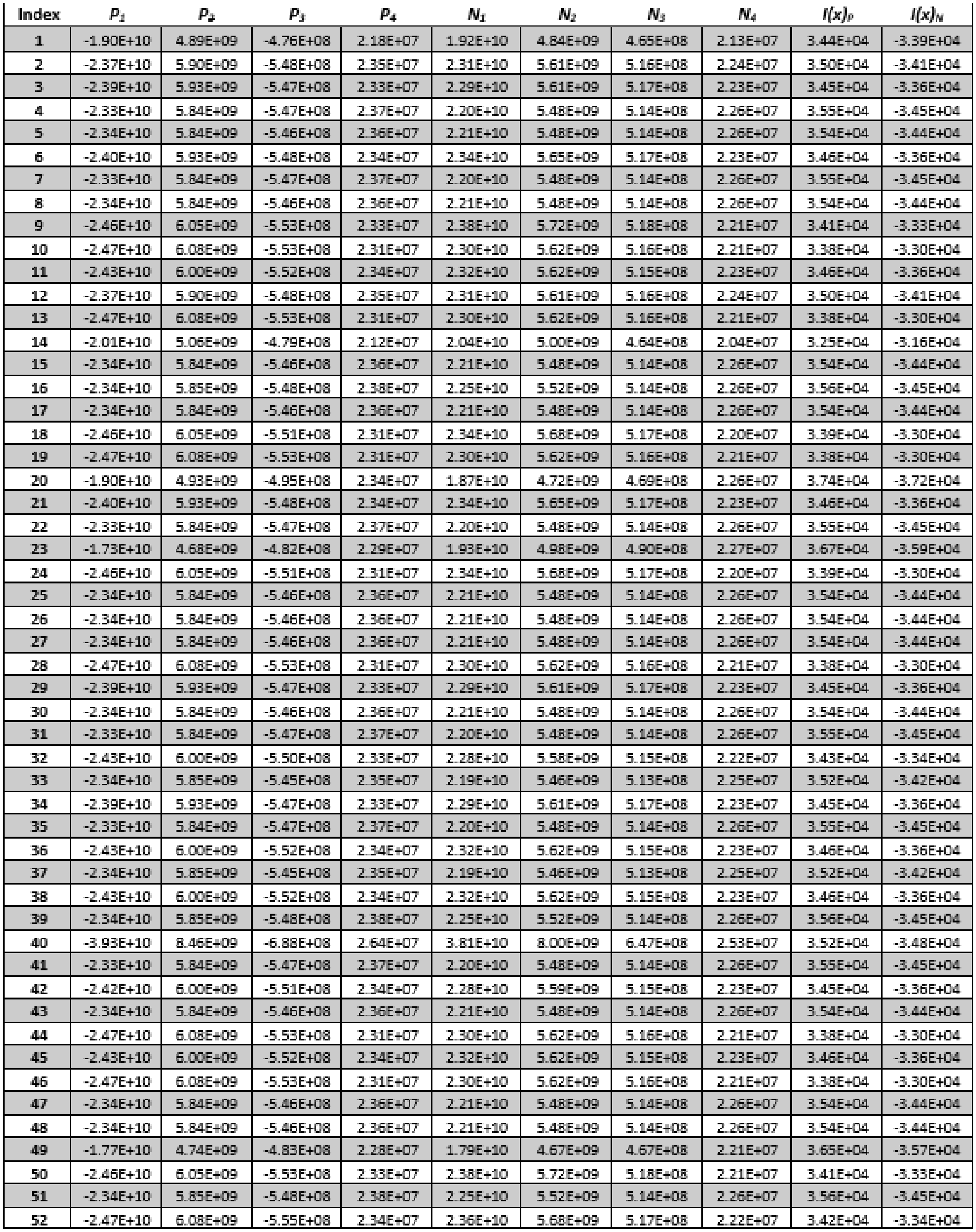
[30] Tiom, A and Bernard Z 2009. Rule Ensembles for Multi-Target Regression*, 2009 Ninth IEEE International Conference on Data Mining Rule*

[31] Jerome. H. Friedman and B. E. Popescu 2008. Predictive learning via rule ensembles, *The Annals of Applied Statistics*, 916–954.

APPENDIX A. ATTRIBUTES OF CAPACITY CURVE DATABASE

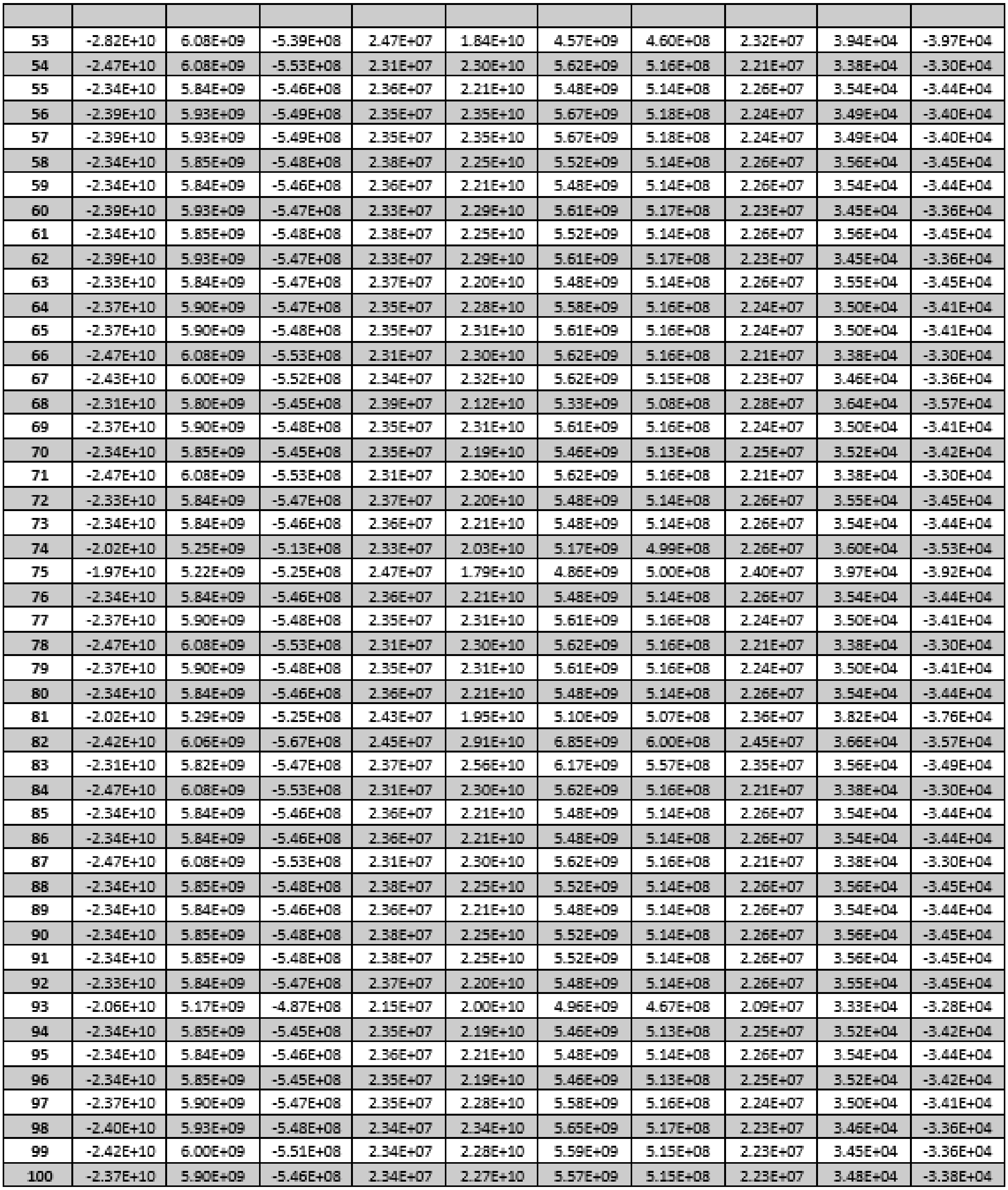
Attributes	Detail
I	Moment of Inertia
Length	Length of shear wall
Thickness	Thickness of shear wall
Height	Height of shear wall
Number of floors	Number of floors
Axial Force Ratio	Axial Force Ratio
Cover thickness	Cover thickness
Concrete_fc	Concrete compressive Strength
Concrete_ft	Concrete Yield Strength
bb	Width of boundary Element
hb	Thickness of boundary Element
cb	Cover Thickness in boundary element
Steel_Vertical1_fy	Yield Strength of boundary longitudinal reinforcement
Steel_Vertical1_fu	Ultimate Stress of boundary longitudinal reinforcement
Steel_Vertical1_Spacing	Spacing of boundary longitudinal reinforcement
Steel_Vertical1_strain at fu	Ultimate Strain of boundary longitudinal reinforcement
Steel_Vertical1_Diameter	Diameter of boundary longitudinal reinforcement
Steel_Vertical2_fy	Yield Strength of web longitudinal reinforcement
Steel_Vertical2_fu	Ultimate Stress of web longitudinal reinforcement
Steel_Vertical2_Diameter	Diameter of web longitudinal reinforcement
Steel_Horizontal1_fy	Yield Strength of boundary transverse reinforcement
Steel_Horizontal1_fu	Ultimate Stress of boundary transverse reinforcement
Steel_Horizontal1_strain at fu	Ultimate Strain of boundary transverse reinforcement
Steel_Horizontal1_Spacing	Spacing of boundary transverse reinforcement
Steel_Horizontal1_Diameter	Diameter of boundary transverse reinforcement
Steel_Stirrup1_fy	Yield Strength of stirrups
Steel_Stirrup1_fu	Ultimate Stress of stirrups
Steel_Stirrup1_strain at fu	Ultimate Strain of stirrups
Steel_Stirrup1_spacing	Spacing of stirrups
Steel_Stirrup1_Diameter	Diameter of stirrups
Number of longitudinal bars at wall boundary	Number of longitudinal bars at wall boundary
P_1	Polynomial function parameter
P_2	Polynomial function parameter
P_3	Polynomial function parameter
P_4	Polynomial function parameter
N_1	Polynomial function parameter
N_2	Polynomial function parameter
N_3	Polynomial function parameter
N_4	Polynomial function parameter

APPENDIX B. RESULT OF 100 INTERATIONS



Index	P_1	P_2	P_3	P_4	N_1	N_2	N_3	N_4	$I(x)_P$	$I(x)_N$
1	-1.90E+10	4.89E+09	-4.76E+08	2.18E+07	1.92E+10	4.84E+09	4.65E+08	2.13E+07	3.44E+04	-3.39E+04
2	-2.37E+10	5.90E+09	-5.48E+08	2.35E+07	2.31E+10	5.61E+09	5.16E+08	2.24E+07	3.50E+04	-3.41E+04
3	-2.39E+10	5.93E+09	-5.47E+08	2.33E+07	2.29E+10	5.61E+09	5.17E+08	2.23E+07	3.45E+04	-3.36E+04
4	-2.33E+10	5.84E+09	-5.47E+08	2.37E+07	2.20E+10	5.48E+09	5.14E+08	2.26E+07	3.55E+04	-3.45E+04
5	-2.34E+10	5.84E+09	-5.46E+08	2.36E+07	2.21E+10	5.48E+09	5.14E+08	2.26E+07	3.54E+04	-3.44E+04
6	-2.40E+10	5.93E+09	-5.48E+08	2.34E+07	2.34E+10	5.65E+09	5.17E+08	2.23E+07	3.46E+04	-3.36E+04
7	-2.33E+10	5.84E+09	-5.47E+08	2.37E+07	2.20E+10	5.48E+09	5.14E+08	2.26E+07	3.55E+04	-3.45E+04
8	-2.34E+10	5.84E+09	-5.46E+08	2.36E+07	2.21E+10	5.48E+09	5.14E+08	2.26E+07	3.54E+04	-3.44E+04
9	-2.46E+10	6.05E+09	-5.53E+08	2.33E+07	2.38E+10	5.72E+09	5.18E+08	2.21E+07	3.41E+04	-3.33E+04
10	-2.47E+10	6.08E+09	-5.53E+08	2.31E+07	2.30E+10	5.62E+09	5.16E+08	2.21E+07	3.38E+04	-3.30E+04
11	-2.43E+10	6.00E+09	-5.52E+08	2.34E+07	2.32E+10	5.62E+09	5.15E+08	2.23E+07	3.46E+04	-3.36E+04
12	-2.37E+10	5.90E+09	-5.48E+08	2.35E+07	2.31E+10	5.61E+09	5.16E+08	2.24E+07	3.50E+04	-3.41E+04
13	-2.47E+10	6.08E+09	-5.53E+08	2.31E+07	2.30E+10	5.62E+09	5.16E+08	2.21E+07	3.38E+04	-3.30E+04
14	-2.01E+10	5.06E+09	-4.79E+08	2.12E+07	2.04E+10	5.00E+09	4.64E+08	2.04E+07	3.25E+04	-3.16E+04
15	-2.34E+10	5.84E+09	-5.46E+08	2.36E+07	2.21E+10	5.48E+09	5.14E+08	2.26E+07	3.54E+04	-3.44E+04
16	-2.34E+10	5.85E+09	-5.48E+08	2.38E+07	2.25E+10	5.52E+09	5.14E+08	2.26E+07	3.56E+04	-3.45E+04
17	-2.34E+10	5.84E+09	-5.46E+08	2.36E+07	2.21E+10	5.48E+09	5.14E+08	2.26E+07	3.54E+04	-3.44E+04
18	-2.46E+10	6.05E+09	-5.51E+08	2.31E+07	2.34E+10	5.68E+09	5.17E+08	2.20E+07	3.39E+04	-3.30E+04
19	-2.47E+10	6.08E+09	-5.53E+08	2.31E+07	2.30E+10	5.62E+09	5.16E+08	2.21E+07	3.38E+04	-3.30E+04
20	-1.90E+10	4.93E+09	-4.95E+08	2.34E+07	1.87E+10	4.72E+09	4.69E+08	2.26E+07	3.74E+04	-3.72E+04
21	-2.40E+10	5.93E+09	-5.48E+08	2.34E+07	2.34E+10	5.65E+09	5.17E+08	2.23E+07	3.46E+04	-3.36E+04
22	-2.33E+10	5.84E+09	-5.47E+08	2.37E+07	2.20E+10	5.48E+09	5.14E+08	2.26E+07	3.55E+04	-3.45E+04
23	-1.73E+10	4.68E+09	-4.82E+08	2.29E+07	1.93E+10	4.98E+09	4.90E+08	2.27E+07	3.67E+04	-3.59E+04
24	-2.46E+10	6.05E+09	-5.51E+08	2.31E+07	2.34E+10	5.68E+09	5.17E+08	2.20E+07	3.39E+04	-3.30E+04
25	-2.34E+10	5.84E+09	-5.46E+08	2.36E+07	2.21E+10	5.48E+09	5.14E+08	2.26E+07	3.54E+04	-3.44E+04
26	-2.34E+10	5.84E+09	-5.46E+08	2.36E+07	2.21E+10	5.48E+09	5.14E+08	2.26E+07	3.54E+04	-3.44E+04
27	-2.34E+10	5.84E+09	-5.46E+08	2.36E+07	2.21E+10	5.48E+09	5.14E+08	2.26E+07	3.54E+04	-3.44E+04
28	-2.47E+10	6.08E+09	-5.53E+08	2.31E+07	2.30E+10	5.62E+09	5.16E+08	2.21E+07	3.38E+04	-3.30E+04
29	-2.39E+10	5.93E+09	-5.47E+08	2.33E+07	2.29E+10	5.61E+09	5.17E+08	2.23E+07	3.45E+04	-3.36E+04
30	-2.34E+10	5.84E+09	-5.46E+08	2.36E+07	2.21E+10	5.48E+09	5.14E+08	2.26E+07	3.54E+04	-3.44E+04
31	-2.33E+10	5.84E+09	-5.47E+08	2.37E+07	2.20E+10	5.48E+09	5.14E+08	2.26E+07	3.55E+04	-3.45E+04
32	-2.43E+10	6.00E+09	-5.50E+08	2.33E+07	2.28E+10	5.58E+09	5.15E+08	2.22E+07	3.43E+04	-3.34E+04
33	-2.34E+10	5.85E+09	-5.45E+08	2.35E+07	2.19E+10	5.46E+09	5.13E+08	2.25E+07	3.52E+04	-3.42E+04
34	-2.39E+10	5.93E+09	-5.47E+08	2.33E+07	2.29E+10	5.61E+09	5.17E+08	2.23E+07	3.45E+04	-3.36E+04
35	-2.33E+10	5.84E+09	-5.47E+08	2.37E+07	2.20E+10	5.48E+09	5.14E+08	2.26E+07	3.55E+04	-3.45E+04
36	-2.43E+10	6.00E+09	-5.52E+08	2.34E+07	2.32E+10	5.62E+09	5.15E+08	2.23E+07	3.46E+04	-3.36E+04
37	-2.34E+10	5.85E+09	-5.45E+08	2.35E+07	2.19E+10	5.46E+09	5.13E+08	2.25E+07	3.52E+04	-3.42E+04
38	-2.43E+10	6.00E+09	-5.52E+08	2.34E+07	2.32E+10	5.62E+09	5.15E+08	2.23E+07	3.46E+04	-3.36E+04
39	-2.34E+10	5.85E+09	-5.48E+08	2.38E+07	2.25E+10	5.52E+09	5.14E+08	2.26E+07	3.56E+04	-3.45E+04
40	-3.93E+10	8.46E+09	-6.88E+08	2.64E+07	3.81E+10	8.00E+09	6.47E+08	2.53E+07	3.52E+04	-3.48E+04
41	-2.33E+10	5.84E+09	-5.47E+08	2.37E+07	2.20E+10	5.48E+09	5.14E+08	2.26E+07	3.55E+04	-3.45E+04
42	-2.42E+10	6.00E+09	-5.51E+08	2.34E+07	2.28E+10	5.59E+09	5.15E+08	2.23E+07	3.45E+04	-3.36E+04
43	-2.34E+10	5.84E+09	-5.46E+08	2.36E+07	2.21E+10	5.48E+09	5.14E+08	2.26E+07	3.54E+04	-3.44E+04
44	-2.47E+10	6.08E+09	-5.53E+08	2.31E+07	2.30E+10	5.62E+09	5.16E+08	2.21E+07	3.38E+04	-3.30E+04
45	-2.43E+10	6.00E+09	-5.52E+08	2.34E+07	2.32E+10	5.62E+09	5.15E+08	2.23E+07	3.46E+04	-3.36E+04
46	-2.47E+10	6.08E+09	-5.53E+08	2.31E+07	2.30E+10	5.62E+09	5.16E+08	2.21E+07	3.38E+04	-3.30E+04
47	-2.34E+10	5.84E+09	-5.46E+08	2.36E+07	2.21E+10	5.48E+09	5.14E+08	2.26E+07	3.54E+04	-3.44E+04
48	-2.34E+10	5.84E+09	-5.46E+08	2.36E+07	2.21E+10	5.48E+09	5.14E+08	2.26E+07	3.54E+04	-3.44E+04
49	-1.77E+10	4.74E+09	-4.83E+08	2.28E+07	1.79E+10	4.67E+09	4.67E+08	2.21E+07	3.65E+04	-3.57E+04
50	-2.46E+10	6.05E+09	-5.53E+08	2.33E+07	2.38E+10	5.72E+09	5.18E+08	2.21E+07	3.41E+04	-3.33E+04
51	-2.34E+10	5.85E+09	-5.48E+08	2.38E+07	2.25E+10	5.52E+09	5.14E+08	2.26E+07	3.56E+04	-3.45E+04
52	-2.47E+10	6.08E+09	-5.55E+08	2.34E+07	2.36E+10	5.68E+09	5.17E+08	2.22E+07	3.42E+04	-3.34E+04

APPENDIX B. CONTINUED



53	-2.82E+10	6.08E+09	-5.39E+08	2.47E+07	1.84E+10	4.57E+09	4.60E+08	2.32E+07	3.94E+04	-3.97E+04
54	-2.47E+10	6.08E+09	-5.53E+08	2.31E+07	2.30E+10	5.62E+09	5.16E+08	2.21E+07	3.38E+04	-3.30E+04
55	-2.34E+10	5.84E+09	-5.46E+08	2.36E+07	2.21E+10	5.48E+09	5.14E+08	2.26E+07	3.54E+04	-3.44E+04
56	-2.39E+10	5.93E+09	-5.49E+08	2.35E+07	2.35E+10	5.67E+09	5.18E+08	2.24E+07	3.49E+04	-3.40E+04
57	-2.39E+10	5.93E+09	-5.49E+08	2.35E+07	2.35E+10	5.67E+09	5.18E+08	2.24E+07	3.49E+04	-3.40E+04
58	-2.34E+10	5.85E+09	-5.48E+08	2.38E+07	2.25E+10	5.52E+09	5.14E+08	2.26E+07	3.56E+04	-3.45E+04
59	-2.34E+10	5.84E+09	-5.46E+08	2.36E+07	2.21E+10	5.48E+09	5.14E+08	2.26E+07	3.54E+04	-3.44E+04
60	-2.39E+10	5.93E+09	-5.47E+08	2.33E+07	2.29E+10	5.61E+09	5.17E+08	2.23E+07	3.45E+04	-3.36E+04
61	-2.34E+10	5.85E+09	-5.48E+08	2.38E+07	2.25E+10	5.52E+09	5.14E+08	2.26E+07	3.56E+04	-3.45E+04
62	-2.39E+10	5.93E+09	-5.47E+08	2.33E+07	2.29E+10	5.61E+09	5.17E+08	2.23E+07	3.45E+04	-3.36E+04
63	-2.33E+10	5.84E+09	-5.47E+08	2.37E+07	2.20E+10	5.48E+09	5.14E+08	2.26E+07	3.55E+04	-3.45E+04
64	-2.37E+10	5.90E+09	-5.47E+08	2.35E+07	2.28E+10	5.58E+09	5.16E+08	2.24E+07	3.50E+04	-3.41E+04
65	-2.37E+10	5.90E+09	-5.48E+08	2.35E+07	2.31E+10	5.61E+09	5.16E+08	2.24E+07	3.50E+04	-3.41E+04
66	-2.47E+10	6.08E+09	-5.53E+08	2.31E+07	2.30E+10	5.62E+09	5.16E+08	2.21E+07	3.38E+04	-3.30E+04
67	-2.43E+10	6.00E+09	-5.52E+08	2.34E+07	2.32E+10	5.62E+09	5.15E+08	2.23E+07	3.46E+04	-3.36E+04
68	-2.31E+10	5.80E+09	-5.45E+08	2.39E+07	2.12E+10	5.33E+09	5.08E+08	2.28E+07	3.64E+04	-3.57E+04
69	-2.37E+10	5.90E+09	-5.48E+08	2.35E+07	2.31E+10	5.61E+09	5.16E+08	2.24E+07	3.50E+04	-3.41E+04
70	-2.34E+10	5.85E+09	-5.45E+08	2.35E+07	2.19E+10	5.46E+09	5.13E+08	2.25E+07	3.52E+04	-3.42E+04
71	-2.47E+10	6.08E+09	-5.53E+08	2.31E+07	2.30E+10	5.62E+09	5.16E+08	2.21E+07	3.38E+04	-3.30E+04
72	-2.33E+10	5.84E+09	-5.47E+08	2.37E+07	2.20E+10	5.48E+09	5.14E+08	2.26E+07	3.55E+04	-3.45E+04
73	-2.34E+10	5.84E+09	-5.46E+08	2.36E+07	2.21E+10	5.48E+09	5.14E+08	2.26E+07	3.54E+04	-3.44E+04
74	-2.02E+10	5.25E+09	-5.13E+08	2.33E+07	2.03E+10	5.17E+09	4.99E+08	2.26E+07	3.60E+04	-3.53E+04
75	-1.97E+10	5.22E+09	-5.25E+08	2.47E+07	1.79E+10	4.86E+09	5.00E+08	2.40E+07	3.97E+04	-3.92E+04
76	-2.34E+10	5.84E+09	-5.46E+08	2.36E+07	2.21E+10	5.48E+09	5.14E+08	2.26E+07	3.54E+04	-3.44E+04
77	-2.37E+10	5.90E+09	-5.48E+08	2.35E+07	2.31E+10	5.61E+09	5.16E+08	2.24E+07	3.50E+04	-3.41E+04
78	-2.47E+10	6.08E+09	-5.53E+08	2.31E+07	2.30E+10	5.62E+09	5.16E+08	2.21E+07	3.38E+04	-3.30E+04
79	-2.37E+10	5.90E+09	-5.48E+08	2.35E+07	2.31E+10	5.61E+09	5.16E+08	2.24E+07	3.50E+04	-3.41E+04
80	-2.34E+10	5.84E+09	-5.46E+08	2.36E+07	2.21E+10	5.48E+09	5.14E+08	2.26E+07	3.54E+04	-3.44E+04
81	-2.02E+10	5.29E+09	-5.25E+08	2.43E+07	1.95E+10	5.10E+09	5.07E+08	2.36E+07	3.82E+04	-3.76E+04
82	-2.42E+10	6.06E+09	-5.67E+08	2.45E+07	2.91E+10	6.85E+09	6.00E+08	2.45E+07	3.66E+04	-3.57E+04
83	-2.31E+10	5.82E+09	-5.47E+08	2.37E+07	2.56E+10	6.17E+09	5.57E+08	2.35E+07	3.56E+04	-3.49E+04
84	-2.47E+10	6.08E+09	-5.53E+08	2.31E+07	2.30E+10	5.62E+09	5.16E+08	2.21E+07	3.38E+04	-3.30E+04
85	-2.34E+10	5.84E+09	-5.46E+08	2.36E+07	2.21E+10	5.48E+09	5.14E+08	2.26E+07	3.54E+04	-3.44E+04
86	-2.34E+10	5.84E+09	-5.46E+08	2.36E+07	2.21E+10	5.48E+09	5.14E+08	2.26E+07	3.54E+04	-3.44E+04
87	-2.47E+10	6.08E+09	-5.53E+08	2.31E+07	2.30E+10	5.62E+09	5.16E+08	2.21E+07	3.38E+04	-3.30E+04
88	-2.34E+10	5.85E+09	-5.48E+08	2.38E+07	2.25E+10	5.52E+09	5.14E+08	2.26E+07	3.56E+04	-3.45E+04
89	-2.34E+10	5.84E+09	-5.46E+08	2.36E+07	2.21E+10	5.48E+09	5.14E+08	2.26E+07	3.54E+04	-3.44E+04
90	-2.34E+10	5.85E+09	-5.48E+08	2.38E+07	2.25E+10	5.52E+09	5.14E+08	2.26E+07	3.56E+04	-3.45E+04
91	-2.34E+10	5.85E+09	-5.48E+08	2.38E+07	2.25E+10	5.52E+09	5.14E+08	2.26E+07	3.56E+04	-3.45E+04
92	-2.33E+10	5.84E+09	-5.47E+08	2.37E+07	2.20E+10	5.48E+09	5.14E+08	2.26E+07	3.55E+04	-3.45E+04
93	-2.06E+10	5.17E+09	-4.87E+08	2.15E+07	2.00E+10	4.96E+09	4.67E+08	2.09E+07	3.33E+04	-3.28E+04
94	-2.34E+10	5.85E+09	-5.45E+08	2.35E+07	2.19E+10	5.46E+09	5.13E+08	2.25E+07	3.52E+04	-3.42E+04
95	-2.34E+10	5.84E+09	-5.46E+08	2.36E+07	2.21E+10	5.48E+09	5.14E+08	2.26E+07	3.54E+04	-3.44E+04
96	-2.34E+10	5.85E+09	-5.45E+08	2.35E+07	2.19E+10	5.46E+09	5.13E+08	2.25E+07	3.52E+04	-3.42E+04
97	-2.37E+10	5.90E+09	-5.47E+08	2.35E+07	2.28E+10	5.58E+09	5.16E+08	2.24E+07	3.50E+04	-3.41E+04
98	-2.40E+10	5.93E+09	-5.48E+08	2.34E+07	2.34E+10	5.65E+09	5.17E+08	2.23E+07	3.46E+04	-3.36E+04
99	-2.42E+10	6.00E+09	-5.51E+08	2.34E+07	2.28E+10	5.59E+09	5.15E+08	2.23E+07	3.45E+04	-3.36E+04
100	-2.37E+10	5.90E+09	-5.46E+08	2.34E+07	2.27E+10	5.57E+09	5.15E+08	2.23E+07	3.48E+04	-3.38E+04

APPENDIX C. FINAL FORMULAS OF DAMAGE PATTERN V

- **Damage Pattern V − Case 1**: $K = K_{P1}^{V1} + K_{P2}^{V1} + K_{P3}^{V1} + K_{P4}^{V1} + K_{P5}^{V1}$

$$K_{P1}^{V1} = \sum_{x=0}^{z_3 \frac{m_\beta}{h_\beta}} \frac{K_T K_{Tc}}{[n-y(x)]K_{Tc}+y(x)K_T};$$

with $z_3 = h_\beta - n\alpha_2$ and $y(x) = \frac{h_\beta(m_\beta - x)}{m_\beta}$ for $x \in [0, z_3 \frac{m_\beta}{h_\beta}]$

$$K_{P2}^{V1} = \sum_{x_2=0}^{n\alpha_2 \frac{m_\beta}{h_\beta}} \frac{K_T K_{Ty} K_{Tc}}{[n\alpha_2 - y(x_2)]K_T K_{Tc} + y(x_2) K_T K_{Ty} + (n - n\alpha_2) K_{Ty} K_{Tc}},$$

with $y(x_2) = \frac{n\alpha_2\left(n\alpha_2 \frac{m_\beta}{h_\beta} - x_2\right)}{n\alpha_2 \frac{m_\beta}{h_\beta}}$ for $x_2 \in [0, n\alpha_2 \frac{m_\beta}{h_\beta}]$;

$$K_{P3}^{V1} = \frac{m - m_\alpha - m_\beta}{n}\left(\frac{K_{Ty} K_T}{\alpha_2\, K_T + (1 - \alpha_2) K_{Ty}}\right);$$

$$K_{P4}^{V1} = \frac{n\alpha_2 \frac{m_\alpha}{h_\alpha}}{n}\left(\frac{K_{Ty} K_T}{\alpha_2 K_T + (1 - \alpha_2) K_{Ty}}\right);$$

$$K_{P5}^{V1} = \sum_{x=0}^{z_4 \frac{m_\alpha}{h_\alpha}} \frac{K_T K_{Ty}}{[n - y(\bar{x})]K_{Ty} + y(\bar{x}) K_T},$$

with $z_4 = (h_\alpha - n\alpha_2)$; $y(\bar{x}) = \frac{h_\alpha(m_\alpha - \bar{x})}{m_\alpha}$ for $\bar{x} \in [0, z_4 \frac{m_\alpha}{h_\alpha}]$.

- **Damage Pattern V − Case 2**: $K = K_{P1}^{V2} + K_{P2}^{V2}$

$$K_{P1}^{V2} = \sum_{x=0}^{m_\beta} \frac{K_T K_{Ty} K_{Tc}}{[n\alpha_2 - y(x)]K_T K_{Tc} + y(x) K_T K_{Ty} + (n - n\alpha_2) K_{Ty} K_{Tc}},$$

with $y(x) = \frac{h_\beta}{m_\beta}(m_\beta - x)$ for $x \in [0, m_\beta]$;

$$K_{P1}^{V2} = \frac{m - m_\beta}{n}\left(\frac{K_{Ty} K_T}{\alpha_2 K_T + (1 - \alpha_2) K_{Ty}}\right).$$

- **Damage Pattern V − Case 3**: $K = K_{P1}^{V3} + K_{P2}^{V3}$

$$K_{P1}^{V3} = \sum_{x=0}^{m_\beta} \frac{K_T K_{Ty} K_{Tc}}{[n\alpha_2 - y(x)]K_T K_{Tc} + y(x) K_T K_{Ty} + (n - n\alpha_2) K_{Ty} K_{Tc}},$$

with $y(x) = \frac{h_\beta}{m_\beta}(m_\beta - x)$ for $x \in [0, m_\beta]$;

$$K_{P2}^{V3} = \frac{m - m_\beta}{n}\left(\frac{K_{Ty} K_T}{\alpha_2 K_T + (1 - \alpha_2) K_{Ty}}\right).$$