

Eigen analysis of a two-storey one-bay frame

固有値解析とモード形状のプロットを実行する方法を示します。2層耐震フレームを目的のため使用します。フレーム構造のジオメトリと材料特性は、図1に表示されます。節点と要素の番号は、図2にしています。

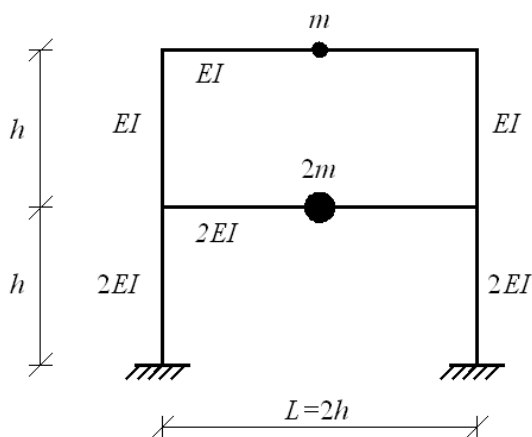


Figure 1

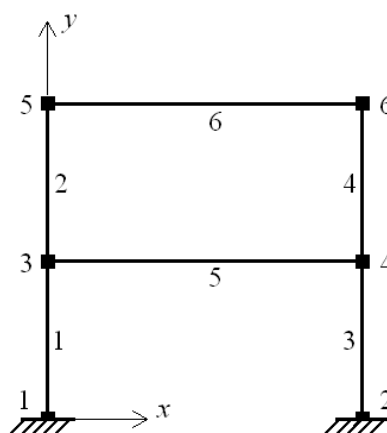


Figure 2

この例を実行する方法

この例題を実行するには、このファイルをダウンロードしてください。

[EigenAnal_twoStoreyFrame.tcl](#)

`EigenAnal_twoStoreyShearFrame.tcl` を `OpenSees.exe` と同じフォルダに保存します。

`OpenSees.exe` を起動し、次のコマンドを入力し、解析を実行します。

```
source EigenAnal_twoStoreyFrame1.tcl
```

出力ファイル(`data` というフォルダの中に保存される)を作成するためには、`[exit]`を入力し、`OpenSees.exe` を閉じる必要があります。

モデルを作成します。

モデルの空間の次元数と各節点の自由度 (DOF) の数の `model` コマンドを使用して定義します。この例は 2D のモデルで各節点での自由度は 3 です。次のように定義します。

```
model BasicBuilder -ndm 2 -ndf 3
```

ジオメトリ、質量、材料特性に図1に示すものに対応するよう変数を割り当てます。(例えば、柱の高さを 144 インチに設定し、変数の `h` を割り当てる)、高さの値は `$h` で引用できます。

```

#mass
set m [expr 100.0/386.0]

#number of modes
set numModes 2

#material
set A 63.41          #柱の断面積
set I 320.0         #断面二次モーメント
set E 29000.0       #ヤング率

#geometry
set L 240.          #長さ
set h 120.          #高さ

```

構造物の節点(図 2)は、[node](#) のコマンドを使用して定義します。

```

node 1 0. 0.;
node 2 $L 0.;
node 3 0. $h;
node 4 $L $h;
node 5 0. [expr 2*$h];
node 6 $L [expr 2*$h];

```

境界条件は、[fix](#) コマンドを使用して定義します。この例では節点 1 および 2 を完全に固定します。

```

fix 1 1 1 1;
fix 2 1 1 1;

```

質量を節点 3、4、5、6 に [mass](#) コマンドを使用して割り当てます。せん断フレームは自由度 2 のみ (1 層、2 層の X 方向の変位) そのため、質量を X 方向のみで割り当てる必要があります。

```

mass 3 $m 0. 0.;
mass 4 $m 0. 0.;
mass 5 [expr $m/2.] 0. 0.;
mass 6 [expr $m/2.] 0. 0.;

```

[geometric transformatio](#) (座標変換)は線形で Id タグ 1 で定義します。

```

set TransfTag 1;
geomTransf Linear $TransfTag;

```

梁と柱のフレームは [elasticBeamColumn](#) 要素を使用して弾性に定義します。梁の慣性モーメントを完全に固定するために (Ib)、非常に高い値に設定する。(10e + 12)。

```
element elasticBeamColumn 1 1 3 $A $E [expr 2.*$I] $TransfTag;
element elasticBeamColumn 2 3 5 $A $E $I $TransfTag;
element elasticBeamColumn 3 2 4 $A $E [expr 2.*$I] $TransfTag;
element elasticBeamColumn 4 4 6 $A $E $I $TransfTag;
element elasticBeamColumn 5 3 4 $A $E [expr 2.*$I] $TransfTag;
element elasticBeamColumn 6 5 6 $A $E $I $TransfTag;
```

レコーダーを定義します。

指定した固有値 (numModes) の数(この例では 2)

[node recorder](#) コマンドを使用してすべての節点でのすべての自由度で固有ベクトルを記録する。

```
for { set k 1 } { $k <= $numModes } { incr k } {
    recorder Node -file [format "modes/mode%i.out" $k] -nodeRange 1 6 -dof 1 2 3
    "eigen $k"
}
```

固有値解析を実行します。

固有値は [eigen commnad](#) を使用して計算して、「lambda」という変数に格納されて計算されます。

```
set lambda [eigen $numModes];
```

構造体の周期と周波数は、次のように計算します。

```
set omega {}
set f {} #周波数
set T {} #固有周期
set pi 3.141593 #円周率
```

```
foreach lam $lambda {
    lappend omega [expr sqrt($lam)]
    lappend f [expr sqrt($lam)/(2*$pi)]
    lappend T [expr (2*$pi)/sqrt($lam)]
}
```

周期は、「modes」というフォルダの中の Periods.txt ファイルに格納されます。

```
set period "modes/Periods.txt"
set Periods [open $period "w"]
foreach t $T {
```

```
        puts $Periods " $t"
    }
close $Periods
```

固有ベクトルを記録する。

固有値を記録するには、[record](#) コマンドは [eigen](#) コマンドを受けて取得する必要があります。

```
record
```

モード形状を表示します。

システムの2つのモード形状は `record display` を使用して、2つのウィンドウが作成します。ウィンドウのタイトルを指定(今回は、"Mode Shape 1")、ウィンドウの左上の x と y の位置、ウィンドウの幅と高さをピクセル単位で指定する。プロジェクターの参照ポイント (`prp`) を定義します。このポイントを投影の中心を定義する。(詳細: [Viewpoint Projections and Specifications](#))。通常は、オブジェクトの中心に配置されます。この例では、構造物の中心である (h, h) 、 h は柱の高さ(図1)。次に、`view-up` ベクトル (`vup`) と `view-plane normal` ベクトル(`vpn`)を定義する。それらは、それぞれ $(0,1,0)$ と $(0,0,1)$ である。

ウィンドウの表示は、`viewWindow` コマンドを使用して4つの座標 ($-x$, x , y , y) を指定し、`prp` を関係したに表示ウィンドウのサイズを定義します。この例題では、 h は 120 なので、表示ウィンドウのベクトルは $(-170, 170, -170, 170)$ に設定され、構造物の周りに 56 単位の空白のスペースができる。最後に、`display` コマンドを使用して、モード形状を表示する。このコマンドの最初の引数は、プロットする応答の種類を指定する。(例 「-1」…は第1のモード形状を表示するためのタグ、「-2」 第2モード形状を表示するためのタグ)。2つ目は節点の応答の拡大率、3つ目は応答量の拡大率を表す。

```
recorder display "Mode Shape 1" 10 10 500 500 -wipe
```

```
prp $h $h 1;
```

```
vup 0 1 0;
```

```
vpn 0 0 1;
```

```
viewWindow -170, 170, -170, 170
```

```
display -1 5 20
```

```
recorder display "Mode Shape 2" 10 510 500 500 -wipe
```

```
prp $h $h 1;
```

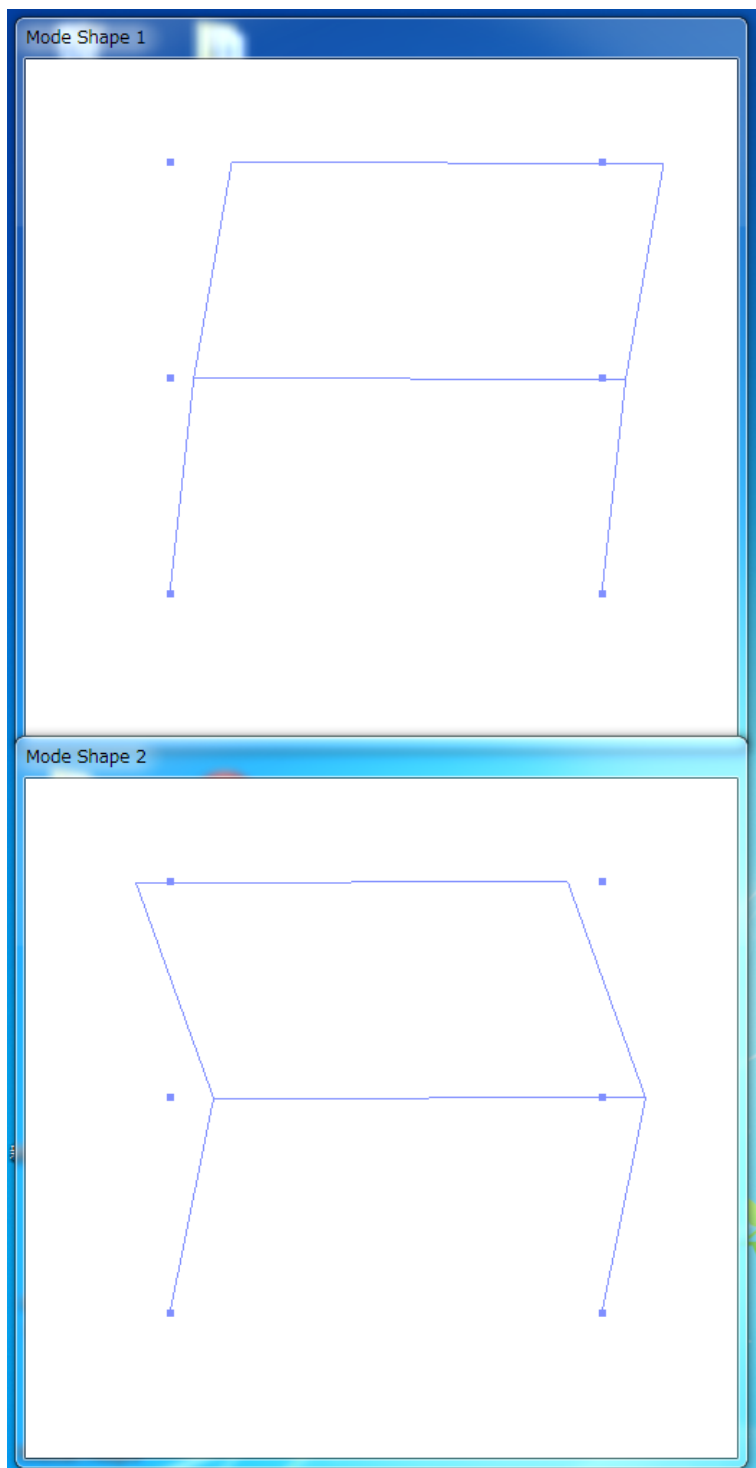
```
vup 0 1 0;
```

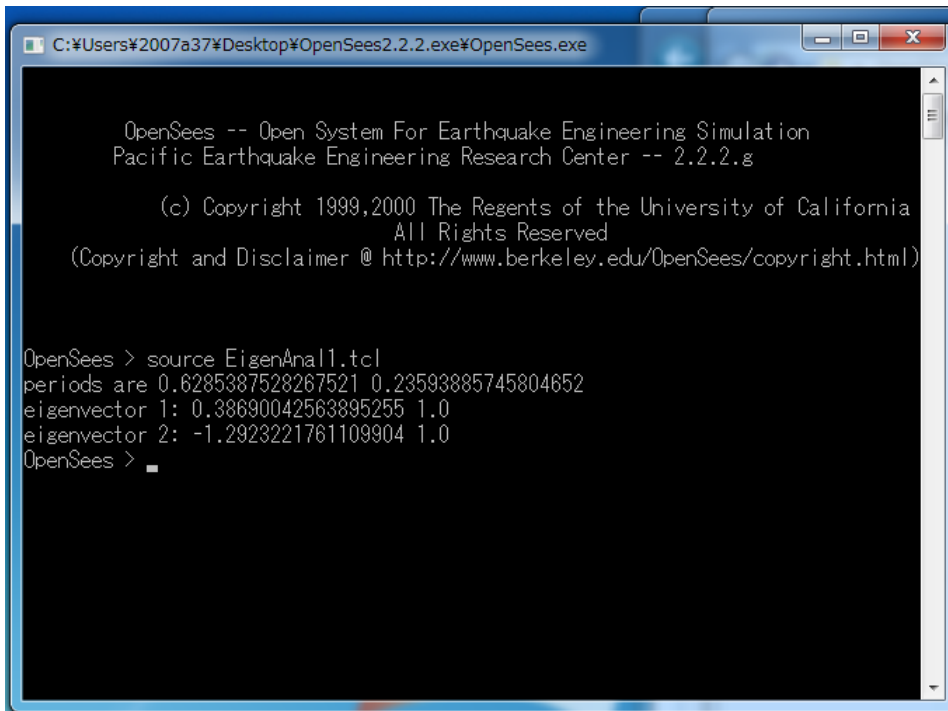
```
vpn 0 0 1;
```

```
viewWindow -170, 170, -170, 170
```

```
display -2 5 20
```

結果





```
C:\Users\2007a37\Desktop\OpenSees2.2.2.exe\OpenSees.exe

OpenSees -- Open System For Earthquake Engineering Simulation
Pacific Earthquake Engineering Research Center -- 2.2.2.g

(c) Copyright 1999,2000 The Regents of the University of California
All Rights Reserved
(Copyright and Disclaimer @ http://www.berkeley.edu/OpenSees/copyright.html)

OpenSees > source EigenAnal1.tcl
periods are 0.6285387528267521 0.23593885745804652
eigenvector 1: 0.38690042563895255 1.0
eigenvector 2: -1.2923221761109904 1.0
OpenSees > █
```