



دانشگاه صنعتی امیرکبیر
(پلی تکنیک تهران)

دانشکده مهندسی عمران و محیط زیست

درس دینامیک سازه‌ها

گزارش پروژه پایانی:

تحلیل استاتیکی و دینامیکی یک ساختمان 10 طبقه با استفاده از نرم افزار

ETABS و برنامه نویسی در MATLAB

استاد:

پروفسور محسن تهرانی زاده

تهیه و تنظیم:

علی میرزابابایی

عنوان پروژه:

پروژه مربوط به ساختمان 10 طبقه با کاربری تجاری واقع در شهر تهران و خاک نوع 3 می باشد. در این ساختمان بار مرده 600 کیلوگرم بر متر مربع، بار زنده 500 کیلوگرم بر متر مربع و بار دیوار پیرامونی 250 کیلوگرم بر متر مربع است. سقف از نوع دال دو طرفه می باشد. طول همه دهانه ها 6 متر و ارتفاع هر طبقه 5 متر می باشد. سیستم باربر جانبی در جهت طولی مهاربندی فولادی هم محور و در جهت عرضی قاب خمشی فولادی متوسط می باشد.

1. ساختمان را با استفاده از آنالیز استاتیکی معادل تحلیل کرده و مقاطع تیرها، ستون ها و بادبند های طراحی شده را گزارش نمایید.

بر اساس آیین نامه 2800 ویرایش سوم ضرایب مربوط به ضریب برش پایه به صورت زیر حساب می گردد:

• ضریب A :

بر اساس پیوست الف آیین نامه 2800 شهر تهران دارای درجه بندی خطر نسبی زلزله خیلی زیاد می باشد. بر اساس آیین نامه این ضریب برابر 0.35 می باشد.

• ضریب B :

برای به دست آوردن ضریب B ابتدا باید پرپود سازه حساب گردد. که به صورت زیر می باشد:

در جهت طولی چون مهاربندی است بر اساس بند 6-3-13 پرپود سازه برابر است با:

$$T = 0.05H^{\frac{3}{4}} = 0.05 \times 50^{0.75} = 0.94 \quad (Sec)$$

در جهت عرضی چون مهاربندی است بر اساس بند 6-3-13 پرپود سازه برابر است با:

$$T = 0.08H^{\frac{3}{4}} = 0.08 \times 50^{0.75} = 1.5 \quad (Sec)$$

حال بر این اساس که خاک سایت مورد نظر از نوع خاک 3 می باشد بنابراین بر اساس جدول 3 آیین نامه 2800 مقادیر T_0 و T_s و S به صورت زیر است:

$$T_s = 0.7, \quad T_0 = 0.15, \quad S = 1.75$$

چون $T > T_s$ بنابراین مقدار B با توجه به رابطه زیر به دست می آید:

$$B = (S + 1) \left(\frac{T_s}{T} \right)^{\frac{2}{3}}$$

با انجام محاسبات مقدار B در جهت طولی برابر 2.56 و در جهت عرضی برابر 1.65 می باشد.

• ضریب I: با توجه به بخش 10 و جدول 5 ساختمان اداری در دسته ساختمان های با اهمیت متوسط دسته بندی می گردد در نتیجه ضریب I برابر 1 می باشد.

• ضریب R: بر اساس جدول 6 آیین نامه 2800 ضریب رفتار برای این ساختمان در جهت طولی برابر 6 و در جهت عرضی برابر 7 می باشد.

با استفاده از مقادیر بالا ضریب برش پایه برای دو راستا به صورت زیر به دست می آید:

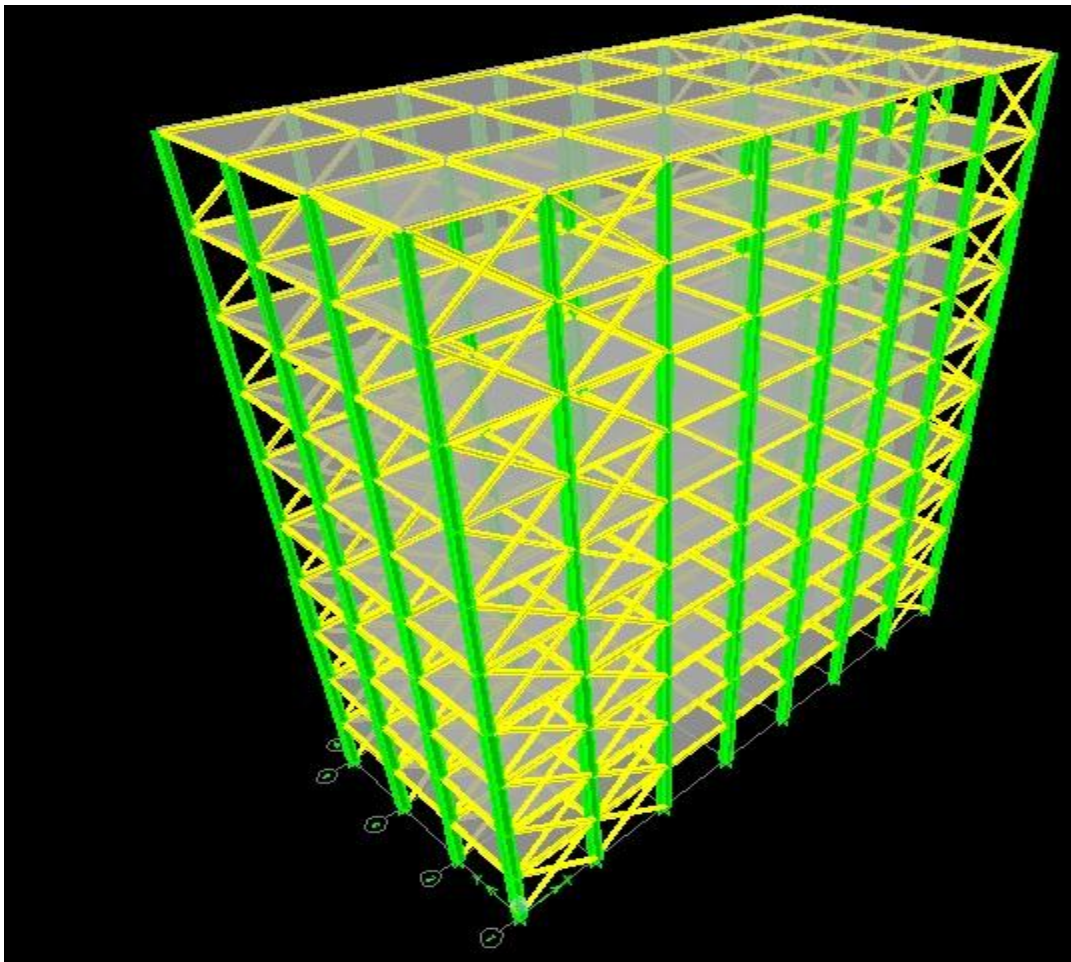
$$C = \frac{ABI}{R_x} = \frac{0.35 \times 2.56 \times 1}{6} = 0.128$$

$$C = \frac{ABI}{R_y} = \frac{0.35 \times 1.65 \times 1}{7} = 0.0825$$

حال با تحلیل با استفاده از نرم افزار ETABS و طراحی سازه، مقاطع به دست آمده با استفاده از این نرم افزار به صورت زیر می باشد: لازم به ذکر است که در این پروژه مقاطع ستون به صورت جعبه ای ساخته شده با استفاده از ورق می باشد که در طبقات اول و دوم و سوم از دو ورق تقویتی در داخل ستون استفاده شده است. مقاطع بادبند نیز به صورت دابل می باشد.

بادبند	تیر	ستون	طبقه
2UNP240	40*20*2*1	60*60*4	اول
2UNP240	40*20*2*1	60*60*4	دوم
2UNP240	40*20*2*1	60*60*4	سوم
2UNP220	40*20*2*1	60*60*4	چهارم
2UNP220	40*20*2*1	50*50*4	پنجم
2UNP200	20*10*2*1	50*50*4	ششم
2UNP200	20*10*2*1	50*50*4	هفتم
2UNP200	20*10*2*1	50*50*4	هشتم
2UNP180	18*9*2*1	40*40*4	نهم
2UNP180	18*9*2*1	40*40*4	دهم

نمای سه بعدی سازه به صورت زیر می باشد:



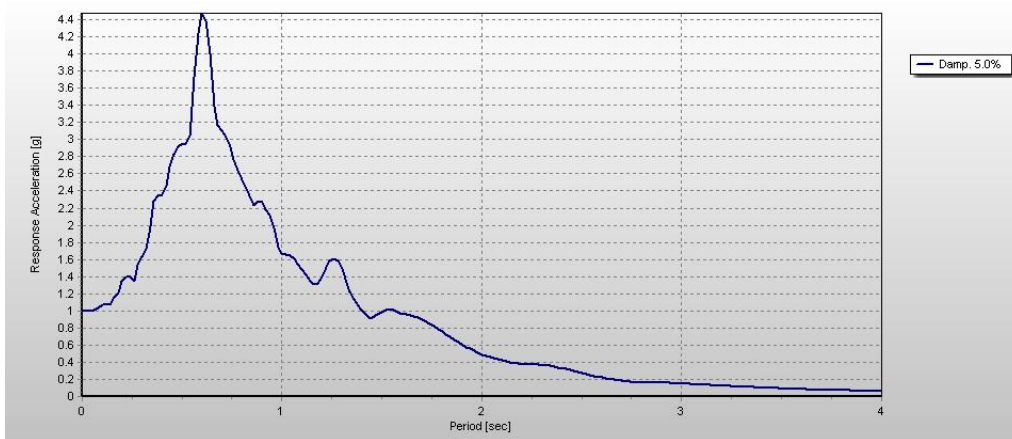
2. تحلیل دینامیکی با استفاده از نرم افزار ETABS :

برای تحلیل با استفاده از نرم افزار ETABS ابتدا باید زلزله‌ها را مقیاس کنیم. بر اساس بند 13-4-1-4-2 در آیین نامه 2800 مراحل مقیاس کردن زلزله‌ها به صورت زیر می‌باشد:

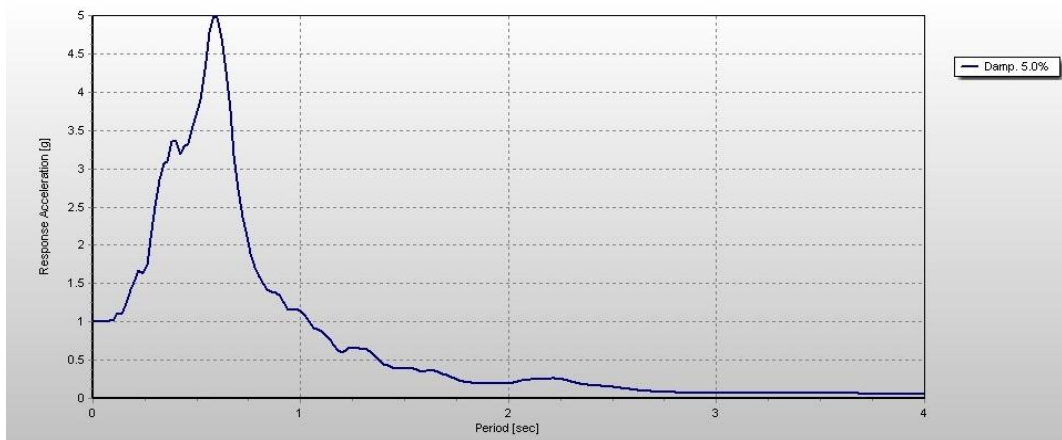
1. کلیه شتاب نگاشت‌ها به ماکزیمم مقدار خود مقیاس شود یعنی حداکثر شتاب هم آنها برابر g گردد.
2. طیف پاسخ هر یک از زوج شتاب نگاشت‌ها با فرض میرایی 5 درصد تعیین گردد.
3. طیف‌های پاسخ هر یک از زوج شتاب نگاشت‌ها با استفاده از روش جذر مجموع مربعات با یکدیگر ترکیب شده و یک طیف ترکیبی واحد برای هر زوج ساخته شود.
4. طیف‌های پاسخ ترکیبی متوسط گیری شده و در محدوده زمان‌های تناوب $0.2T$ و $1.5T$ با طیف طرز استاندارد مقایسه می‌شود. ضریب مقیاس آنچنان تعیین می‌گردد که در این محدوده مقادیر متوسط‌ها در هیچ حالت کمتر از 1.4 برابر مقدار نظیر آن در طیف استاندارد نباشد. T زمان تناوب اصلی سازه می‌باشد.

زلزله‌های مورد بررسی در این پروژه زلزله‌های کوبه و لوما پرایتا و طیس می‌باشد که مراحل مقیاس کردن آنها بر اساس بند آیین نامه 2800 به صورت زیر می‌باشد.

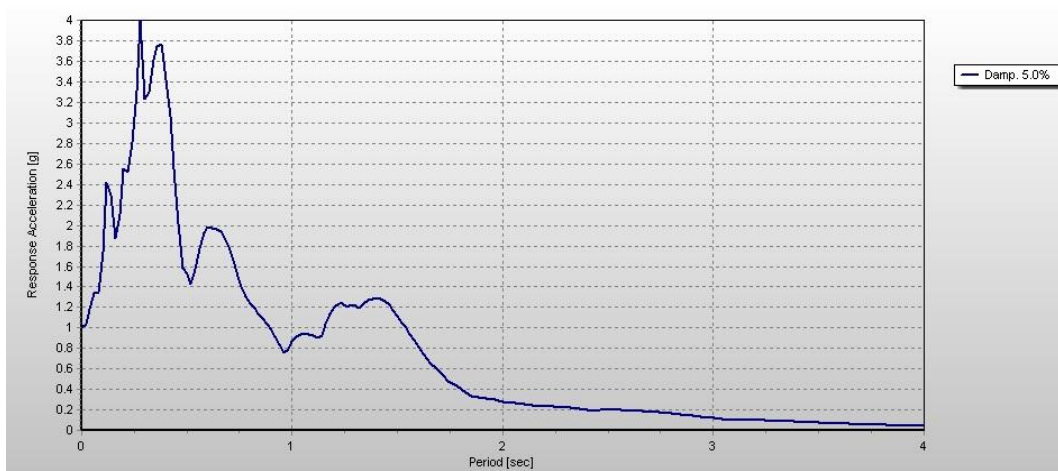
- به دست آوردن طیف پاسخ برای هر زوج شتاب نگاشت (قبل از آن به حداکثر مقدار خود مقیاس شده است). این طیف با استفاده از نرم‌افزار Seismo Signal به دست آمده است.



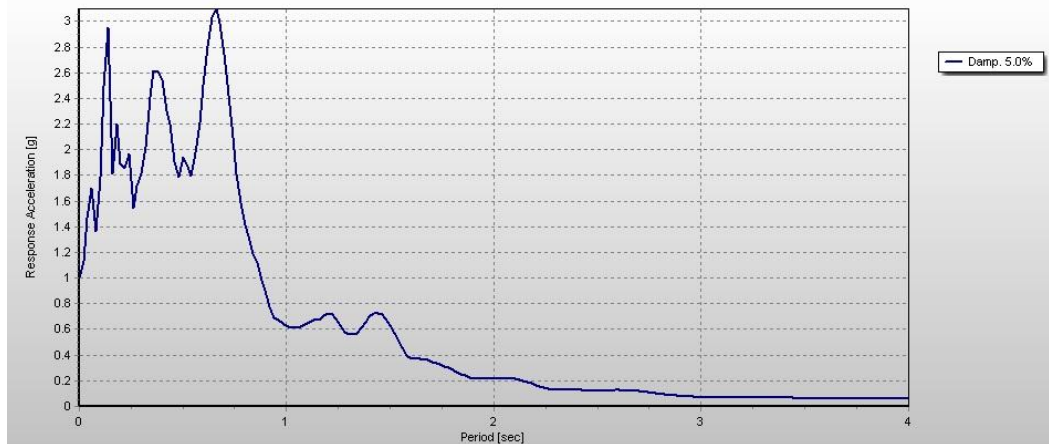
طیف پاسخ برای زلزله کوبه در جهت X



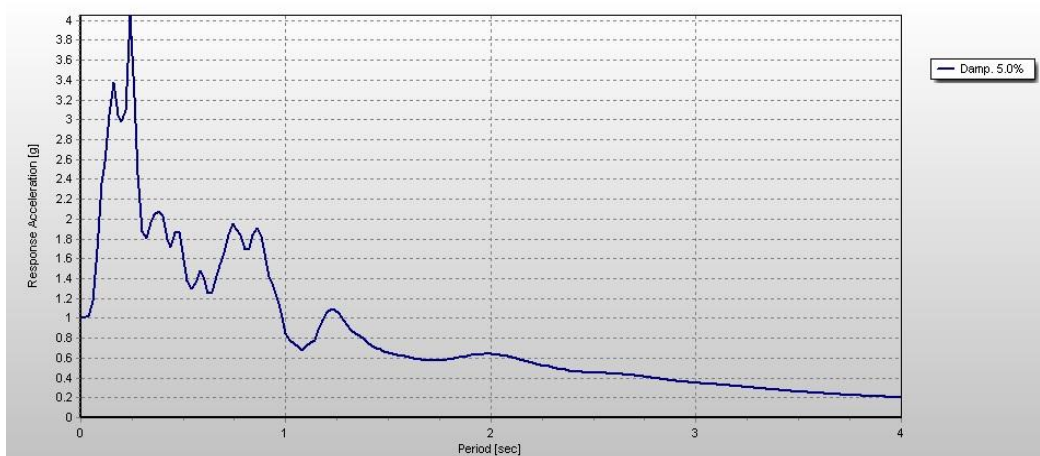
طیف پاسخ برای زلزله کوبه در جهت Y



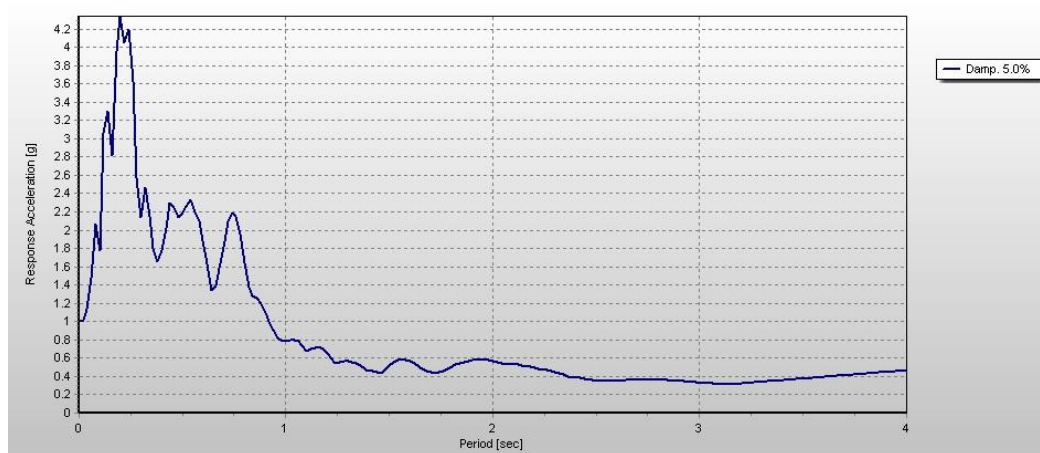
طیف پاسخ برای زلزله لوما پرایتا در جهت X



طیف پاسخ برای زلزله لوما پرایتا در جهت Y

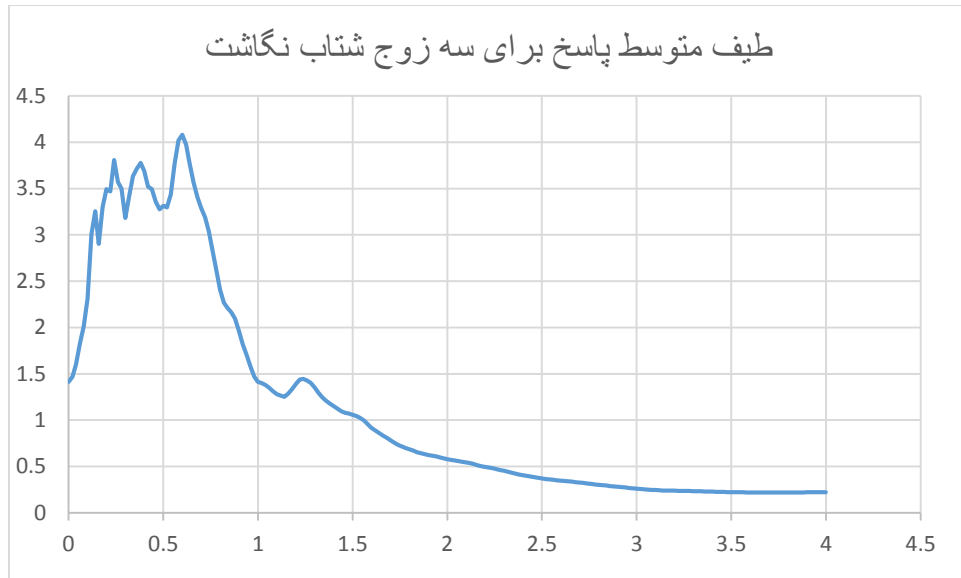


طیف پاسخ برای زلزله طیس در جهت X

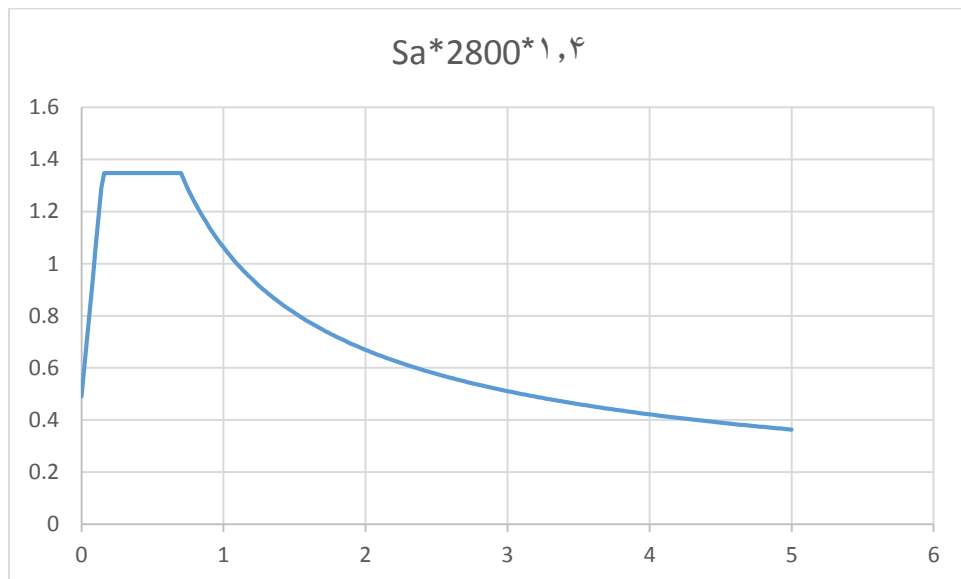


طیف پاسخ برای زلزله طیس در جهت Y

- با ترکیب هر یک از زوج شتاب نگاشت‌ها با استفاده از روش SRSS و متوسط‌گیری از بین سه طیف ترکیبی، طیف میانگین به صورت به دست می‌آید.



حال با ضرب ضریب A در طیف طرح خاک نوع 3 و ضرب عدد 1.4 بر آن طیف طرح اصلاح شده به صورت زیر به دست می‌آید.



با مقایسه دو طیف بالا در محدوده 0.2T و 1.5T ضرایب مقیاس به ترتیب در جهت طولی و عرضی برابر است با 2.13 و 2.94.

حال این ضرایب را در شتاب نگاشت مرحله 1 آیین نامه ضرب کرده و به برنامه ETABS وارد کرده و سپس اقدام به تحلیل دینامیکی می‌کنیم.

3. معادله تحلیل دینامیکی ساختمان را در دو جهت X و Y :

برای به دست آوردن معادله تحلیل دینامیکی در دو جهت ابتدا باید جرم سازه و سختی و میرایی سازه را به دست آوریم. که این مقادیر را به ترتیب می‌توانیم به صورت زیر به دست آوریم:

جرم هر طبقه از سازه برابر است با :

$$M_i = (A \times 600) + (A \times 0.4 \times 500) + (120 \times 5 \times 250) = 754800 \quad Kg$$

برای طبقات اول و آخر چون نصف دیوار پیرامونی در محاسبه جرم لحاظ می‌گردد بنابراین جرم این طبقات برابر است با:

$$M_i = (A \times 600) + (A \times 0.4 \times 500) + (120 \times 2.5 \times 250) = 679800 \quad Kg$$

بنابر این ماتریس جرم برای پروژ به صورت زیر می باشد:

[illegible]

برای به دست آوردن سختی چون جرم کل را حساب کرده‌ایم سختی کل را برای هر طبقه به دست می‌آوریم.

سختی ستون هر طبقه برابر است با $\frac{12EI}{L^3}$ و سختی مهاربند برابر است با $\frac{EA}{L} \cos^2 \alpha$ که زاویه به

دلیل برابر بودن ارتفاع هر طبقه برابر است با $\alpha = \tan^{-1}(\frac{5}{6})$ یا همان 39.8° می‌باشد.

بنابراین سختی هر طبقه در راستای X به صورت جدول زیر می باشد.

طبقه	ستون $\frac{N}{m}$	بادبند $\frac{N}{m}$	سختی کل $\frac{N}{m}$
اول	1.13×10^8	1.336×10^8	62.872×10^8
دوم	1.13×10^8	1.336×10^8	62.872×10^8
سوم	1.13×10^8	1.336×10^8	62.872×10^8
چهارم	1.13×10^8	1.183×10^8	59.82×10^8
پنجم	0.627×10^8	1.183×10^8	43.724×10^8
ششم	0.627×10^8	1.02×10^8	40.46×10^8
هفتم	0.627×10^8	1.02×10^8	40.46×10^8
هشتم	0.627×10^8	1.02×10^8	40.46×10^8
نهم	0.148×10^8	0.886×10^8	22.416×10^8
دهم	0.148×10^8	0.886×10^8	22.416×10^8

همچنین سختی کل هر طبقه در راستای Y به صورت جدول زیر می باشد:

طبقه	ستون $\frac{N}{m}$	بادبند $\frac{N}{m}$	سختی کل $\frac{N}{m}$
اول	1.13×10^8	-----	36.16×10^8
دوم	1.13×10^8	-----	36.16×10^8
سوم	1.13×10^8	-----	36.16×10^8
چهارم	1.13×10^8	-----	36.16×10^8
پنجم	0.627×10^8	-----	20.064×10^8
ششم	0.627×10^8	-----	20.064×10^8
هفتم	0.627×10^8	-----	20.064×10^8
هشتم	0.627×10^8	-----	20.064×10^8
نهم	0.148×10^8	-----	4.736×10^8
دهم	0.148×10^8	-----	4.736×10^8

بنابراین ماتریس‌های سختی در راستای طولی و عرضی به صورت زیر می‌باشد.
ماتریس سختی در راستای X :

1.26E+10	-6.29E+09	0	0	0	0	0	0	0	0
-6.29E+09	1.26E+10	-6.29E+09	0	0	0	0	0	0	0
0	-6.29E+09	1.23E+10	-5.98E+09	0	0	0	0	0	0
0	0	-5.98E+09	1.04E+10	-4.37E+09	0	0	0	0	0
0	0	0	-4.37E+09	8.42E+09	-4.05E+09	0	0	0	0
0	0	0	0	-4.05E+09	8.09E+09	-4.05E+09	0	0	0
0	0	0	0	0	-4.05E+09	8.09E+09	-4.05E+09	0	0
0	0	0	0	0	0	-4.05E+09	6.28E+09	-2.24E+09	0
0	0	0	0	0	0	0	-2.24E+09	4.48E+09	-2.24E+09
0	0	0	0	0	0	0	0	-2.24E+09	2.24E+09

ماتریس سختی در راستای Y :

7.23E+09	-3.62E+09	0	0	0	0	0	0	0	0	0
-3.62E+09	7.23E+09	-3.62E+09	0	0	0	0	0	0	0	0
0	-3.62E+09	7.23E+09	-3.62E+09	0	0	0	0	0	0	0
0	0	-3.62E+09	5.62E+09	-2.01E+09	0	0	0	0	0	0
0	0	0	-2.01E+09	4.01E+09	-2.01E+09	0	0	0	0	0
0	0	0	0	-2.01E+09	4.01E+09	-2.01E+09	0	0	0	0
0	0	0	0	0	-2.01E+09	4.01E+09	-2.01E+09	0	0	0
0	0	0	0	0	0	-2.01E+09	6.28E+09	-2.24E+09	0	0
0	0	0	0	0	0	0	-2.24E+09	2.48E+09	-4.74E+08	0
0	0	0	0	0	0	0	0	-4.74E+08	4.74E+08	0

برای به دست آوردن ماتریس میرایی با استفاده از روش گفته شده در کتاب چوپرا، رابطه ارائه شده برای ماتریس میرایی به صورت زیر می‌باشد:

$$[C] = a_0[M] + a_1[K]$$

که ضرایب با استفاده از رابطه زیر به دست می‌آید:

$$[a] = 2[Q]^{-1}[\xi]$$

و در رابطه بالا ماتریس Q به صورت زیر می‌باشد:

$$[Q] = \begin{bmatrix} \frac{1}{\omega_1} & \omega_1 & \omega_1^3 \\ \frac{1}{\omega_2} & \omega_2 & \omega_2^3 \\ \frac{1}{\omega_3} & \omega_3 & \omega_3^3 \end{bmatrix}$$

در نتیجه ماتریس میرایی به صورت زیر می‌باشد:

54414808.9	-2.8E+07	0	0	0	0	0	0	0	0
-27768421	54414809	-2.8E+07	0	0	0	0	0	0	0
0	-2.8E+07	54414809	-2.8E+07	0	0	0	0	0	0
0	0	-2.8E+07	42054175	-1.5E+07	0	0	0	0	0
0	0	0	-1.5E+07	29693542	-1.5E+07	0	0	0	0
0	0	0	0	-1.5E+07	29693542	-1.5E+07	0	0	0
0	0	0	0	0	-1.5E+07	29693542	-1.5E+07	0	0
0	0	0	0	0	0	-1.5E+07	47093345	-1.7E+07	0
0	0	0	0	0	0	0	-1.7E+07	17922680	-3636926
0	0	0	0	0	0	0	0	-3636926	2514892

ماتریس نیرو به این دلیل که سازه تحت اثر شتاب زلزله قرار می‌گیرد را می‌توان به صورت زیر نوشت:

$$\{P(t)\} = -P_{eff} = -[M]\{I\}\ddot{X}_g$$

بنابراین با استفاده مقادیر به دست آمده در بالا معادله تعادل دینامیکی در هر دو جهت به صورت زیر می‌باشد:

$$[M]\{\ddot{u}(t)\} + [C]\{\dot{u}(t)\} + [K_x]\{u\} = \{p(t)\}$$

که در رابطه به بالا ماتریس سختی برابر ماتریس سختی به دست آمده در بالا در جهت طولی می‌باشد. ولی ماتریس جرم و میرایی تغییری نخواهد کرد.

$$[M]\{\ddot{u}(t)\} + [C]\{\dot{u}(t)\} + [K_y]\{u\} = \{p(t)\}$$

در رابطه بالا ماتریس سختی برابر ماتریس سختی در راستای عرضی می‌باشد که در بالا محاسبه گشته است.

4. تعیین معادلات مودال برای هر مد:

برای به دست آوردن معادلات مودال برای هر مد ابتدا باید فرکانس و بردار مدی را برای هر مد به دست آوریم.

برای به دست آوردن فرکانس هر مد با استفاده از روش عددی که در کتاب دینامیک سازه‌ها تألیف دکتر محسن تهرانی زاده و با استفاده از برنامه نویسی در نرم افزار MATLAB که در کد برنامه در قسمت پیوست آمده می‌توان فرکانس را برای هر مد به صورت جدول زیر نمایش دهیم.

مد	1	2	3	4	5	6	7	8	9	10
فرکانس $\frac{rad}{s}$	17.11	22.9	37.42	48.65	65.43	79.73	93.03	105.33	110.64	131

و بردارهای مدی نیز با استفاده از برنامه نویسی در نرم افزار MATLAB و با استفاده از روش عددی ارائه شده در کتاب دینامیک سازه ها تالیف دکتر محسن تهرانی زاده می توان بردار مدی را به صورت جدول زیر به دست نمایش داد.

1	1	1	1	1	1	1	1	1	1	1
1.938851	1.890532	1.707645	1.505929	1.106319	0.672848	0.192862	-0.31738	-0.55553	-1.58176	
2.759144	2.574116	1.916043	1.267865	0.223876	-0.5472	-0.96282	-0.89924	-0.69132	1.501651	
3.410722	2.975928	1.564244	0.403502	-0.85878	-1.04093	-0.37869	0.602591	0.939369	-0.79281	
4.201611	3.107444	0.10573	-1.51058	-1.4244	0.557925	1.903953	0.79128	-0.44694	0.188439	
4.51527	2.615956	-1.40897	-2.0765	0.305514	0.822204	-2.00895	-2.32019	0.22275	-0.04486	
4.316091	1.600037	-2.17776	-0.78964	1.543475	-0.87921	0.614905	4.244208	-0.13205	0.010907	
3.626719	0.263375	-1.79344	1.201442	0.294182	-0.47813	1.237916	-6.88899	0.120284	-0.00356	
5.93042	-0.74351	-2.22469	3.115596	-0.98432	0.472673	-0.69117	2.63291	-0.04009	0.000769	
11.12083	-4.51868	1.805798	-1.12377	0.169091	-0.05174	0.054033	-0.15785	0.002168	-2.94E-05	

در مرحله بعد باید جرم مودی را برای هر مد به دست آوریم. جرم مودی برای هر مد را با استفاده از رابطه زیر می توانیم به دست بیاوریم:

$$\tilde{X}_i^T [M] \tilde{X}_i$$

با استفاده از فرمول بالا جرم مودی برای هر مد را با استفاده از برنامه نویسی در نرم افزار MATLAB که در فایل پیوست آمده است، می توان به صورت جدول زیر نمایش داد:

مد	1	2	3	4	5	6	7	8	9	10
جرم مدی $\times 10^8$	1.907	0.454	0.213	0.1862	0.0649	0.03812	0.0918	0.06092	0.02228	0.04848

ماتریس نیروی مودی را با استفاده از فرمول زیر به دست می آوریم:

$$f_{k(t)} = -\tilde{X}^T [M] \{\ddot{I}\} \ddot{X}_g$$

ضرایب زلزله هر مد $\{I\} [M] X_i^T$ در جدول زیر نشان داده شده است.

مد	1	2	3	4	5	6	7	8	9	10
ضریب زلزله مد $\times 10^{+7}$	3.232	0.813	0.0373	0.226	0.1038	0.0398	0.0726	- 0.099	0.0316	0.02104

میرایی مدی برای هر مد را با استفاده از رابطه زیر به دست می آوریم:

$$\tilde{X}_i^T [C] \tilde{X}_i$$

میرایی مدی را برای هر مد را می توان به صورت جدول زیر نشان داد:

مد	1	2	3	4	5	6	7	8	9	10
میرایی مد $\times 10^{+8}$	1.45	1.15	1.97	3.107	2.038	1.804	5.96	5.1	2.06	6.32

سختی مدی برای هر مد را با استفاده از رابطه زیر به دست می آوریم:

$$\tilde{X}_i^T [K] \tilde{X}_i$$

سختی مدی را برای هر مد در هر جهت را می توان به صورت جدول زیر نشان داد:

سختی مدی در جهت X :

مد	1	2	3	4	5	6	7	8	9	10
سختی مد $\times 10^{+10}$	9.57	6.18	6.87	10.125	5.09	5.424	14.45	96.85	4.857	14.39

سختی مدی در جهت Y:

مد	1	2	3	4	5	6	7	8	9	10
سختی مد $\times 10^{+10}$	5.586	2.38	2.98	4.407	2.78	2.42	7.94	67.58	2.73	8.32

بنابراین معادلات مودال برای هر مد را می‌توان به صورت زیر نشان داد:

• مد اول:

$$1.907 \times 10^8 \ddot{Y}(t) + 1.45 \times 10^8 \dot{Y}(t) + 9.57 \times 10^{10} \times Y(t) = -3.232 \times 10^7 \ddot{X}_g \quad X \text{ direction}$$

$$1.907 \times 10^8 \ddot{Y}(t) + 1.45 \times 10^8 \dot{Y}(t) + 5.586 \times 10^{10} \times Y(t) = -3.232 \times 10^7 \ddot{X}_g \quad Y \text{ direction}$$

• مد دوم:

$$0.454 \times 10^8 \ddot{Y}(t) + 1.15 \times 10^8 \dot{Y}(t) + 6.18 \times 10^{10} \times Y(t) = -0.813 \times 10^7 \ddot{X}_g \quad X \text{ direction}$$

$$0.454 \times 10^8 \ddot{Y}(t) + 1.45 \times 10^8 \dot{Y}(t) + 2.38 \times 10^{10} \times Y(t) = -0.813 \times 10^7 \ddot{X}_g \quad Y \text{ direction}$$

• مد سوم:

$$0.213 \times 10^8 \ddot{Y}(t) + 1.97 \times 10^8 \dot{Y}(t) + 6.87 \times 10^{10} \times Y(t) = -0.0373 \times 10^7 \ddot{X}_g \quad X \text{ direction}$$

$$0.213 \times 10^8 \ddot{Y}(t) + 1.97 \times 10^8 \dot{Y}(t) + 2.98 \times 10^{10} \times Y(t) = -0.0373 \times 10^7 \ddot{X}_g \quad Y \text{ direction}$$

• مد چهارم:

$$0.1862 \times 10^8 \ddot{Y}(t) + 3.107 \times 10^8 \dot{Y}(t) + 10.125 \times 10^{10} \times Y(t) = -0.226 \times 10^7 \ddot{X}_g \quad X \text{ direction}$$

$$0.1862 \times 10^8 \ddot{Y}(t) + 3.107 \times 10^8 \dot{Y}(t) + 4.407 \times 10^{10} \times Y(t) = -0.226 \times 10^7 \ddot{X}_g \quad Y \text{ direction}$$

• مد پنجم:

$$0.0649 \times 10^8 \ddot{Y}(t) + 2.038 \times 10^8 \dot{Y}(t) + 14.39 \times 10^{10} \times Y(t) = -0.1038 \times 10^7 \ddot{X}_g \quad X \text{ direction}$$

$$0.0649 \times 10^8 \ddot{Y}(t) + 2.038 \times 10^8 \dot{Y}(t) + 2.78 \times 10^{10} \times Y(t) = -0.1038 \times 10^7 \ddot{X}_g \quad Y \text{ direction}$$

• مد ششم:

$$0.03812 \times 10^8 \ddot{Y}(t) + 1.804 \times 10^8 \dot{Y}(t) + 5.424 \times 10^{10} \times Y(t) = -0.0398 \times 10^7 \ddot{X}_g \quad X \text{ direction}$$

$$0.03812 \times 10^8 \ddot{Y}(t) + 1.804 \times 10^8 \dot{Y}(t) + 2.42 \times 10^{10} \times Y(t) = -0.0398 \times 10^7 \ddot{X}_g \quad Y \text{ direction}$$

• مد هفتم:

$$0.0918 \times 10^8 \ddot{Y}(t) + 5.96 \times 10^8 \dot{Y}(t) + 14.45 \times 10^{10} \times Y(t) = -0.0726 \times 10^7 \ddot{X}_g \quad X \text{ direction}$$

$$0.0918 \times 10^8 \ddot{Y}(t) + 5.96 \times 10^8 \dot{Y}(t) + 7.94 \times 10^{10} \times Y(t) = -0.0726 \times 10^7 \ddot{X}_g \quad Y \text{ direction}$$

• مد هشتم:

$$0.061 \times 10^8 \ddot{Y}(t) + 5.1 \times 10^8 \dot{Y}(t) + 96.85 \times 10^{10} \times Y(t) = +0.099 \times 10^7 \ddot{X}_g \quad X \text{ direction}$$

$$0.061 \times 10^8 \ddot{Y}(t) + 5.1 \times 10^8 \dot{Y}(t) + 67.58 \times 10^{10} \times Y(t) = +0.099 \times 10^7 \ddot{X}_g \quad Y \text{ direction}$$

• مد نهم:

$$0.0223 \times 10^8 \ddot{Y}(t) + 2.06 \times 10^8 \dot{Y}(t) + 4.857 \times 10^{10} \times Y(t) = -0.0316 \times 10^7 \ddot{X}_g \quad X \text{ direction}$$

$$0.0223 \times 10^8 \ddot{Y}(t) + 2.06 \times 10^8 \dot{Y}(t) + 2.73 \times 10^{10} \times Y(t) = -0.0316 \times 10^7 \ddot{X}_g \quad Y \text{ direction}$$

• مد دهم:

$$0.0485 \times 10^8 \ddot{Y}(t) + 6.32 \times 10^8 \dot{Y}(t) + 9.57 \times 10^{10} \times Y(t) = -0.021 \times 10^7 \ddot{X}_g \quad X \text{ direction}$$

$$0.0485 \times 10^8 \ddot{Y}(t) + 6.32 \times 10^8 \dot{Y}(t) + 8.32 \times 10^{10} \times Y(t) = -0.021 \times 10^7 \ddot{X}_g \quad Y \text{ direction}$$

5. حل معادلات با استفاده از انتگرال دوهمل و روش عددی نیومارک و روش عددی ویلسون

برای حل معادلات مودال با استفاده از انتگرال دوهمل با استفاده از رابطه زیر و برنامه نویسی در نرم افزار MATLAB تغییر مکان هر طبقه به دست می‌آید.

$$X_i(t) = \sum_{k=1}^{\infty} \ddot{X}_g [(C_k \cos(\omega_k t) + D_k \sin(\omega_k t)) - \frac{\alpha_k}{\omega_k} \int_0^t \ddot{X}_g * \sin(\omega_k(t-\tau)) * e^{-\xi \omega_k(t-\tau)} d\tau]$$

همچنین برای روش نیومارک از روش شتاب متوسط استفاده شده است که در این حالت $\alpha = 0.25$ $\delta = 0.5$

$$\delta = 0.5 \quad \alpha = \frac{1}{6} \quad \theta = 1.35 \quad \text{و در روش عددی ویلسون از روش شتاب خطی}$$

استفاده شده است. برای حل معادلات مودال با استفاده از روش‌های عددی از برنامه نویسی در نرم افزار MATLAB استفاده شده است.

کد برنامه نویسی برای سه روش فوق در پیوست آمده است.

6. نتایج تحلیل دینامیکی و مقایسه آنها با یکدیگر:

با انجام تحلیل دینامیکی در نرم افزار ETABS و برنامه نویسی در نرم افزار MATLAB نتایج به دست آمده برای هر یک از روش‌های فوق در جدول زیر نشان داده شده است.

• زلزله کوبه در جهت طولی:

ویلسون	نیومارک	دوهمل	ETABS	طبقه
0.0039	0.0017	0.0015	0.001	1
0.0078	0.0034	0.0033	0.0025	2
0.0115	0.0050	0.0041	0.0031	3
0.0151	0.0066	0.0075	0.0054	4
0.0198	0.0086	0.0165	0.0072	5

6	0.0098	0.0198	0.0105	0.0242
7	0.0112	0.0098	0.0121	0.028
8	0.0124	0.011	0.0134	0.0310
9	0.0135	0.0125	0.0149	0.0346
10	0.0166	0.0154	0.0156	0.0365

• زلزله کوبه در جهت عرضی:

طابقه	ETABS	دوهمل	نیومارک	ویلسون
1	0.0015	0.001	0.0011	0.0013
2	0.0019	0.0015	0.0022	0.0026
3	0.0028	0.0025	0.0031	0.0036
4	0.0033	0.0030	0.0038	0.0044
5	0.0036	0.0039	0.0046	0.0053
6	0.0040	0.0043	0.0048	0.0055
7	0.0043	0.0045	0.0044	0.005
8	0.0033	0.0030	0.0035	0.0041
9	0.0060	0.0059	0.0057	0.0066
10	0.0098	0.0105	0.01128	0.01233

- زلزله لوما پرایتا در جهت طولی:

ویلسون	نیومارک	دوهمل	ETABS	طبقه
0.026	0.0346	0.0198	0.022	1
0.0522	0.0691	0.0395	0.045	2
0.0762	0.1027	0.0874	0.0965	3
0.0989	0.1367	0.0102	0.1125	4
0.1259	0.1804	0.1154	0.1365	5
0.1504	0.2227	0.1564	0.1854	6
0.1703	0.2583	0.1954	0.2233	7
0.1856	0.2858	0.2254	0.2467	8
0.2048	0.3181	0.2478	0.2741	9
0.2151	0.3342	0.2987	0.3002	10

- زلزله لوما پرایتا در جهت عرضی:

ویلسون	نیومارک	دوهمل	ETABS	طبقه
0.0321	0.0686	0.0569	0.0456	1
0.0617	0.1329	0.0987	0.1035	2
0.0869	0.1890	0.1125	0.1457	3
0.1058	0.2333	0.1347	0.1856	4
0.1260	0.2865	0.1621	0.2144	5
0.1319	0.3068	0.1936	0.2568	6
0.1317	0.2919	0.2147	0.2657	7
0.1230	0.2435	0.1987	0.2358	8
0.2146	0.3964	0.2587	0.3355	9
0.4377	0.7394	0.4525	0.5565	10

• زلزله طبس در جهت طولی

طبقه	ETABS	دوهمل	نیومارک	ویلسون
1	0.0325	0.0298	0.043	0.0375
2	0.0547	0.0498	0.08	0.0743
3	0.0987	0.0784	0.128	0.1098
4	0.136	0.125	0.170	0.1449
5	0.184	0.154	0.224	0.1891
6	0.234	0.225	0.276	0.2314
7	0.298	0.274	0.320	0.2694
8	0.314	0.294	0.356	0.3010
9	0.356	0.335	0.399	0.3409
10	0.398	0.365	0.422	0.3618

• زلزله طبس در جهت عرضی

طبقه	ETABS	دوهمل	نیومارک	ویلسون
1	0.0325	0.0356	0.0437	0.0642
2	0.0988	0.0985	0.1065	0.1241
3	0.1254	0.1365	0.1477	0.1756
4	0.1547	0.1565	0.1866	0.2152
5	0.1987	0.2256	0.2271	0.2598
6	0.2189	0.2365	0.2414	0.2722
7	0.1856	0.2236	0.2292	0.2514
8	0.1547	0.1745	0.1804	0.2024
9	0.2365	0.2856	0.2731	0.3399
10	0.4578	0.5547	0.5576	0.6656

پیوست:

کد برنامه نویسی برای به دست آوردن فرکانس و اشکال مدی:

```

%Mode shapes and frequency of multiple degree of freedom system
% Dynamic of Structures: Dr Mohsen Tehranizadeh
% This program was Written by Vahid Shahmoradi-93124107
clc;
clear all;
m=xlsread('mass.xlsx');
k=xlsread('stiffness.xlsx');
k1=inv(k);
d=k1*m;
n=size(m,1);
n1=1;
n2=n;
ep=1;
w=[];
frequency={};
Mody_Mass=[];
mod={};
sweeping={};
j=1;
y=n1+(n2-n1)*rand(n,1);
yn=d*y;
w(1,j)=sqrt(norm(y)/norm(yn));
yn=yn/yn(1,1);
y=yn;
while (abs(ep)>0.000001)
j=j+1;
yn=d*y;
w(1,j)=sqrt(norm(y)/norm(yn));
ep=w(1,j)-w(1,j-1);
yn=yn/yn(1,1);
y=yn;
end
frequency{1}=w(1,:);
mod{1}=yn;
Mody_Mass(1,1)=(yn')*m*yn;
j=1;
sweeping{1}=eye(n)-(yn*yn'*m)/Mody_Mass(1,1);
y=n1+(n2-n1)*rand(n,1);
yn=d*cell2mat(sweeping(1,1))*y;
w(2,j)=sqrt(norm(y)/norm(yn));
yn=yn/yn(1,1);
y=yn;
ep=1;
while (abs(ep)>0.000001)
j=j+1;
yn=d*cell2mat(sweeping(1,1))*y;

```



```

w(2,j)=sqrt(norm(y)/norm(yn));
ep=w(2,j)-w(2,j-1);
yn=yn/yn(1,1);
y=yn;
end
mod{2}=yn;
frequency{2}=w(2,:);
Mody_Mass(1,2)=(yn')*m*yn;
w(1,:)=0;
for i=3:n
    ep=1;
    j=1;
    w(1,:)=0;
    sweeping{i-1}=sweeping{i-2}-(yn*yn'*m)/Mody_Mass(1,i-1);
    y=n1+(n2-n1)*rand(n,1);
    yn=d*cell2mat(sweeping(1,i-1))*y;
    w(i,j)=sqrt(norm(y)/norm(yn));
    yn=yn/yn(1,1);
    y=yn;
    while (abs(ep)>0.000001)
        j=j+1;
        yn=d*cell2mat(sweeping(1,i-1))*y;
        w(i,j)=sqrt(norm(y)/norm(yn));
        ep=w(i,j)-w(i,j-1);
        yn=yn/yn(1,1);
        y=yn;
    end
    mod{i}=yn;
    frequency{i}=w(i,:);
    Mody_Mass(1,i)=(yn')*m*yn;
end
%Reporting the Analysis
%Frequency
frequency1=[];
for i=1:n
    fr=cell2mat(frequency(1,i));
    ans=1;
    j=1;
    while (ans==1)
        if fr(1,j)==0
            frequency1(1,i)=fr(1,j-1);
            break;
        end
        j=j+1;
        if j>size(fr,2) && ans==1
            ans=0;
            frequency1(1,i)=fr(1,(size(fr,2)));
        end
    end
end
end

```

```

end
xlswrite('Frequency.xlsx', frequency1);
%Mode Shapes
ms=cell2mat(mod);
mss=zeros(n+1,n);
xlswrite('Mode Shapes',ms);
for i=1:size(ms,1)
    mss(i+1,:)=ms(i,:);
end
f=(0:1:n);
x=zeros(1,size(f,2));
for i=1:n
    figure(i);
    plot(mss(:,i),f,'ko-');hold on;
    plot(x,f,'k');
    xlabel('Displacement of Story');
    ylabel('Story number');
    grid on;
end

```

کد برنامه نویسی نرم افزار MATLAB برای حل معادلات مودال برای روش های عددی نیومارک و ویلسون و روش انتگرال دوهمل به صورت زیر می باشد:

- در جهت طولی:

```

% Project of Dynamic of Structures: Dr Mohsen Tehranizadeh
% Amirkabir University of Technology(Tehran Polytechnic Press)
% Written By : Vahid Shahmoradi-93124107
clear all;
clc;
m=xlsread('mass.xlsx');
k=xlsread('stiffness.xlsx');
%% Damping Matrix
a=[];
c=[];
freq=xlsread('Frequency.xlsx');
omega1=freq(1,1);
omega2=freq(1,2);
omega3=freq(1,3);
Q=[(1/omega1) omega1 omega1^3;(1/omega2) omega2
omega2^3;(1/omega3) omega3 omega3^3];
Q=inv(Q);
zeyta=[0.02;0.05;0.1];
a=2*Q*zeyta;
c=a(1,1)*m+a(2,1)*k;

%% Niyomark Analysis

```

```
%del=niyomark method paramrters for constant acceleration=0 ,
average
%constant acceleration=0.25 , linear acceleration=0.5
%alfa=niyomark method paramrters for constant acceleration=0 ,
average
%constant acceleration=0.5 , linear acceleration=1/6
%delt=Delta t,sca=Scale factor
%-----
%Kobe Earthquake
eart1=xlsread('kobe x.xlsx');
delt=eart1(2,1);
I=ones(size(m,1),1);
for i=1:size(eart1,1)
    force{1,i}=-1*m*I*(eart1(i,2)*9.81);
end
x{1,1}=zeros((3*size(m,1)),1);
%sca=input('Please enter Scale Factor');
del=0.5;alfa=0.25;
B=m+(del*c*delt)*(alfa*k*delt^2);
G=((1-del)*c*delt)+((0.5-alfa)*k*delt^2);
% calculate FN Matrix
FN{1,1}=eye(size(m,1))-(alfa*inv(B)*k*delt^2);
FN{1,2}=(eye(size(m,1))*delt)-(alfa*inv(B)*c*delt^2)-
(alfa*inv(B)*k*delt^3);
FN{1,3}=((0.5-alfa)*eye(size(m,1))*delt^2)-
(alfa*inv(B)*G*delt^2);
FN{2,1}=(-1*del*inv(B)*k*delt);
FN{2,2}=eye(size(m,1))-(del*inv(B)*c*delt)-
(del*inv(B)*k*delt^2);
FN{2,3}=((1-del)*eye(size(m,1))*delt)-(del*inv(B)*G*delt);
FN{3,1}=(-1*inv(B)*k);
FN{3,2}=(-1*inv(B)*c)-(inv(B)*k*delt);
FN{3,3}=(-1*inv(B)*G);
Fn=cell2mat(FN);
% calculate HN Matrix
HN{1,1}=alfa*inv(B)*delt^2;
HN{2,1}=del*inv(B)*delt;
HN{3,1}=inv(B);
Hn=cell2mat(HN);
x{1,1}=(Fn*x{1,1})+(Hn*force{1,1});
for i=2:size(eart1,1)
    x{1,i}=(Fn*x{1,i-1})+(Hn*force{1,i});
end
%-----
xxx=cell2mat(x);
maxx1=[];
for i=1:size(xxx,1)
    maxx1(i,1)=max(xxx(i,:));
end
```

```
%-----
%Loma Earthquake
eart2=xlsread('loma x.xlsx');
delt=eart2(2,1);
I=ones(size(m,1),1);
for i=1:size(eart2,1)
    forcel{1,i}=-1*m*I*(eart2(i,2)*9.81);
end
x1{1,1}=zeros((3*size(m,1)),1);
%sca=input('Please enter Scale Factor');
del=0.5;alfa=0.25;
B=m+(del*c*delt)*(alfa*k*delt^2);
G=((1-del)*c*delt)+((0.5-alfa)*k*delt^2);
% calculate FN Matrix
FN{1,1}=eye(size(m,1))-(alfa*inv(B)*k*delt^2);
FN{1,2}=(eye(size(m,1))*delt)-(alfa*inv(B)*c*delt^2)-(
    alfa*inv(B)*k*delt^3);
FN{1,3}=(0.5-alfa)*eye(size(m,1))*delt^2-(
    alfa*inv(B)*G*delt^2);
FN{2,1}=(-1*del*inv(B)*k*delt);
FN{2,2}=eye(size(m,1))-(del*inv(B)*c*delt)-(
    del*inv(B)*k*delt^2);
FN{2,3}=(1-del)*eye(size(m,1))*delt-(del*inv(B)*G*delt);
FN{3,1}=(-1*inv(B)*k);
FN{3,2}=(-1*inv(B)*c)-(inv(B)*k*delt);
FN{3,3}=(-1*inv(B)*G);
Fn=cell2mat(FN);
% calculate HN Matrix
HN{1,1}=alfa*inv(B)*(delt^2);
HN{2,1}=del*inv(B)*delt;
HN{3,1}=inv(B);
Hn=cell2mat(HN);
x1{1,1}=(Fn*x1{1,1})+(Hn*forcel{1,1});
for i=2:size(eart2,1)
    x1{1,i}=(Fn*x1{1,i-1})+(Hn*forcel{1,i});
end
%-----
xxx=cell2mat(x1);
maxx2=[];
for i=1:size(xxx,1)
    maxx2(i,1)=max(xxx(i,:));
end
%-----
%Tabas Earthquake
eart3=xlsread('tabas x.xlsx');
delt=eart3(2,1);
I=ones(size(m,1),1);
for i=1:size(eart3,1)
    force2{1,i}=-1*m*I*(eart3(i,2)*9.81);
```

```

end
x2{1,1}=zeros((3*size(m,1)),1);
%sca=input('Please enter Scale Factor');
del=0.5;alfa=0.25;
B=m+(del*c*delt)*(alfa*k*delt^2);
G=((1-del)*c*delt)+((0.5-alfa)*k*delt^2);
% calculate FN Matrix
FN{1,1}=eye(size(m,1))-(alfa*inv(B)*k*delt^2);
FN{1,2}=(eye(size(m,1))*delt)-(alfa*inv(B)*c*delt^2)-(
    alfa*inv(B)*k*delt^3);
FN{1,3}=((0.5-alfa)*eye(size(m,1))*delt^2)-(
    alfa*inv(B)*G*delt^2);
FN{2,1}=(-1*del*inv(B)*k*delt);
FN{2,2}=eye(size(m,1))-(del*inv(B)*c*delt)-(
    del*inv(B)*k*delt^2);
FN{2,3}=((1-del)*eye(size(m,1))*delt)-(del*inv(B)*G*delt);
FN{3,1}=(-1*inv(B)*k);
FN{3,2}=(-1*inv(B)*c)-(inv(B)*k*delt);
FN{3,3}=(-1*inv(B)*G);
Fn=cell2mat(FN);
% calculate HN Matrix
HN{1,1}=alfa*inv(B)*delt^2;
HN{2,1}=del*inv(B)*delt;
HN{3,1}=inv(B);
Hn=cell2mat(HN);
x2{1,1}=(Fn*x2{1,1})+(Hn*force2{1,1});
for i=2:size(eart3,1)
    x2{1,i}=(Fn*x2{1,i-1})+(Hn*force2{1,i});
end
%-----
xxx=cell2mat(x2);
maxx3=[];
for i=1:size(xxx,1)
    maxx3(i,1)=max(xxx(i,:));
end
%% Wilson Analysis
%del=niyomark method paramrters for constant acceleration=0 ,
average
%constant acceleration=0.25 , linear acceleration=0.5
%alfa=niyomark method paramrters for constant acceleration=0 ,
average
%constant acceleration=0.5 , linear acceleration=1/6
%delt=Delta t,sca=Scale factor
teta=1.35 ;
%-----
%Kobe Earthquake
eart1=xlsread('kobe x.xlsx');
delt=eart1(2,1);
I=ones(size(m,1),1);

```

```

for i=1:size(eart1,1)
    force{1,i}=-1*m*I*(eart1(i,2)*9.81);
end
x21{1,1}=zeros((3*size(m,1)),1);
del=0.5;alfa=(1/6);
B=m+(del*c*delt)*(alfa*k*delt^2);
BB=(inv(B)*G)-((1-teta)*inv(B)*m);
FN{1,1}=eye(size(m,1))-(alfa*inv(B)*k*(delt^2)*(teta^2));
FN{1,2}=(eye(size(m,1))*delt)-(alfa*inv(B)*c*(delt^2)*(teta^2))-
(alfa*inv(B)*k*(delt^3)*(teta^2));
FN{1,3}=alfa*teta*(delt^2)*(2*eye(size(m,1))-BB);
FN{2,1}=(-1*del*inv(B)*k*delt*teta);
FN{2,2}=eye(size(m,1))-(inv(B)*c*delt*teta)-
(del*inv(B)*k*(delt^2)*teta);
FN{2,3}=del*delt*(eye(size(m,1))-BB);
FN{3,1}=(-1*inv(B)*k);
FN{3,2}=(-1*inv(B)*c)-(inv(B)*k*delt);
FN{3,3}=(1-(1/teta))*(eye(size(m,1))-BB);
Fn=cell2mat(FN);
% calculate HN Matrix
HN{1,1}=alfa*inv(B)*(delt^2)*(teta^2);
HN{2,1}=del*inv(B)*delt*teta;
HN{3,1}=inv(B);
Hn=cell2mat(HN);
x21{1,1}=(Fn*x21{1,1})+(Hn*force{1,1});
for i=2:size(eart1,1)
    x21{1,i}=(Fn*x21{1,i-1})+(Hn*force{1,i});
end
%-----
xxx=cell2mat(x21);
maxx4=[];
for i=1:size(xxx,1)
    maxx4(i,1)=max(xxx(i,:));
end
%-----
%Loma Earthquake
eart2=xlsread('loma x.xlsx');
delt=eart2(2,1);
I=ones(size(m,1),1);
for i=1:size(eart2,1)
    forcel{1,i}=-1*m*I*(eart2(i,2)*9.81);
end
x22{1,1}=zeros((3*size(m,1)),1);
del=0.5;alfa=(1/6);
B=m+(del*c*delt)*(alfa*k*delt^2);
BB=(inv(B)*G)-((1-teta)*inv(B)*m);
FN{1,1}=eye(size(m,1))-(alfa*inv(B)*k*(delt^2)*(teta^2));
FN{1,2}=(eye(size(m,1))*delt)-(alfa*inv(B)*c*(delt^2)*(teta^2))-
(alfa*inv(B)*k*(delt^3)*(teta^2));

```

```

FN{1,3}=alfa*teta*(delt^2)*(2*eye(size(m,1))-BB);
FN{2,1}=(-1*del*inv(B)*k*delt*teta);
FN{2,2}=eye(size(m,1))-(inv(B)*c*delt*teta)-
(delt*inv(B)*k*(delt^2)*teta);
FN{2,3}=del*delt*(eye(size(m,1))-BB);
FN{3,1}=(-1*inv(B)*k);
FN{3,2}=(-1*inv(B)*c)-(inv(B)*k*delt);
FN{3,3}=(1-(1/teta))*(eye(size(m,1))-BB);
Fn=cell2mat(FN);
% calculate HN Matrix
HN{1,1}=alfa*inv(B)*(delt^2)*(teta^2);
HN{2,1}=del*inv(B)*delt*teta;
HN{3,1}=inv(B);
Hn=cell2mat(HN);
x22{1,1}=(Fn*x22{1,1})+(Hn*force1{1,1});
for i=2:size(eart2,1)
    x22{1,i}=(Fn*x22{1,i-1})+(Hn*force1{1,i});
end
%-----
xxx=cell2mat(x22);
maxx5=[];
for i=1:size(xxx,1)
    maxx5(i,1)=max(xxx(i,:));
end
%-----
%Tabas Earthquake
eart3=xlsread('tabas x.xlsx');
delt=eart3(2,1);
I=ones(size(m,1),1);
for i=1:size(eart3,1)
    force2{1,i}=-1*m*I*(eart3(i,2)*9.81);
end
x23{1,1}=zeros((3*size(m,1)),1);
del=0.5;alfa=(1/6);
B=m+(del*c*delt)*(alfa*k*delt^2);
BB=(inv(B)*G)-((1-teta)*inv(B)*m);
FN{1,1}=eye(size(m,1))-(alfa*inv(B)*k*(delt^2)*(teta^2));
FN{1,2}=(eye(size(m,1))*delt)-(alfa*inv(B)*c*(delt^2)*(teta^2))-
(alfa*inv(B)*k*(delt^3)*(teta^2));
FN{1,3}=alfa*teta*(delt^2)*(2*eye(size(m,1))-BB);
FN{2,1}=(-1*del*inv(B)*k*delt*teta);
FN{2,2}=eye(size(m,1))-(inv(B)*c*delt*teta)-
(delt*inv(B)*k*(delt^2)*teta);
FN{2,3}=del*delt*(eye(size(m,1))-BB);
FN{3,1}=(-1*inv(B)*k);
FN{3,2}=(-1*inv(B)*c)-(inv(B)*k*delt);
FN{3,3}=(1-(1/teta))*(eye(size(m,1))-BB);
Fn=cell2mat(FN);
% calculate HN Matrix

```

```

HN{1,1}=alfa*inv(B)*(delt^2)*(teta^2);
HN{2,1}=del*inv(B)*delt*teta;
HN{3,1}=inv(B);
Hn=cell2mat(HN);
x23{1,1}=(Fn*x23{1,1})+(Hn*force1{1,1});
for i=2:size(eart2,1)
    x23{1,i}=(Fn*x23{1,i-1})+(Hn*force1{1,i});
end
%-----
xxx=cell2mat(x23);
maxx6=[];
for i=1:size(xxx,1)
    maxx6(i,1)=max(xxx(i,:));
end

%% Dohamel Analysis
%md=Mode shape matrix;freq=frequency matrix
%dd=Mode shape matrix*dohamel
%alf=(alfa/frequency)
%Mm=Mody Mass Matrix
%-----
%KOBExEarthquake
md=xlsread('mode.xlsx');
dohamel2=[];
dohamel=[];
zeyta=[];
displ2=[];
alf=[];
eart1=xlsread('kobe x.xlsx');
% eart1=eart1*1.23;
t1=eart1(:,1);
accl=eart1(:,2);
delt=eart1(2,1);
for i=1:size(m,1)

zeyta(1,i)=(a(1,1)/(2*freq(1,i)))+(a(2,1)*freq(1,i)/2)+(a(3,1)*(
freq(1,i)^3));
end
for k=1:size(m,1)
%     dohamel2(1,i)=0;
    for j=1:(size(eart1,1)-1)
        ta=sym('ta');
        shib=(accl((j+1),1)-accl(j,1))/(t1((j+1),1)-t1(j,1));
        f=((shib*ta)-(shib*t1(j,1))+accl(j,1))*9.81*(exp(-
1*zeyta(1,k)*freq(1,k)*(t1((j+1),1)-
ta)))*(sin(freq(1,k)*(t1((j+1),1)-ta))));
        a=t1(j,1);
        b=t1((j+1),1);
        xmin=min(a,b);
    end
end

```



```

        xmax=max(a,b);
        a=xmin;
        b=xmax;
        n=0.000001;
        %-----
        n=2*floor(n/2);
if(n<2)
    n=2;
end

h=(b-a)/n;
S=0;
for i=1:n+1
    xx=a+(i-1)*h;
    ff=subs(f,xx);

    if i==1 || i==n+1
        S=S+ff;
    else
        if mod(i,2)==0
            S=S+4*ff;
        else
            S=S+2*ff;
        end
    end
end
S=S*h/3;
dd=S(1,1);
    disp(j);
    disp(k);
    dohamel(k,j)=dd;
end
end

% for i=1:size(m,1)
%     dohamel2(1,i)=0;
%     for j=1:(size(eart1,1)-1)
%         disp(j);
%         disp(i);
%
dohamel2(1,i)=dohamel2(1,i)+(int((acc1((j+1),1)*9.81*(exp(-
1*zeyta(1,i)*freq(1,i)*(t1((j+1),1)-
ta)))*(sin(freq(1,i)*(t1((j+1),1)-ta)))),t1(j,1),t1((j+1),1)))));
%     end
% end
for i=1:size(m,1)
    dohamel2(1,i)=0;
    for j=1:size(dohamel,2)
        dohamel2(1,i)=dohamel2(1,i)+abs(dohamel(i,j));
    end
end

```

```

        end
    end
    for i=1:size(m,1)

alf(1,i)=(md(:,i))'*m*ones(size(m,1),1))/(md(:,i))'*m*md(:,i))
;
        alf(1,i)=-1*alf(1,i)/freq(1,i);
    end
    dd=md;
    for i=1:size(m,1)
        dd(:,i)=md(:,i)*(alf(1,i)*dohamel2(1,i));
    end
    for i=1:size(m,1)
        displ2(1,i)=0;
        for j=1:size(m,2)
            displ2(1,i)=displ2(1,i)+dd(i,j);
        end
    end
end
%-----
%Loma x Earhtquake
md=xlsread('mode.xlsx');
dohamel2=[];
dohamel=[];
zeyta=[];
displ2=[];
alf=[];
earth1=xlsread('loma x.xlsx');
% earth1=earth1*1.23;
t1=earth1(:,1);
acc1=earth1(:,2);
delt=earth1(2,1);
for i=1:size(m,1)

zeyta(1,i)=(a(1,1)/(2*freq(1,i)))+(a(2,1)*freq(1,i)/2)+(a(3,1)*(
freq(1,i)^3));
end
for k=1:size(m,1)
%     dohamel2(1,i)=0;
    for j=1:(size(earth1,1)-1)
        ta=sym('ta');
        shib=(acc1((j+1),1)-acc1(j,1))/(t1((j+1),1)-t1(j,1));
        f=((shib*ta)-(shib*t1(j,1))+acc1(j,1))*9.81*(exp(-
1*zeyta(1,k)*freq(1,k)*(t1((j+1),1)-
ta)))*(sin(freq(1,k)*(t1((j+1),1)-ta))));
        a=t1(j,1);
        b=t1((j+1),1);
        xmin=min(a,b);
        xmax=max(a,b);
        a=xmin;
    end
end
end

```

```

        b=xmax;
        n=0.000001;
        %-----
        n=2*floor(n/2);
    if (n<2)
        n=2;
    end

    h=(b-a)/n;
    S=0;
    for i=1:n+1
        xx=a+(i-1)*h;
        ff=subs(f,xx);

        if i==1 || i==n+1
            S=S+ff;
        else
            if mod(i,2)==0
                S=S+4*ff;
            else
                S=S+2*ff;
            end
        end
    end
    S=S*h/3;
    dd=S(1,1);
    disp(j);
    disp(k);
    dohamel(k,j)=dd;
    end
end

% for i=1:size(m,1)
%     dohamel2(1,i)=0;
%     for j=1:(size(eart1,1)-1)
%         disp(j);
%         disp(i);
%     end
dohamel2(1,i)=dohamel2(1,i)+(int((acc1((j+1),1)*9.81*(exp(-
1*zeyta(1,i)*freq(1,i)*(t1((j+1),1)-
ta)))*(sin(freq(1,i)*(t1((j+1),1)-ta)))),t1(j,1),t1((j+1),1)))));
%     end
% end
for i=1:size(m,1)
    dohamel2(1,i)=0;
    for j=1:size(dohamel,2)
        dohamel2(1,i)=dohamel2(1,i)+abs(dohamel(i,j));
    end
end
end

```

```

for i=1:size(m,1)

alf(1,i)=(md(:,i))'*m*ones(size(m,1),1)/((md(:,i))'*m*md(:,i))
;
    alf(1,i)=-1*alf(1,i)/freq(1,i);
end
dd=md;
for i=1:size(m,1)
    dd(:,i)=md(:,i)*(alf(1,i)*dohamel2(1,i));
end
for i=1:size(m,1)
    displ2(1,i)=0;
    for j=1:size(m,2)
        displ2(1,i)=displ2(1,i)+dd(i,j);
    end
end
end
%-----
%Tabas y Earhtquake
md=xlsread('mode.xlsx');
dohamel2=[];
dohamel=[];
zeyta=[];
displ2=[];
alf=[];
eart1=xlsread('tabas x.xlsx');
% eart1=eart1*1.23;
t1=eart1(:,1);
acc1=eart1(:,2);
delt=eart1(2,1);
for i=1:size(m,1)

zeyta(1,i)=(a(1,1)/(2*freq(1,i)))+(a(2,1)*freq(1,i)/2)+(a(3,1)*(
freq(1,i)^3));
end
for k=1:size(m,1)
%     dohamel2(1,i)=0;
    for j=1:(size(eart1,1)-1)
        ta=sym('ta');
        shib=(acc1((j+1),1)-acc1(j,1))/(t1((j+1),1)-t1(j,1));
        f=((shib*ta)-(shib*t1(j,1))+acc1(j,1))*9.81*(exp(-
1*zeyta(1,k)*freq(1,k)*(t1((j+1),1)-
ta)))*(sin(freq(1,k)*(t1((j+1),1)-ta))));
        a=t1(j,1);
        b=t1((j+1),1);
        xmin=min(a,b);
        xmax=max(a,b);
        a=xmin;
        b=xmax;
        n=0.000001;
    end
end

```

```

%-----
n=2*floor(n/2);
if(n<2)
    n=2;
end

h=(b-a)/n;
S=0;
for i=1:n+1
    xx=a+(i-1)*h;
    ff=subs(f,xx);

    if i==1 || i==n+1
        S=S+ff;
    else
        if mod(i,2)==0
            S=S+4*ff;
        else
            S=S+2*ff;
        end
    end
end
S=S*h/3;
dd=S(1,1);
disp(j);
disp(k);
dohamel(k,j)=dd;
end
end

% for i=1:size(m,1)
%     dohamel2(1,i)=0;
%     for j=1:(size(eart1,1)-1)
%         disp(j);
%         disp(i);
%     end
dohamel2(1,i)=dohamel2(1,i)+(int((acc1((j+1),1)*9.81*(exp(-
1*zeyta(1,i)*freq(1,i)*(t1((j+1),1)-
ta)))*(sin(freq(1,i)*(t1((j+1),1)-ta))))),t1(j,1),t1((j+1),1)));
%     end
% end
for i=1:size(m,1)
    dohamel2(1,i)=0;
    for j=1:size(dohamel,2)
        dohamel2(1,i)=dohamel2(1,i)+abs(dohamel(i,j));
    end
end
for i=1:size(m,1)

```

```
alf(1,i)=(md(:,i))'*m*ones(size(m,1),1)/((md(:,i))'*m*md(:,i))
;
alf(1,i)=-1*alf(1,i)/freq(1,i);
end
dd=md;
for i=1:size(m,1)
dd(:,i)=md(:,i)*(alf(1,i)*dohamel2(1,i));
end
for i=1:size(m,1)
displ2(1,i)=0;
for j=1:size(m,2)
displ2(1,i)=displ2(1,i)+dd(i,j);
end
end
```

• در جهت عرضی:

```
% Project of Dynamic of Structures: Dr Mohsen Tehranizadeh
% Amirkabir University of Technology(Tehran Polytechnic Press)
% Written By : Vahid Shahmoradi-93124107
clear all;
clc;
m=xlsread('mass.xlsx');
k=xlsread('stiffness.xlsx');
%% Damping Matrix
a=[];
c=[];
freq=xlsread('Frequency.xlsx');
omega1=freq(1,1);
omega2=freq(1,2);
omega3=freq(1,3);
Q=[(1/omega1) omega1 omega1^3;(1/omega2) omega2
omega2^3;(1/omega3) omega3 omega3^3];
Q=inv(Q);
zeyta=[0.02;0.05;0.1];
a=2*Q*zeyta;
c=a(1,1)*m+a(2,1)*k;

%% Niyomark Analysis
%del=niyomark method paramrters for constant acceleration=0 ,
average
%constant acceleration=0.25 , linear acceleration=0.5
%alfa=niyomark method paramrters for constant acceleration=0 ,
average
%constant acceleration=0.5 , linear acceleration=1/6
%delt=Delta t,sca=Scale factor
```

```
%-----
%Kobe Earthquake
eart1=xlsread('kobe y.xlsx');
delt=eart1(2,1);
I=ones(size(m,1),1);
for i=1:size(eart1,1)
    force{1,i}=-1*m*I*(eart1(i,2)*9.81);
end
x{1,1}=zeros((3*size(m,1)),1);
%sca=input('Please enter Scale Factor');
del=0.5;alfa=0.25;
B=m+(del*c*delt)*(alfa*k*delt^2);
G=((1-del)*c*delt)+((0.5-alfa)*k*delt^2);
% calculate FN Matrix
FN{1,1}=eye(size(m,1))-(alfa*inv(B)*k*delt^2);
FN{1,2}=(eye(size(m,1))*delt)-(alfa*inv(B)*c*delt^2)-(
    alfa*inv(B)*k*delt^3);
FN{1,3}=((0.5-alfa)*eye(size(m,1))*delt^2)-
    (alfa*inv(B)*G*delt^2);
FN{2,1}=(-1*del*inv(B)*k*delt);
FN{2,2}=eye(size(m,1))-(del*inv(B)*c*delt)-
    (del*inv(B)*k*delt^2);
FN{2,3}=((1-del)*eye(size(m,1))*delt)-(del*inv(B)*G*delt);
FN{3,1}=(-1*inv(B)*k);
FN{3,2}=(-1*inv(B)*c)-(inv(B)*k*delt);
FN{3,3}=(-1*inv(B)*G);
Fn=cell2mat(FN);
% calculate HN Matrix
HN{1,1}=alfa*inv(B)*delt^2;
HN{2,1}=del*inv(B)*delt;
HN{3,1}=inv(B);
Hn=cell2mat(HN);
x{1,1}=(Fn*x{1,1})+(Hn*force{1,1});
for i=2:size(eart1,1)
    x{1,i}=(Fn*x{1,i-1})+(Hn*force{1,i});
end
%-----
xxx=cell2mat(x);
maxx1=[];
for i=1:size(xxx,1)
    maxx1(i,1)=max(xxx(i,:));
end
%-----
%Loma Earthquake
eart2=xlsread('loma y.xlsx');
delt=eart2(2,1);
I=ones(size(m,1),1);
for i=1:size(eart2,1)
    force1{1,i}=-1*m*I*(eart2(i,2)*9.81);
```

```

end
x1{1,1}=zeros((3*size(m,1)),1);
%sca=input('Please enter Scale Factor');
del=0.5;alfa=0.25;
B=m+(del*c*delt)*(alfa*k*delt^2);
G=((1-del)*c*delt)+((0.5-alfa)*k*delt^2);
% calculate FN Matrix
FN{1,1}=eye(size(m,1))-(alfa*inv(B)*k*delt^2);
FN{1,2}=(eye(size(m,1))*delt)-(alfa*inv(B)*c*delt^2)-(
    alfa*inv(B)*k*delt^3);
FN{1,3}=((0.5-alfa)*eye(size(m,1))*delt^2)-
    (alfa*inv(B)*G*delt^2);
FN{2,1}=(-1*del*inv(B)*k*delt);
FN{2,2}=eye(size(m,1))-(del*inv(B)*c*delt)-
    (del*inv(B)*k*delt^2);
FN{2,3}=((1-del)*eye(size(m,1))*delt)-(del*inv(B)*G*delt);
FN{3,1}=(-1*inv(B)*k);
FN{3,2}=(-1*inv(B)*c)-(inv(B)*k*delt);
FN{3,3}=(-1*inv(B)*G);
Fn=cell2mat(FN);
% calculate HN Matrix
HN{1,1}=alfa*inv(B)*(delt^2);
HN{2,1}=del*inv(B)*delt;
HN{3,1}=inv(B);
Hn=cell2mat(HN);
x1{1,1}=(Fn*x1{1,1})+(Hn*forcel{1,1});
for i=2:size(eart2,1)
    x1{1,i}=(Fn*x1{1,i-1})+(Hn*forcel{1,i});
end
%-----
xxx=cell2mat(x1);
maxx2=[];
for i=1:size(xxx,1)
    maxx2(i,1)=max(xxx(i,:));
end
%-----
%Tabas Earthquake
eart3=xlsread('tabas y.xlsx');
delt=eart3(2,1);
I=ones(size(m,1),1);
for i=1:size(eart3,1)
    force2{1,i}=-1*m*I*(eart3(i,2)*9.81);
end
x2{1,1}=zeros((3*size(m,1)),1);
%sca=input('Please enter Scale Factor');
del=0.5;alfa=0.25;
B=m+(del*c*delt)*(alfa*k*delt^2);
G=((1-del)*c*delt)+((0.5-alfa)*k*delt^2);
% calculate FN Matrix

```



```

FN{1,1}=eye(size(m,1))-(alfa*inv(B)*k*delt^2);
FN{1,2}=(eye(size(m,1))*delt)-(alfa*inv(B)*c*delt^2)-(
    alfa*inv(B)*k*delt^3);
FN{1,3}=( (0.5-alfa)*eye(size(m,1))*delt^2)-(
    alfa*inv(B)*G*delt^2);
FN{2,1}=(-1*del*inv(B)*k*delt);
FN{2,2}=eye(size(m,1))-(del*inv(B)*c*delt)-(
    del*inv(B)*k*delt^2);
FN{2,3}=( (1-del)*eye(size(m,1))*delt)-(del*inv(B)*G*delt);
FN{3,1}=(-1*inv(B)*k);
FN{3,2}=(-1*inv(B)*c)-(inv(B)*k*delt);
FN{3,3}=(-1*inv(B)*G);
Fn=cell2mat(FN);
% calculate HN Matrix
HN{1,1}=alfa*inv(B)*delt^2;
HN{2,1}=del*inv(B)*delt;
HN{3,1}=inv(B);
Hn=cell2mat(HN);
x2{1,1}=(Fn*x2{1,1})+(Hn*force2{1,1});
for i=2:size(eart3,1)
    x2{1,i}=(Fn*x2{1,i-1})+(Hn*force2{1,i});
end
%-----
xxx=cell2mat(x2);
maxx3=[];
for i=1:size(xxx,1)
    maxx3(i,1)=max(xxx(i,:));
end
%% Wilson Analysis
%del=niyomark method paramrters for constant acceleration=0 ,
average
%constant acceleration=0.25 , linear acceleration=0.5
%alfa=niyomark method paramrters for constant acceleration=0 ,
average
%constant acceleration=0.5 , linear acceleration=1/6
%delt=Delta t,sca=Scale factor
teta=1.35 ;
%-----
%Kobe Earthquake
eart1=xlsread('kobe y.xlsx');
delt=eart1(2,1);
I=ones(size(m,1),1);
for i=1:size(eart1,1)
    force{1,i}=-1*m*I*(eart1(i,2)*9.81);
end
x21{1,1}=zeros((3*size(m,1)),1);
del=0.5;alfa=(1/6);
B=m+(del*c*delt)*(alfa*k*delt^2);
BB=(inv(B)*G)-((1-teta)*inv(B)*m);

```

```

FN{1,1}=eye(size(m,1))-(alfa*inv(B)*k*(delt^2)*(teta^2));
FN{1,2}=(eye(size(m,1))*delt)-(alfa*inv(B)*c*(delt^2)*(teta^2))-(
(alfa*inv(B)*k*(delt^3)*(teta^2));
FN{1,3}=alfa*teta*(delt^2)*(2*eye(size(m,1))-BB);
FN{2,1}=(-1*del*inv(B)*k*delt*teta);
FN{2,2}=eye(size(m,1))-(inv(B)*c*delt*teta)-(
(del*inv(B)*k*(delt^2)*teta);
FN{2,3}=del*delt*(eye(size(m,1))-BB);
FN{3,1}=(-1*inv(B)*k);
FN{3,2}=(-1*inv(B)*c)-(inv(B)*k*delt);
FN{3,3}=(1-(1/teta))*(eye(size(m,1))-BB);
Fn=cell2mat(FN);
% calculate HN Matrix
HN{1,1}=alfa*inv(B)*(delt^2)*(teta^2);
HN{2,1}=del*inv(B)*delt*teta;
HN{3,1}=inv(B);
Hn=cell2mat(HN);
x21{1,1}=(Fn*x21{1,1})+(Hn*force{1,1});
for i=2:size(eart1,1)
    x21{1,i}=(Fn*x21{1,i-1})+(Hn*force{1,i});
end
%-----
xxx=cell2mat(x21);
maxx4=[];
for i=1:size(xxx,1)
    maxx4(i,1)=max(xxx(i,:));
end
%-----
%Loma Earthquake
eart2=xlsread('loma y.xlsx');
delt=eart2(2,1);
I=ones(size(m,1),1);
for i=1:size(eart2,1)
    forcel{1,i}=-1*m*I*(eart2(i,2)*9.81);
end
x22{1,1}=zeros((3*size(m,1)),1);
del=0.5;alfa=(1/6);
B=m+(del*c*delt)*(alfa*k*delt^2);
BB=(inv(B)*G)-((1-teta)*inv(B)*m);
FN{1,1}=eye(size(m,1))-(alfa*inv(B)*k*(delt^2)*(teta^2));
FN{1,2}=(eye(size(m,1))*delt)-(alfa*inv(B)*c*(delt^2)*(teta^2))-(
(alfa*inv(B)*k*(delt^3)*(teta^2));
FN{1,3}=alfa*teta*(delt^2)*(2*eye(size(m,1))-BB);
FN{2,1}=(-1*del*inv(B)*k*delt*teta);
FN{2,2}=eye(size(m,1))-(inv(B)*c*delt*teta)-(
(del*inv(B)*k*(delt^2)*teta);
FN{2,3}=del*delt*(eye(size(m,1))-BB);
FN{3,1}=(-1*inv(B)*k);
FN{3,2}=(-1*inv(B)*c)-(inv(B)*k*delt);

```

```

FN{3,3}=(1-(1/teta))*(eye(size(m,1))-BB);
Fn=cell2mat(FN);
% calculate HN Matrix
HN{1,1}=alfa*inv(B)*(delt^2)*(teta^2);
HN{2,1}=del*inv(B)*delt*teta;
HN{3,1}=inv(B);
Hn=cell2mat(HN);
x22{1,1}=(Fn*x22{1,1})+(Hn*force1{1,1});
for i=2:size(eart2,1)
    x22{1,i}=(Fn*x22{1,i-1})+(Hn*force1{1,i});
end
%-----
xxx=cell2mat(x22);
maxx5=[];
for i=1:size(xxx,1)
    maxx5(i,1)=max(xxx(i,:));
end
%-----
%Tabas Earthquake
eart3=xlsread('tabas y.xlsx');
delt=eart3(2,1);
I=ones(size(m,1),1);
for i=1:size(eart3,1)
    force2{1,i}=-1*m*I*(eart3(i,2)*9.81);
end
x23{1,1}=zeros((3*size(m,1)),1);
del=0.5;alfa=(1/6);
B=m+(del*c*delt)*(alfa*k*delt^2);
BB=(inv(B)*G)-((1-teta)*inv(B)*m);
FN{1,1}=eye(size(m,1))-(alfa*inv(B)*k*(delt^2)*(teta^2));
FN{1,2}=(eye(size(m,1))*delt)-(alfa*inv(B)*c*(delt^2)*(teta^2))-
(alfa*inv(B)*k*(delt^3)*(teta^2));
FN{1,3}=alfa*teta*(delt^2)*(2*eye(size(m,1))-BB);
FN{2,1}=(-1*del*inv(B)*k*delt*teta);
FN{2,2}=eye(size(m,1))-(inv(B)*c*delt*teta)-
(del*inv(B)*k*(delt^2)*teta);
FN{2,3}=del*delt*(eye(size(m,1))-BB);
FN{3,1}=(-1*inv(B)*k);
FN{3,2}=(-1*inv(B)*c)-(inv(B)*k*delt);
FN{3,3}=(1-(1/teta))*(eye(size(m,1))-BB);
Fn=cell2mat(FN);
% calculate HN Matrix
HN{1,1}=alfa*inv(B)*(delt^2)*(teta^2);
HN{2,1}=del*inv(B)*delt*teta;
HN{3,1}=inv(B);
Hn=cell2mat(HN);
x23{1,1}=(Fn*x23{1,1})+(Hn*force1{1,1});
for i=2:size(eart2,1)
    x23{1,i}=(Fn*x23{1,i-1})+(Hn*force1{1,i});

```

```

end
%-----
xxx=cell2mat(x23);
maxx6=[];
for i=1:size(xxx,1)
    maxx6(i,1)=max(xxx(i,:));
end

%% Dohamel Analysis
%md=Mode shape matrix;freq=frequency matrix
%dd=Mode shape matrix*dohamel
%alf=(alfa/frequency)
%Mm=Mody Mass Matrix
%-----
%Kobe y Earhtquake
md=xlsread('mode.xlsx');
dohamel2=[];
dohamel=[];
zeyta=[];
displ2=[];
alf=[];
earth1=xlsread('kobe y.xlsx');
% earth1=earth1*1.23;
t1=earth1(:,1);
acc1=earth1(:,2);
delt=earth1(2,1);
for i=1:size(m,1)

zeyta(1,i)=(a(1,1)/(2*freq(1,i)))+(a(2,1)*freq(1,i)/2)+(a(3,1)*(
freq(1,i)^3));
end
for k=1:size(m,1)
%     dohamel2(1,i)=0;
    for j=1:(size(earth1,1)-1)
        ta=sym('ta');
        shib=(acc1((j+1),1)-acc1(j,1))/(t1((j+1),1)-t1(j,1));
        f=((shib*ta)-(shib*t1(j,1))+acc1(j,1))*9.81*(exp(-
1*zeyta(1,k)*freq(1,k)*(t1((j+1),1)-
ta)))*(sin(freq(1,k)*(t1((j+1),1)-ta))));
        a=t1(j,1);
        b=t1((j+1),1);
        xmin=min(a,b);
        xmax=max(a,b);
        a=xmin;
        b=xmax;
        n=0.000001;
        %-----
        n=2*floor(n/2);
    end
end
if (n<2)

```

```

        n=2;
end

h=(b-a)/n;
S=0;
for i=1:n+1
    xx=a+(i-1)*h;
    ff=subs(f,xx);

    if i==1 || i==n+1
        S=S+ff;
    else
        if mod(i,2)==0
            S=S+4*ff;
        else
            S=S+2*ff;
        end
    end
end
S=S*h/3;
dd=S(1,1);
disp(j);
disp(k);
dohamel(k,j)=dd;
end
end
for i=1:size(m,1)
    dohamel2(1,i)=0;
    for j=1:size(dohamel,2)
        dohamel2(1,i)=dohamel2(1,i)+abs(dohamel(i,j));
    end
end
for i=1:size(m,1)

alf(1,i)=(md(:,i))'*m*ones(size(m,1),1)/((md(:,i))'*m*md(:,i))
;
    alf(1,i)=-1*alf(1,i)/freq(1,i);
end
dd=md;
for i=1:size(m,1)
    dd(:,i)=md(:,i)*(alf(1,i)*dohamel2(1,i));
end
for i=1:size(m,1)
    displ2(1,i)=0;
    for j=1:size(m,2)
        displ2(1,i)=displ2(1,i)+dd(i,j);
    end
end
end
%-----

```

```
%Loma y Earhtquake
md=xlsread('mode.xlsx');
dohamel2=[];
dohamel=[];
zeyta=[];
displ2=[];
alf=[];
earth1=xlsread('loma y.xlsx');
% earth1=earth1*1.23;
t1=earth1(:,1);
acc1=earth1(:,2);
delt=earth1(2,1);
for i=1:size(m,1)

zeyta(1,i)=(a(1,1)/(2*freq(1,i)))+(a(2,1)*freq(1,i)/2)+(a(3,1)*(
freq(1,i)^3));
end
for k=1:size(m,1)
%     dohamel2(1,i)=0;
    for j=1:(size(earth1,1)-1)
        ta=sym('ta');
        shib=(acc1((j+1),1)-acc1(j,1))/(t1((j+1),1)-t1(j,1));
        f=((shib*ta)-(shib*t1(j,1))+acc1(j,1))*9.81*(exp(-
1*zeyta(1,k)*freq(1,k)*(t1((j+1),1)-
ta)))*(sin(freq(1,k)*(t1((j+1),1)-ta))));
        a=t1(j,1);
        b=t1((j+1),1);
        xmin=min(a,b);
        xmax=max(a,b);
        a=xmin;
        b=xmax;
        n=0.000001;
        %-----
        n=2*floor(n/2);
    if (n<2)
        n=2;
    end

h=(b-a)/n;
S=0;
for i=1:n+1
    xx=a+(i-1)*h;
    ff=subs(f,xx);

    if i==1 || i==n+1
        S=S+ff;
    else
        if mod(i,2)==0
            S=S+4*ff;

```

```

        else
            S=S+2*ff;
        end
    end
end
S=S*h/3;
dd=S(1,1);
    disp(j);
    disp(k);
    dohamel(k,j)=dd;
end
end
for i=1:size(m,1)
    dohamel2(1,i)=0;
    for j=1:size(dohamel,2)
        dohamel2(1,i)=dohamel2(1,i)+abs(dohamel(i,j));
    end
end
for i=1:size(m,1)

alf(1,i)=(md(:,i))'*m*ones(size(m,1),1)/((md(:,i))'*m*md(:,i))
;
    alf(1,i)=-1*alf(1,i)/freq(1,i);
end
dd=md;
for i=1:size(m,1)
    dd(:,i)=md(:,i)*(alf(1,i)*dohamel2(1,i));
end
for i=1:size(m,1)
    displ2(1,i)=0;
    for j=1:size(m,2)
        displ2(1,i)=displ2(1,i)+dd(i,j);
    end
end
end
%-----
%Tabas y Earhtquake
md=xlsread('mode.xlsx');
dohamel2=[];
dohamel=[];
zeyta=[];
displ2=[];
alf=[];
earth1=xlsread('tabas y.xlsx');
% earth1=earth1*1.23;
t1=earth1(:,1);
accl=earth1(:,2);
delt=earth1(2,1);
for i=1:size(m,1)

```

```

zeyta(1,i)=(a(1,1)/(2*freq(1,i)))+(a(2,1)*freq(1,i)/2)+(a(3,1)*(
freq(1,i)^3));
end
for k=1:size(m,1)
%     dohamel2(1,i)=0;
    for j=1:(size(eart1,1)-1)
        ta=sym('ta');
        shib=(acc1((j+1),1)-acc1(j,1))/(t1((j+1),1)-t1(j,1));
        f=((shib*ta)-(shib*t1(j,1))+acc1(j,1))*9.81*(exp(-
1*zeyta(1,k)*freq(1,k)*(t1((j+1),1)-
ta)))*(sin(freq(1,k)*(t1((j+1),1)-ta))));
        a=t1(j,1);
        b=t1((j+1),1);
        xmin=min(a,b);
        xmax=max(a,b);
        a=xmin;
        b=xmax;
        n=0.000001;
        %-----
        n=2*floor(n/2);
    if(n<2)
        n=2;
    end

    h=(b-a)/n;
    S=0;
    for i=1:n+1
        xx=a+(i-1)*h;
        ff=subs(f,xx);

        if i==1 || i==n+1
            S=S+ff;
        else
            if mod(i,2)==0
                S=S+4*ff;
            else
                S=S+2*ff;
            end
        end
    end
    S=S*h/3;
    dd=S(1,1);
    disp(j);
    disp(k);
    dohamel(k,j)=dd;
end
end
for i=1:size(m,1)

```



```

    dohamel2(1,i)=0;
    for j=1:size(dohamel,2)
        dohamel2(1,i)=dohamel2(1,i)+abs(dohamel(i,j));
    end
end
for i=1:size(m,1)

alf(1,i)=(md(:,i))'*m*ones(size(m,1),1)/(md(:,i))'*m*md(:,i))
;
    alf(1,i)=-1*alf(1,i)/freq(1,i);
end
dd=md;
for i=1:size(m,1)
    dd(:,i)=md(:,i)*(alf(1,i)*dohamel2(1,i));
end
for i=1:size(m,1)
    displ2(1,i)=0;
    for j=1:size(m,2)
        displ2(1,i)=displ2(1,i)+dd(i,j);
    end
end
end

```

منابع:

1. دینامیک سازه‌ها و کاربرد آن در مهندسی زلزله، دکتر محسن تهرانی‌زاده، انتشارات دانشگاه صنعتی امیرکبیر
2. تحلیل و طراحی ساختمان‌های فولادی بر مبنای مقررات ملی ساختمان با برنامه EATBS، مهدی هادیزاده-علی سیفی، مرکز آموزش مهندسين خانه عمران شریف، چاپ پانزدهم، پاییز 1392
3. محاسبات پروژه‌های ساختمانی با استفاده از ETABS و SAFE، حسن باجی، نشر علم عمران
4. آیین نامه طراحی ساختمان‌ها در برابر زلزله-استاندارد 2800، ویرایش سوم، موسسه استاندارد و تحقیقات صنعتی