Calculation of reliability of structures using random sets

Andrzej Pownuk
Department of Theoretical Mechanics, Silesian University of Technology,
Krzywoustego 7, 44-100 Gliwice
E-mail: pownuk@zeus.polsl.gliwice.pl
URL: http://zeus.polsl.gliwice.pl/~pownuk

Abstract

There are many problems in which the available information is not precise enough to justify the use of numbers. In many engineering problems we cannot precisely measure the exact values of loads (e.g. wind loads, weight of vehicle etc.), material constant (particularly in bimechanics, geomechanics, wall structures, concrete structures and composite structures) and geometric characteristic (tolerances).

In this paper to modelling of uncertain parameters random set theory was applied. Additionally a new algorithm of calculation bending moment envelope was presented. This algorithm is based on the finite element method.

Keywords : reliability, random sets, uncertain parameters, bending moment envelop

1. Introduction

Let us consider mechanical system, which is shown in the Fig. 1.

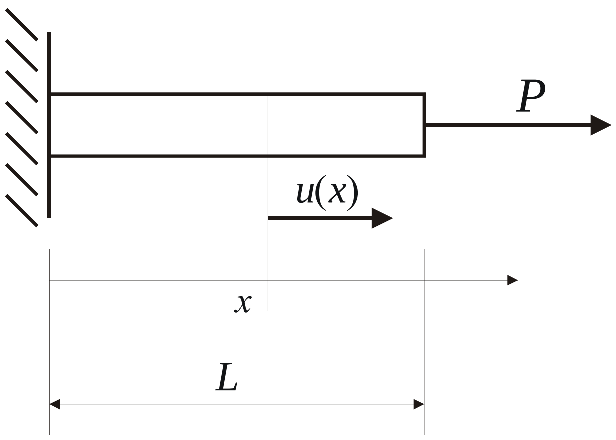

Fig. 1

This is well known that the displacement in the rod can be calculated using the following formula

$$u(x)=\frac{P\cdot x}{E\cdot A} \tag{1}$$

Now we can calculate the logarithmic derivative.

$$\frac{du}{u}=\frac{dP}{P}+\frac{dx}{x}-\frac{dE}{E}-\frac{dA}{A} \tag{2}$$

The increments have different sign. Because of that the extreme relative error can be calculated in the following way

$$\left|\frac{\Delta u}{u}\right|=\left|\frac{\Delta P}{P}\right|+\left|\frac{\Delta x}{x}\right|+\left|\frac{\Delta E}{E}\right|+\left|\frac{\Delta A}{A}\right| \tag{3}$$

Let's assume that the relative error of each parameter is equal to 5%. We can see that the relative error of displacement is equal to 20%. This is a big error and it cannot be neglect in calculations.
This simply example shows that the uncertainties cannot be neglected in calculation.
The approximation errors in the calculation can be reduce (using adaptive methods) to 0.1% in linear elasticity [2] and 5% in plasticity [1].
The error cased by the uncertain parameters is much bigger than the approximation errors.
In geomechanics the uncertain parameters are known with accuracy, which is equal to 20%. If we apply civil engineering code the error is even bigger (about 100-200% in the worst case).

2. Modelling of uncertain parameters using intervals

One of the simplest ways of representation of uncertain or inexact data, as well as inexact computations with them, is based on intervals. In this approach, an uncertain (real) number is represented by an interval (a continuous bounded subset) of real numbers which pre-sumably contains the unknown exact value of the number in question. Despite its simplicity, it conforms very well to many practical situations, like tolerance handling or managing rounding errors in numerical computations. Additionally, estimation of upper and lower bound of some physical quantity is sufficiently simple. If we want to calculate a probability density function we have to know a lot of measurement. If we want to define an interval we have to know only two numbers (upper and lower bound).

It can be shown that the method which is based on interval parameters give the same result that the semi-probabilistic methods [3] (in some cases).

3. Calculation of bending moment envelop using sensitivity analysis

3.1. Calculation of displacement

Let us consider linear elastic model of the structure with interval parameters. The finite element method lead to the following system of equilibrium equations

$$\mathbf{K}(\mathbf{h})\mathbf{q} = \mathbf{Q}(\mathbf{h}) , \quad \mathbf{h} \in \hat{\mathbf{h}} \tag{4}$$

where
K – stiffness matrix,
Q – load vector,
q – displacement vector,
h – vector of uncertain parameters,
$\hat{\mathbf{h}} = [h_1^-, h_1^+] \times ... \times [h_m^-, h_m^+]$ vector of interval parameters.

Upper and lower bound of the displacement can be calculated using the following formula:

$$q_i^- = inf\{q_i(\mathbf{h}) : \mathbf{h} \in \hat{\mathbf{h}}\} \tag{5}$$

$$q_i^+ = sup\{q_i(\mathbf{h}) : \mathbf{h} \in \hat{\mathbf{h}}\} \tag{6}$$

It can be shown that in some cases relation between the displacement q_i and uncertain parameter h_j is monotone [4]. If the functions $q_i = q_i(..., h_j, ...)$ are monotone, then the extreme values of displacements (5, 6) can be calculated using only endpoints of the intervals $\hat{h}_i = [h_i^-, h_i^+]$.
Sensitivity of displacements can be calculated using the following equation

$$\mathbf{K}\frac{\partial \mathbf{q}}{\partial h_i} = \frac{\partial \mathbf{Q}}{\partial h_i} - \frac{\partial \mathbf{K}}{\partial h_i}\mathbf{q} \tag{7}$$

Now we can define the following sign vectors

$$\mathbf{s}^i = sign\left(\frac{\partial q_i}{\partial \mathbf{h}}\right) \quad i=1,...,n \tag{8}$$

where

$$s_j^i = sign\left(\frac{\partial q_i}{\partial h_j}\right) \quad j=1,...,m \tag{9}$$

Now we can calculate the number of different vectors $\mathbf{s}^i = [s_1^i \ \ ... \ \ s_m^i]$. We can define the "Index" vector in the following way:

```
NumberOfIntervalSolution=1;
Index(1)=1;
for(i=2; i<=n; i++)
{
  j=0;  st=0;
  while ( (j<=i)&&(st==0) )
  {
    j++;
    if ( S(i) == S(j) )
    {
      Index(i)=j;
      st=1;
    };
  };
  if( st==0 )
  {
    Index(i)=i;
    NumberOfIntervalSolution++;
  }
};
```

Now we can calculate the interval displacement, which are connected witch each independent sign vector S(i).

```
j=0;
for(i=1; i<=n; i++)
{
  if( Index(i) == i )
  {
    j++
    IntervalSolution(j)=CalculateIntervalSolution(S(i));
    IntervalSolutionIndex(i) = j;
  }
  else
    IntervalSolutionIndex(i) = IntervalSolutionIndex(Index(i));
};
```

The vector „IntervalSolution" contains all interval solutions. (S(i) = $\mathbf{s}^i$, S(i,j) = s_j^i).
Procedure "IntervalVector CalculateIntervalSolution(**int** i)" calculate interval solution using given vector $\mathbf{s}^i$.

```
IntervalVector CalculateIntervalSolution(IntVector S)
{
  IntervalVector IntervalSolution(n);
  long j;
  /* Calculate upper bound */
  for(j=1; j<=m ; j++)
  {
    if( S(j)>=0 )
      h(i)=IntervalH(i).upper;
    else
      h(i)=IntervalH(i).lower;
  };
  CalculateGlobalStiffnessMatrix(K,h);
  CalculateGlobalLoadVector(Q,h);
  /* Solution of the equation Kq=Q */
  q=Solution(K,Q);
  for(j=1;j<=n;j++)
    IntervalSolution(j).upper = q(j);

  /* Calculate lower bound */
  for(j=1; j<=m ; j++)
  {
    if( S(j)>=0 )
      h(j)=IntervalH(j).lower;
    else
      h(j)=IntervalH(j).upper;
```

```
  };
  CalculateGlobalStiffnessMatrix(K,h);
  CalculateGlobalLoadVector(Q,h);
  /* Solution of the equation Kq=Q */
  q=Solution(K,Q);
  for(j=1;j<=n;j++)
    IntervalSolution(j).lower = q(j);

  return IntervalSolution;
};
```

The interval displacement can be calculated using the following algorithm:

```
for(i=1; i<=n; i++)
{
  Interval temp;
  temp = IntervalSolution(IntervalSolutionIndex(i), i);
  IntervalDisplacement(i) = temp;
};
```

3.2. Calculation of bending moment envelop

Let us consider a rod structure. A bending moment in the straight element "e" can be calculated in the following way

$$M^e(x) = M^e(0) - \int_0^x q^e(t)(x-t)dt \tag{10}$$

where $M^e(0)$ is a value of the bending moment in the first node in the element and $q^e(t)$ is a continuos load.
If we consider uncertain parameters we can write the following equation

$$M^e(x,\mathbf{h}) = M^e(0,\mathbf{h}) - \int_0^x q^e(t,\mathbf{h}) \cdot (x-t)dt \tag{11}$$

The bending moment envelope can be defined using the following equations

$$M^{e-}(x) = inf\{M^e(x,\mathbf{h}) : \mathbf{h} \in \hat{\mathbf{h}}\} \tag{12}$$

$$M^{e+}(x) = sup\{M^e(x,\mathbf{h}) : \mathbf{h} \in \hat{\mathbf{h}}\} \tag{13}$$

Nodal forces in the element "e" $\mathbf{Q}_R^e$ can be calculated using local stiffness matrix $\mathbf{K}^e$, local load vector $\mathbf{Q}^e$ and local displacement vector $\mathbf{q}^e$

$$\mathbf{Q}_R^e = \mathbf{K}^e \mathbf{q}^e - \mathbf{Q}^e \tag{14}$$

One of the coefficients of the vector $\mathbf{Q}_R^e$ is equal to $\pm M^e(0,\mathbf{h})$. Because of this, if we know the vector $\mathbf{Q}_R^e$ we know the number $M^e(0,\mathbf{h})$. Extreme value of the moment can be calculated using sensitivity analysis.

$$\frac{\partial M^e(x,\mathbf{h})}{\partial h_i} = \frac{\partial M^e(0,\mathbf{h})}{\partial h_i} - \int_0^x \frac{\partial q^e(t,\mathbf{h})}{\partial h_i}(x-t)dt \tag{15}$$

$\frac{\partial M^e(0,\mathbf{h})}{\partial h_i}$ can be calculated using equation (14)

$$\frac{\partial \mathbf{Q}_R^e}{\partial h_i} = \frac{\partial \mathbf{K}^e}{\partial h_i}\mathbf{q}^e + \mathbf{K}^e \frac{\partial \mathbf{q}^e}{\partial h_i} - \frac{\partial \mathbf{Q}^e}{\partial h_i} \tag{16}$$

Number

$$\int_0^x \frac{\partial q^e(t,\mathbf{h})}{\partial h}(x-t)dt \tag{17}$$

can be calculated directly.
Extreme values of the bending moment can be calculated in the following way.

```
/* Calculate derivative of bending moment - equation (15) */
for(j=1; j<=m.; j++)
{
  DKe = DerivativeOfLocalStiffnessMatrix(e,j);
  Ke  = LocalStiffnessMatrix(e);
  DQe = DerivativeOfLocalLoadVector(e,j);
  Dqe = DerivativeOfLocalDisplacementVector(e,j);
  /* Derivative of nodal forces - equation (16)*/
  DQR = DKe*qe+Ke*Dqe-DQe;
  /* Calculate ∂M^e(0,h)/∂h_i using ∂Q_R^e(0,h)/∂h_i */
  DM0 = DQR(3); /* 2D problem */
  DMq = CalculateDerivativeOfLoad(x,e,j);
  /* Calculate derivative of M^e(0) - equation (15) */
  DM(j)=DM0+DMq;
  SignOfDerivativeM(j)=Sign( DM(j) );
};

/* Calculate upper value of the bending moment */
j=0;  st=0;
while( ( j<=NumberOfIntervalSolution )&&(st==0) )
{
  j++;
  if ( SignOfDerivativeM == S(j) )
  {
    st=1;
    int IndexTemp;
    IndexTemp = IntervalSolutionIndex( j );
    IntervalDisplacement=IntervalSolution(IndexTemp);
  }
  else
  {
    NumberOfIntervalSolution++;
    IntervalDisplacement=
      CalculateIntervalSolution(SignOfDerivativeM);
```

```
    IntervalSolution(NumberOfIntervalSolution)=
      IntervalDisplacement;
  };
};
qe  = LocalDisplacement( IntervalDisplacement, e);
Ke = CalculateLocalStiffnessMatrix(e);
Qe = CalculateLocalLoadVector(e);
/* Calculate nodal forces – equation (14) */
QRe=Ke*qe-Qe;
M0e=QRe(3); /* 2D problems */
/* Equation (17) */
Mx=CalculateBendingMoment(x,e);
/* Extreme value of bending moment – equation (10) */
Mmax=M0e+Mx;
```

4. Calculation of reliability of structures using random sets

Let U be a universal nonempty set of a variable u under consideration and $P(U)$ the power set of U. In random set theory (and Dempster-Shafner theory), available evidence can be expressed by a *basic probability assignment* on $P(U)$, i.e. by a set of function $m: P(U) \to [0,1]$ such that $m(\varnothing)=0$ and $\sum_{A\in P(U)} m(A)=1$. This function can be considered as a probability measure defined on a universal set Z related to U through a multivalued mapping $\Gamma: Z \to P(U)$. So for each set $A\in P(U)$, the value $m(A)$ express the probability of $z=\Gamma^{-1}(A)$ and it does not exclude that the singletons of A can get additional probability deriving from other subsets B of U so that $A\cap B\neq\varnothing$. Each set $A\in P(U)$ for which $m(A)>0$ is called *focal element*. A *finite support random set* on U is a pair (Ξ, m), where m is a basic probability assignment on $P(U)$ and Ξ is the family of focal elements induced by m [6].

As Ξ is a class of subset of U, we cannot generally calculate the probability of singleton $u\in U$ or of generic subset $E\subseteq U$. Nevertheless we can calculate an upper and lower bound of this probability, called Plausibility $Pl(.)$ and Belief $Bel(.)$ respectively:

$$Bel(E)\le Pro(E)\le Pl(E), \quad \forall E\subseteq U \tag{18}$$

where

$$Bel(E)=\sum_{A:A\subseteq E} m(A) \tag{19}$$

$$Pl(E)=\sum_{A:A\cap E\neq\varnothing} m(A) \tag{20}$$

When the experimental data are uncertain, then we can calculate upper probability of failure [5]

$$P_f^{+}=Pl(g(\hat{\mathbf{x}})\le 0)=\sum_{\hat{\mathbf{x}}:g(\hat{\mathbf{x}})\cap(-\infty,0]\neq\varnothing} m(\hat{\mathbf{x}}) \tag{21}$$

where $\hat{\mathbf{x}}\in\Xi$. Ξ is a family of uncertain measurements [5]. The number P_f^{+} can be calculated using Monte Carlo simulations.

The condition $g(\hat{\mathbf{x}})\cap(-\infty,0]\neq\varnothing$ can be check using the procedures, which were described above. The extreme value of limit state function $g(\mathbf{x})$ can be find using sensitivity analysis.

$$g^{-}=g^{-}\left(\hat{\mathbf{x}}, sign\left(\frac{\partial g}{\partial \mathbf{x}}\right)\right),\quad g^{+}=g^{+}\left(\hat{\mathbf{x}}, sign\left(\frac{\partial g}{\partial \mathbf{x}}\right)\right) \tag{22}$$

5. Conclusions

Taking into account uncertainty of parameters is a very important matter. The error, which is generated by the uncertainty of parameters (in many cases) is greater than approximation errors. In this paper a new method of calculation of reliability is presented. This method is based on the theory of random set and Monte Carlo simulation. In computation of the extreme values of bending moment and displacements sensitivity analysis were applied. The examples of application will be presented on the conference. Presented method can be applied to nonlinear problems of computational mechanics.

References

[1] L. Gallimard, P. Ladeveze, J.P. Pelle, Error estimation and adaptivty in elastoplasticity, International Journal for Numerical Methods in Engineering, 39(1996)*189-217*.

[2] A.I. Abbo, S.W. Sloan, An automatic load stepping algorithm with error control, International Journal for Numerical Methods in Engineering, 39(1996)*1737-1759*.

[3] A. Pownuk, Reliability of structures with interval and random parameters. The Second Conference of PHD-students. Wisła 2001 (submitted to publications)

[4] A. Pownuk, Applications of sensitivity analysis for modelling of structures with uncertain parameters. International Conference on Interval Methods in Science and Engineering Interval'2000, Karlsruhe, Germany, 2000

[5] A. Pownuk, Calculation of reliability of structures by using interval probability, AI-MECH 2000, Symposium on Methods of Artificial Intelligence in Mechanics and Mechanical Engineering , Gliwice 2000, 273-276

[6] F. Tonon, A. Bernardini, A random set approach to the optimisation of uncertain structures, Computers and Structures, 68(1998)583-600