Preprint
UCRL-JC-144765

Determination of an initial mesh density for finite element computations via data mining

R. Kanapady, S. K. Bathina, K.K. Tamma, C. Kamath, and V. Kumar

This article was submitted to
The Fourth Workshop on Mining Scientific Datasets, KDD01, San Francisco, August 26, 2001

U.S. Department of Energy

Lawrence
Livermore
National
Laboratory

July 23, 2001

UCRL-JC-144765. The work of Chandrika Kamath was performed under the auspicies of the U.S. Department of Energy by University of California Lawrence Livermore National Laboratory under contract No. W-74050-Eng-48.

Determination of an Initial Mesh Density for Finite Element Computations via Data Mining*

R. Kanapady, S. K. Bathina and K. K. Tamma
ramdev@me.umn.edu, sai@cs.umn.edu and ktamma@tc.umn.edu
Dept. of Mechanical Engineering, University of Minnesota, 111 Church St. S.E, Minneapolis MN 55455

C. Kamath
kamath2@llnl.gov
Center for Applied Scientific Computing, Lawrence Livermore National Laboratory, Livermore, CA 94551

V. Kumar
kumar@cs.umn.edu
Dept. of Computer Science and Engineering, University of Minnesota, 200 Union Street S.E, Minneapolis MN 55455

ABSTRACT

Numerical analysis software packages which employ a coarse first mesh or an inadequate initial mesh need to undergo a cumbersome and time consuming mesh refinement studies to obtain solutions with acceptable accuracy. Hence, it is critical for numerical methods such as finite element analysis to be able to determine a good initial mesh density for the subsequent finite element computations or as an input to a subsequent adaptive mesh generator. This paper explores the use of data mining techniques for obtaining an initial approximate finite element density that avoids significant trial and error to start finite element computations. As an illustration of proof of concept, a square plate which is simply supported at its edges and is subjected to a concentrated load is employed for the test case. Although simplistic, the present study provides insight into addressing the above considerations.

1. INTRODUCTION

It is widely recognized that the finite element method is the choice of many analysts for performing structural analysis simulations. It is a viable computational tool due to the various inherent advantages, namely, the capability of programming the method in a general purpose manner, the ability to handle natural boundary conditions and arbitrary loads acting on the structure, and the ability to model complex geometries. Various methods of generating finite element meshes exist in the literature. Some are based on prescribed mesh density values at various sample points in the geometry. Other approaches such as adaptive h, p, h-p refinements also exist. The so-called r-method of relocation of the nodes is yet another strategy for developing a suitable finite element mesh.

Numerical methods employing a coarse initial mesh suffer from the drawback of needing several successive mesh refinements for acceptable accuracy of results which tend to be cumbersome and expensive. It is well known that the procedures which start with a coarse mesh and attempt serious repetitive refinements, as is the case in most finite-element packages, are time consuming and costly. An approach of overcoming this limitation involves the use of some type of adaptive re-meshing scheme to guarantee convergence in the finite element solution. Whilst this approach is attractive, it can be slow to converge to ideal finite element meshes since the initial mesh for these adaptive schemes has zero knowledge of the problem apriori. Hence, close to ideal initial meshes of these adaptive re-meshing schemes may accelerate the convergence and guarantee sufficient accuracy in the finite element solution. Consequently this reduces the overall solution times for both serial and parallel architectures. Recent works [1] and [2] involved the application of Artificial Neural Networks (ANN) for the prediction of the finite element mesh density in order to estimate the magnetic field in a body. The present study builds upon previous work and provides a detailed study as related to structural mechanics applications. We specifically outline details in obtaining an ideal mesh densities at selected sampling points and an approach to enhance the quality of the results by asymmetric scaling of training samples.

In case of the structural mechanics, for illustration, Fig. 1 describes an elastic body Ω with boundary Γ which is de-

*The authors are very pleased to acknowledge support in part by the Department of Energy DOE/LLNL W-7045-ENG-48 and by the Army High Performance Computing Research Center (AHPCRC) under the auspices of the Department of the Army, Army Research Laboratory (ARL) cooperative agreement number DAAH04-95-2-0003/contract number DAAH04-95-C-0008. The content does not necessarily reflect the position or the policy of the government, and no official endorsement should be inferred. Access to computing facilities was provided by AHPCRC and Minnesota Supercomputer Institute

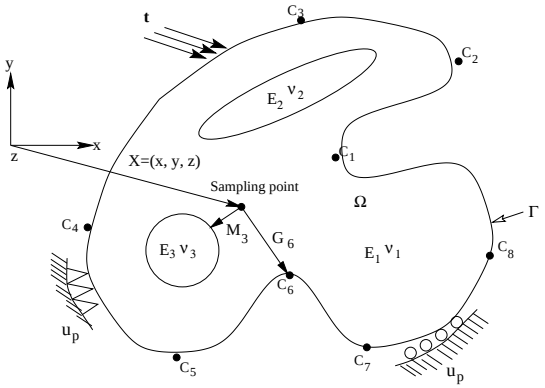


Figure 1: Illustrative application problem description for predicting finite element mesh density using data mining.

fined by representative critical points c ={c_1,c_2, ... } with respect to a fixed co-ordinate system. The elastic body is also comprised of materials with properties set M = {(E_1,v_1, ...),(E_2,v_2, ...), ... } where E_i, v_i are the Youngs modulus and the Poison's ratio respectively. The body is subjected to traction loads, **t**, and variety boundary constraints. The response of such a structure includes determination of field variables such as displacements, stresses and strain data for use in subsequent data mining models. Fig. 2 describes the finite element discretization with different material set M_i, loads t_i and boundary conditions u_{Pi}, i = 1,2, ... ,n where n is the number of finite element analyses carried out to generate the training data for the data mining model. From these analyses, it is postulated that one could predict an approximate mesh density for the analysis by employing error indicators formulated from data mining models. Hence for a given geometry, the objective is to predict the iniital finite element mesh density for arbitrary material distributions, loads, and boundary conditions as described in Fig. 3. Pictorially Fig. 2 describes the training examples to generate training data and Fig. 3 describes the test example for which the data mining model is required to predict the desired initial mesh density.

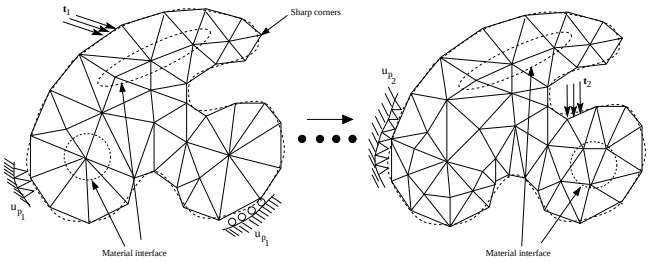


Figure 2: Illustrative application problem with finite element models for training example.

In this paper, as a proof-of-concept, we explore the calculation of the mesh density for a square plate, which is simply supported at its edges, with a concentrated load acting on it. This simplistic test example was selected since we already know the exact theoretical solution to the problem. From this, we can immediately assess if data mining techniques are indeed helpful for predicting the mesh density. The mesh density is predicted by training a simple feed forward neural network and making it learn the relationship between the mesh density and geometric features of the model. In Section 2, the preliminaries are discussed, followed by a discussion of the methodology used in predicting the mesh density in Section 3. In Section 4 and 5 the results obtained and conclusions of this study are discussed. In Section 6, future directions and the challenges involved are highlighted.

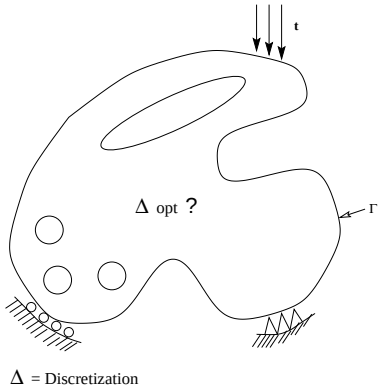


Figure 3: Overall goal of data mining illustrting application problem for predicting finite element mesh density.

2. PRELIMINARIES

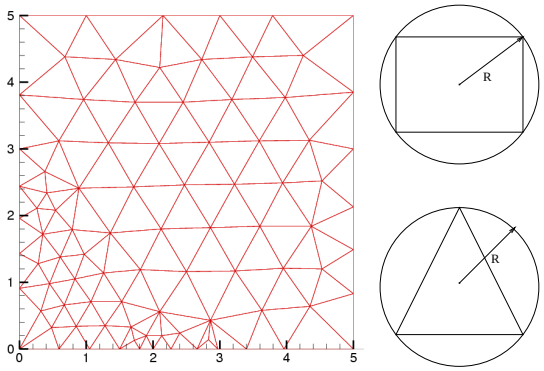
Finite element modeling involves discretizing the original domain into finite elements such as triangles. Such a typical process is shown in Fig. 4(a) using triangular elements for illustration, though other element types could also have been used. This is accomplished by a mesh generator for which the input is the mesh density at selected points in the domain. This mesh density can be defined in many ways. One such definition could be the number of nodes in the vicinity of a point [3]. Another definition could be the value of the radius (R) of the circle which is circumscribed over the triangle as shown in Fig. 4(b). This determines the triangle size and hence the element size in a finite element discretization. The mesh density value is the target variable of the classifier and the features can typically consist of geometry descriptions, loads applied, etc., depending on the problem at hand.

3. METHODOLOGY

In this section we discuss the various steps followed in calculating the initial mesh density for the problem at hand, the neural network architecture used to train the data, and feature selection required for training.

3.1 Generating the data

We start by training the predictive data mining models using example data, which pertains to "ideal" meshes of the representative geometries or domains. Here a square plate, is simply supported at its edges as shown in Fig. 5. A concentrated load, is applied at a point whose coordinates are (x_l,y_l). For this situation an analytical solution is available in [4], [5] which gives the displacement at any point (x_s,y_s)



(a) (b)

Figure 4: (a) Typical finite element mesh discretization and (b) ideal element representation for triangular and quadrilateral elements.

for any load at (x_l,y_l). The displacement w at (x_s,y_s) is given by

$$w = \frac{4P}{\pi^4 Dab} \sum_{m=1}^{\infty} \sum_{n=1}^{\infty} \frac{\sin(m\pi x_l)\sin(n\pi y_l)\sin(m\pi \frac{x_s}{a})\sin(n\pi \frac{y_s}{b})}{(\frac{x_s}{a})^2 + (\frac{y_s}{b})^2} \tag{1}$$

where P is the load applied at load coordinates (x_l, y_l), D is the flexural rigidity of the plate given by $\frac{Et^3}{12(1-v^2)}$, where E is the Young's modulus, t is the thickness, v is the Poisson's ratio, a is the length of the plate and b is the width of the plate. Once the displacement is obtained from this equation for a point (x_s,y_s), the mesh density value (h_{ideal}), which is the radius of the circle circumscribing the triangle (as shown in Fig. 4(b)), is determined for this point via the following steps:

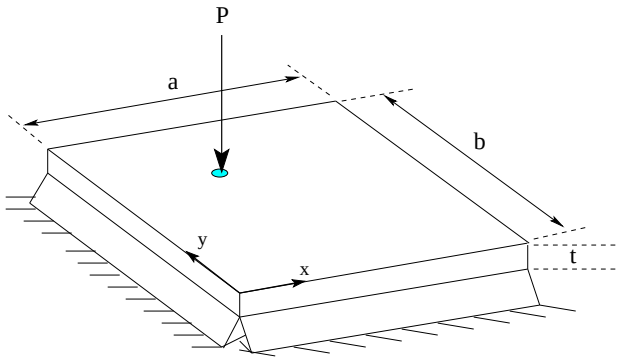
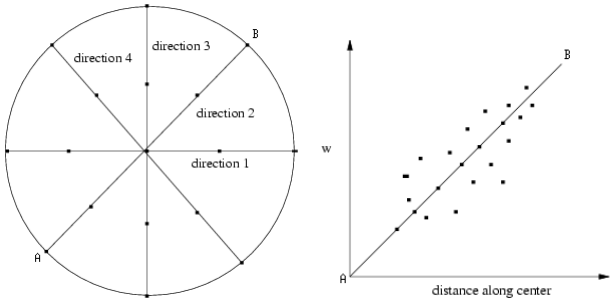


Figure 5: Problem description: a simply supported at all edges of square plate with concentrated load.

- First, choose a circular magnifier (radius of influence) of radius R_{ini} unit about the sample point (x_s,y_s). Next, start with and choose different directions in the circle at equal angular displacements. Then choose points along each direction such that they are equidistant from each other as shown in the Fig. 6(a). The displacements, w, are then computed at each of these points using Eq.(1).

- Next all the points that are on the line in a particular direction are chosen. The method of least squares is used to make the best linear fit as shown in Fig. 6(b) using various w's at the chosen points along this direction. The process is repeated for all the directions. Note that one could employ the best quadratic fit, best cubic fit, etc., depending on the application and the type of the finite element employed.

- The error is then estimated between the analytical solution and the numerical solution obtained from the best fit line, for each direction. Here, the error is defined by the L_2 norm on the solution vector in each direction. This norm should be less than the predefined tolerance limit, ϵ, for all the directions. Then, this value of the radius of the circle forms the h_{ideal} value, otherwise the radius is reduced. Note that the data mining model developed only holds for predetermined values of the tolerance limit, ϵ, for all other values the above steps have to be repeated till convergence to a numerical solution whose error lies within the tolerance limit.



(a) (b)

Figure 6: Ideal mesh density h_{ideal} computations for situation with known analytical response: (a) circular magnifier with points in chosen direction and (b) best fit line to exact values along a selected direction A-B.

Finally after obtaining the various h_{ideal} values at different sampling points in the domain, the mesh generator draws the approximate finite element mesh as shown Fig. 4(a).

3.2 Architecture of the Artificial Neural Network (ANN)

The artificial neural network considered in this work consists of a one input layer with 7 processing units corresponding one hidden layer with 19 processing units, and one output layer with a single processing unit. The ANN is trained to the corresponding target vector on the output layer. This

	Features							Target variable
Sampling points	Projxload	Projyload	ProjX	ProjY	d	P	t	h_{ideal}
1								
2								
⋮								
n								

Table 1: Training and test data layout for the problem.

target vector is the mesh density value for the finite element model.

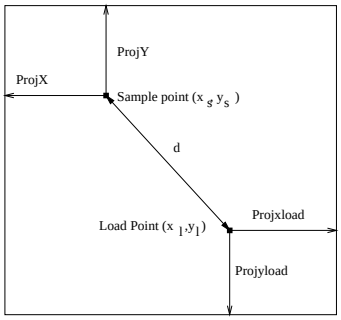


Figure 7: Features description for simply supported square plate with concentrated load.

3.3 Features selection

Since the model seeks to predict the element size, h_{ideal} at a point in the domain, the training data can have data sampled at many points in the domain of a single representative geometry. It is essential and critical that the features chosen for training have all of the characteristics that can be encountered in real applications such that the data-mining model becomes almost problem independent. The various features considered here for training the model are shown in Fig. 3.2 and include:

- Projxload, Projyload - projections of the load point (x_l,y_l) to the nearest adjacent edges,
- ProjX, ProjY - projections of sample point (x_s,y_s), where the mesh density is being determined, to the nearest adjacent edges,
- d - distance of the point (x_s,y_s) to the load point (x_l,y_l),
- P - load value,
- t - thickness of the plate.

For illustration, the corresponding feature table is described in Table 1.

4. RESULTS

Sample points were chosen randomly in the plate and the displacements were found for a load applied at different points which were again chosen randomly on the plate. The training data set consists of values of h_{ideal} in the range 0.01 to 1. This results in a finite element size scale ratio 1:100 which is the case in more realistic finite element applications. The distributions of h_{ideal} values are identical for training instances and testing instances.

load	0.68214
projpyload	0.56408
projpxload	0.50756
distance	0.32830
thickness	0.21520
projpx1	0.17636
projpy1	0.17247

Table 2: Reported relative importance of features to the developed neural network model.

The values of loads and thicknesses chosen for the testing case are different from the training case and, were chosen such that they are within the range of training case values. The training set consisted of 36,600 records and the test set consisted of 18,300 records. A neural net model was created using the training set. The relative importance of each of the features to the neural network using Clementine software [1] is listed in the Table 2. The plot between the predicted mesh value and the actual mesh value is shown in Fig. 8. In Fig. 8 the points above the diagonal represent predicted mesh sizes which are larger than the actual mesh size. This is detrimental for the finite element solution accuracy. Similarly, the points below the diagonal represent the predicted mesh sizes which are smaller than the actual mesh size. This is not detrimental for the the finite element solution. However, it is critical as the number of finite elements increases, it increases the computational cost. Restricting the attention only to the points above the diagonal, a scaling technique is used to bring down the points close to the diagonal by increasing the number of records pertaining to these points in the training set. The scaling technique used for this is given by

$$W = \frac{h_{ideal}^{pred} - h_{ideal}^{actual}}{h_{ideal}^{actual}} \times k \qquad (2)$$

where W is the number of records and k is a constant with a value of 2 in our case. Also, it is observed that most of these points pertain to the case where the load was close to the boundary of the plate. Therefore, in conjunction with the above scaling procedure load points that were within 0.5 units distance from the boundary of the plate were not considered in the new training set. A new neural net model was created with this new training set. The performance

[1] ©1999 SPSS Inc., Version 5.0.1

measures of the developed neural network model are listed in the Table 3. The plot between the predicted h_{ideal} and the actual h_{ideal} is shown in Fig. 9. Figures 10 – 13 show the finite element mesh generated employing these actual and predicted mesh densities for four loading conditions with 36 sampling points for each of them. The location of the loading point in each of the case is illustrated with a box on the plate. The actual mesh sizes for these four cases are obtained as mentioned in Section 3.1. The predicted mesh sizes are obtained from the neural network model. As the Figs. 10 – 13 show, the meshes for both the actual and predicted cases resemble each other very closely. As shown in table 4 the predicted number of mesh elements is very close to the actual number for all four cases.

Minimum Error	- 0.74468
Maximum Error	0.53632
Mean Error	0.0028778
Mean Absolute Error	0.019690
Standard deviation	0.051059
Linear Correlation	0.98256
Occurrences	18300

Table 3: Performance measures of the developed neural network model.

MESH	$\text{Elements}_{\text{Act}}$	$\text{Elements}_{\text{Pred}}$	% increase
Fig. 10	5667	5780	2.15
Fig. 11	7307	7414	1.46
Fig. 12	7270	7258	-0.17
Fig. 13	7318	7021	-4.06

Table 4: Actual and predicted number of elements for mesh in Figs. 10 – 13.

5. CONCLUDING REMARKS

The present work concentrated on a proof-of-concept application. A relatively simple problem where we know the theoretical solution was employed to assess the performance of data mining models. A simply supported square plate with a concentrated load was considered as a test case. Close to "ideal" mesh density h_{ideal} at various points in the plate were predicted with different load values, location and plate thickness. The training set was created without finite element discretization and this allows to create a data mining model. An ANN is employed and the initial results for predicting the appropriate mesh density are encouraging.

6. CHALLENGES AND FUTURE WORK

There are challenges involved in getting the training sets for a complex geometry subject to different loading conditions, composed of different materials and different boundary conditions (Ref. Figs. 1 – 3). Analytical solution are generally not known and we need to devise a strategy to determine mesh density values for a general case. Assessments of feasibility and performance are planned to be undertaken for various other data mining methods like Classification and Regression Trees (CART) and Multivariate adaptive regression splines (MARS) to find an alternative to neural network techniques. The overall goal is to create an effective system intended to provide an ideal initial mesh for a finite element simulation code or an initial "close to ideal" mesh for a subsequent adaptive solver employed for the finite element computations. Such a system will enable a knowledge-based approach for the pre-processing phase of finite element simulation codes.

7. REFERENCES

[1] D. N. Dyck, D. A. Lowther, and S. McFee. Determining an approximate finite element mesh density using neural network techniques. *IEEE Trans. Magn.*, 28, Mar 1992.

[2] C. H. Ahn. A self-organizing neural network approach for automatic mesh generation. *IEEE Trans. Magn.*, 27, 1990.

[3] R. Chedid and N. Najjar. Automatic finite-element mesh generation using artificial neural networks-part1: Prediction of mesh density. *IEEE Trans. Magn.*, 32(5), 1996.

[4] A. C. Ugural. *Stresses in Plates and Shells.* McGraw Hill, Inc, 1981.

[5] A. C. Ugural and D. Fenster. *Advanced applied stresses in Plates and Shells.* McGraw Hill, Inc, 1994.

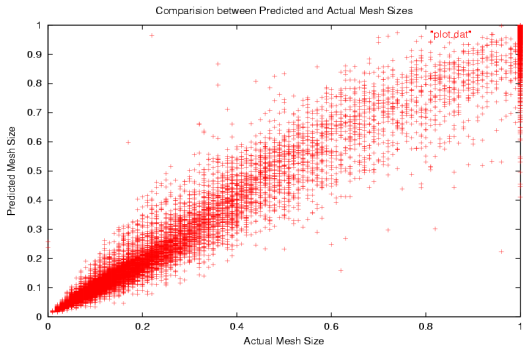


Figure 8: Comparison between predicted and actual mesh size before scaling.

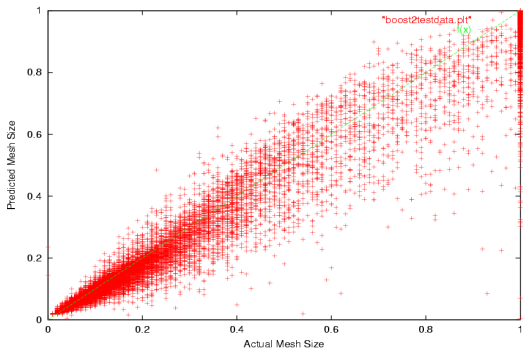
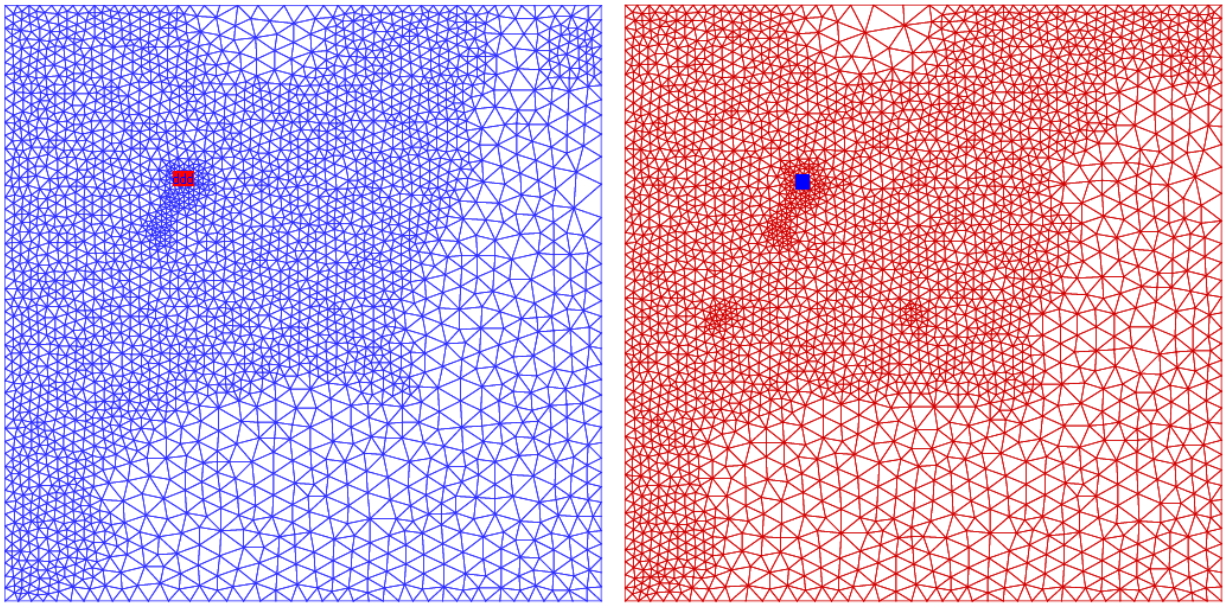
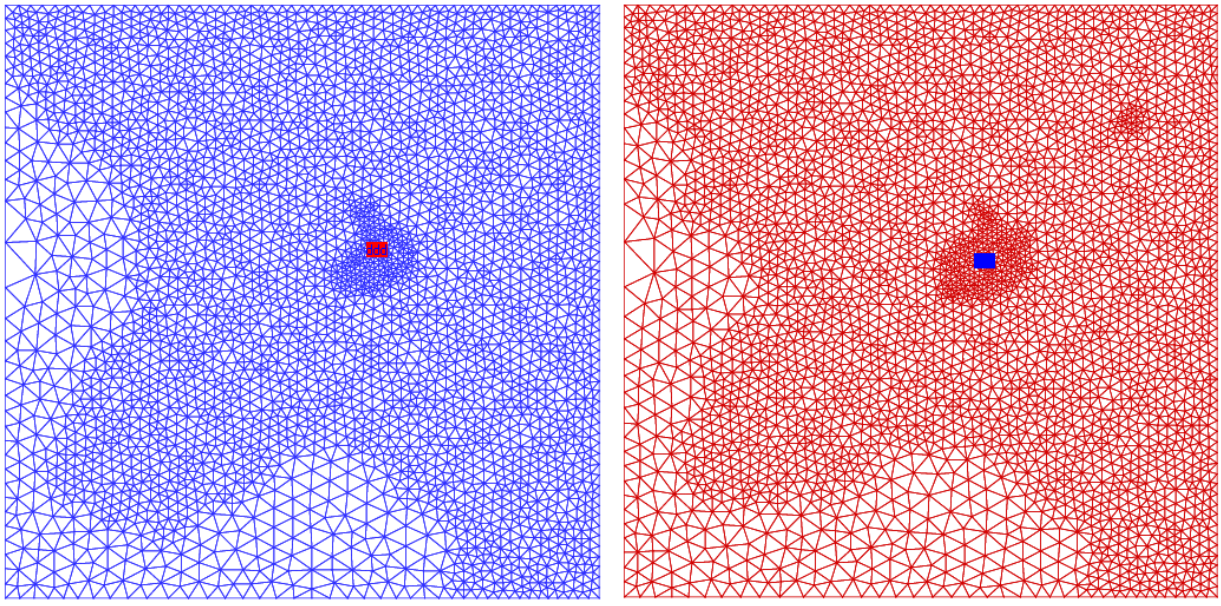


Figure 9: Comparison between predicted and actual mesh size after scaling.



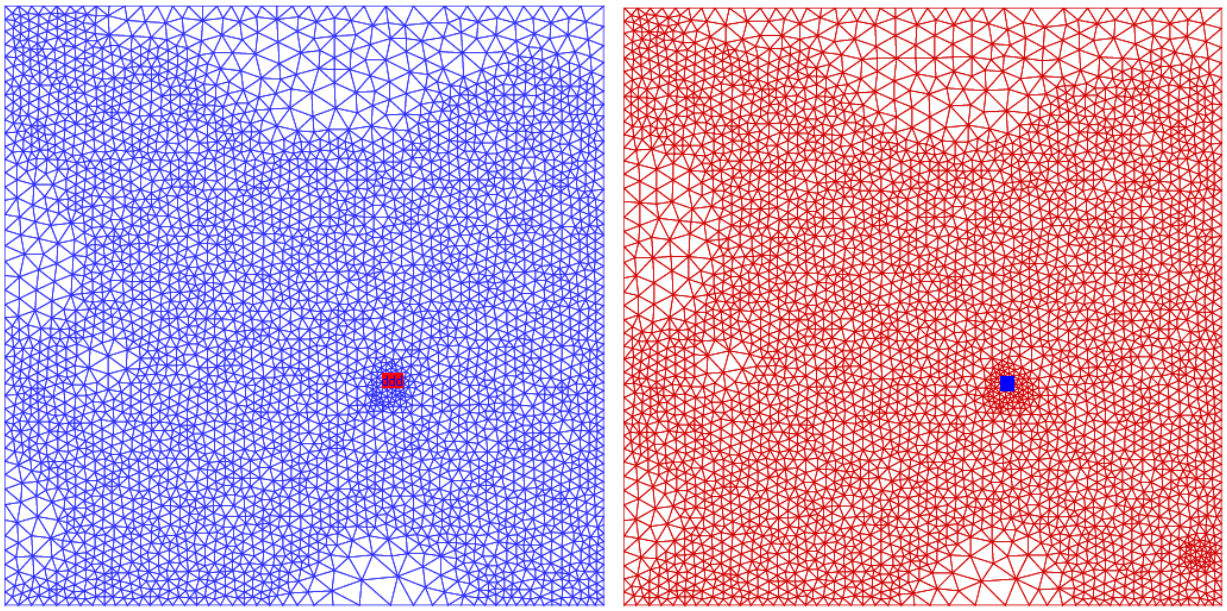
(a) Actual mesh (b) Predicted mesh

Figure 10: Finite element mesh for the load location (1.5,3.5).



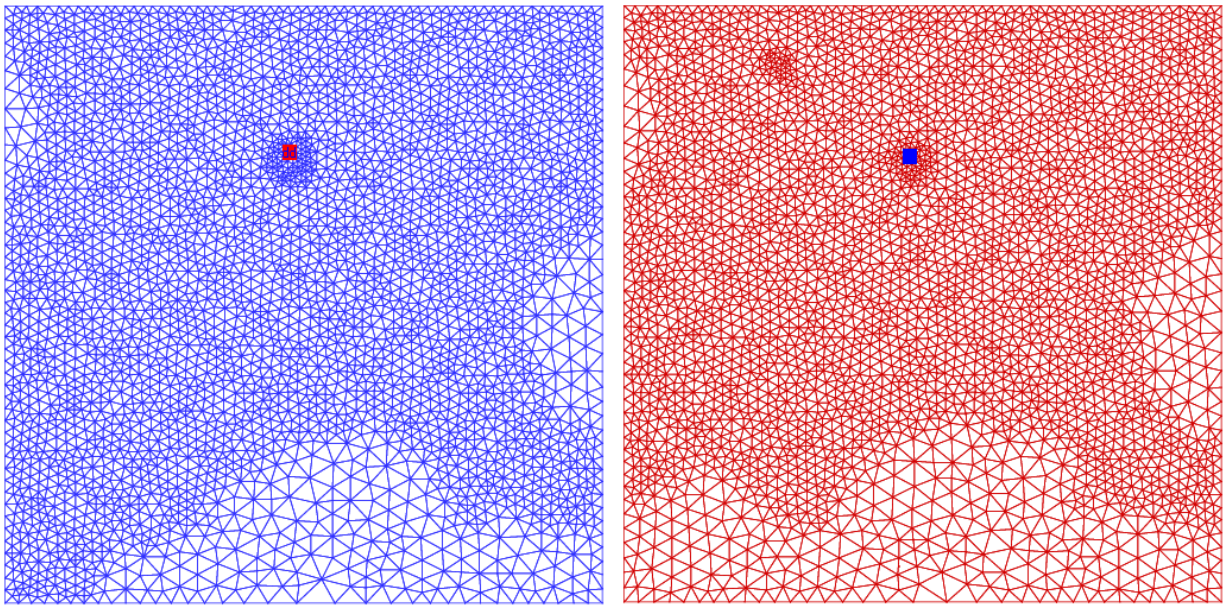
(a) Actual mesh (b) Predicted mesh

Figure 11: Finite element mesh for the load location (3.25,3).



(a) Actual mesh (b) Predicted mesh

Figure 12: Finite element mesh for the load location (3.3,1.9).



(a) Actual mesh (b) Predicted mesh

Figure 13: Finite element mesh for the load location (2.4,3.5).

University of California
Lawrence Livermore National Laboratory
Technical Information Department
Livermore, CA 94551