

Міністерство освіти і науки України
Національний технічний університет України
«Київський політехнічний інститут ім. І. Сікорського»

Кафедра інженерії програмного забезпечення в енергетиці

Практична робота № 5
з курсу: «*Основи Веб-програмування*»

Виконав:

студентка 2-го курсу,
групи ТВ-31

Ященко Анастасія Антонівна

Посилання на GitHub репозиторій:

https://github.com/Yashchen/PW5TB-31_Yashchenko_Anastasiia_Antonivna

Перевірив:

Недашківський О.Л.

Київ 2024/2025

1. Короткий теоретичний матеріал

Надійність у сфері електропостачання є одним із головних параметрів, що визначають ефективність і безпеку функціонування енергосистем. Вона відображає здатність системи забезпечувати безперервне постачання електроенергії в межах заданих нормативами умов. У разі збоїв або пошкодження складових енергетичної інфраструктури можуть виникати значні фінансові втрати й негативні наслідки для суспільства. Це й пояснює необхідність точного оцінювання надійності систем передачі електроенергії.

Ключові поняття та параметри

Сучасна енергосистема складається з великої кількості компонентів — ліній електропередач (ЛЕП), трансформаторів, вимикачів тощо. Для кожного з них, а також для системи загалом, розраховують показники надійності. Основними з них є:

- **Інтенсивність відмов (ϕ)** — показник, що визначає частоту виходу з ладу за певний проміжок часу;
- **Середній час відновлення (t)** — період, необхідний для повернення обладнання до нормального функціонування;
- **Коефіцієнти простою:** аварійного (k_a) та планового (k_p) — характеризують, скільки часу система не працює через аварії або обслуговування.

У розрахунках надійності використовують математичні моделі, зокрема ті, що базуються на розподілі Пуассона. Для систем із послідовним з'єднанням елементів (як у випадку одноколових схем) загальна частота відмов — це сума інтенсивностей відмов усіх складових. У паралельних або резервованих структурах (як у двоколових схемах) враховуються ймовірності одночасних збоїв кількох компонентів.

Порівняння одноколової та двоколової конфігурацій

Проведені розрахунки демонструють, що двоколові схеми значно перевершують одноколові за рівнем надійності. Наприклад, для одноколової структури, яка включає елегазовий вимикач, ЛЕП, трансформатор, ввідний

вимикач і приєднання 10 кВ, інтенсивність відмов становить приблизно **0,295 відмов/рік**, а середній час відновлення — близько **10,7 годин**. У свою чергу, для аналогічної двокової схеми частота відмов знижується до **0,0237 відмов/рік**, що свідчить про суттєве підвищення її стійкості.

Економічні наслідки перерв у постачанні

Фінансові втрати, спричинені відключенням електроенергії, залежать від багатьох чинників — несподіваності, тривалості, кількості споживачів, які залишилися без живлення тощо. Вартість таких збитків розраховується з урахуванням питомих втрат при аварійних і планових зупинках. Наприклад, у випадку використання однострансформаторної підстанції, вартість недопоставленої енергії може становити **23,6 грн/кВт·год** у разі аварії та **17,6 грн/кВт·год** при плановому відключенні. З урахуванням масштабів споживання навіть короточасні простої здатні спричинити втрати на рівні мільйонів гривень.

2. Опис програмної реалізації з необхідними поясненнями та скріншотами програмного коду;

2.1 Завдання 1

Текст

Створіть Веб калькулятор для порівняння надійності однокової та двокової систем електропередачі та розрахунку збитків від перерв електропостачання у разі застосування однострансформаторної ГТП у складі:

1. Порівняти надійність однокової та двокової систем електропередачі (див. Приклад 3.1.);
2. Розрахувати збитки від перерв електропостачання у разі застосування однострансформаторної ГТП (див. Приклад 3.2.).

Опис реалізації

Згідно з реалізованим кодом, інтерфейс користувача створено у файлі `index.html`, де передбачено форму з полями для введення вхідних параметрів. Кожне поле має унікальний атрибут `id`, за яким ми потім отримаємо дані.

Результати обчислень відображаються у виділеному блоці сторінки, що розташований у спеціальному контейнері з `id="output-text"`. Візуальне оформлення форми, кнопок і результатів реалізовано в окремому файлі стилів `style.css`, що забезпечує зручний і охайний вигляд вебсторінки.

```
1  <!DOCTYPE html>
2  <html lang="uk">
3  <head>
4    <meta charset="UTF-8">
5    <meta name="viewport" content="width=device-width, initial-scale=1.0">
6    <title>Аналіз Надійності ЕПС</title>
7    <link rel="stylesheet" href="style.css">
8  </head>
9  <body>
10   <main class="wrapper">
11     <h1>Оцінка Надійності та Перерв</h1>
12
13     <form id="reliability-form">
14       <div class="field">
15         <label for="lambda">Інтенсивність відмов ( $\lambda$ ):</label>
16         <input type="number" id="lambda" step="0.001" required>
17       </div>
18
19       <div class="field">
20         <label for="restore">Час відновлення (t, год):</label>
21         <input type="number" id="restore" step="0.1" required>
22       </div>
23
24       <div class="field">
25         <label for="maintenance">Плановий ремонт ( $t_p$ , год):</label>
26         <input type="number" id="maintenance" step="0.1">
27       </div>
28
29       <div class="field">
30         <label for="type">Конфігурація мережі:</label>
31         <select id="type" required>
32           <option value="single">Одноколова</option>
33           <option value="double">Двоколова</option>
34         </select>
35       </div>
36
37       <button type="submit">Обчислити</button>
38     </form>
```

```

39
40     <section id="result-block">
41         <h2>Результат:</h2>
42         <div id="output-text"></div>
43     </section>
44 </main>
45 <script src="script.js"></script>
46 </body>
47 </html>

```

Рисунок 2.1.1 - Код сторінки калькулятора

У розробленому скрипті, після натискання кнопки **"Обчислити"** спрацьовує обробник події, який перехоплює стандартну поведінку форми. У цей момент відбувається перевірка коректності введених даних, а після цього — виконуються відповідні розрахунки на основі отриманих значень. Усі дії відбуваються без перезавантаження сторінки, а результати миттєво виводяться у призначену для цього область на сторінці.

```

1 document.addEventListener("DOMContentLoaded", () => {
2   const form = document.getElementById("reliability-form");
3   const resultBox = document.getElementById("output-text");
4
5   form.addEventListener("submit", async (e) => {
6     e.preventDefault();
7
8     const lambda = parseFloat(document.getElementById("lambda").value);
9     const tRestore = parseFloat(document.getElementById("restore").value);
10    const tPlanned = parseFloat(document.getElementById("maintenance").value) || 0;
11    const system = document.getElementById("type").value;
12
13    if (lambda <= 0 || tRestore <= 0 || tPlanned < 0) {
14      resultBox.textContent = "❗ значення повинні бути додатні!";
15      return;
16    }
17
18    let data;
19    try {
20      const res = await fetch("data.json");
21      data = await res.json();
22    } catch (err) {
23      resultBox.textContent = "Помилка при отриманні даних.";
24      return;
25    }
26
27    if (!data || !data.elements) {
28      resultBox.textContent = "Неможливо обробити файл даних.";
29      return;
30    }
31
32    const oneCircuit = data.elements.reduce(
33      (sum, el) => {
34        sum.lambda += el.rate;
35        sum.total += el.rate * el.repair;
36        return sum;
37      },

```

```

38     { lambda: 0, total: 0 }
39   );
40
41   const avgRepair = oneCircuit.total / oneCircuit.lambda;
42   let totalFailureRate, downtime;
43
44   if (system === "single") {
45     totalFailureRate = oneCircuit.lambda * avgRepair;
46     downtime = 1.2 * (tPlanned / 8760);
47   } else {
48     const coeff1 = 3.6e-4, coeff2 = 58.9e-4;
49     totalFailureRate = 2 * oneCircuit.lambda * (coeff1 + coeff2) + data.backupBreakerRate;
50     downtime = 0;
51   }
52
53   resultBox.innerHTML = `
54     <p>Сумарна частота збоїв: <strong>${totalFailureRate.toFixed(4)} пік-1</strong></p>
55     <p>Планова перерва: <strong>${downtime.toFixed(4)} годин</strong></p>
56   `;
57   });
58 });

```

Рисунок 2.1.2 - Функція для обрахунку результатів

У згаданій функції спочатку виконується перевірка правильності введених користувачем значень. Після цього, за допомогою асинхронного запиту, завантажуються дані з зовнішнього файлу data.json. Отримана інформація використовується для проведення обчислень, а блок із результатами оновлюється відповідно до розрахованих значень. Усе це відбувається динамічно, без оновлення сторінки.

Результат виконання

Оцінка Надійності та Перерв

Інтенсивність відмов (λ):

Час відновлення (t , год):

Плановий ремонт (t_p , год):

Конфігурація мережі:

Двоколова

Обчислити

Результат:

Сумарна частота збоїв: **0.0237 рік⁻¹**

Планова перерва: **0.0000 років**

Рисунок 2.1.3 - Обрахунок для двоколової системи

Оцінка Надійності та Перерв

Інтенсивність відмов (λ):

Час відновлення (t , год):

Плановий ремонт (t_p , год):

Конфігурація мережі:

Одноколова

Обчислити

Результат:

Сумарна частота збоїв: **3.1600 рік⁻¹**

Планова перерва: **0.0137 років**

Рисунок 2.1.4 - Обрахунок для одноколової системи

Очевидно, що двоколова система демонструє вищу надійність, що відповідає нашим очікуванням.

Висновок

Аналіз надійності систем електропостачання підтверджує переваги резервування у двоколових схемах, які забезпечують підвищену стійкість до відмов та допомагають зменшити економічні втрати. Впровадження цифрових інструментів, зокрема веб-калькуляторів для розрахунку показників надійності та збитків, є важливим кроком у модернізації сучасних енергетичних систем. Це дає змогу інженерам швидко й ефективно оцінювати різні варіанти та вибирати найбільш оптимальну структуру електропостачання.