

4

Коливання.

Математичний маятник.

Мета: отримати уявлення про моделювання перебігу коливального процесу.

4.1 Загальна теорія

Найбільш наочними й зручними для вивчення є механічні коливання – рух тіла (матеріальної точки), при якому його положення у просторі та інші характеристики руху з плином часу періодично повторюються.

Характер руху всякого тіла визначається властивостями діючих на нього сил. Зокрема, якщо прикладена “спонукальна” сила періодично змінює свій напрям на протилежний і здатна подолати сили тертя та опору, то тіло здійснює вимушені коливання. Вільні

механічні коливання тіла спостерігаються, коли рівнодійна прикладених сил весь час намагається повернути його в положення рівноваги. Таку силу інколи називають “повертаючою силою”. Але слід пам’ятати, що цей термін є умовним. У природі не існує ніякої спеціальної повертаючої сили — вона створюється сумісною дією якихось інших сил. До прикладу, при вільних коливаннях кульки на нитці повертаюча сила є рівнодійною сил тяжіння та натягу нитки.

В цій практичній роботі ми вивчаємо коливання на прикладі математичного маятника. Математичний маятник — це *теоретична* модель маятника, в якій матеріальна точка масою m підвішена на невагомій нерозтяжній нитці або стрижні. Модель нехтує розмірами тіла, деформацією підвісу та тертям в точці підвісу. Зазвичай розглядають коливання маятника в одній площині. В загальному випадку, якщо відхилити маятник від положення рівноваги та штовхнути його вбік, рух маятника буде складатися з коливань в вертикальних площинах та руху по горизонталі

При малому відхиленні математичний маятник здійснює гармонічні коливання. Якщо початкове

відхилення є великим, то коливання маятника періодичні, але не гармонічні.

Математичний маятник має два положення рівноваги: стійке та нестійке. В стійкому положенні рівноваги маятник висить непорушно строго вертикально, сила тяжіння врівноважується силою пружності стрижня. Якщо відвести маятник від положення рівноваги, або надати йому початкової швидкості, виникають коливання. Сили тертя, що діють на реальний маятник але не враховані в даній моделі, приводять до загасання коливань та знов повертають маятник в початкове положення. Саме тому це положення має назву стійкого. Інше положення рівноваги математичного маятника знаходиться в точці $\varphi = \pi$, тобто коли стрижень (підвіс) орієнтований вертикально вгору. В цьому положенні сили тяжіння та пружності стрижня, як і в точці стійкої рівноваги, зрівноважені, проте дана рівновага є нестійкою. При найменшому відхиленні від вертикального положення рівнодійна сил, що діють на маятник, виводить його з рівноваги. Реальний маятник вже ніколи не повернеться в це положення. Підтримати маятник у вертикальному положенні можна за допомогою балансування, яке зводиться до особливих рухів точки опори.

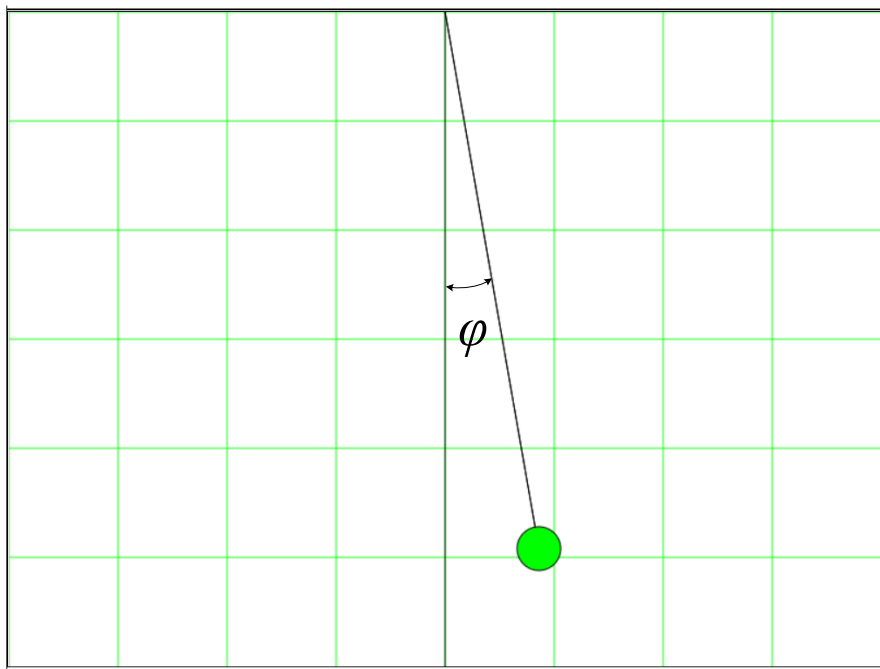


Рис. 4.1 – Кут відхилення в математичному маятнику

Якщо вибрати систему координат так, щоб її початок був у точці підвісу маятника, то його координати в кожен момент часу будуть:

$$x = l \sin \varphi; \quad y = l \cos \varphi,$$

де φ – миттєвий кут відхилення, l – довжина підвісу.

Сам кут відхилення φ при цьому залежить від часу:

$$\varphi = \varphi_0 \cos(\omega t),$$

де φ_0 – початковий кут відхилення, а ω – колова частота коливань маятника:

$$\omega = \sqrt{\frac{g}{l}},$$

де g – прискорення вільного падіння. Важливо відзначити, що по цій формулі ω отримується одразу в рад/с, тому в коді переводити цю величину в радіани для обчислення косинуса від неї не потрібно.

4.2 Моделювання траєкторії руху математичного маятника

Для цієї практичної роботи ми будемо використовувати наробітки з попередньої практичної роботи – про кінематику і рух на площині.

У папці з робочими файлами повинні бути ті ж самі файли `index.html`, `styles.css` та `whowroteit.js`.

Вхідними даними для моделювання будуть довжина підвісу та початковий кут відхилення. Результат – анімація коливального руху. Також передбічені три кнопки – запуск процесу, його зупинка та повний перезапуск із новими вхідними даними. Далі наводяться лістинг основного скрипту:

```
let canvas = document.getElementById('Pendulum');
let ctx = canvas.getContext('2d');

const X = canvas.width;
```

```

const Y = canvas.height;
const X0 = Math.floor(X/2);
const Y0 = Math.floor(Y/2);
const GRID = 100;

drawGrid(GRID);

const R = 20;
const SCALE = 100;
const g = 9.81;

let l, phi0, omega, t=0;

let raf;

//=====
function drawGrid(GRID) {
  ctx.clearRect(0, 0, X, Y);
  ctx.strokeStyle = "#00FF00";
  ctx.lineWidth = 0.5;
  let x, y;
  for (x=0; x<X; x+=GRID) {
    ctx.beginPath();
    ctx.moveTo(x, 0);
    ctx.lineTo(x, Y);
    ctx.closePath();
    ctx.stroke();
  }
  for (y=0; y<Y; y+=GRID) {
    ctx.beginPath();

```

```

    ctx.moveTo(0, y);
    ctx.lineTo(X, y);
    ctx.closePath();
    ctx.stroke();
}
ctx.strokeStyle = "#000000";
ctx.beginPath();
ctx.moveTo(X0, 0);
ctx.lineTo(X0, Y);
ctx.closePath();
ctx.stroke();
}

//=====
function drawPendulum(length, angle) {
    let x, y;
    ctx.lineWidth = 1;
    ctx.strokeStyle = "#000000";
    ctx.fillStyle = "#00FF00";
    x = X0 + Math.floor(length*SCALE*Math.sin(angle));
    y = Math.floor(length*SCALE*Math.cos(angle));
    ctx.beginPath();
    ctx.moveTo(X0, 0);
    ctx.lineTo(x, y);
    ctx.closePath();
    ctx.stroke();
    ctx.beginPath();
    ctx.arc(x, y, R, 0, Math.PI*2);
    ctx.closePath();
    ctx.fill();
}

```

```

    ctx.stroke();
}

//=====
function oscilations() {
    let phi;
    drawGrid(GRID);
    t += .05
    phi = phi0 * Math.cos(omega*t);
    drawPendulum(l, phi);
    raf = window.requestAnimationFrame(oscilations);
}

//=====
function go() {
    l=Number(document.getElementById('Length').value);
    phi0=Number(document.getElementById('Angle').value)
        *Math.PI/180;
    omega = Math.sqrt(g/l);
    drawPendulum(l, phi0);
    document.getElementById('stopButton').disabled
        = false;
    document.getElementById('goButton').disabled
        = true;
    document.getElementById('beginButton').disabled
        = false;
    oscilations();
}

```



```

//=====
function stop() {
    window.cancelAnimationFrame(raf);
    document.getElementById('goButton').disabled
        = false;
    document.getElementById('stopButton').disabled
        = true;
}

//=====
function begin() {
    window.cancelAnimationFrame(raf);
    drawGrid(GRID);
    drawPendulum(l, phi0);
    document.getElementById('goButton').disabled
        = false;
    document.getElementById('stopButton').disabled
        = true;
}

```

Даний скрипт реалізує незатухаючі коливання, чого, зрозуміло, насправді у природі бути не може. Тому завданням цієї практичної роботи буде реалізувати затухаючі коливання. Мірою затухання коливання є *логарифмічний декремент затухання* – безрозмірна фізична величина, що описує зменшення амплітуди коливального процесу і дорівнює натуральному

логарифму відношення двох послідовних амплітуд величини x в один і той же бік (див. рис. 4.2):

$$\lambda = \ln \frac{x_0}{x_1}$$

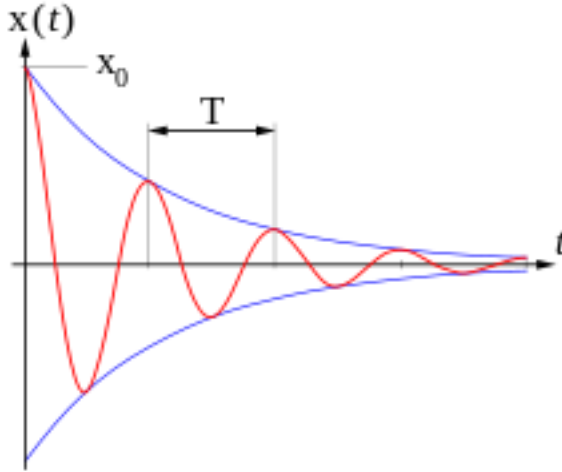


Рис. 4.2 – До визначення логарифмічного декременту затухання

Логарифмічний декремент затухання також дорівнює добутку коефіцієнта затухання β на період коливань:

$$\lambda = \beta T,$$
$$T = 2\pi \sqrt{\frac{l}{g}}.$$

Тоді рівняння затухоаючих коливань буде таким:

$$u = u_0 e^{-\lambda t} \cos(\omega t),$$

де u – величина, що виконує коливання, а u_0 – її максимальне (амплітудне) значення.

Таким чином, остаточним завданням цієї роботи є наступне: потрібно додати ще одне поле для вводу коефіцієнта затухання або логарифмічного декременту затухання, отримання відповідного числа та доповнення вище наведеного коду функцією, яка буде реалізовувати саме затухоаючі коливання.

4.3 Зміст звіту

Звіт з цієї практичної роботи повинен включати наступне:

- 0) Титульна сторінка.
- 1) Назва і мета практичної роботи.
- 2) Лістинг остаточного варіанту js-файлу зі скриптом, що реалізує затухоаючі коливання.
- 3) Опис результатів експериментів із коливальною системою: що буде коли коефіцієнт затухання (або декремент затухання) дорівнює нулю? Дорівнює одиниці? Більше одиниці?
- 4) Висновки. Чому математичний маятник взагалі коливається? Чому коливання

затухають? Який фізичний зміст коефіцієнта затухання? Логарифмічного декременту затухання? Яку величину легше запрограмувати – коефіцієнт затухання чи декремент затухання?

Звіт з практики зберегти у форматі pdf і надіслати на пошту krt_kro@ztu.edu.ua.