ICCEPM 2020
The 8th International Conference on Construction Engineering and Project Management
Dec. 7-8, 2020, Hong Kong SAR

A Framework to Automate Reliability-based Structural Optimization based on Visual Programming and OpenSees

Jia-Rui Lin[1]*, Jian Xiao[2], Yi Zhang[3]

[1] *Department of Civil Engineering, Tsinghua University, China,* E-mail address: lin611@tsinghua.edu.cn

[2] *Department of Civil Engineering, Tsinghua University, China,* E-mail address: xiao-j15@mails.tsinghua.edu.cn

[3] *Department of Civil Engineering, Tsinghua University, China,* E-mail address: zhang-yi@tsinghua.edu.cn

Abstract: Reliability-based structural optimization usually requires designers or engineers model different designs manually, which is considered very time consuming and all possibilities cannot be fully explored. Otherwise, a lot of time are needed for designers or engineers to learn mathematical modeling and programming skills. Therefore, a framework that integrates generative design, structural simulation and reliability theory is proposed. With the proposed framework, various designs are generated based on a set of rules and parameters defined based on visual programming, and their structural performance are simulated by OpenSees. Then, reliability of each design is evaluated based on the simulation results, and an optimal design can be found. The proposed framework and prototype are tested in the optimization of a steel frame structure, and results illustrate that generative design based on visual programming is user friendly and different design possibilities can be explored in an efficient way. It is also reported that structural reliability can be assessed in an automatic way by integrating Dynamo and OpenSees. This research contributes to the body of knowledge by providing a novel framework for automatic reliability evaluation and structural optimization.

Key words: Reliability-based Structural Optimization, Visual Programming, Generative Design, OpenSees, Dynamo

1. INTRODUCTION

The main objective of structural design is create an engineering system with specific safety, functional, and performance requirements under conditions of uncertainty [1]. In deterministic design optimization, the uncertainties of the structural system (i.e. dimensions, models, materials, loads, etc.) are taken into account in a subjective and indirect way, by means of partial safety factors specified in design codes. As a consequence, deterministic optimal solutions may lead to reduced reliability levels [2]. Reliability Based Design Optimization (RBDO) has emerged as an alternative to properly model the safety-under-uncertainty aspect of the optimization problem [3].

Thus, probabilistic analyses are needed in the development of such reliability-based design of structures [4]. Since they provide consistent levels of safety over various types of structural components, the reliability-based structural design methods are more flexible and rational than their counterparts [5].

Such a design procedure takes into account more information than the deterministic methods in the design of ship structural components [2]. This information includes uncertainties in the strength of various ship structural elements, in loads, and modeling errors in analysis procedures [4].

Since decades ago, various methods were proposed, including first and second order reliability method [3], Monte Carlo simulation [6], etc. However, most of these efforts focus on theory, computational complexity and design optimization methods [3, 7, 8], few of them have considered how to automate the approach and to improve the applicability of these methods. In other words, the process of reliability evaluation is time-consuming, tedious and not all the possibilities are explored most of the time.

Recently, together with building information modeling (BIM) which captures all the data of a building structure in a single model, generative design (GD) based on parametrization is proposed [9, 10]. Currently, GD utilizes visual programming (VP) language to provide direct and convenient way for parameter calculation and process automation [11]. Nowadays, VP tools like Rhinoceros 3d and Dynamo are widely used in different applications, and both of them are quite easy to learn.

To automate the optimization process of reliability-based design, this paper proposes an integrated approach based on visual programming and OpenSees. The approach consists 3 parts, namely, parametrization, simulation model generation and simulation, reliability evaluation, which are described in section 2. Then, demonstration of the proposed approach is implemented and discussed in section 3. Finally, feasibility and potential benefits of the approach is then concluded in section 4.

2. METHODOLOGY

To automatic the model generation, performance simulation, and reliability evaluation process of structure optimization, an integrated framework is proposed as Figure 1. Generally, the framework consists three parts: parameterization, OpenSees-based simulation, and reliability evaluation. The first part takes visual programming approach to represent structural design in an parametric way, which makes it possible to generate different models by changing parameters. While the middle part utilizes OpenSees as the underlying performance simulation module, and automatic simulation file generation and OpenSees invocation are implemented. Finally, post processing of the simulation results and automatic reliability calculation are provided in the last part, which could evaluate the reliability of a structure automatically. Details of each part are as follows.

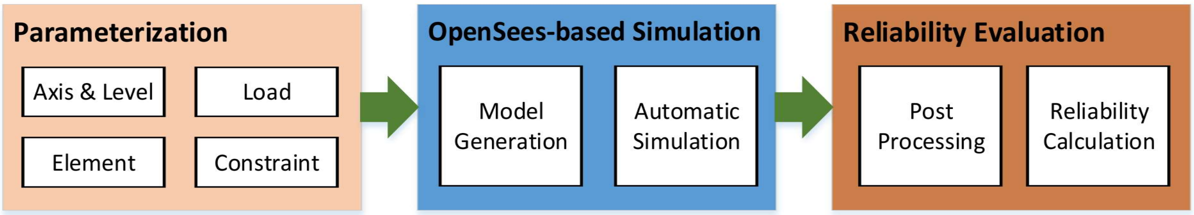


Figure 1. The proposed framework for automatic reliability-based structure optimization

2.1. Parameterization

Parameterization is a way that uses parametric equations to represent and generate design models. Generally, in a parametric model, a few parameters are selected and their relationships are carefully defined and created based on equations and programs, and all the properties of a design model can be derived from these parameters. That is, a design model is easily created by changing the pre-defined parameters, and the model generation and modification process can be taken as near real-time.

Therefore, this research adopts parameterization to automate the structural model generation process. As Shown in Figure 1, when creating a parametric structural model, four parts of information are considered, namely, axis and level, element, constraint and load. Parameters of the first 3 parts used for defining a parametric structural model are listed in Table 1. For axes and levels, count and offset/elevation are enough to create the axis grid and different levels. While, for a structural element, not only the category, material, section, but also the location is needed. And for the beam element, we use line as its location, while for the column element, point is used to express its location. In addition, modulus, weight per volume, etc. are defined to generate different materials.

Table 1. Parameters used for model generation

Group	Parameter	Data type
Axis	X offset,Y offset	float
Axis	X count, Y count	integer
Level	Elevation	float

Level	count	integer
Element	Category, Material	string
Element	Location	Point, Line
Element	Section	string
Element	Connection	string
Element	Constraint	string
Section	Size, Dimension	float
Material	Modulus, Weight per volume	float

After capturing all the needed parameters, their relationships are defined based on visual programming, or more concisely, Dynamo. Together Autodesk Revit, Building Information Modeling (BIM) –based structural model can be generated for further collaboration and manufacturing purpose.

For example, when offset and count of the axes are provided, a series of intersection points and the corresponding grid can be generated (Figure 2). Given that column and element in Revit and Dynamo are considered as *StructuralFraming* elements, once location point and location line are provided, it is possible to create column and beam element based on *StructuralFraming.ColumnByPoint* and *StructuralFraming.BeamByCurve* respectively.

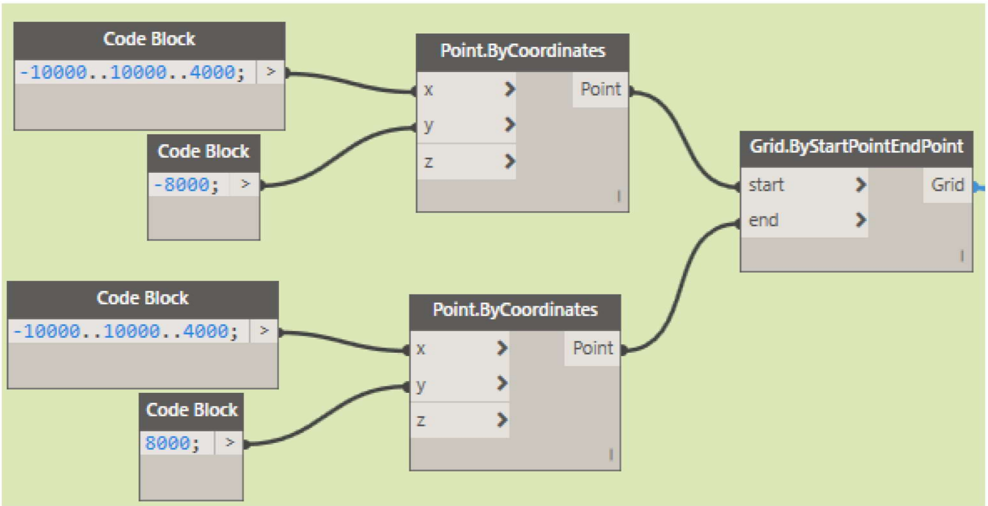


Figure 2. Generating grid based on parameters

In this way, when setting the X-direction count as 4, the Y-direction count as 2, the level count as 10, section of beams and columns as H type, a model as showed in the left part of Figure 3 is generated in seconds. Similarly, changing the X/Y-direction count and the level count, model like the right part of Figure 3 can be generated.

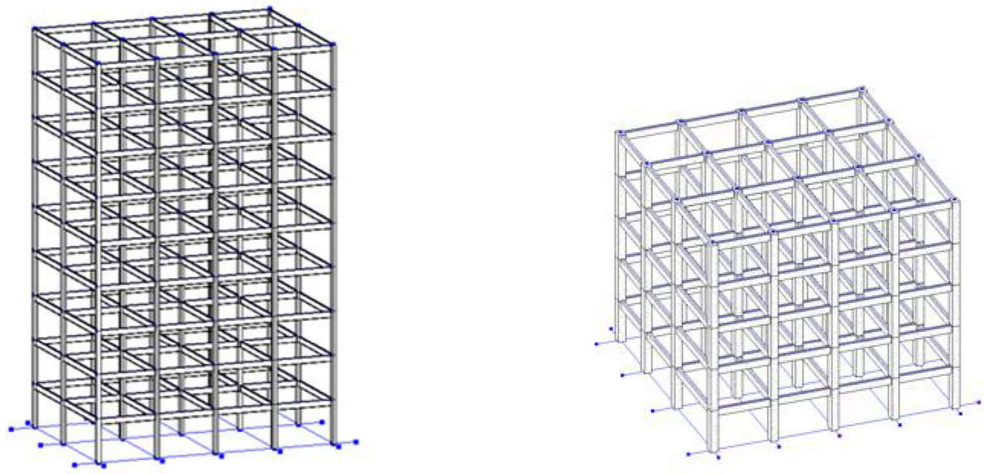


Figure 3. Generated models based on different parameters

Specially, when considering the reliability of a structure design, distribution of different loads should be taken account. In this paper, Monte Carlo simulation method is adopted. According to the design codes of China, extreme wind speed each year obeys the Gumbel distribution, and wind pressure obeys similar distribution. In literature [12], it is suggested that we can use the following function to represent the distribution of wind pressure in a 50-year period:

$$F_{\mathrm{W}}(x) = \exp[-\exp(-\frac{x-0.982W_{0k}}{0.158W_{0k}})] \tag{1}$$

Where W_{0k} is the basic wind pressure defined in building code [13], and can be calculated using the following equation:

$$w_{0k} = \beta_z \mu_s \mu_z w_0 \tag{2}$$

While w_0 stands for the basic wind pressure, and μ_z , μ_s , β_z are the wind pressure height coefficient, shape factor of wind load, and wind vibration coefficient respectively.

Following the same way, distribution of live load of each level can be defined as:

$$F_{\mathrm{LT}}(x) = \exp[-\exp(-\frac{x-0.423L_k}{0.084L_k})] \tag{3}$$

Where L_k is the standard live load defined in building code.

However, dead load of a structure obeys normal distribution, and is noted as:

$$N(1.06\mathrm{G}_k, (0.074\mathrm{G}_k)^2) \tag{4}$$

Where G_k is the standard dead load defined in building code.

Finally, based on the statistics collected from 45 cities and counties in China, it is suggested that the seismic action obeys Fréchet distribution, and in a 50-year period, distribution of the base shear force is defined as [14]:

$$F_{\mathrm{Q}}(\mathrm{Q}) = \exp[-(\frac{Q}{0.385Q_K})^{-K}] \tag{5}$$

Where Q_K is the standard base shear force defined in building code, K is the location parameter of the distribution and is usually taken as 2.35.

For a random variable X, given its distribution function $F(X)$ and suppose u_i is uniformly distributed in the range of $(0,1)$, we can say that $x_i = F^{-1}(u_i)$ is a sample of $F(X)$. therefore, for the Gumbel distribution:

$$F(x_i) = \exp[-\exp(-\frac{x_i - \alpha}{\beta})] \tag{6}$$

The following can be derived:

$$x_i = \alpha\text{-}\beta \ln(-\ln u_i) \tag{7}$$

Therefore, according to the above-mentioned equations, the wind load and live load can be calculated and exported to the OpenSees file by custom defined Dynamo nodes based on some scripts (Figure 3). Then, using different coefficients, all the loads mentioned are combined together as a load combination applied to the structure. For a demonstration purpose, all coefficients are taken as 1.0 in this research.

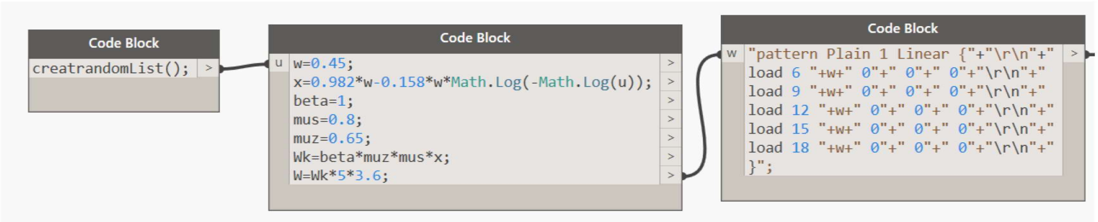


Figure 3. Generating load combination for structural simulation in OpenSees

2.2. OpenSees-based simulation

With the proposed parameterization method, structural design models can be generated automatically in an efficient way, and it saves quite a lot of time in design modeling. To further automate the performance simulation process, this section proposes an OpenSees-based simulation method.

Given that the OpenSees application is an open-source software, and adopts Tcl (tool command language) for data modeling and input, thus this research uses its Tcl specification to generate structural models as Tcl files for further simulation. Main definitions provided in OpenSees are listed in Table 2. Not only the dimension and degree of freedom, but also nodes, constraints, elements, and load can be defined in a Tcl file. Moreover, it is also possible to specify the output of the simulation.

Table 2. Data definition of OpenSees

Data type	Definition
model	model basic -ndm $ndm -ndf $ndf
node	node $nodeTag (ndm $coords)
constraint	fix $nodeTag (ndf $constrValues)
transformation	geomTransf Linear $transfTag
frame element	element elasticBeamColumn $eleTag $iNode $jNode $A $E $Iz $transfTag
load pattern	pattern Plain $patternTag $tsTag{}
uniform load	eleLoad -ele $eleTag1 -type -beamUniform $Wy
point load	load $nodeTag (ndf $LoadValues)

Since all the data needed in a Tcl file are already defined or generated based on the parameters and corresponding functions and equations, it is quite straightforward to extract relevant information and generate a Tcl file based on customized Dynamo nodes and scripts. Figure 4 shows an example of generated Tcl file.

```
wipe
puts "System"
model basic -ndm 2 -ndf 3
puts "restraint"
node 1 6400.000000 0.000000
node 2 0.000000 0.000000
...
fix 1 1 1 1
...
puts "transformation"
geomTransf Linear 1
element elasticBeamColumn 1 1 4 17044.000000 206000 395061761.333333 1
element elasticBeamColumn 2 2 5 16440.000000 206000 495553000.000000 1
...
```

Figure 4. An exemplary Tcl file generated

What's more, since OpenSees is usually delivered as a command line application, it is possible to invoke the application based on command line commands. Therefore, according to the command line guide of OpenSees, this research created a Dynamo node based on some python scripts (Figure 5). With this Dynamo node, paths of generated Tcl files, configuration for performance simulation can be passed to OpenSees by command line, and the simulation can be started automatically in Dynamo. Thus, it is possible to generate required Tcl files and run the performance simulation in a fully automatic way, and manual working efforts of engineers are eliminated.

Python Script

```
import clr
clr.AddReference('ProtoGeometry')
from Autodesk.DesignScript.Geometry import *
from System.Diagnostics import Process
#该节点的输入内容将存储为 IN 变量中的一个列表。
dataEnteringNode = IN

#将输出内容指定给 OUT 变量。
adr=IN[0]
str=IN[1]
ans=IN[2]
openseesadr=IN[3]
word=[]
for i in range(len(adr)):
    f=open(adr[i],'w')
    f.write(str[i])
    f.close()

p=Process()
p.StartInfo.UseShellExecute=False
p.StartInfo.CreateNoWindow=True
#p.StartInfo.RedirectStandardOutput=True
p.StartInfo.FileName='put your path Opensees.exe here'
for i in range(len(openseesadr)):
    p.StartInfo.Arguments=openseesadr[i]
    p.Start()
p.WaitForExit()
p.ExitCode
for i in range(len(ans)):
    g=open(ans[i],'r')
    word.append(g.read())
OUT = word
```

运行 保存更改 还原

Figure 5. Python script for invoking OpenSees in Dynamo

2.3. Reliability evaluation

According to the reliability theory, different variables can be chosen to reflect the function of a structure. And if the function of a structure is influenced by variables X_i ($i = 1, 2, \ldots, \mathrm{n}$), then the function of the structure can be defined as:

$$Z = g(\mathrm{X}_1, \mathrm{X}_2, \ldots, \mathrm{X}_n) \tag{8}$$

While $\mathrm{Z} > 0$ means the structure can provide its function in a reliable way. When $\mathrm{Z} < 0$, the structures loses its function or fails, and $\mathrm{Z}{=}0$ is a transfer state between the two. Then the probability of $\mathrm{Z} < 0$ is defined as the failure probability of a structure, and is denoted as P_f. Similarly, P_r is the probability of $\mathrm{Z} > 0$, which is called the reliability of a structure. Obviously, the sum of P_f and P_r always equals 1. Generally, the function of a structure's bearing capacity is defined as:

$$Z = g(\mathrm{R}, \mathrm{S}) = \mathrm{R} - \mathrm{S} \tag{9}$$

Where R is the structure resistance, S is the load effect, and they obey the normal distribution $N(\mu_R, \sigma_R)$ and $N(\mu_S, \sigma_S)$ respectively. So we have $\mu_Z{=}\mu_R - \mu_S$, $\sigma_Z{=}\sqrt{{\sigma_R}^2{+}{\sigma_S}^2}$, and then the reliability of structure failure can be derived:

$$P_r = 1 - P_f = 1 - p(\mathrm{Z} < 0) = 1 - \int_{-\infty}^{0} f_Z(\mathrm{Z})\,\mathrm{dZ} = 1 - \Phi(-\frac{\mu_Z}{\sigma_Z}) = \Phi(\frac{\mu_Z}{\sigma_Z}) \tag{10}$$

Most of the time, we just use $\beta{=}\dfrac{\mu_Z}{\sigma_Z}$ as the indicator of reliability of a structure.

If taken displacement as the criterion for structural failure, maximum lateral displacement (MXD) of the structure is usually considered an appropriate indictor of structural performance according the building code. Thus, collected MXD of all the samples from output generated by OpenSees, average μ and standard deviation σ can be obtained. Therefore, let w denotes the maximum allowable lateral displacement, the reliability of the structure is then calculated as:

$$\beta = \frac{w - \mu}{\sigma} \tag{11}$$

Converting the above equations into customized Dynamo nodes, and together with Dynamo nodes for data extraction, the reliability of a structure is automatically calculated (Figure 6).

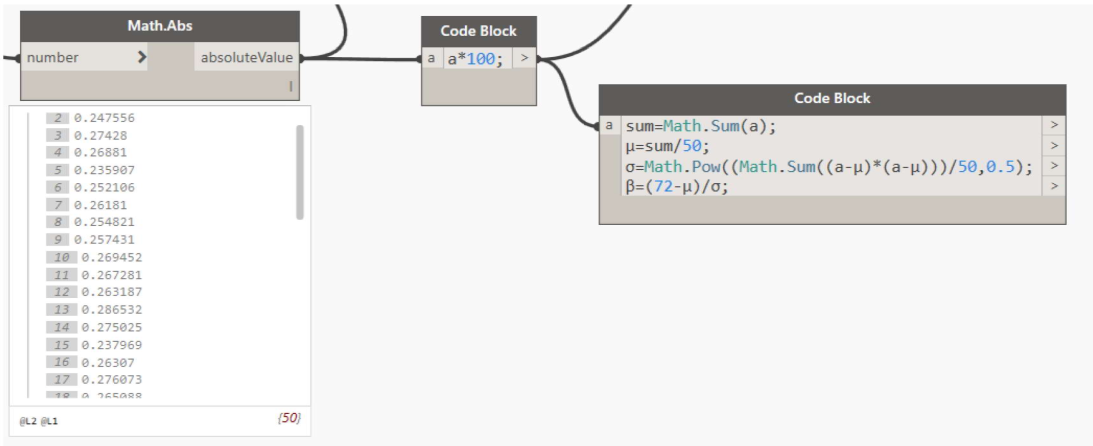
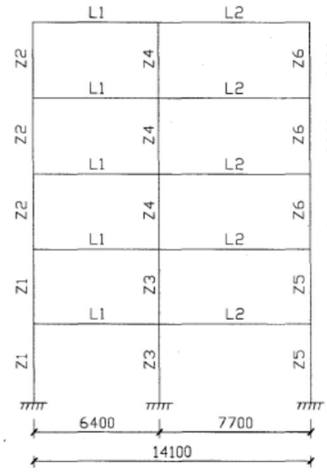


Figure 6. Customized Dynamo node for calculating reliability of a structure

3. DEMONSTRATION

Based on the above-proposed approach, automated structural model generation, simulation and reliability evaluation can be achieved. To further demonstrate and validate the feasibility and flexibility of the proposed approach, a five-story steel frame structure is used. Y-direction offsets of the frame are 6.4m and 7.7m respectively, while X-direction offset is 5m, height of each story is 3.6m, and total height of the frame is 18m (left of Figure 7). As listed at the right of Figure 7, sections of all the columns and beams are H sections, and their material is Q345B. When generating the load samples, 50-year design period is considered. Given that the shape is simple and the height of the structure is lower than 30m, wind vibration effect can be omitted, thus μ_s, β_z are both taken as 1.0 and μ_z is taken as 0.65. In addition, only elastic simulation is conducted for simplification.

When doing the optimization, total weight of the structure is taken as the objective function as an example, and only sections of columns and beams are changeable.



Element ID	Section
L1	HN496×199×10×14
L2	HN500×200×10×16
Z1	HW350×350×12×19
Z2	HW300×300×10×15
Z3	HW400×400×13×21
Z4	HW350×350×12×19
Z5	HW350×350×12×19
Z6	HW300×300×10×15

Figure 7. Parameters of the test case

Following the proposed parametric modeling approach, similar model like Figure 3 is created, and different load samples are generated based on their distribution (Figure 8).

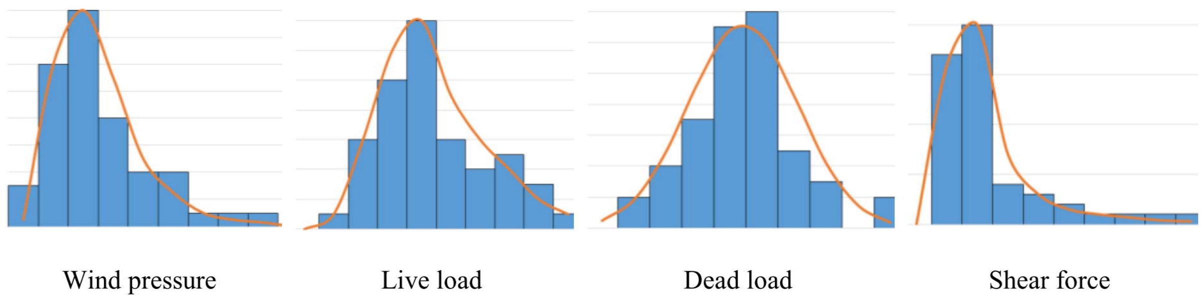


Figure 8. Generated load distributions

Then, following the proposed approach, structural performance is automated simulated and reliability of the structure can be calculated. Meanwhile, total weight of the structure is calculated at the same time. Therefore, the designer can see how the reliability and total weight change when different sections of structural elements are chosen (Figure 9). Finally, the designer can select appropriate section combinations considering the reliability requirements and the cost constraints (based on total weight).

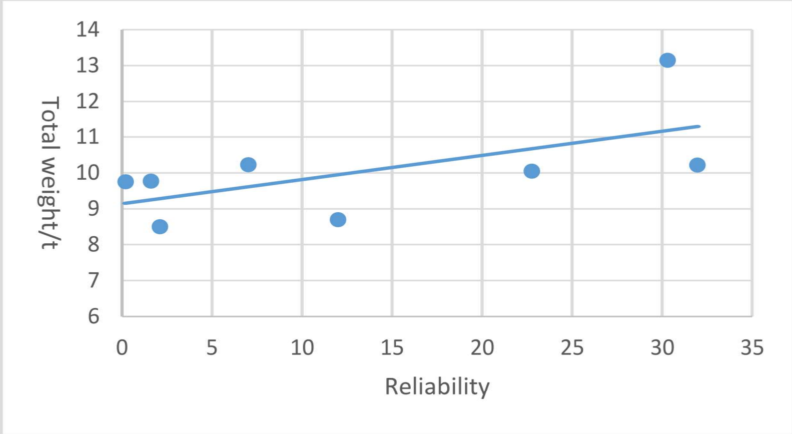


Figure 9. Reliability and total weight of different section combinations

Since Dynamo embeds complex programming codes and scripts as visual nodes, it is intuitive and easy to understand for non-experts. If an engineer or designer could understand the input and output of a Dynamo node, he or she can use just drag the node into work space and connect it with other nodes to create a computing process. Meanwhile, since Dynamo is also integrated with Autodesk Revit, it is possible to generate BIM models and share them with other stakeholders, thus improving the value of the generate models.

4. CONCLUSION

Reliability-based structural optimization is a well-known method for structural design. However, due to the complexity of structure, reliability is usually hard and time-consuming to calculate and approximate methods are always used. This research proposes a visual programming and OpenSees based approach to automate the model generation, simulation, and reliability evaluation process, therefore making to possible to get the reliability of a structure in a more time efficient way. Demonstration with a steel frame structural model shows that the proposed method is feasible and has the potential to automate performance-based reliability evaluation. Moreover, since the proposed method is implemented in Dynamo, it is also possible to generate BIM models for collaboration purpose. However, this research is still proof of concept, more improvements and extensions are needed in the future for complex structures and practical applications.

ACKNOWLEGEMENTS

This research is supported by the Beijing Natural Science Foundation (No. 8194067), the National Science Foundation of China (No. 51908323), the National Key R&D Program of China (No. 2018YFD1100900), and the Young Elite Scientists Sponsorship Program by China Association for Science and Technology (No. QNRC2016001).

REFERENCES

[1] B.M. Ayyub, A. Haldar, “Practical structural reliability techniques”, Journal of Structural Engineering, vol. 110, no. 8, pp. 1707-1724, 1984.

[2] A.T. Beck, W.J. de Santana Gomes, “A comparison of deterministic, reliability-based and risk-based structural optimization under uncertainty”, Probabilistic Engineering Mechanics, no. 28, pp. 18-29, 2012.

[3] R.H. Lopez, A.E.T.O. Beck, “Reliability-based design optimization strategies based on FORM: a review”, Journal of the Brazilian Society of Mechanical Sciences and Engineering, vol. 34, no. 4, pp. 506-514, 2012.

[4] I. Assakkaf, B. Ayyub, “Reliability-Based Structural Design”, in Ship Design and Construction, The Society of Naval Architects and Marine Engineers, pp. 19-1 to 19-36, 2004.

[5] B.M. Ayyub, R.H. McCuen, “Probability, statistics, and reliability for engineers and scientists”, 1st ed., CRC press, 2016.

[6] J.B. Cardoso, J.R. de Almeida, J.M. Dias, P.G. Coelho, "Structural reliability analysis using Monte Carlo simulation and neural networks", Advances in Engineering Software, vol. 39, no. 6, pp. 505-513, 2008.

[7] A.A. Chojaczyk, A.P. Teixeira, L.I.S.C. Neves, J.B. Cardoso, C.G. Soares, "Review and application of artificial neural networks models in reliability analysis of steel structures", Structural Safety, no. 52, pp. 78-89, 2015.

[8] R. Rackwitz, "Reliability analysis - a review and some perspectives", Structural safety, vol. 23, no. 4, pp. 365-395, 2001.

[9] S. Abrishami, J. Goulding, F.P. Rahimian, A. Ganah, "Integration of BIM and generative design to exploit AEC conceptual design innovation", Information Technology in Construction, no. 19, pp. 350-359, 2014.

[10] M.R. Asl, M. Bergin, A. Menter, W. Yan, "BIM-based parametric building energy performance multi-objective optimization", Proceedings of the 32nd eCAADe Conference, Newcastle upon Tyne, England, UK, pp. 455-464, 2014.

[11] B. Ferreira, A. Leitão, "Generative Design for Building Information Modeling", Proceedings of the 33rd eCAADe Conference, Vienna University of Technology: Vienna, Austria, pp. 635-644, 2015.

[12] X. Tu, "Statistics of Wind Load", Industrial Buildings, pp. 57-60, 1982.

[13]. Ministry of Construction of the People's Republic of China, "Load Code for the Design of Building Structures (GB50009)", Beijing, China, 2012.

[14] X. Gao, A. Bao, "Probablistic Model and Its Statistical Parameters of Seismic Load", Eqarthquake Engineering and Engineering Vibration, vol. 5, no. 1, pp. 13-22, 1985.