Санкт-Петербургский 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. М.А.Ведров

Руководитель

ассистент

«\_\_\_» \_\_\_\_\_\_\_\_\_\_ 2018 г.

Санкт-Петербург

2018

**СОДЕРЖАНИЕ**

|  |  |
| --- | --- |
| **ВВЕДЕНИЕ**. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . | 4 |
| 1. Классификация колебаний. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . | 5 |
| 1. Three.js. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .. . . . . . . | 6 |
| 3.1. Постановка задачи. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . | 8 |
| 3.2. Решение задачи . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . | 9 |
| 1. Моделирование в программной среде js. . . . . . . . . . . . . . . . . . . . . . . . . | 11 |
| **ЗАКЛЮЧЕНИЕ**. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . | 15 |
| **СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ** . . . . . . . . . . . . . . . . . . | 16 |

**ВВЕДЕНИЕ**

**Колебания [2]** – это движение тела, в ходе которого оно многократно движется по одной и той же траектории и проходит при этом одни и те же точки пространства. Примерами колеблющихся объектов могут служить - маятник часов, струна скрипки или фортепиано, вибрации автомобиля.

Колебания играют важную роль во многих физических явлениях за пределами области механики. Например, напряжение и сила тока в электрических цепях 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 сходным образом. Физическая величина, которая изменяется со временем при колебательном движении, называется смещением. Амплитуда представляет собой максимальное смещение колеблющегося объекта от положения равновесия. Полное колебание, или цикл – это движение, при котором тело, выведенное из положения равновесия на некоторую амплитуду, возвращается в это положение, отклоняется до максимального смещения в противоположную сторону и возвращается в свое первоначальное положение. Период колебания T – время, необходимое для осуществления одного полного цикла. Число колебаний за единицу времени - это частота колебаний.

**КЛАССИФИКАЦИЯ КОЛЕБАНИЙ**

Различают несколько видов колебаний, зависящих от подчёркиваемых свойств колеблющихся систем (осцилляторов)

По физической природе:

• Механические (звук, вибрация)

• Электромагнитные (свет, радиоволны, тепловые)

• Смешанного типа — комбинации вышеперечисленных

По характеру взаимодействия с окружающей средой:

• Вынужденные — колебания, протекающие в системе под влиянием внешнего периодического воздействия. Примеры: листья на деревьях, поднятие и опускание руки. При вынужденных колебаниях может возникнуть явление резонанса: резкое возрастание амплитуды колебаний при совпадении собственной частоты осциллятора и частоты внешнего воздействия.

• Свободные (или собственные) — это колебания в системе под действием внутренних сил после того, как система выведена из состояния равновесия (в реальных условиях свободные колебания всегда затухающие). Простейшими примерами свободных колебаний являются колебания груза, прикреплённого к пружине, или груза, подвешенного на нити.

• Автоколебания — колебания, при которых система имеет запас потенциальной энергии, расходующейся на совершение колебаний (пример такой системы — механические часы). Характерным отличием автоколебаний от вынужденных колебаний является то, что их амплитуда определяется свойствами самой системы, а не начальными условиями.

• Параметрические — колебания, возникающие при изменении какого-либо параметра колебательной системы в результате внешнего воздействия.

• Случайные — колебания, при которых внешняя или параметрическая нагрузка является случайным процессом.

**БИБЛИОТЕКА THREE.JS**

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

Three.js позволяет создавать ускоренную на GPU 3D графику, используя язык **JavaScript** как часть сайта без подключения проприетарных плагинов для браузера. Это возможно благодаря использованию технологии WebGL. Поддерживает трёхмерные модели формата Collada.

**JavaScript** — мультипарадигменный язык программирования. Поддерживает объектно-ориентированный, императивный и функциональный стили. Является реализацией языка ECMAScript (стандарт ECMA-262).

JavaScript обычно используется как встраиваемый язык для программного доступа к объектам приложений. Наиболее широкое применение находит в браузерах как язык сценариев для придания интерактивности веб-страницам.

Основные архитектурные черты: динамическая типизация, слабая типизация, автоматическое управление памятью, прототипное программирование, функции как объекты первого класса.

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 :

• Рендереры: Canvas, SVG или WebGL

• Сцена: добавление и удаление объектов в режиме реального времени; туман

• Камеры: перспективная или ортографическая

• Анимация: каркасы, быстрая кинематика, обратная кинематика, покадровая анимация

• Источники света: внешний, направленный, точечный; тени: брошенные и полученные

• Шейдеры: полный доступ ко всем OpenGL шейдерам (GLSL)

• Объекты: сети, частицы, спрайты, линии, скелетная анимация и другое

• Геометрия: плоскость, куб, сфера, тор, 3D текст и другое; модификаторы: ткань, выдавливание

• Загрузчики данных: двоичный, изображения, JSON и сцена

• Экспорт и импорт: утилиты, создающие Three.js-совместимые JSON файлы из форматов: Blender, openCTM, FBX, 3D Studio Max, и Wavefront .obj файл

• 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; также может работать с «чистым» интерфейсом элемента CANVAS, благодаря чему работает и на многих мобильных устройствах. Three.js распространяется под лицензией MIT license.

**ПОСТАНОВКА ЗАДАЧИ**

**Вариант. 17.Задача из сборника Яблонского [1].**

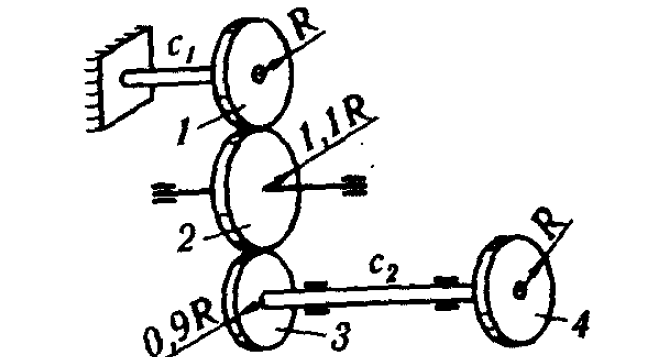


Рис.1. Рисунок к задаче

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

(Так как главной задачей в данной работе является в разработке компьютерной модели с регулируемыми условиями, поэтому мы будем искать решение в общем виде.)

Дано: (рис.1)

**РЕШЕНИЕ ЗАДАЧИ**

Вычисления проводятся относительно координат и , относящихся к валу 1 и 2 соответственно.

1. **Кинетическая энергия**
2. **Потенциальная энергия**
3. **Коэффициенты инерции и жесткости**
4. **Система уравнений в виде Лагранжа 2-ого рода**

Будем искать частное решение этой системы в виде

Получаем уравнения частот

Полученная система решается относительно k

Коэффициенты распределения

1. **Главные колебания**

Первые главные колебания

Вторые главные колебания

Общее решение

**МОДЕЛИРОВАНИЕ В ПРОГРАММНОЙ СРЕДЕ JS [5]**

При моделировании задачи были использованы библиотеки:

* Three.js
* graphicBuilder.js
* dat.gui.min.js

Данные (рис.2) регулируются с помощью библиотеки dat.gui.min.js. Данные оказывают влияние не только на прорисовку анимации движения валов, но и на графики (рис.3, 4), работающие за счёт библиотеки graphicBuilder.js.

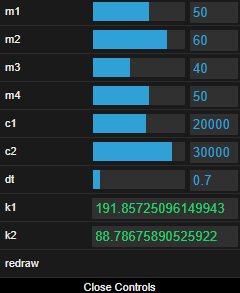


Рис.2. Регулируемые данные

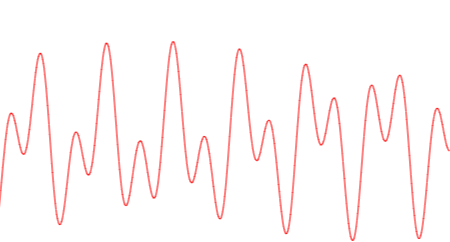


Рис.3. График по координате

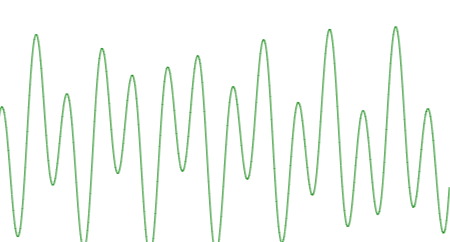


Рис.4. График по координате

* Код отвечающий за данные:

const solution = new function () {  
 this.c1 = 20000;  
 this.c2 = 30000;  
 this.c3 = 10000;  
 this.R = .2;  
 this.t = 0;  
 this.dt = .7;  
  
 this.m1 = 50;  
 this.m2 = 60;  
 this.m3 = 40;  
 this.m4 = 50;  
  
  
 this.a11 = .5\*(this.m1+this.m2)\*Math.pow(this.R,2);  
 this.a22 = .5\*(this.m3+this.m4)\*Math.pow(this.R,2);  
 this.c11 = this.c1+this.c2/1.44;  
 this.c22 = this.c2/.81 + this.c3;  
 this.c12 = this.c2/(1.2\*.9);  
 this.D = Math.sqrt(Math.pow(this.a11 \* this.c22 + this.a22 \* this.c11,2) -4 \* this.a11 \* this.a22 \* (this.c11 \* this.c22 - Math.pow(this.c12,2)));  
 this.k1 = '' + Math.sqrt((this.a11 \* this.c22 + this.a22 \* this.c11 + this.D)/(2\*this.a11\*this.a22));  
 this.k2 = '' + Math.sqrt((this.a11 \* this.c22 + this.a22 \* this.c11 - this.D)/(2\*this.a11\*this.a22));  
  
 this.p1 = -(this.c11 - this.a11\*Math.pow(this.k1,2))/this.c12;  
 this.p2 = -(this.c11 - this.a11\*Math.pow(this.k2,2))/this.c12;  
  
 let self = this;  
  
 this.redraw = function() {  
 self.t = 0;  
 ctx.clearRect(0,-242,500,500);  
 ctx1.clearRect(0,-242,500,500);  
 };  
};

const gui = new dat.GUI();  
gui.add(solution, 'm1',20,70).onChange(solution.redraw);  
gui.add(solution, 'm2',20,70).onChange(solution.redraw);  
gui.add(solution, 'm3',20,70).onChange(solution.redraw);  
gui.add(solution, 'm4',20,70).onChange(solution.redraw);  
gui.add(solution, 'c1',0,35000).onChange(solution.redraw);  
gui.add(solution, 'c2',0,35000).onChange(solution.redraw);  
gui.add(solution, 'dt',0,10).onChange(solution.redraw);

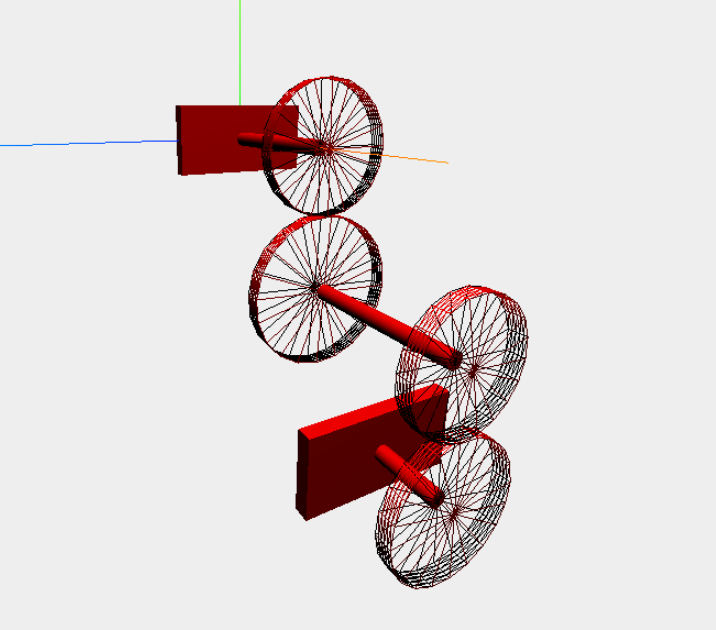


Рис.5. Движение валов

* Прорисовка движения валов (рис.5) и графиков:

animate = function () {  
 requestAnimationFrame(animate);  
 T = solution.dt;  
 solution.a11 = .5 \* (solution.m1 + solution.m2) \* Math.pow(solution.R, 2);  
 solution.a22 = .5 \* (solution.m3 + solution.m4) \* Math.pow(solution.R, 2);  
 solution.c11 = solution.c1 + solution.c2 / 1.44;  
 solution.c22 = solution.c2 / .81 + solution.c3;  
 solution.c12 = (1 / (1.2 \* .9)) \* solution.c2;  
 solution.D = Math.sqrt(Math.pow(solution.a11 \* solution.c22 + solution.a22 \* solution.c11, 2) - 4 \* solution.a11 \* solution.a22 \* (solution.c11 \* solution.c22 - Math.pow(solution.c12, 2)));  
 solution.k1 = '' + Math.sqrt((solution.a11 \* solution.c22 + solution.a22 \* solution.c11 + solution.D) / (2 \* solution.a11 \* solution.a22));  
 solution.k2 = '' + Math.sqrt((solution.a11 \* solution.c22 + solution.a22 \* solution.c11 - solution.D) / (2 \* solution.a11 \* solution.a22));  
 solution.p1 = -(solution.c11 - solution.a11 \* Math.pow(solution.k1, 2)) / solution.c12;  
 solution.p2 = -(solution.c11 - solution.a11 \* Math.pow(solution.k2, 2)) / solution.c12;  
 wheel\_1.spin(solution, 1, 1, 1);  
 wheel\_2.spin(solution, -1, 1.2, 1);  
 wheel\_3.spin(solution, -1, .9, 2);  
 wheel\_4.spin(solution, 1, 1, 2);  
  
 if (solution.t === 0) {  
 z = 0;  
 }  
 if (solution.t > 450) {  
 z -= T;  
 }  
 ctx.clearRect(0, 0, buffer.width, 400);  
 ctx.drawImage(GraphicBulider.bufferDraw(buffer, solution.t, T, Moving\_parts.equation(solution, 1, 1, 1, solution.t - T), Moving\_parts.equation(solution, 1, 1, 1, solution.t), 'red'), z, 0);  
 ctx1.clearRect(0, 0, buffer.width, 400);  
 ctx1.drawImage(GraphicBulider.bufferDraw(buffer1, solution.t, T, Moving\_parts.equation(solution, 1, 1, 0, solution.t - T), Moving\_parts.equation(solution, 1, 1, 0, solution.t), 'green'), z, 0);  
 solution.t += T;  
 renderer.render(scene, camera);  
};  
animate();

**ЗАКЛЮЧЕНИЕ**

В данной курсовой работе было реализовано решение задачи теоретической механики по нахождение уравнений движений систем с двумя степенями свободы, также с помощью средств языка программирования JS смоделировано наглядная модель решённой задачи.

**СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ**

1. Яблонский А. А. и др. Сборник заданий для курсовых работ по теоретической механике: Учебное пособие. – Высшая школа, 1985.
2. Алдошин Г. Т. Теория линейных и нелинейных колебаний: Учебное пособие. 2-е изд., стер. – 2013.
3. Флэнаган Д. JavaScript. Подробное руководство, 6-е издание //Санкт-Перетбург: Символ-Плюс. – 2012. – С. 21.