Санкт-Петербургский государственный политехнический университет

 Петра Великого

Институт прикладной математики и механики

**Кафедра «Теоретическая механика»**

**КУРСОВАЯ РАБОТА**

**Исследование и визуализация свободных колебаний системы с двумя степенями свободы**

по дисциплине «Языки программирования»

Выполнил

студент гр. 23632/2 В.Д.Тур

Руководитель

Ассистент А.Ю.Панченко

«\_\_\_» \_\_\_\_\_\_\_\_\_\_ 2018г.

Санкт-Петербург

2018

**СОДЕРЖАНИЕ**

Постановка задачи……………………………………………………………………. 3 Введение……………………………………………………………………………… 4

 1. Исследование свободных колебаний………………………………………… 6

 1.1. Аналитическое решение задачи……………………………………………….. 6

 1.2. Визуализация…………………………………………………………………… 9

 1.3. Код……………………………………………………………………………….. 11

Заключение……………………………………………………………………………. 14

Список использованных источников………………………………………………… 15

**ПОСТАНОВКА ЗАДАЧИ**

Определить частоты малых свободных колебаний и формы главных колебаний системы с двумя степенями свободы, пренебрегая силами сопротивления, массами пружин и моментами инерции скручиваемых валов [1].

**Дано:**



Также даны массы $m\_{1}$ и $m\_{2}$ двух дисков и жесткости $с\_{1}$,$ с\_{2}$,$ с\_{3}$ трех пружин.

**ВВЕДЕНИЕ**

**Виды и характеристики колебаний**

**Колебания** – это периодическое движение тела, при котором оно многократно движется по одной и той же траектории или проходит одни и те же точки пространства. Примерами колеблющихся объектов могут служить самые разнообразные вещи – маятник часов, струна скрипки, рессоры автомобиля. По сути, большинство механических систем можно рассматривать как объект изучения для теории колебаний.

Колебания, впрочем, играют важную роль в рассмотрении многих физических явлений и за пределами механики. Например, сила тока и напряжение в электрических контурах могут колебаться. Биологическими примерами колебаний могут служить сердечные сокращения, артериальный пульс и производство звука голосовыми связками.

Хотя физическая природа колеблющихся систем может существенно отличаться, разнообразные типы колебаний могут быть охарактеризованы количественно и качественно сходным образом. Основные характеристики колебаний это:

* **Смещение** ***x*** от положения равновесия(координата) – физическая величина, которая изменяется со временем при колебательном движении.
* **Амплитуда** ***А*** – это максимальное смещение колеблющегося объекта от положения равновесия.
* **Период колебания** ***T*** – время, необходимое для осуществления одного полного цикла. Полное колебание, или цикл – это движение, при котором тело, выведенное из положения равновесия, возвращается в это положение, отклоняется до максимального смещения в противоположную сторону и возвращается в свое первоначальное положение.
* **Частота *k***– число колебаний за единицу времени.

Различают несколько видов колебаний, зависящих от учитываемых свойств колеблющихся систем:

* **Собственные** — это колебания в системе под действием внутренних сил после того, как система выведена из состояния равновесия. В данном контексте система— это группа тел, движение которых мы изучаем, а внутренние силы — силы, действующие между телами системы. В этом виде колебаний внешние силы не учитываются в силу их малости. Частоту (частоты) колебаний системы, в отсутствии внешних воздействий и при наличии начальной энергии в системе, называют собственной.
* **Вынужденные** — колебания, протекающие в системе под влиянием внешнего периодического воздействия. При вынужденных колебаниях может возникнуть явление резонанса: резкое и неограниченное (в теории, без учета трения) возрастание амплитуды колебаний при совпадении собственной частоты и частоты вынуждающей силы.
* **Параметрические** — колебания, возникающие при изменении какого-либо параметра (инерционного, диссипативного или квазиупругого коэффициента) колебательной системы в результате внешнего воздействия или с течением времени.
* **Автоколебания** — саморегулирующиеся колебания, при которых система сама имеет запас потенциальной энергии и расходует его на совершение колебаний. Характерным отличием автоколебаний от вынужденных колебаний является то, что их амплитуда определяется свойствами самой системы, а не начальными условиями.
* **Случайные** — колебания, при которых внешняя или параметрическая нагрузка является случайной величиной.

**Свободные колебания**

В данной работе нас прежде всего интересуют свободные колебания.

Условия возникновения свободных колебаний:

1. При выведении тела из положения равновесия в системе возникает сила, направленная к положению равновесия и, следовательно, стремящаяся возвратить тело в положение равновесия.
2. Трение в системе достаточно мало, иначе колебания быстро затухнут или вовсе не возникнут. Незатухающие колебания возможны лишь при отсутствии трения.

**ИССЛЕДОВАНИЕ СВОБОДНЫХ КОЛЕБАНИЙ**

**Аналитическое решение задачи**

Рассмотрим свободные колебания механической системы, имеющей две степени свободы, положение которой определяется двумя обобщенными координатами $x\_{1}$и $x\_{2}$, отсчитываемыми от положения устойчивого равновесия системы.



Полагая все связи системы стационарными и голономными, а силы потенциальными, вычисляем кинетическую энергию системы.

Кинетическая энергия дисков состоит из кинетической энергии поступательного и вращательного движения.

Для первого диска:$T\_{1}=\frac{1}{2}\frac{m\_{1}R\_{1}^{2}}{2}\dot{φ}\_{1}^{2}+\frac{1}{2}m\_{1}\dot{x}\_{1}^{2}$, при этом $\dot{φ}=\frac{\dot{x\_{1}}}{R\_{1}}$, исходя из отсутствия проскальзывания диска по поверхности. Поэтому $T\_{1}=\frac{3}{4}m\_{1}\dot{x}\_{1}^{2}$. Для второго диска аналогично получается $T\_{2}=\frac{3}{4}m\_{2}\dot{x}\_{2}^{2}$.

Кинетическая энергия системы:

$$T=\frac{3}{4}(m\_{1}\dot{x}\_{1}^{2}+m\_{2}\dot{x}\_{2}^{2})$$

Потенциальная энергия пружин в линейном случае без диссипации выглядит как $П\_{i}=\frac{1}{2}c\_{i}Δl\_{i}^{2}$, где $Δl\_{i}$– удлинение пружины от положения равновесия.

Потенциальная энергия системы:

$$П=\frac{1}{2}(c\_{1}x\_{1}^{2}+c\_{2}(x\_{1}+x\_{2})^{2}+c\_{3}x\_{2}^{2})$$

Для каждой из обобщенной координат можно составить уравнение Лагранжа вида:$\frac{d}{dt}(\frac{∂T}{∂\dot{x}\_{i}})-\frac{∂T}{∂x\_{i}}=-\frac{∂П}{∂x\_{i}}$

Беря частные производные и производную по времени, получаем систему уравнений Лагранжа:

$$\begin{array}{c}\frac{3}{2}m\_{1}\ddot{x}\_{1}+(c\_{1}+c\_{2})x\_{1}+c\_{2}x\_{2}=0\\\frac{3}{2}m\_{2}\ddot{x}\_{2}+(c\_{2}+c\_{3})x\_{2}+c\_{2}x\_{1}=0\end{array}$$

Будет искать решение этой однородной системы дифференциальных уравнений второго порядка в виде

$$\begin{array}{c}x\_{1}=A\_{1}sin(kt+α\_{1})\\x\_{2}=A\_{2}sin(kt+α\_{2})\end{array}$$

Подставляя $x\_{1}$и $x\_{2}$в систему дифференциальных уравнений, получаем однородную систему уже алгебраических уравнений относительно амплитуд.

$$\begin{array}{c}-\frac{3}{2}m\_{1}k^{2}A\_{1}+(c\_{1}+c\_{2})A\_{1}+c\_{2}A\_{2}=0\\-\frac{3}{2}m\_{2}k^{2}A\_{2}+(c\_{1}+c\_{2})A\_{2}+c\_{2}A\_{1}=0\end{array}$$

Для существования нетривиального решения этой системы ее определитель $Δ(k^{2})$должен быть нулевым:

$$\left∣\begin{matrix}-\frac{3}{2}m\_{1}k^{2}+(c\_{1}+c\_{2})&c\_{2}\\c\_{2}&-\frac{3}{2}m\_{2}k^{2}+(c\_{2}+c\_{3})\end{matrix}\right∣=0$$

$$\frac{9}{4}m\_{1}m\_{2}k^{4}-\frac{3}{2}(c\_{1}+c\_{2})m\_{2}k^{2}-\frac{3}{2}(c\_{2}+c\_{3})m\_{1}k^{2}+c\_{1}c\_{2}+c\_{2}c\_{3}+c\_{3}c\_{1}=0$$

Это выражение является квадратным уравнением относительно $k^{2}$. Решив его, находим собственные частоты системы. Отношения амплитуд могут быть найдены из однородной системы уравнений:

$$\begin{array}{c}γ\_{1}=\frac{A\_{2}^{(1)}}{A\_{1}^{(1)}}=\frac{\frac{3}{2}m\_{1}k\_{1}^{2}-c\_{1}-c\_{2}}{c\_{2}}\\γ\_{1}=\frac{A\_{2}^{(1)}}{A\_{1}^{(1)}}=\frac{\frac{3}{2}m\_{1}k\_{2}^{2}-c\_{1}-c\_{2}}{c\_{2}}\end{array}$$

Тогда общее решение однородного дифференциального уравнения:

$$\begin{array}{c}x\_{1}=A\_{1}^{(1)}sin(k\_{1}t+α\_{1})+A\_{1}^{(2)}sin(k\_{2}t+α\_{2})\\x\_{2}=γ\_{1}A\_{1}^{(1)}sin(k\_{1}t+α\_{1})+γ\_{2}A\_{1}^{(2)}sin(k\_{2}t+α\_{2})\end{array}$$

Эта зависимость координат от времени по сути и является выражением для главных колебаний системы.

Теперь можно построить графики для обобщенных координат или амплитуд или визуализировать данное решение с помощью какого-нибудь языка программирования и соответствующих библиотек.

**Визуализация**

Визуализировать данную задачу будем с помощью библиотеки Three.js.

**Three.js** — легковесная кроссбраузерная библиотека JavaScript, используемая для создания и отображения анимированной компьютерной 3D графики при разработке веб-приложений. Three.js скрипты могут использоваться совместно с элементами HTML5 и WebGL.

**HTML5** — язык для структурирования и представления интернет-страниц.

**WebGL** (Web-based Graphics Library) — программная библиотека для языка программирования JavaScript, позволяющая создавать на JavaScript интерактивную 3D-графику, функционирующую в широком спектре совместимых с ней веб-браузеров.

**JavaScript** — [мультипарадигменный](https://ru.wikipedia.org/wiki/%D0%9C%D1%83%D0%BB%D1%8C%D1%82%D0%B8%D0%BF%D0%B0%D1%80%D0%B0%D0%B4%D0%B8%D0%B3%D0%BC%D0%B5%D0%BD%D0%BD%D0%BE%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5) [язык программирования](https://ru.wikipedia.org/wiki/%D0%AF%D0%B7%D1%8B%D0%BA_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F). Поддерживает [объектно-ориентированный](https://ru.wikipedia.org/wiki/%D0%9E%D0%B1%D1%8A%D0%B5%D0%BA%D1%82%D0%BD%D0%BE-%D0%BE%D1%80%D0%B8%D0%B5%D0%BD%D1%82%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%BD%D0%BE%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5), императивный и функциональный стили. Является реализацией языка ECMAScript. JavaScript обычно используется как встраиваемый язык для программного доступа к объектам [приложений](https://ru.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BC%D0%BF%D1%8C%D1%8E%D1%82%D0%B5%D1%80%D0%BD%D0%B0%D1%8F_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B0). Наиболее широкое применение находит в браузерах как язык сценариев для придания интерактивности веб-страницам. На JavaScript оказали влияние многие языки, при разработке была цель сделать язык похожим на Java, но при этом лёгким для использования непрограммистами. Языком JavaScript не владеет какая-либо компания или организация, что отличает его от ряда языков программирования, используемых в веб-разработке.

**Особенности Three.js:**

* Рендеры: Canvas или WebGL.
* Сцена: добавление и удаление объектов в режиме реального времени.
* Камеры: перспективная или ортографическая.
* Анимация: каркасы, быстрая кинематика, обратная кинематика, покадровая анимация.
* Источники света: внешний, направленный, точечный; тени: брошенные и полученные.
* Объекты: сети, частицы, спрайты, линии, скелетная анимация и другое.
* Геометрия: плоскость, куб, сфера, тор, 3D текст и другое; модификаторы: ткань, выдавливание.
* Поддержка: документация по API библиотеки находится в процессе постоянного расширения и дополнения, есть публичный форум и обширное сообщество.
* Примеры: на официальном сайте можно найти более 150 примеров работы со шрифтами, моделями, текстурами, звуком и другими элементами сцены.

Библиотека Three.js работает во всех браузерах, которые поддерживают технологию WebGL, а также может работать с «чистым» интерфейсом элемента CANVAS, благодаря чему работает и на многих мобильных устройствах. Three.js распространяется под лицензией [MIT license](https://ru.wikipedia.org/wiki/MIT_license) [2].

**Особенности численного решения** данной задачи заключаются, в основном, в постепенном увеличении (так как используется явная схема) собственной энергии системы вследствие численных ошибок. Но при взятии шага $dt$ довольно малым (менее одной десятитысячной от времени работы программы) сильного расхождения, заметного невооруженным глазом, с аналитическим решением не будет.

**Код**

Для визуализации данной задачи использовались файлы:

* three.js
* dat.gui.min.js
* stats.min.js

Код программы:

<!DOCTYPE html>
<html>

<head>
 <title>Example 01.03 - Materials and light</title>
 <script type="text/javascript" src="http://tm.spbstu.ru/htmlets/libs/three.min.js"></script>
 <script type="text/javascript" src="http://tm.spbstu.ru/htmlets/libs/stats.min.js"></script>
 <script type="text/javascript" src="http://tm.spbstu.ru/htmlets/libs/dat.gui.min.js"></script>
</head>

<body>
 <div id="Stats-output">
 </div>
 <div id="WebGL-output">
 </div>

 <script type="text/javascript">
 function init(){
 var stats=initStats();
 var scene=new THREE.Scene();
 var camera=new THREE.PerspectiveCamera(45,window.innerWidth/window.innerHeight,0.1,1000);
 var renderer=new THREE.WebGLRenderer();
 scene.background=new THREE.Color(0xDDFFFF);
 camera.position.set(-15,5,30);
 camera.lookAt(new THREE.Vector3(0,5,0));
 renderer.setClearColor(new THREE.Color(0xFFFFFF,1.0));
 renderer.setSize(window.innerWidth, window.innerHeight);
 renderer.shadowMap.enabled=true;

 var rc=2,lc=1,bw=40,bh=3,n=20,sl=1.5;//размеры оснований и дисков
 var base=new THREE.Mesh(new THREE.BoxGeometry(bw,bh,bh),new THREE.MeshLambertMaterial({color:0xF9FFF5}));//создаем основание
 var lbase=new THREE.Mesh(new THREE.BoxGeometry(bh,3\*bh,bh),new THRE rtMaterial({color:0xF9FFF5}));//левый блок
 var rbase=new THREE.Mesh(new THREE.BoxGeometry(bh,3\*bh,bh),new THREE.MeshLambertMaterial({color:0xF9FFF5}));//правый блок
 var disk1=new THREE.Mesh(new THREE.CylinderGeometry(rc,rc,lc,15,0),new THREE.MeshLambertMaterial({color:0xF9FFF5}));//создаем левый(первый) диск
 var disk2=new THREE.Mesh(new THREE.CylinderGeometry(rc,rc,lc,15,0),new THREE.MeshLambertMaterial({color:0xF9FFF5}));//создаем правый(второй) диск
 for(var k=1;k<=3;k++)//цикл для создания пружин
 for(var i=0;i<=n;i++){
 var nexus=new THREE.Mesh(new THREE.CylinderGeometry(0.05,0.05,sl),new THREE.MeshLambertMaterial({color:0xF9FFF5}));//создаем звено пружины
 nexus.name=k+"-"+i;
 nexus.castShadow=true;
 scene.add(nexus);
 }
 //источники освещения
 var ambientLight=new THREE.AmbientLight(0x0F0F0F);
 var spotLight=new THREE.SpotLight(0xFFFFFF);
 //настраиваем тени
 base.receiveShadow=true;
 lbase.receiveShadow=true;
 rbase.receiveShadow=true;
 lbase.castShadow=true;
 rbase.castShadow=true;
 disk1.castShadow=true;
 disk2.castShadow=true;
 spotLight.castShadow=true;
 //прикрепляем объекты к сцене
 scene.add(base);
 scene.add(lbase);
 scene.add(rbase);
 scene.add(disk1);
 scene.add(disk2);
 scene.add(ambientLight);
 scene.add(spotLight);
 //прикрепляем объекты к сцене
 document.getElementById("WebGL-output").appendChild(renderer.domElement);
 //создаем пользовательский интерфейс
 var m1,m2,c1,c2,c3,speed;//параметры системы
 var k1,k2,g1,g2;//собственные частоты и отношения амплитуд
 var A11,A21,a1,a2;//начальные условия для амплитуд и фаз
 var x01,x02,v01,v02;//начальные условия для смещений и скоростей
 var t=0,dt,fl=true;//обнуление времени
 var cnt=new function(){
 this.dt=0.002;this.x01=-5;this.x02=5;this.v01=0;this.v02=0; this.m1=6; this.m2=4;his.c1=2000;this.c2=1000;this.c3=3000;
 this.pause=function(){fl=false;}//функция паузы
 this.resume=function(){fl=true;}//функция снятия с паузы
 this.redraw=function(){//функция перезапуска
 t=0;x01=cnt.x01;x02=cnt.x02;v01=cnt.v01;v02=cnt.v02;dt=cnt.dt;m1=cnt.m1;m2=cnt.m2;c1=cnt.c1;c2=cnt.c2;

 c3=cnt.c3;init();
 };
 };

 var gui=new dat.GUI();
 gui.add(cnt,'dt',0,0.005);gui.add(cnt,'x01',-5,-rc);gui.add(cnt,'x02',rc,5);gui.add(cnt,'v01',-10,10);gui.add(cnt,'v02',-10,10);
 gui.add(cnt,'m1',1,12);gui.add(cnt,'m2',1,12);gui.add(cnt,'c1',500,5000);gui.add(cnt,'c2',500,5000);
 gui.add(cnt,'c3',500,5000);gui.add(cnt,'pause');gui.add(cnt,'resume');gui.add(cnt,'redraw');cnt.redraw();render();

 function render(){
 stats.update();
 if(fl)t+=dt;//шаг времени и обновление расположений и ориентаций дисков
 disk1.position.x=x01-rc+A11\*Math.sin(k1\*t+a1)+A21\*Math.sin(k2\*t+a2);
 disk2.position.x=x02+rc+g1\*A11\*Math.sin(k1\*t+a1)+g2\*A21\*Math.sin(k2\*t+a2);
 disk1.rotation.y=(x01-rc-disk1.position.x)/rc;
 disk2.rotation.y=(x02-rc-disk2.position.x)/rc;

 for(var k=1;k<=3;k++){//обновление звеньев пружин
 var xst,xfn;
 if(k==1){xst=-bw/2;xfn=disk1.position.x-rc/2;}
 if(k==2){xst=disk1.position.x+rc/2;xfn=disk2.position.x-rc/2;}
 if(k==3){xst=disk2.position.x+rc/2;xfn=bw/2;}
 var length=Math.abs(xfn-xst);
 var h=length/n;
 for(var i=0;i<=n;i++){
 var nexus=scene.getObjectByName(k+"-"+i);
 nexus.position.x=xst+i\*h;
 if(i%2==0)nexus.rotation.z=Math.asin(h/sl);
 else nexus.rotation.z=-Math.asin(h/sl);
 }
 }

 renderer.render(scene,camera);
 requestAnimationFrame(render);
 }
 function init(){
 var tk1=(c2+c3)/m2;//вспомогательные параметры
 var tk2=(c1+c2)/m1;
 k1=Math.sqrt((tk1+tk2-Math.sqrt((tk1-tk2)\*(tk1-tk2)+4\*c2\*c2/m1/m2))/3);//собственные частоты системы
 k2=Math.sqrt((tk1+tk2+Math.sqrt((tk1-tk2)\*(tk1-tk2)+4\*c2\*c2/m1/m2))/3);
 g1=(1.5\*m1\*k1\*k1-c1-c2)/c2;//отношения амплитуд
 g2=(1.5\*m1\*k2\*k2-c1-c2)/c2;
 a1=Math.atan(k1\*(x02-g2\*x01)/(v02-g2\*v01));//начальные условия фаз
 a2=Math.atan(k2\*(x02-g1\*x01)/(v02-g1\*v01));
 A11=(x02-g2\*x01)/(g1-g2)/Math.sin(a1);//начальные условия амплитуд
 A21=(x02-g1\*x01)/(g2-g1)/Math.sin(a2);
 base.position.set(0,0,0);
 lbase.position.set(-bw/2,0,0);
 rbase.position.set(bw/2,0,0);
 disk1.position.set(x01,rc+bh/2,0);
 disk2.position.set(x02,rc+bh/2,0);
 disk1.rotation.x=Math.PI/2;
 disk2.rotation.x=Math.PI/2;
 spotLight.position.set(50,60,50);
 for(var k=1;k<=3;k++)//первоначальные условия для звеньев пружин
 for(var i=0;i<=n;i++){
 var nexus=scene.getObjectByName(k+"-"+i);
 nexus.position.set(0,rc+bh/2,0);
 }
 }
 function initStats() {
 var stats = new Stats();
 stats.setMode(0);
 stats.domElement.style.position='absolute';
 stats.domElement.style.left='0px';
 stats.domElement.style.top='0px';
 document.getElementById("Stats-output").appendChild(stats.domElement);
 return stats;
 }
 }
 window.onload=init;
</script>
</body>
</html>

**ЗАКЛЮЧЕНИЕ**

В данной работе я провел исследование свободных колебаний системы с двумя степенями свободы, а затем визуализировал решение с помощью библиотеки THREE.js.

Полученные результаты выглядят так:



Опции программы:

* dt – шаг времени, который определяет скорость анимации
* $x\_{1}$и $x\_{2}$ ,$v\_{1}$и $v\_{2}$ – начальные положения и скорости дисков
* $m\_{1}$,$m\_{2}$,$c\_{1}$,$c\_{2}$,$c\_{3}$ – параметры системы
* Pause, resume, redraw – кнопки для управления программой

**СПИСОК ИСПОЛЗОВАННЫХ ИСТОЧНИКОВ**

1.Яблонский А.А. Сборник задач для курсовых работ по теоретической механике: Учебное

пособие для технических вузов. – 5-е изд., исправленное – М.: Интеграл-Пресс, 2000. – 384 с.

2.Dirksen J. – «Learning Three.js. The JavaScript 3D Library for WebGL (2nd Edition)» – М.:

Packt Publishing, 2015. – 382 с.