

Одеський національний університет імені І. І. Мечникова
Факультет математики, фізики та інформаційних технологій
Кафедра методів математичної фізики

Кваліфікаційна робота

на здобуття ступеня вищої освіти «бакалавр»

«Коливання підвішеної нитки»

«Oscillations of the suspended thread»

Виконала: здобувачка денної форми навчання
спеціальності 113 Прикладна математика
Освітня програма «Прикладна математика»
Шарковська Марія Олександрівна

Керівник: канд. фіз.-мат. наук, доц. Фесенко Г. О. _____

Рецензент: доктор фіз.-мат. наук, проф. Вайсфельд Н.Д.

Рекомендовано до захисту:

Протокол засідання кафедри

№ ____ від _____ 2024 р.

Завідувач кафедри

Захищено на засіданні ЕК № _____

Протокол № ____ від _____ 2024 р.

Оцінка _____ / _____ / _____

Голова ЕК

ЗМІСТ

Вступ		3
1	Вільні коливання підвішеної важкої однорідної гнучкої нитки	5
1.1	Отримання рівняння коливань	5
1.2	Розв'язання рівняння коливань	8
1.3	Результати моделювання	11
2	Вимушені коливання важкої однорідної гнучкої нитки	14
2.1	Отримання рівняння коливань	14
2.2	Розв'язання рівняння коливань	15
2.3	Результати моделювання	18
3	Складні коливання важкої однорідної гнучкої нитки	23
3.1	Отримання рівняння коливань	23
3.2	Розв'язання рівняння коливань	24
3.3	Результати моделювання	24
	Додаток А	26
	Додаток Б	30
	Додаток В	37
	Висновки	43
	Список літератури	44

ВСТУП

Коливання підвішеної важкої однорідної гнучкої нитки є важливою задачею в механіці та математичній фізиці. Вивчення коливань таких систем має значне практичне застосування, від інженерних споруд до комп'ютерної анімації. У цій роботі розглядаються вимушені, вільні та складні коливання підвішеної нитки.

Функції Бесселя відіграють ключову роль у розв'язанні диференціальних рівнянь, які описують коливання гнучкої нитки. Завдяки своїм властивостям, функції Бесселя знаходять застосування у багатьох галузях науки і техніки, включаючи акустику, електродинаміку, теплопередачу та інші. Вони виникають у вирішенні диференціальних рівнянь другого порядку, таких як рівняння Лапласа та рівняння Гельмгольца в циліндричних координатах.

Диференціальне рівняння, яке описує коливання нитки, є фундаментальним для розуміння механіки деформованих тіл. Його розв'язок дозволяє прогнозувати поведінку системи під впливом різних збурень. Застосування таких математичних моделей є незамінним у сучасній комп'ютерній графіці та анімації. Наприклад, для створення реалістичних візуальних ефектів у фільмах, відеоіграх, анімація коливання ниток або мотузок використовує розв'язки диференціальних рівнянь.

У кваліфікаційній роботі розглядається побудова математичної моделі, що описує коливання важкої однорідної гнучкої нитки закріпленої у верхній точці, розв'язання диференціальних рівнянь, що описують динаміку коливань, із застосуванням рядів Бесселя-Фур'є, аналіз отриманих розв'язків та дослідження впливу параметрів системи, таких як довжина нитки, лінійна густина, сила тяжіння, початкове відхилення, на коливальний процес, графічне представлення результатів та їх інтерпретація.

Метою даного дослідження є розробка математичної моделі, яка описує коливання важкої однорідної гнучкої нитки, закріпленої у верхній точці, та вивчення впливу основних параметрів системи на її динамічні характеристики.

Об'єктом дослідження є важка однорідна гнучка нитка довжини l , що здійснює коливання під дією сили тяжіння.

Предметом дослідження є динамічні характеристики коливань підвішеної гнучкої нитки, такі як амплітуда, частота та період коливань, а також вплив основних параметрів системи на ці характеристики.

У роботі використовувалися метод математичного аналізу для побудови диференціальних рівнянь, що описують коливання гнучкої нитки, ряди Фур'є-Бесселя для розв'язання отриманих диференціальних рівнянь, чисельні методи для моделювання та аналізу коливального процесу, графічне представлення даних для візуалізації результатів дослідження, програмні засоби, такі як MATLAB, для реалізації розрахунків та візуалізації динаміки коливань.

Важливість цієї роботи полягає у розробці чисельного алгоритму для моделювання коливань нитки, що дозволяє глибше зрозуміти і проаналізувати різні аспекти цієї фізичної системи. Моделювання таких процесів сприяє покращенню методів проєктування інженерних конструкцій та створенню більш реалістичних анімацій у розважальних та освітніх медіа.

РОЗДІЛ 1

ВІЛЬНІ КОЛИВАННЯ ПІДВІШЕНОЇ ВАЖКОЇ
ОДНОРІДНОЇ ГНУЧКОЇ НИТКИ

1.1 Отримання рівняння коливань

Розглядається важка однорідна гнучка нитка довжини l . Нитка закріплена верхнім кінцем у точці $x = l$ та чинить коливання під дією сили тяжіння. За вісь x приймаємо вертикальний напрямок, уздовж якого розташується нитка, коли під дією своєї ваги вона займе прямолінійне положення. Якщо нитка максимально витягнута вздовж OX , нижній кінець нитки перебуває в точці $x = 0$. Розтяг нитки в точці x відповідає вазі нитки під цією точкою - $m_x g$. Якщо через ρ позначити лінійну густину (маса, що приходить на одиницю довжини), тоді отримаємо $T(x) = \rho x g$.

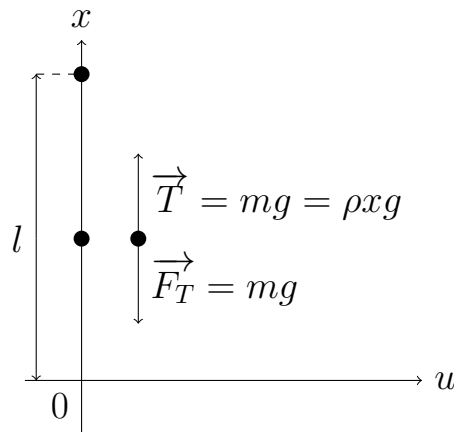


Рис. 1 - положення нитки під дією сили тяжіння

Нитка виконує коливання у площині XOU (рисунок 2). Якщо на осі OX обрати точку x та провести перпендикуляр до перетину з ниткою, після чого опустити перпендикуляр на вісь OU , отримаємо u , котре можна розглядати, як функцію від x . Таким чином, форму нитки можна описувати функцією $u(x)$. Якщо форма нитки змінюється з часом у процесі коливань, додається також аргумент часу. Разом, отримуємо функцію $u(x,t)$.

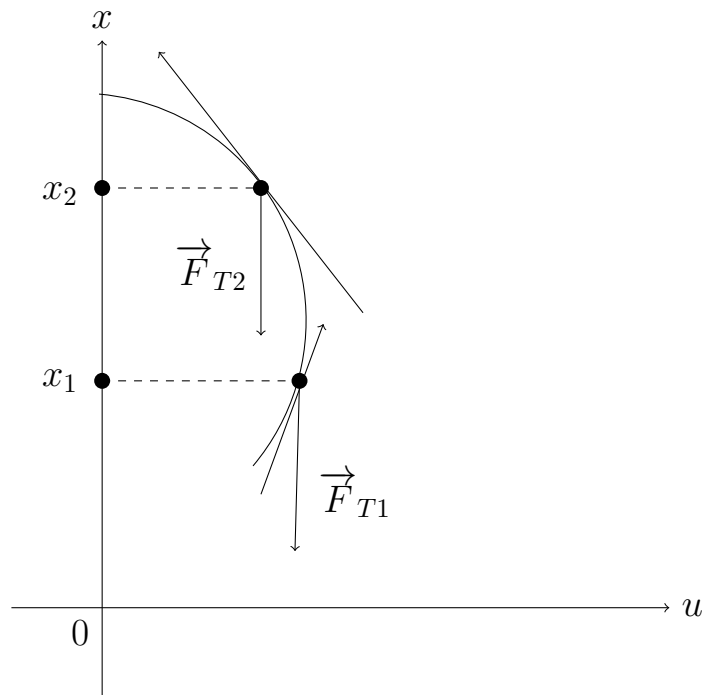


Рис. 2 - відхилення нитки від положення рівноваги

Нехай, нитка відхилена від положення рівноваги. Виділимо на нитці дві близько розташовані точки - Н та М, і проведемо перпендикуляри на OX , отримавши точки x_1, x_2 відповідно. Проведемо дотичні до кривої, відповідної нитки, у точках Н та М. Сила натягу нитки буде спрямована вздовж дотичної. При цьому вектор сили можна розкласти за базисом, знайшовши проєкції на відповідні осі:

$$T_{1x} = T_1 \cos \alpha_1 = \rho l_1 g \cos \alpha_1$$

$$T_{1u} = T_1 \sin \alpha_1 = \rho l_1 g \sin \alpha_1$$

$$T_{2x} = T_2 \cos \alpha_2 = \rho l_2 g \cos \alpha_2$$

$$T_{2u} = T_2 \sin \alpha_2 = \rho l_2 g \sin \alpha_2$$

Різниці значень проєкцій сили натягу у різних точках:

$$T_{2x} - T_{1x} = \rho l_2 g \cos \alpha_2 - \rho l_1 g \cos \alpha_1 = \rho (l_2 \cos \alpha_2 - l_1 \cos \alpha_1)$$

$$T_{2u} - T_{1u} = \rho l_2 g \sin \alpha_2 - \rho l_1 g \sin \alpha_1 = \rho (l_2 \sin \alpha_2 - l_1 \sin \alpha_1)$$

Якщо амплітуда коливань мала, кути α_1, α_2 також малі. Тому, можна

прийняти $\cos(\alpha) \approx 1$, звідки

$$T_{2x} - T_{1x} = \rho l_2 \cos \alpha_2 - \rho l_1 \cos \alpha_1 = \rho g (l_2 \cos \alpha_2 - l_1 \cos \alpha_1) \approx \rho g (l_2 - l_1) = \rho g \Delta l$$

Якщо прийняти, що $l(x) \approx x$, тоді $\Delta l \approx \Delta x$ і $\Delta T_x \approx \rho g \Delta x$

Вертикальна складова сили натягу компенсуватиметься силою тяжіння, тому першу можна не брати до уваги. Похідна функції - це тангенс кута нахилу дотичної. Таким чином,

$$\operatorname{tg} \alpha = \frac{du}{dx}$$

При малих кутах $\operatorname{tg} \alpha \approx \sin \alpha$, тому

$$\sin \alpha \approx \frac{du}{dx}$$

Горизонтальна складова різниці сил натягу:

$$\Delta T_u \approx \rho g (x_2 \sin \alpha_2 - x_1 \sin \alpha_1) - \text{з урахування } l(x) \approx x.$$

$$\Delta T_u \approx \rho g (x_2 \operatorname{tg} \alpha_2 - x_1 \operatorname{tg} \alpha_1) - \text{з урахуванням } \operatorname{tg} \alpha \approx \sin \alpha.$$

$$\Delta T_u \approx \rho g \left(x_2 \frac{du(x_2)}{dx} - x_1 \frac{du(x_1)}{dx} \right)$$

Відношення

$$\frac{\rho g \left(x_2 \frac{du(x_2)}{dx} - x_1 \frac{du(x_1)}{dx} \right)}{\Delta x} - \text{приблизне значення } \rho g \frac{d}{dx} \left(x \frac{du}{dx} \right).$$

$$\Delta T_u \approx \rho g \frac{d}{dx} \left(x \frac{du}{dx} \right) dx$$

Отже,

- вертикальна складова різниці сил натягу компенсується відповідною різницею значень сили тяжіння;
- нитка рухається за рахунок наявності некомпенсованої горизонтальної різниці сил натягу.

Відповідно, можна розглядати лише деякий горизонтальний рух точки з координатою x з часом.

Згідно з другим законом Ньютона,

$$F = ma = \rho xa$$

$$\frac{\partial}{\partial x}(F) = \frac{\partial}{\partial x}(\rho xa) = \rho a$$

Прискорення – друга похідна за часом. Тому:

$$a = \frac{\partial^2 u(x,t)}{dt^2} \text{ – прискорення руху нитки в горизонтальній площині}$$

$$\frac{\partial}{\partial x}(F) = \rho \frac{\partial^2 u(x,t)}{dt^2}$$

$$\rho g \frac{\partial}{\partial x} \left(x \frac{\partial u}{\partial x} \right) = \rho \frac{\partial^2 u(x,t)}{dt^2}$$

– похідна сили, що діє на нитку, у разі вільних коливань

$$\text{Скоротимо на } \rho : g \frac{\partial}{\partial x} \left(x \frac{\partial u}{\partial x} \right) = \frac{\partial^2 u(x,t)}{dt^2}$$

$$\frac{\partial}{\partial x} \left(x \frac{\partial u}{\partial x} \right) = \frac{1}{g} \frac{\partial^2 u(x,t)}{dt^2}$$

Отже, отримаємо диференціальне рівняння, як математичну модель:

$$\frac{\partial}{\partial x} \left(x \frac{\partial u}{\partial x} \right) = \frac{1}{a^2} \frac{\partial^2 u(x,t)}{dt^2}, \text{ де } a = \sqrt{g} \quad (1.1)$$

1.2 Розв'язання рівняння коливань

Задача про коливання підвішеної нитки зводиться до інтегрування рівняння (1.1) з крайовою умовою

$$u|_{x=l} = 0$$

з початковими умовами

$$u|_{t=0} = f(x), \quad \frac{\partial u}{\partial t}|_{t=0} = F(x).$$

Зробимо заміну

$$\xi = \sqrt{x}$$

$$\frac{1}{4\xi} \frac{\partial}{\partial \xi} \left(\xi \frac{\partial u}{\partial \xi} \right) = \frac{1}{a^2} \frac{\partial^2 u}{\partial t^2}.$$

Знайдемо розв'язок цього рівняння у вигляді

$$u = \omega(\xi)T(t).$$

Отримаємо

$$\frac{1}{\xi\omega(\xi)} \cdot \frac{d}{d\xi} = \left(\frac{2}{a}\right)^2 \frac{T''(t)}{T(t)}$$

Обидві частини рівняння позначимо через $-\lambda^2$

$$\frac{d}{d\xi} \left(\xi \frac{d\omega}{d\xi} \right) + \lambda^2 \xi \omega = 0,$$

$$T''(t) + \left(\frac{a\lambda}{2}\right)^2 T(t) = 0.$$

Загальний розв'язок рівняння $\frac{d}{d\xi} \left(\xi \frac{d\omega}{d\xi} \right) + \lambda^2 \xi \omega = 0$ матиме вигляд:

$$\omega(\xi) = C_1 J_0(\lambda\xi) + C_2 Y_0(\lambda\xi).$$

Так як $Y_0(\xi) \rightarrow \infty$ при $\xi \rightarrow 0 \Rightarrow C_2 = 0$

Підставляючи до крайової умови, отримаємо

$$J_0(\lambda\sqrt{l}) = 0.$$

Власні значення задачі матимуть вигляд: $\lambda_k^2 = \frac{\mu_k^2}{l}$ ($k = 1, 2, 3, \dots$).

Відповідні власні функції матимуть вигляд: $\omega_k(x) = J_0(\mu_k \sqrt{\frac{x}{l}})$.

Загальний розв'язок рівняння $T''(t) + \left(\frac{a\lambda}{2}\right)^2 T(t) = 0$ матиме вигляд:

$$T_k(t) = A_k \cos \frac{a\mu_k t}{2\sqrt{l}} + B_k \sin \frac{a\mu_k t}{2\sqrt{l}}$$

Тоді, розв'язок рівняння $\frac{\partial}{\partial x} \left(x \frac{\partial u}{\partial x} \right) = \frac{1}{a^2} \frac{\partial^2 u(x,t)}{\partial t^2}$ при крайовій умові $u|_{x=l} = 0$ можна записати у вигляді ряду

$$u(x,t) = \sum_{k=1}^{\infty} \left(A_k \cos \frac{a\mu_k t}{2\sqrt{l}} + B_k \sin \frac{a\mu_k t}{2\sqrt{l}} \right) J_0 \left(\mu_k \sqrt{\frac{x}{l}} \right)$$

Визначимо коефіцієнти розкладання A_k і B_k

$$f(x) = \sum_{k=1}^{\infty} A_k J_0\left(\mu_k \sqrt{\frac{x}{l}}\right)$$

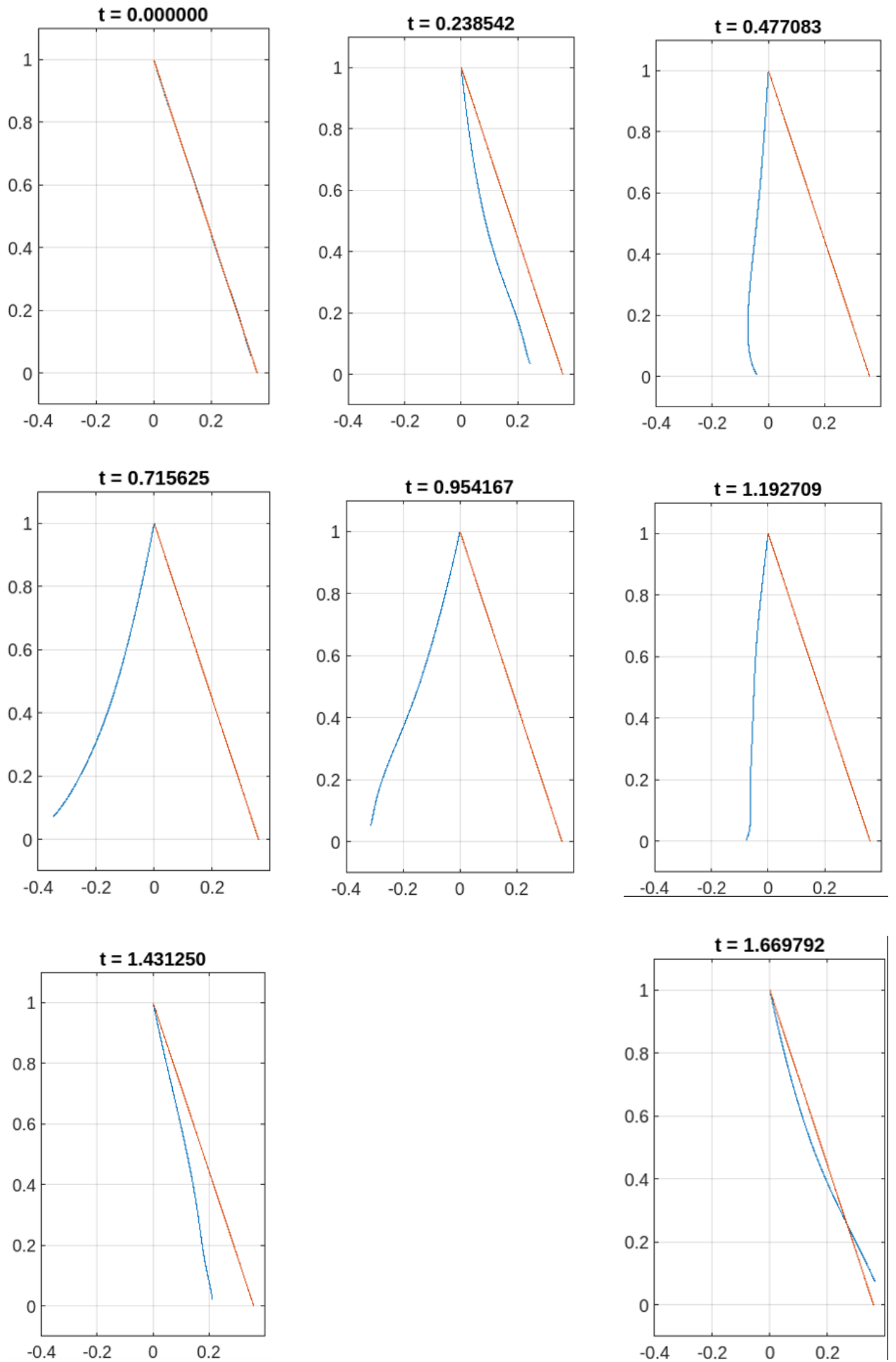
$$A_k = \frac{1}{l J_1^2(\mu_k)} \int_0^1 f(x) J_0\left(\mu_k \sqrt{\frac{x}{l}}\right) dx \quad (1.2)$$

$$B_k = \frac{2}{a \sqrt{l} \mu_k J_1^2(\mu_k)} \int_0^1 f(x) J_0\left(\mu_k \sqrt{\frac{x}{l}}\right) dx \quad (1.3)$$

1.3 Результати моделювання

На графіках (*рисунок 3*) можна спостерігати коливання нитки в залежності від часу. Кожен графік показує форму нитки у певний момент часу t . Синя крива представляє форму нитки у момент часу t , червона пряма - початкову форму нитки. Видно, що амплітуда коливань змінюється з часом.

При збільшенні за модулем кутового коефіцієнта, збільшенні початкового нахилу нитки ($k = -0.66$), спостерігається збільшення амплітуди коливань (*рисунок 4*).

Рис.3 - Зміни положення нитки з часом при $k = -0.36$

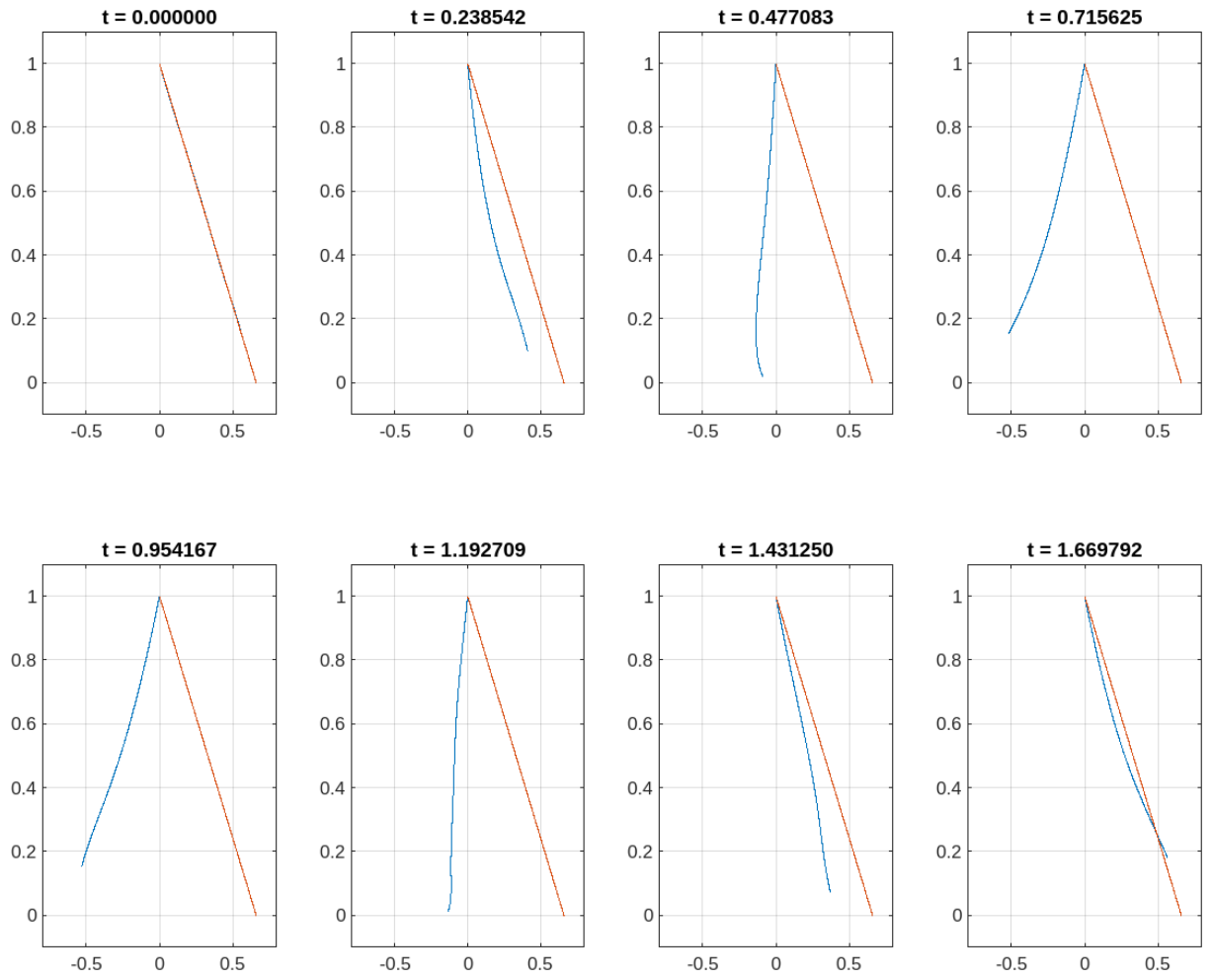


Рис. 4 - Зміни положення нитки з часом при $k = -0.66$

РОЗДІЛ 2

ВИМУШЕНІ КОЛИВАННЯ ВАЖКОЇ ОДНОРІДНОЇ ГНУЧКОЇ НИТКИ

2.1 Отримання рівняння коливань

У разі вимушених коливань на нитку діє зовнішня сила. Знайдемо проєкції зовнішньої сили на вісь OU , і, в результаті, отримаємо функцію, $F_{\text{зовн}}$ що відповідає горизонтальній компоненті зовнішньої сили в точці x . Отже, отримаємо рівнодіючу горизонтальної компоненти сили натягу та горизонтальну компоненту зовнішньої сили - $F_{\text{зовн}}$. Під цією рівнодіючою нитка чинить коливання.

$$F_{\text{зовн}}(x) + T(x) \sin \alpha = ma$$

$$d(F_{\text{зовн}}(x)) + \rho g \frac{d}{dx} \left(x \frac{du}{dx} \right) = d(ma)$$

$$\frac{d}{dx} (F_{\text{зовн}}(x)) + \rho g \frac{d}{dx} \left(x \frac{du}{dx} \right) = \frac{d}{dx} (ma)$$

$$\frac{d}{dx} (F_{\text{зовн}}(x)) + \rho g \frac{d}{dx} \left(x \frac{du}{dx} \right) = \rho \frac{\partial^2 u(x,t)}{dt^2}$$

Зауважимо: $\rho g \frac{d}{dx} \left(x \frac{du}{dx} \right)$ і $\rho \frac{\partial^2 u(x,t)}{dt^2}$ - похідні від сили.

Нехай, похідна по x від сили $F_{\text{зовн}}(x)$ - це $\Phi(x)$. Тоді:

$$\frac{\partial F_{\text{зовн}}}{\partial x} = \Phi(x)$$

$$\Phi(x) + \rho g \frac{d}{dx} \left(x \frac{du}{dx} \right) = \rho \frac{\partial^2 u(x,t)}{dt^2}$$

Розділимо все на ρ :

$$\frac{\Phi(x)}{\rho} + g \frac{d}{dx} \left(x \frac{du}{dx} \right) = \frac{\partial^2 u(x,t)}{dt^2}$$

$$\frac{\partial^2 u(x,t)}{dt^2} = g \frac{\partial}{\partial x} \left(x \frac{\partial u}{\partial x} \right) + Y(x,t)$$

Перепозначимо g як a^2 :

$$\frac{\partial^2 u(x,t)}{dt^2} = a^2 \frac{\partial}{\partial x} \left(x \frac{\partial u}{\partial x} \right) + Y(x,t), \text{ де } Y(x,t) = \frac{\Phi(x)}{\rho} \quad (2.1)$$

Крайова умова:

$$u|_{x=l} = 0, \text{ де } l - \text{ довжина нитки.} \quad (2.2)$$

Початкові умови:

$$u|_{t=0} = f(x) - \text{ початкова форма нитки} \quad (2.3)$$

$$\frac{\partial u}{\partial t}|_{t=0} = F(x) - \text{ початкові швидкості точок нитки.} \quad (2.4)$$

2.2 Розв'язання рівняння коливань

Розв'язок задачі будемо шукати у вигляді суми:

$$u = u_1 + u_2,$$

де $u|_{t=0}$ – розв'язок неоднорідного рівняння (2.1), який задовольняє крайовій умові (2.2) і нульовим початковим умовам $u_1|_{t=0}, \frac{\partial u_1}{\partial t}|_{t=0} = 0$, а функція $u_2(x,t)$ – розв'язок однорідного рівняння

$$\frac{\partial^2 u}{dt^2} = a^2 \frac{\partial}{\partial x} \left(x \frac{\partial u}{\partial x} \right), \quad (2.5)$$

який задовольняє крайовій умові (2.2) та початковим умовам.

Отримаємо розв'язок задачі (2.2-2.5):

Похідна складеної функції - це похідна зовнішньої функції за внутрішньою функцією, помножена на похідну внутрішньої функції за незалежною змінною:

$$\frac{df(\xi)}{dx} = \frac{df(\xi)}{d\xi} \frac{d\xi}{dx} = \frac{df(x)}{dx}$$

Отримаємо

$$\sum_{k=1}^{\infty} [T_k'' + \omega_k^2 T_k] J_0\left(\mu_k \sqrt{\frac{x}{l}}\right) = Y(x, t), \text{ де } \omega_k = \frac{\mu_k a}{2\sqrt{l}} \quad (2.7)$$

Розкладемо функцію $Y(x, t)$ в ряд по власним функціям $J_0\left(\mu_k \sqrt{\frac{x}{l}}\right)$, тобто покладемо

$$Y(x, t) = \sum_{k=1}^{\infty} H_k(t) J_0\left(\mu_k \sqrt{\frac{x}{l}}\right). \quad (2.8)$$

Цей розклад за видом співпадає з розкладом функції у ряд Бесселя-Фур'є. Тоді, коефіцієнти $H_k(t)$ визначаються за формулою

$$H_k(t) = \frac{1}{l J_1^2(\mu_k)} \int_0^l Y(\xi, t) J_0\left(\mu_k \sqrt{\frac{\xi}{l}}\right) d\xi \quad (2.9)$$

Порівнюючи розклади (2.7) і (2.8) для однієї і тієї ж функції $Y(x, t)$, отримаємо рівняння

$$T_k''(t) + \omega_k^2 T_k(t) = H_k(t),$$

якому повинні задовольняти коефіцієнти $T_k(t)$. За таким визначенням коефіцієнтів $T_k(t)$, функція (2.8) задовольняє диференціальне рівняння (2.1) і крайовій умові (2.2). Для задовільнення початкових умов (2.3), (2.4) достатньо підпорядкувати функції $T_k(t)$ умовам

$$T_k(t) = \frac{1}{\omega_k} \int_0^t H_k(\tau) \sin \omega_k(t - \tau) d\tau \quad (2.10)$$

Підставимо вираз (2.9) для $H_k(\tau)$:

$$T_k(t) = \frac{1}{l \omega_k J_1^2(\mu_k)} \int_0^t d\tau \int_0^l Y(\xi, \tau) J_0\left(\mu_k \sqrt{\frac{\xi}{l}}\right) \sin \omega_k(t - \tau) d\xi.$$

Звідси випливає, що відхилення підвішеної нитки від вертикального положення рівноваги виражається формулою

$$u(x,t) = \sum_{k=1}^{\infty} T_k(t) J_0\left(\mu \sqrt{\frac{x}{l}}\right) + \sum_{k=1}^{\infty} \left(A_k \cos \frac{a\mu_k t}{2\sqrt{l}} + B_k \sin \frac{a\mu_k t}{2\sqrt{l}} \right) J_0\left(\mu_k \sqrt{\frac{x}{l}}\right) \quad (2.11)$$

де коефіцієнт $T_k(t)$, A_k і B_k визначаються рівностями (2.10), (1.2) і (1.3), а $\mu_1, \mu_2, \mu_3, \dots$ — додатні корені рівняння $J_0(\mu) = 0$.

Розглянемо випадок, коли зовнішня сила діє гармонічно, тобто

$$Y(x,t) = A \sin \omega t.$$

У цьому випадку коефіцієнти $T_k(t)$ визначаються за формулою

$$T_k(t) = \frac{A}{l\omega_k J_1(\mu_k)} \int_0^t \sin \omega_k(t - \eta) \sin \omega \eta d\eta \int_0^l J_0\left(\mu_k \sqrt{\frac{\xi}{l}}\right) d\xi$$

За допомогою формули $\int_0^x J_0(\sqrt{x}) dx = 2\sqrt{x} J_1(\sqrt{x})$, знайдемо, що

$$\int_0^l J_0\left(\mu_k \sqrt{\frac{\xi}{l}}\right) d\xi = \frac{2l}{\mu_k} J_1(\mu_k)$$

Так як $\int_0^t \sin \omega_k(t - \tau) \sin \omega \tau d\tau = \frac{\omega_k \sin \omega t}{\omega_k^2 - \omega^2}$,

Тоді

$$T_k(t) = 4\sqrt{\frac{l}{g}} \cdot \frac{A}{\mu_k^2 J_1(\mu_k) \left[\frac{\omega_k \sin \omega t}{\omega_k^2} - \frac{\omega \sin \omega_k t}{\omega_k^2 - \omega^2} \right]} \quad (2.12)$$

Припустимо, що початкові відхилення і початкові швидкості в даному випадку відсутні і нитка коливається тільки внаслідок збудження. Тоді з формули (2.11) та (2.12) випливає, що відхилення нитки від вертикального положення рівноваги буде виражатися формулою

$$u(x,t) = 4\sqrt{\frac{l}{g}} A \sin \omega t \sum_{k=1}^{\infty} \frac{\omega_k J_0\left(\mu_k \sqrt{\frac{x}{l}}\right)}{(\omega_k^2 - \omega^2) \mu_k^2 J_1(\mu_k)} - 4A\omega \sqrt{\frac{l}{g}} \sum_{k=1}^{\infty} \frac{J_0\left(\mu_k \sqrt{\frac{x}{l}}\right) \sin \omega_k t}{(\omega_k^2 - \omega^2) \mu_k^2 J_1(\mu_k)}. \quad (2.13)$$

Знайдемо розв'язок рівняння

$$\frac{\partial^2 u}{\partial t^2} = a^2 \frac{\partial}{\partial x} \left(x \frac{\partial u}{\partial x} \right) + A \sin \omega t, \quad (2.14)$$

яке задовольняє умовам

$$u|_{x=0} = \text{кінцева величина}, u|_{x=l} = 0, \quad (2.15)$$

у вигляді добутку

$$u = X(x) \sin \omega t. \quad (2.16)$$

Підставимо (2.16) у рівняння (2.14):

$$\frac{d}{dx} \left(x \frac{dX}{dx} \right) + \left(\frac{\omega}{a} \right)^2 X + \frac{A}{\omega^2} \quad (2.17)$$

Враховуючи крайові умови (1.20)

$$C_1 = \frac{A}{\omega^2} \frac{1}{J_0 \left(1\omega \sqrt{\frac{l}{g}} \right)}, C_2 = 0. \quad (2.18)$$

Таким чином, розв'язок (2.13) має вигляд:

$$u(x,t) = \frac{A}{\omega^2} \left[\frac{J_0 \left(2\omega \sqrt{\frac{x}{g}} \right)}{J_0 \left(2\omega \sqrt{\frac{l}{g}} \right)} \right] \sin \omega t - \\ - 4A\omega \sqrt{\frac{l}{g}} \sum_{k=1}^{\infty} \frac{J_0 \left(\mu_k \sqrt{\frac{x}{l}} \right) \sin \omega_k t}{(\omega_k^2 - \omega^2) \mu_k^2 J_1(\mu_k)} \quad (2.19)$$

2.3 Результати моделювання

Можна побачити, що за однакової довжини нитки ($l = 1$), початкового положення нитки (вздовж осі OX), початкових швидкостей точок нитки (дорівнюють нулеві) та амплітуди зовнішніх коливань ($A = 1$), змінюючи тільки кутову частоту, спостерігаємо зміну амплітуди коливань нитки.

На *рисунку 5* коливання здійснюються із зовнішньою кутовою частотою $\omega = 1$.

На *рисунку 6* кутова частота коливань збільшується до 2π радіан в секунду ($\omega = 2\pi$) \equiv 1Гц, тобто зовнішні коливання відбуваються значно швидше.

При цьому, у випадку резонансу, частота зовнішніх коливань співпадає з частотою першої гармоніки ($\omega = 3.76$), значення функції $u(x,t)$ дорівнюють нескінченності, тобто амплітуда прагне до нескінченності.

У випадку обрання частоти близької до резонансної ($\omega = \omega_1 - 0.1 * \omega_1$), можна спостерігати значне збільшення амплітуди коливання нитки, що проілюстровано на результатах моделювання *рисунок 7*.

Роблячи висновок, можна стверджувати, що при незмінній амплітуді зовнішніх коливань, амплітуда коливань нитки значно зростає з наближенням значення кутової частоти зовнішніх коливань до резонансної.

Код програми наведено у додатку Б.

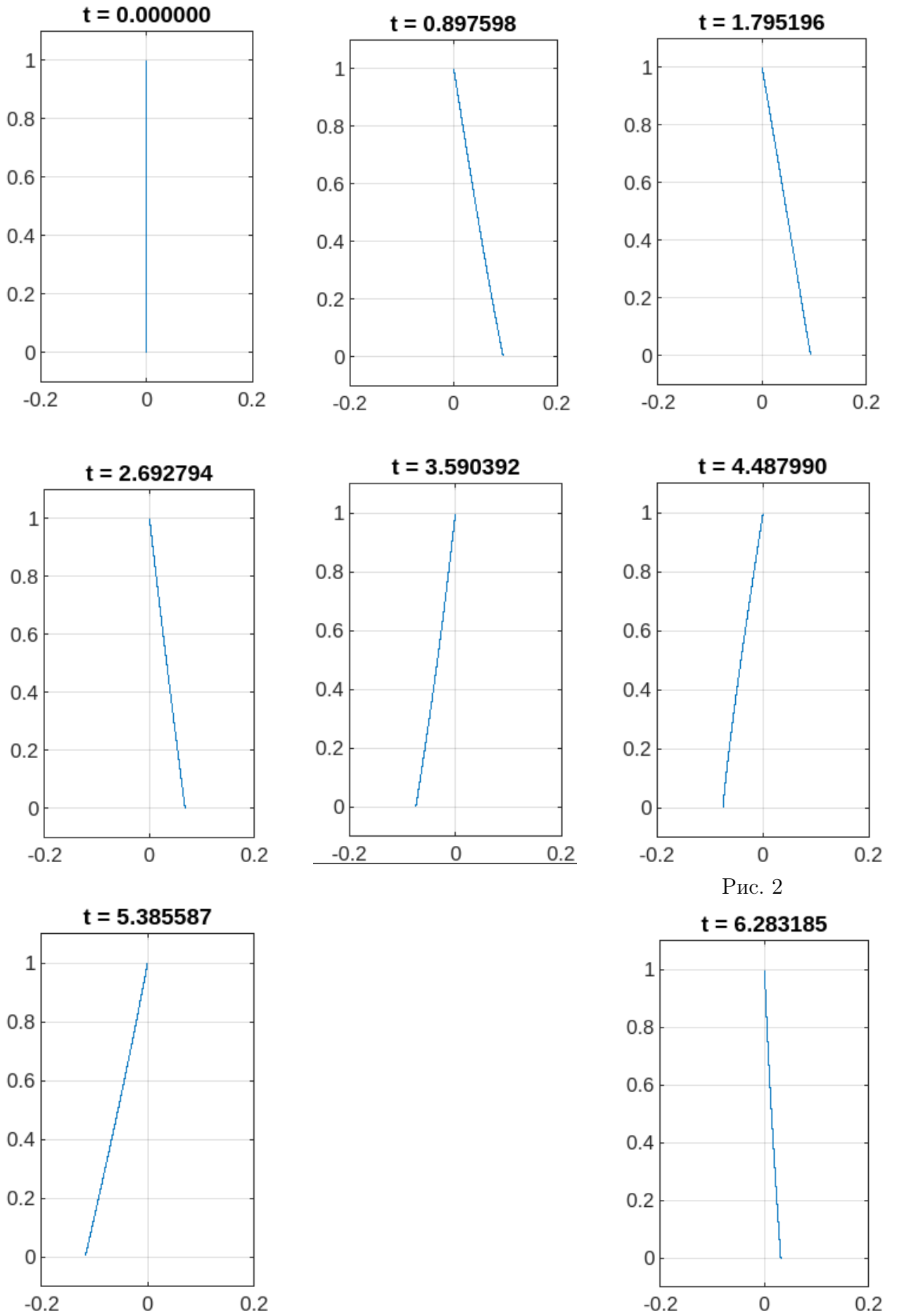
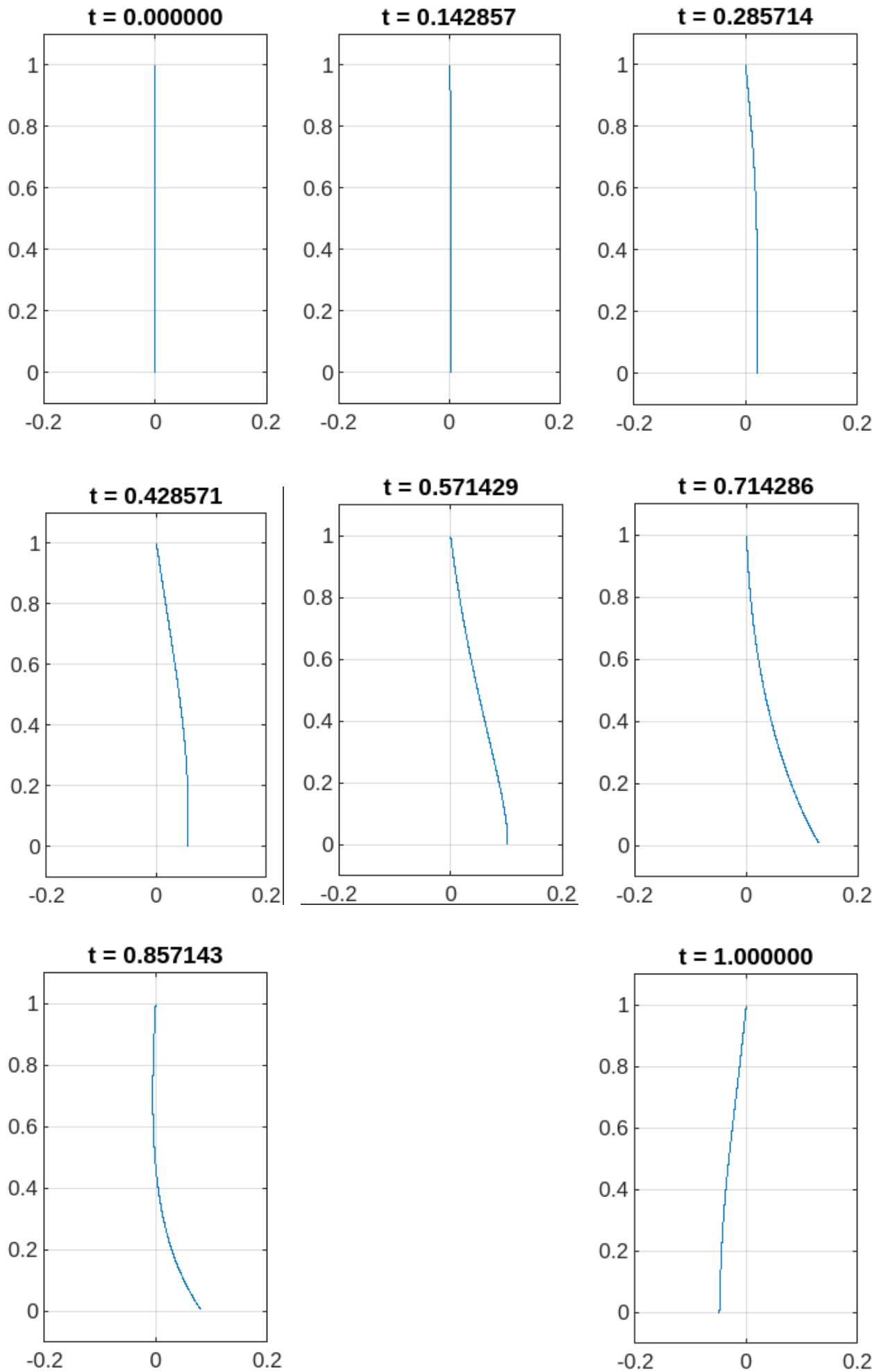


Рис. 2

Рис. 5 - Вимушені коливання нитки при $\omega = 1$

Рис. 6 - Вимушені коливання нитки при $\omega = 2\pi$

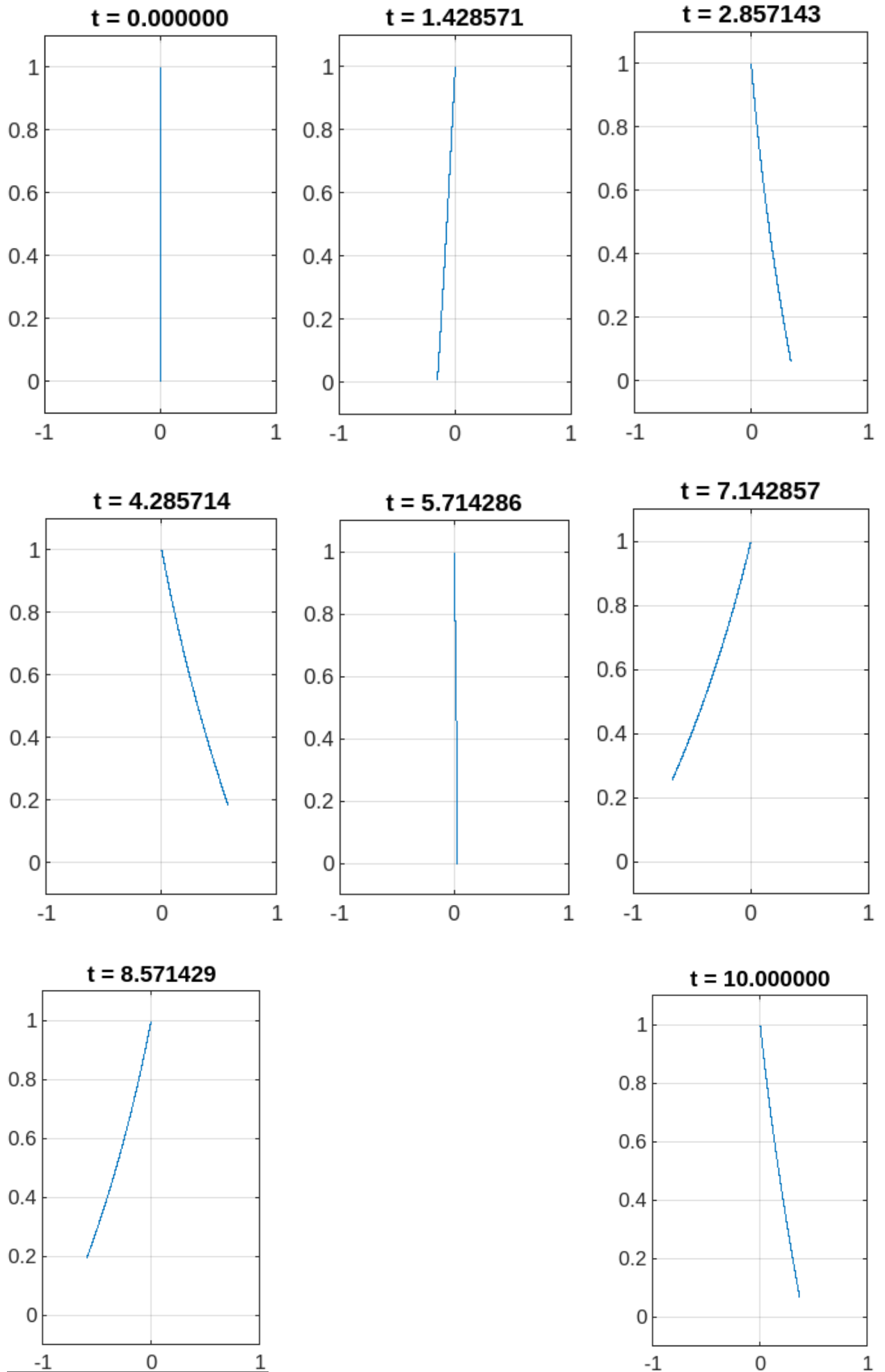


Рис. 7 - Вимушені коливання нитки при $\omega = \omega_1 - 0.1 * \omega_1$

РОЗДІЛ 3

СКЛАДНІ КОЛИВАННЯ ВАЖКОЇ ОДНОРІДНОЇ
ГНУЧКОЇ НИТКИ

3.1 Отримання рівняння коливань

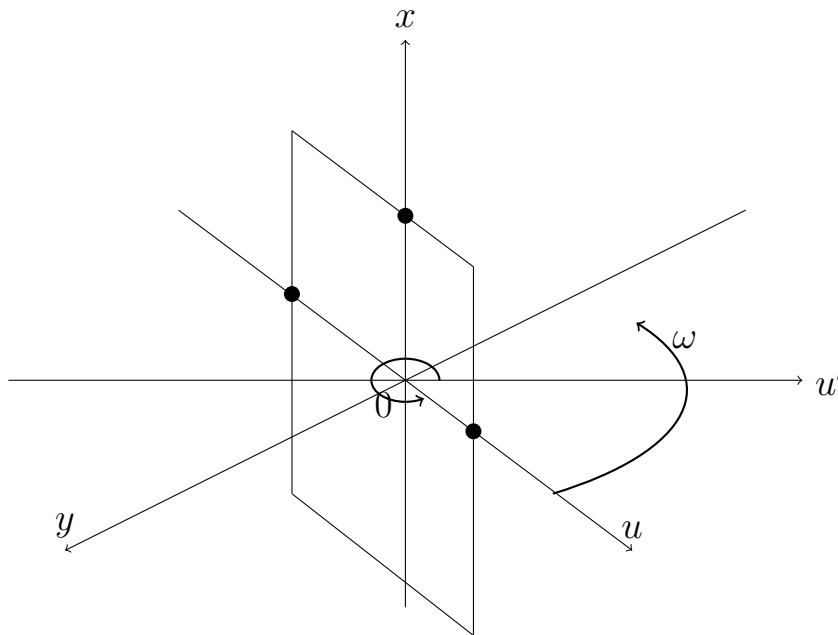


Рис.8

Коливання нитки здійснюються у площині XOU . У випадку вимушених коливань маємо зовнішню силу та неоднорідне диференціальне рівняння. Обертання нитки розглядатиметься, як обертання площини XOU навколо осі X у нерухомій системі координат $XU'Y$. Внаслідок обертання виникає доцентрове прискорення та тангенціальне прискорення. Зовнішньою силою буде доцентрове прискорення - $a_n = \omega^2 R$, R – радіус – $u(x,t)$. Використовуючи рівняння, отримане при розгляданні вимушених коливань:

$$\frac{\partial^2 u(x,t)}{\partial t^2} = g \frac{\partial}{\partial x} \left(x \frac{\partial u}{\partial x} \right) + Y(x,t), Y(x,t) = \frac{\Phi(x)}{\rho}$$

$$\Phi(x) = \frac{\partial F_n(x)}{\partial x} = \frac{\partial}{\partial x} m a_n = \frac{\partial}{\partial x} \rho x a_n = \rho a_n$$

$$Y(x,t) = a_n = \omega^2 R = \omega^2 u$$

$$\frac{\partial^2 u(x,t)}{dt^2} = g \frac{\partial}{\partial x} \left(x \frac{\partial u}{\partial x} \right) + \omega^2 u \quad (3.1)$$

3.2 Розв'язання рівняння коливань

Використовуючи отримане раніше, розв'язок рівняння при крайовій умові $u|_{x=l} = 0$ можна записати у вигляді ряду

$$u(x,t) = \sum_{k=1}^{\infty} \left(A_k \cos a \sqrt{\frac{\mu_k^2}{4l} - \left(\frac{w}{a}\right)^2} + B_k \sin a \sqrt{\frac{\mu_k^2}{4l} - \left(\frac{w}{a}\right)^2} \right) J_0 \left(\mu_k \sqrt{\frac{x}{l}} \right),$$

$$A_k = \frac{1}{l J_1^2(\mu_k)} \int_0^1 f(x) J_0 \left(\mu_k \sqrt{\frac{x}{l}} \right) dx$$

$$B_k = \frac{2}{a \sqrt{l} \mu_k J_1^2(\mu_k)} \int_0^1 f(x) J_0 \left(\mu_k \sqrt{\frac{x}{l}} \right) dx$$

3.3 Результати моделювання

Код програми наведено у додатку В

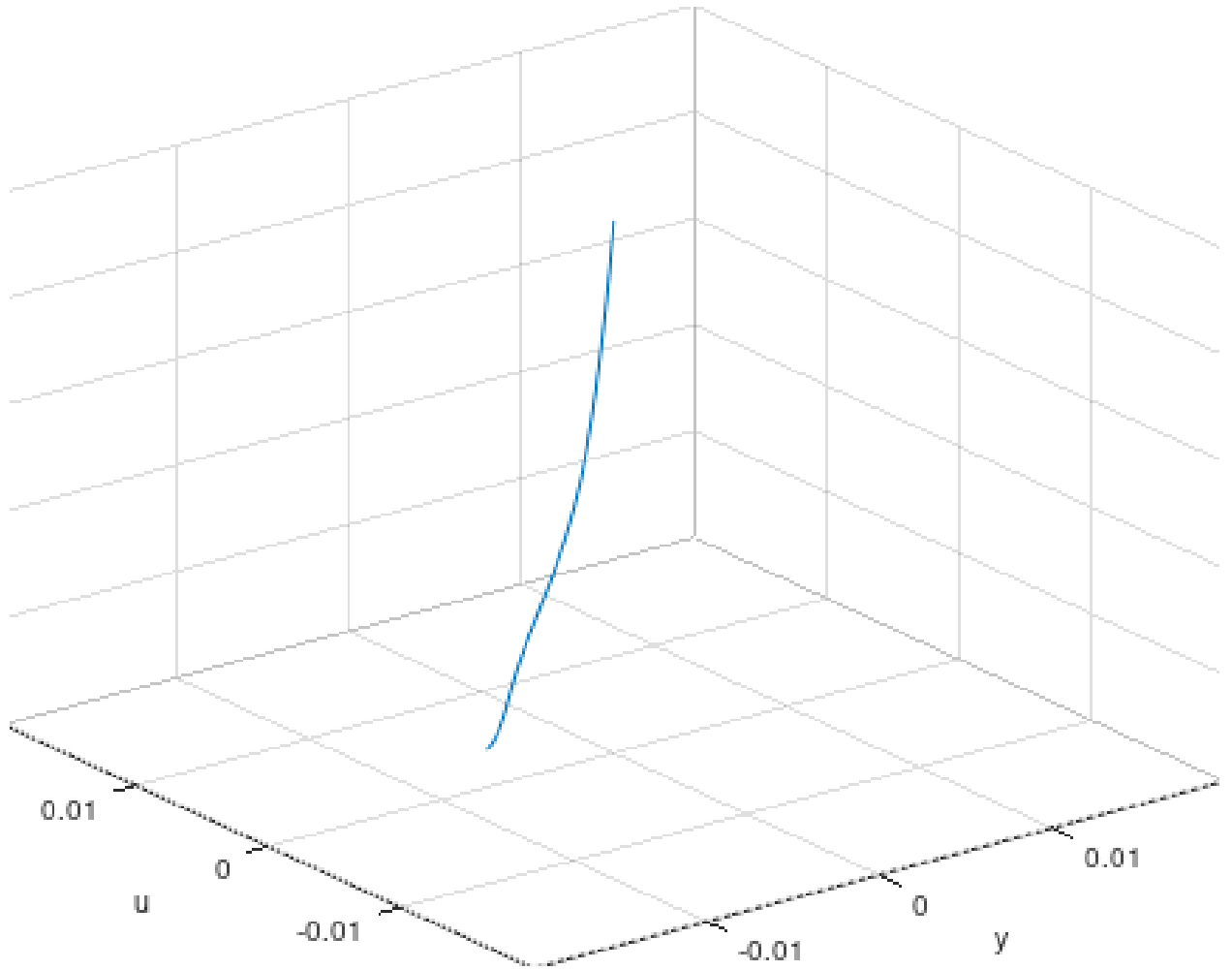


Рис.9 - Кадр анімації складних коливань нитки

Вихідний код програми для моделювання вільних коливань нитки

```

1 X_MAX = 20;
2 dx = 1e-3;
3 xv = 0:dx:X_MAX;
4 yv = besselj(0, xv);
5
6 N = length(xv);
7 bessel_roots = [];
8
9 for k = 1:N-1
10     if yv(k) * yv(k + 1) <= 0
11         bessel_roots = [bessel_roots xv(k)];
12     end
13 end
14
15 ROOTS_COUNT = length(bessel_roots);
16
17 function result = fun_f(x, k, b)
18     result = k .* x + b;
19 end
20
21 function result = get_ak(n, l, dx, bessel_roots, k_coef,
22     b)
23     result = 0;
24     mk = bessel_roots(n);
25     for x = 0:dx:l-dx
26         result = result + fun_f(x, k_coef, b) .* besselj(0,
27             mk .* sqrt(x ./ l)) .* dx;
28     end
29     result = result ./ (1 .* besselj(1, mk).^2);
30 end

```

```

30 function result = get_item(x, l, dx, bessell_roots, akv, t
    , a)
31     result = 0;
32     n_max = length(bessell_roots);
33
34     for n = 1:n_max
35         mk = bessell_roots(n);
36         result = result + akv(n) * besselj(0, mk * sqrt(x / l
            )) * ...
37         sin(a * mk * t / (2 * sqrt(l)) + pi/2);
38     end
39 end
40
41 function result = get_expansion(xv, l, dx, bessell_roots,
    t, a, k_coef, b)
42     result = [];
43     n_max = length(bessell_roots);
44     akv = get_ak(1:n_max, l, dx, bessell_roots, k_coef, b
        );
45     for x = xv
46         result = [result get_item(x, l, dx, bessell_roots, akv
            , t, a)];
47     endfor
48 end
49
50 function [xn, yn] = get_end(x, y, dx)
51     s = 0;
52     n = length(y);
53     index = 0;
54     for k = 1:n-1
55         yn2 = y(n - k + 1);
56         yn1 = y(n - k);
57         dy = abs(yn2 - yn1);
58         dl = sqrt(dy^2 + dx^2);

```

```

59     s = s + dl;
60     if s >= 1.0
61         index = n - k;
62         break;
63     end
64 end
65 xn = x(index:n);
66 yn = y(index:n);
67 end
68
69 l = 1;
70 dx = 0.001;
71 xv = 0:dx:l;
72 k_coef = -0.36;
73 b = -k_coef;
74 a = sqrt(9.8);
75
76 f0 = fun_f(xv, k_coef, b);
77
78 T = 4 * pi * sqrt(l) / (a * bessell_roots(1));
79
80 figure();
81
82 k = 1;
83 for t = 0:T/7:T
84     y = get_expansion(xv, l, dx, bessell_roots, t,
85         a, k_coef, b);
86     [xn, yn] = get_end(xv, y, dx);
87
88     subplot(2, 4, k);
89     plot(yn, xn, f0, xv);
90     ylim([-0.1 1.1]);
91     xlim([-0.4 0.4]);
92     grid on;

```

```
93     title(sprintf('%s%f', 't = ', t));
94
95     k = k + 1;
96 end
```

Вихідний код програми для моделювання вимушених коливань нитки

```

1
2 X_MAX = 20;
3 dx = 1e-3;
4 xv = 0:dx:X_MAX;
5 yv = besselj(0, xv);
6
7
8 N = length(xv);
9 bessell_roots = [];
10
11 for k = 1:N-1
12     if yv(k) * yv(k + 1) <= 0
13         bessell_roots = [bessell_roots xv(k)];
14     end
15 end
16
17 ROOTS_COUNT = length(bessell_roots);
18
19 function result = get_u(w, x, g, l, A, t, bessell_roots)
20
21     k_max = length(bessell_roots);
22
23     p1 = 2 * w * sqrt(x / g);
24     p2 = 2 * w * sqrt(l / g);
25     p3 = besselj(0, p1);
26     p4 = besselj(0, p2);
27     p5 = (p3 / p4 - 1) * A / w^2;
28     p6 = p5 * sin(w*t);
29
30     s = 0;
31     for k = 1:k_max

```

```

32     p7 = bessell_roots(k) * sqrt(x / l);
33     wk = bessell_roots(k) * sqrt(g) / (2 * sqrt(l));
34     p8 = bessellj(0, p7) * sin(wk*t);
35     p9 = (wk^2 - w^2) * bessell_roots(k)^2 * bessellj(1,
        bessell_roots(k));
36     s = s + p8 / p9;
37 end
38
39     p10 = s * 4 * A * w * sqrt(1 / g);
40     result = p6 - p10;
41 end
42
43 function result = get_expansion(xv, w, g, l, A, t,
        bessell_roots, dx)
44     result = [];
45     for x = xv
46         result = [result get_u(w, x, g, l, A, t, bessell_roots
            )];
47     end
48 end
49
50 function [xn, yn] = get_end(x, y, dx)
51     s = 0;
52     n = length(y);
53     index = 0;
54     for k = 1:n-1
55         yn2 = y(n - k + 1);
56         yn1 = y(n - k);
57         dy = abs(yn2 - yn1);
58         dl = sqrt(dy^2 + dx^2);
59         s = s + dl;
60         if s >= 1.0
61             index = n - k;
62             break;

```

```

63     end
64 end
65 xn = x(index:n);
66 yn = y(index:n);
67 end
68
69 l = 1;
70 dx = 0.001;
71 xv = 0:dx:l;
72 g = 9.8;
73
74 A = 1;
75
76 T1 = 4 * pi * sqrt(l) / (sqrt(g) * bessel_roots(1));
77
78 figure();
79
80 w = 1;
81 f = w / (2 * pi);
82 T = 1 / f;
83
84 k = 1;
85 for t = 0:T/7:T
86     y = get_expansion(xv, w, g, l, A, t, bessel_roots, dx);
87     [xn, yn] = get_end(xv, y, dx);
88
89     subplot(2, 4, k);
90     plot(yn, xn);
91     ylim([-0.1 1.1]);
92     xlim([-0.2 0.2]);
93     grid on;
94     title(sprintf('%s%f', 't = ', t));
95
96     k = k + 1;

```

```

97 end
98
99 % w = 2*pi
100 figure ();
101
102 w = 2*pi;
103 T = 1;
104
105 k = 1;
106 for t = 0:T/7:T
107     y = get_expansion(xv, w, g, l, A, t, bessell_roots, dx);
108     [xn, yn] = get_end(xv, y, dx);
109
110     subplot(2, 4, k);
111     plot(yn, xn);
112     ylim([-0.1 1.1]);
113     xlim([-0.2 0.2]);
114     grid on;
115     title(sprintf('%s%f', 't = ', t));
116
117     k = k + 1;
118 end
119
120 figure ();
121
122 w1 = sqrt(g) * bessell_roots(1) / (sqrt(1) * 2);
123
124 w = w1;
125 T = 10;
126
127 k = 1;
128 for t = 0:T/7:T
129     y = get_expansion(xv, w, g, l, A, t, bessell_roots, dx);
130

```

```

131 subplot(2, 4, k);
132 plot(y, xv);
133 ylim([-0.1 1.1]);
134 grid on;
135 title(sprintf('%s%f', 't = ', t));
136
137 k = k + 1;
138 end
139
140 figure();
141
142 w1 = sqrt(g) * bessell_roots(1) / (sqrt(1) * 2);
143
144 w = w1 - w1 * 0.1;
145 T = 10;
146
147 k = 1;
148 for t = 0:T/7:T
149     y = get_expansion(xv, w, g, l, A, t, bessell_roots, dx);
150     [xn, yn] = get_end(xv, y, dx);
151
152     subplot(2, 4, k);
153     plot(yn, xn);
154     ylim([-0.1 1.1]);
155     xlim([-1 1]);
156     grid on;
157     title(sprintf('%s%f', 't = ', t));
158
159     k = k + 1;
160 end
161
162 figure();
163 t = 0.5;
164 m = 500;

```

```

165 r = [];
166
167 for w = 0:0.1:8*pi
168     y = get_expansion(xv, w, g, l, A, t, bessell_roots, dx);
169     r = [r y(m)];
170 end
171
172 w = 0:0.1:8*pi;
173 plot(w, r)
174 ylabel('u(0.5,0.5)');
175 xlabel('w');
176
177 w = 1;
178
179 fn = 'oscillations.gif'
180
181 for t = 0:0.1:10
182     y = get_expansion(xv, w, g, l, A, t, bessell_roots, dx
183     );
184     [xn, yn] = get_end(xv, y, dx);
185     plot(yn, xn);
186     ylim([-0.1 1.1]);
187     xlim([-0.2 0.2]);
188     text(-0.1, 1, sprintf('%s%f', 't = ', t))
189     grid on;
190
191     frame = getframe;
192     image = frame2im(frame);
193     [X, cmap] = rgb2ind(image);
194
195     if t == 0
196         imwrite(X, cmap, fn, 'gif', 'LoopCount', Inf, ...
197             'DelayTime', 1/10);
198     else

```

```
198         imwrite(X, cmap, fn, 'gif', 'WriteMode', 'append'  
199             , ...  
200             'DelayTime', 1/10);  
201     end  
end
```

Вихідний код програми для моделювання складних коливань нитки

```

1
2 X_MAX = 20;
3 dx = 1e-3;
4 xv = 0:dx:X_MAX;
5 yv = besselj(0, xv);
6
7 N = length(xv);
8 bessel_roots = [];
9
10 for k = 1:N-1
11     if yv(k) * yv(k + 1) <= 0
12         bessel_roots = [bessel_roots xv(k)];
13     endif
14 end
15
16 ROOTS_COUNT = length(bessel_roots);
17
18 function result = fun_f(x, k, b)
19     result = k .* x + b;
20 end
21
22 function result = get_ak(n, l, dx, bessel_roots, k_coef,
23     b)
24     result = 0;
25     mk      = bessel_roots(n);
26     for x = 0:dx:l-dx
27         result = result + fun_f(x, k_coef, b) .* besselj(0,
28             mk .* sqrt(x ./ l)) .* dx;
29     endfor
30     result = result ./ (1 .* besselj(1, mk).^2);
31 end

```

```

30
31 function result = get_u(x, l, dx, bessell_roots, akv, t, a
    , w)
32     result = 0;
33     n_max = length(bessell_roots);
34
35     for n = 1:n_max
36         mk = bessell_roots(n);
37         lk = sqrt(mk^2 / (4*l) - (w/a)^2);
38         result = result + akv(n) * bessellj(0, mk * sqrt(x / l
            )) * cos(a * lk * t);
39     endfor
40 end
41
42 function result = get_uv(xv, l, dx, bessell_roots, t, a,
    k_coef, b, w)
43     result = [];
44     n_max = length(bessell_roots);
45     akv = get_ak(1:n_max, l, dx, bessell_roots, k_coef, b
        );
46     for x = xv
47         result = [result get_u(x, l, dx, bessell_roots, akv, t
            , a, w)];
48     endfor
49 end
50
51 function [xn, yn] = get_end(x, y, dx)
52     s = 0;
53     n = length(y);
54     index = 0;
55     for k = 1:n-1
56         yn2 = y(n - k + 1);
57         yn1 = y(n - k);
58         dy = abs(yn2 - yn1);

```

```

59     dl = sqrt(dy^2 + dx^2);
60     s = s + dl;
61     if s >= 1.0
62         index = n - k;
63         break;
64     endif
65 endfor
66 xn = x(index:n);
67 yn = y(index:n);
68 end
69
70 l = 1;
71 dx = 0.001;
72 xv = 0:dx:l;
73 k_coef = -0.01;
74 b = -k_coef;
75 t = 1;
76 a = sqrt(9.8);
77
78 figure();
79 for w = 0:2*pi/4:2*pi;
80     y = get_uv(xv, l, dx, bessell_roots, t, a, k_coef, b, w)
81         ;
82     [xn, yn] = get_end(xv, y, dx);
83     plot(yn, xn)
84     xlim([-0.02 0.02]);
85     ylim([-0.1 1.1]);
86     hold on;
87 end
88 function [xn, yn, zn] = get_end_3d(x, y, z, dx)
89     s = 0;
90     n = length(z);
91     index = 0;

```

```
92  for k = 1:n-1
93
94      xn2 = x(k + 1);
95      xn1 = x(k);
96
97      yn2 = y(k + 1);
98      yn1 = y(k);
99
100     zn2 = z(k + 1);
101     zn1 = z(k);
102
103     dx = xn2 - xn1;
104     dy = yn2 - yn1;
105     dz = zn2 - zn1;
106
107     d1 = sqrt(dx^2 + dy^2);
108     d2 = sqrt(d1^2 + dz^2);
109
110     r = d2;
111
112     s = s + r;
113
114     if s >= 1.0
115         index = k;
116         break;
117     endif
118 endfor
119 xn = x(1:index);
120 yn = y(1:index);
121 zn = z(1:index);
122 end
123
124 figure ();
125
```

```
126 l = 1;
127 dx = 0.001;
128 xv = 0:dx:l;
129 k_coef = -0.01;
130 b = -k_coef;
131 a = sqrt(9.8);
132
133 w = 1;
134 fn = 'task.gif';
135 dt = 0.1;
136
137 for t = 0:dt:10
138
139     z = xv;
140     r = get_uv(xv, l, dx, bessel_roots, t, a, k_coef, b,
141              w);
142     y = r .* cos(w * t);
143     x = r .* sin(w * t);
144
145     plot3(x, y, z);
146     grid on;
147
148     xlim([-0.02 0.02]);
149     ylim([-0.02 0.02]);
150     zlim([0 1]);
151
152     xlabel('y');
153     ylabel('u');
154     zlabel('x');
155
156     frame = getframe;
157     image = frame2im(frame);
158     [X, cmap] = rgb2ind(image);
159     if t == 0
```

```
159         imwrite(X, cmap, fn, 'gif', 'LoopCount', Inf, '
           DelayTime', dt);
160     else
161         imwrite(X, cmap, fn, 'gif', 'WriteMode', 'append'
           , 'DelayTime', dt);
162     end
163 end
```

ВИСНОВКИ

Отримані рівняння вільних, вимушених та складних коливань.

Розв'язані рівняння вільних, вимушених та складних коливань.

Застосовано метод Фур'є для розв'язання диференціального рівняння.

Застосовано ряд Бесселя-Фур'є для розкладання вихідної функції у відповідний ряд, що дозволило знайти розв'язок диференціального рівняння та отримати чисельні значення відхилень нитки від положення рівноваги з часом.

Розроблено чисельний алгоритм для моделювання коливань нитки, які розглядаються як рух, складений із множини гармонічних коливань. У алгоритмі було використано функції Бесселя нульового та першого порядків, загальний розв'язок рівняння у вигляді ряду.

Змодельовано процес коливання нитки в залежності від часу, частоти, коефіцієнтів форми нитки та їхній вплив на амплітуду та форму коливань.

СПИСОК ЛІТЕРАТУРИ

1. Вайсфельд Н.Д. Рівняння математичної фізики : навч-метод. посібн. для студентів "Прикладна математика" / Н.Д. Вайсфельд, В.В. Реут. - Одеса : Одеськ. нац. ун-т. ім. І.І.Мечникова, 2018 —194 с.
2. Попов Г.Я. Рівняння математичної фізики. Метод інтегральних перетворень : учб.посібн./Г.Я.Попов, В.В.Реут,Н.Д.Вайсфельд. - Одеса : Астропринт, 1999 — 67 с.
3. Bowman, F., Introduction to Bessel Functions — Dover New York, 1958.— 142 с.
4. Markel Epelde García Bessel Functions and Equations of Mathematical Physics — 69 с.
5. Jennifer Niedziela — University of Tennessee – Knoxville— 4 с.
6. Bronson, R., and Costa, G.B. (2006). Differential Equations, 3rd edition. — New York: McGraw-Hill.— 404 с.