UNITECH 2017 GABROVO

INTERNATIONAL SCIENTIFIC CONFERENCE
17-18 November 2017, GABROVO

ИЗСЛЕДВАНЕ НА ПРЕХОДЕН ПРОЦЕС В ЕЛЕКТРИЧЕСКА ВЕРИГА В ПРОГРАМНА СРЕДА МАТЛАБ

RESEARCH OF THE TRANSITIONAL PROCESS IN A ELECTRICAL CIRCUITS VIA MATLAB ENVIRONMENT

Velislava Raydovska
UNIVERSITY OF PLOVDIV PAISII HILENDARSKI
FACULTY OF PHYSICS AND TECHNOLOGY

Velichko Manev
UNIVERSITY OF PLOVDIV PAISII HILENDARSKI
FACULTY OF PHYSICS AND TECHNOLOGY

Anton Haritev
UNIVERSITY OF PLOVDIV PAISII HILENDARSKI
FACULTY OF PHYSICS AND TECHNOLOGY

Angel Chekichev
UNIVERSITY OF PLOVDIV PAISII HILENDARSKI
FACULTY OF PHYSICS AND TECHNOLOGY

Abstract
The paper provides an example for easiness of mathematics at solving and research transitional processes in a complex electrical circuits through the software Matlab. The use of the Matlab programme software which is presented in the paper will be used in the educational process in the Electrical Engineering course, included in the curriculum of the specialty "Electricity technology" or "Computer and Communication Systems" for the Bachelor educational degree.

Keywords: transitional process; electrical circuit; MatLab.

ВЪВЕДЕНИЕ

Изучаването на преходните процеси позволява да се предвидят появата на пренапрежения в участъци от веригата, които да са опасни за изолацията. При преминаването от един установен режим към друг е възможно амплитудите на токове многократно да превишат амплитудата на тока в установен режим. Решаването на преходен процес дава възможност да се установи как се деформират сигналите по форма и амплитуда при преминаването им през усилватели, филтри или други устройства. Затова изучаването на преходните процеси е важна част от обучението на студентите от областта на електротехниката, електрониката и комуникациите.

Решаването на преходен процес в сложна електрическа верига изисква познания както в областта на електротехниката, така и на математиката. Познавайки добре методиката за класически или операторен метод за решение, прилагайки основните закони от теоретичната електротехника, студентът получава квадратни уравнения, системи от уравнения с реални или комплексни числа. Възможност за опростяване на математическите пресмятания е използването на програмния продукт MathCad [1].

Методът с променливите на състоянието е модификация на класическия метод. Уравненията с променливите на състоянието лесно се записват в матрична форма и могат да се обработят в програмната среда MatLab. Известно е, че MatLab е диалогова програмна система, която разполага с множество вградени функции за решаване на задачи от областта на матричната алгебра, комплексната аритметика, линейните системи уравнения, диференциалното и интегрално смятане и др.

Използването на симулации на различните процеси в обучението помагат на студентите да придобиват знания и умения по активен начин. Те се използват за по-добро разбиране на същността на процесите в електрическите схеми или устройства, създаването на възможности за експерементиране с различни сценарии за подпомагане на процеса на обучение при сложни обекти.

Предимство на програмната среда MatLab е приложението Simulink. Това е графична среда за моделиране, симулиране и анализ на динамични системи. Неговият първичен интерфейс е графичен инструмент за блокова диаграма и адаптивен набор от библиотеки. Приложението Simulink дава възможност да се моделира електрическата верига и да се изследва преходния процес в нея като моделираният обект може да се изследва при различни режими на работа.

Известно е, че преходните процеси са бързо протичащи процеси – продължителността им е десетки, стотици или дори милиардни части от секундата. Възможността да се проектират различни по сложност схеми в програмната среда MatLab и след това да се проверява дали те работят според очакванията, позволява симулациите да се използват вместо реалната практика, когато се изучават опасни явления, като възникването на пренапрежения при работата на високоволтови и високомощни електрически схеми или при трудно достижими обекти.

ИЗЛОЖЕНИЕ

Програма за решаване на преходен процес в произволна схема

Преходните процеси се решават като се използват класическия или операторния метод. Те са широко известни [2,3], не са цел на настоящия доклад и няма да бъдат разглеждани подробно. За да се намерят корените на характеристичното уравнение може да се използва командата *roots(p)*, където *p* е полиномът на характеристичното уравнение. При съставянето на характеристичното уравнение може да се използват уравненията, написани по законите на Кирхоф за следкомутационната схема. Необходимо е да се напомни, че символът $\frac{d...}{dt}$ се заменя, например, с λ, а $\int ...dt$ – с λ^{-1}. Когато се търсят интеграционните константи и е необходимо да се реши система уравнения може да се използва следния сорс код програмна среда MatLab:

>>A=input('коефициентите пред неизвестните: A='); % да се използват квадратни скоби при въвеждане на коефициентите пред неизвестните, матрицата се вкарва като се изписват първо коефициентите в първото уравнение, после във второ, отделени с „;”

B=input('свободните членове: B='); % да се използват квадратни скоби при въвеждането на вектор-стълб, стойностите да се отделят с „;”

X=A\B.

Методът на променливите на състоянието е модификация на класическия метод. При него се избират индуктивните токове и капацитивните напрежения за променливи на състоянието. Всички останали величини се изразяват чрез тях. С помощта на законите на Кирхоф се съставя нехомогенна система уравнения (1), която описва преходния процес във веригата. Уравненията се записват в нормална форма на Коши, като се решат спрямо първите производни на променливите на състоянието [2,3]. Броят на уравненията в (1) е s, където s е броят на реактивните елементи. Други означения в (1) са: p_L – брой на индуктивни клонове, p_{LC} и p_C – съответно брой на резонансните и на капацивните клонове. Общият брой на реактивните клонове е отбелязан с p_r, а на клоновете във веригата p. Функциите $f_1, f_2, ..., f_s$ са линейни еднородни форми на аргументите си.

$$\left|\begin{array}{l}
\frac{di_{L1}}{dt} = f_1(i_{L_1}, i_{L2}, ..., u_{C(p_L+1)}, ..., u_{Cp_L}, e_1, ..., e_p) \\
\frac{di_{L2}}{dt} = f_2(i_{L_1}, i_{L2}, ..., u_{C(p_L+1)}, ..., u_{Cp_L}, e_1, ..., e_p) \\
... \\
\frac{di_{L(p_L+p_{LC})}}{dt} = f_{(p_L+p_{LC})}(i_{L_1}, i_{L2}, ..., u_{C(p_L+1)}, ..., u_{Cp_L}, e_1, ..., e_p) \\
\frac{du_{C(p_L+1)}}{dt} = f_{(p_L+p_{LC}+1)}(i_{L_1}, i_{L2}, ..., u_{C(p_L+1)}, ..., u_{Cp_L}, e_1, ..., e_p) \\
... \\
\frac{du_{Cpr}}{dt} = f_s(i_{L_1}, i_{L2}, ..., u_{C(p_L+1)}, ..., u_{Cp_L}, e_1, ..., e_p)
\end{array}\right. \quad (1)$$

Удобно е променливите на състоянието да се означават с един и същ символ, независимо дали са ток или напрежение. Полага се [3] $x_i = i_{L_i}$ за $i = 1,2,..., p_L + p_{LC}$ и $x_i = u_{C(i-p_{LC})}$ за $i = p_L + p_{LC} + 1,..., s$.

Системата (1) добива вида:

$$\begin{cases} \frac{dx_1}{dt} = f_1(x_1, x_2, \dots, x_s, e_1, \dots, e_p) \\ \frac{dx_2}{dt} = f_2(x_1, x_2, \dots, x_s, e_1, \dots, e_p) \\ \dots \\ \frac{dx_s}{dt} = f_s(x_1, x_2, \dots, x_s, e_1, \dots, e_p) \end{cases} \quad (2)$$

Началните стойности на променливите на състоянието са независимите начални условия.

Системата (2) може да бъде представена в матрична форма:

$$\dot{X} = A.X + B.u \quad (3)$$

В (3) с X е отбелязана матрицата на променливите на състоянието, тя е вектор-стълб. A е матрицата на параметрите на веригата. Тя е квадратна и е от ред s. B е матрица на параметрите на въздействията. $\dot{X}$ е производната на матрицата X по отношение на времето. Математическото решение е свързано първо с определяне на преходната матрица e^{At}. Нека A е матрица с размерност nxn и I е единична матрица със същата размерност [4]. По определение собствените стойности λ_i на матрицата A, където $i=1,2,\dots,n$, са корените на полинома (4) от n^{-ma} степен.

$$\det[A - \lambda I] = 0 \quad (4)$$

Корените на полинома (4) могат да са различни реални или кратни, или комплексни числа. Оценяването на преходната матрица e^{At} се основава на теоремата на Хамилтон-Кейли. Тя гласи, че матрицата може да бъде представена като полином от $(n-1)^{-ва}$ степен като

$$e^{At} = a_0 I + a_1 A + a_2 A^2 + \dots + a_{n-1} A^{n-1}, \quad (5)$$

където коефициентите a_i са функция на собствените стойности λ_i. Когато корените са различни реални или комплексни числа, коефициентите a_i се намират от решението на системата:

$$\begin{cases} a_0 + a_1\lambda_1 + a_2\lambda_1^2 + \dots + a_{n-1}\lambda_1^{n-1} = e^{\lambda_1 t} \\ a_0 + a_1\lambda_2 + a_2\lambda_2^2 + \dots + a_{n-1}\lambda_2^{n-1} = e^{\lambda_2 t} \\ \dots \\ a_0 + a_1\lambda_n + a_2\lambda_n^2 + \dots + a_{n-1}\lambda_n^{n-1} = e^{\lambda_n t} \end{cases} \quad (6)$$

Когато част от корените са m-кратно число, а останалите са различни, коефициентите се намират като решение на системата:

$$\begin{cases} a_0 + a_1\lambda_1 + a_2\lambda_1^2 + \dots + a_{n-1}\lambda_1^{n-1} = e^{\lambda_1 t} \\ \frac{d(a_0 + a_1\lambda_1 + a_2\lambda_1^2 + \dots + a_{n-1}\lambda_1^{n-1})}{d\lambda_1} = \frac{d(e^{\lambda_1 t})}{d\lambda_1} \\ \dots \\ \frac{d^{m-1}(a_0 + a_1\lambda_1 + a_2\lambda_1^2 + \dots + a_{n-1}\lambda_1^{n-1})}{d\lambda^{m-1}{}_1} = \frac{d^{m-1}(e^{\lambda_1 t})}{d\lambda^{m-1}{}_1} \\ a_0 + a_1\lambda_{m+1} + a_2\lambda_{m+1}^2 + \dots + a_{n-1}\lambda_{m+1}^{n-1} = e^{\lambda_{m+1} t} \\ \dots \\ a_0 + a_1\lambda_n + a_2\lambda_n^2 + \dots + a_{n-1}\lambda_n^{n-1} = e^{\lambda_n t} \end{cases} \quad (7)$$

Намерените коефициенти се заместват в преходната матрица. Решението на (3) е подробно обяснено в [4] и е от вида:

$$x(t) = e^{A(t-t_0)} x_0 + e^{At} \int_{t_0}^{t} e^{-A\tau} bu(\tau) d\tau \quad (8)$$

Въз основа на съставените уравнения по метода на променливите на състоянието и описаното решение се съставя програма *prehoden_proces.m* за решаване преходния процес. Сорс кодът на програмата е:

1. disp('Програма за решаване на преходен процес в ЛЕВ')
2. disp('******')
3. disp('Съставете системата уравнения по метода на променливите на състоянието')
4. clear
5. format long
6. A=input('Въведете матрицата на параметрите на веригата (използвайте квадратни скоби) A=');
7. B=input('Въведете като вектор-стълб свободните членове (използвайте квадратни скоби) B=');
8. X0=input('Въведете като вектор-стълб независимите начални условия (използвайте квадратни скоби) Xo=');
9. p=poly(A); %характеристично уравнение
10. r=roots(poly(A)); %корени на характеристичното уравнение
11. disp('Корените на характеристичното уравнение са:')
12. fprintf('\n');
13. fprintf('lambda1=%5.2f\t',r(1));
14. fprintf('lambda2=%5.2f\t',r(2));
15. syms a0,

16. syms a1,
17. syms t
18. disp('Намиране на коефициентите a_i. Броят i зависи от размерността на матрицата A.') %в примера тя е е 2x2, затова ще се търсят първите 2 члена a_0 и a_1
19. disp('Съставете система уравнения от вида посочен в теорията: (6) или (7)')
20. fprintf('\n');
21. p2=input('Въведете коефициентите пред a0 и a1 в матрица p2=');
22. p3=input('Въведете свободните членове като вектор-стълб p3=');
23. eA=p2\p3
24. A0=input('Въведете израз за A0 (първия ред на eA) A0=');
25. A1=input('Въведете израз за A1 (втория ред на eA) A1=');
26. I=eye(2,2);
27. eAt=(A0*I)+(A1*A);
28. eAtx=eAt*X0;
29. X1=eAt*B
30. X2=input('Въведете горния резултат като матрица, като използвате квадратни скоби и отделите с точка и запетая двата реда X2=');
31. X3=int(X2,t,0,t);
32. X=eAtx+X3;
33. iL=input('Въведете първия ред от получения израз X iL=')
34. uC=input('Въведете втория ред от получения израз X uC=')

Приложение на създадената програма

Задачата, която се поставя, е да се намери изразът за напрежението $u_C(t)$ след превключването на ключа k от положение *1* в положение *2* в електрическата верига (фиг.1) със следните параметри:

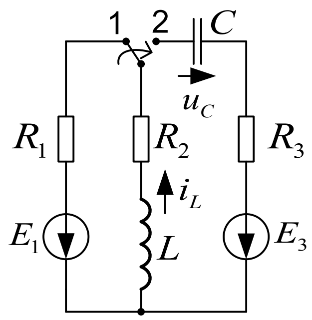
$R_1 = 20\Omega$, $R_2 = 80\Omega$, $R_3 = 10\Omega$, $L = 5mH$, $C = 50\mu F$, $E_1 = 100V$, $E_3 = 50V$.

Задачата се решава като се изпълняват стъпките от методиката за решаване по класически метод [2, 3]. Тъй като те не са цел на настоящия доклад няма да бъдат разглеждани подробно.

Независими начални условия са: $i_L(0) = 1A$ и $u_C(0) = 0V$.

Установените съставки са:

$i_L' = 0A$; $u_C' = 50V$.



Фиг. 1. *Схема на изследваната верига*

За контура по закона на Кирхоф се записва:

$$L\frac{di_L}{dt} + (R_2 + R_3)i_L + \frac{1}{C}\int i_L dt = E_3. \qquad (9)$$

Характеристичното уравнение е

$$LC\lambda^2 + (R_2 + R_3)C\lambda + 1 = 0.$$

Неговите корени са $\lambda_1 = -2596.9$ и $\lambda_1 = -15403$.

Видът на свободните съставки е:

$i_{L св} = A_{i1}e^{-2596.9t} + A_{i2}e^{-15403t}, A$;

$u_{Cсв}(0) = A_{u1}e^{-2596.9t} + A_{u2}e^{-15403t}, V$.

Решение на задачата са:

$i_L(0) = -0.578e^{-2596.9t} - 0.4219e^{-15403t}, A$

$u_C(0) = 50 - 44.52e^{-2596.9t} - 5.478e^{-15403t}, V$

Решение по метода с променливите на състоянието

Използвайки уравнение (9) се съставя системата уравнения:

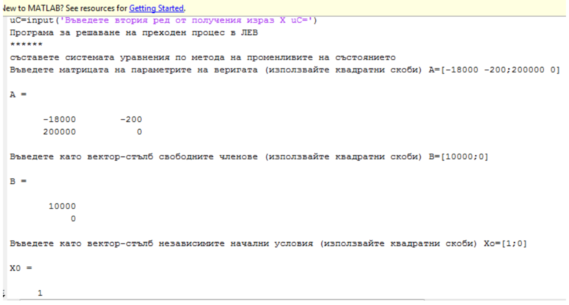
$$\begin{cases} \dfrac{di_L}{dt} = -\dfrac{1}{L}(R_2 + R_3)i_L - \dfrac{u_C}{L} + \dfrac{E_3}{L} \\ \dfrac{du_C}{dt} = \dfrac{1}{C}i_L \end{cases} \qquad (10)$$

Като се замести с числените стойности се получава:

$$X = \begin{bmatrix} i_L \\ u_C \end{bmatrix}; \quad X0 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}; \quad B = \begin{bmatrix} 10000 \\ 0 \end{bmatrix};$$

$$A = \begin{bmatrix} -18000, -200 \\ 200000, 0 \end{bmatrix}.$$

Стартира се сорс кодът на програмата *prehoden_proces.m*. На фиг.2 е показан фрагмент от изпълнението на командите в програмната среда MatLab.



Фиг. 2. *Фрагмент от изпълнението на командите в програмната среда MatLab*

Крайните резултати, получени при работата на програмата са

*iL=(92539*exp(-(64922*t)/25))/160078 + (5719171973470894642O9*exp(-(8467954772014531*t)/549755813888))/1355533263994542093418*

*Въведете втория ред от получения израз X uC=11000476396945408687500000000/22000982641263415447220849 - (3712995791418163750000*exp(-(8467954772014531*t)/549755813888))/6777666319972710467O9 - (115673750000*exp(-(64922*t)/25))/2598145979.*

При преработката на горните изрази се получава:

$$i_L(0) = -0.578e^{-2596.9t} - 0.4219e^{-15403t}, A$$

$$u_C(0) = 50 - 44.52e^{-2596.9t} - 5.478e^{-15403t}, V \quad (11)$$

Като се съпоставят резултатите, получени по класическия метод и по метода на променливите на състоянието се вижда, че съвпадат напълно.

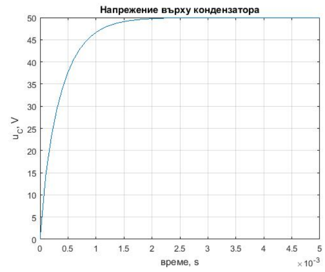
За да се изчертае графиката на зависимост (11) се въвеждат следните команди:

```
t1=0:0.0001:0.005;
y=50-45.*exp(-2597.*t1)-5.*exp(-15403.*t1);
plot(t1,y)
grid on;
title('Напрежение върху кондензатора')
xlabel('време, s')
ylabel('u_{C}, V')
```

На фиг.3 е показана изчертаната зависимост (11). Тъй като преходният процес е много кратък, за да се получи добра визуализация е необходимо да се подбере малка стъпка на нарастване на времето.

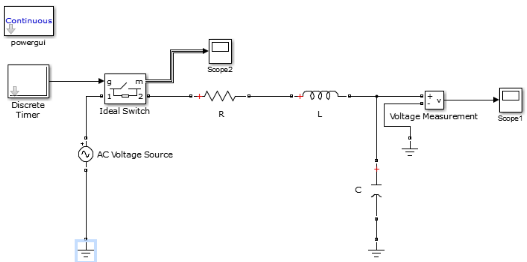
Създаване на модел на веригата в SimPowerSystem

Приложението *SimPowerSystem* дава възможност да се създаде модел на схемата в програмната среда MatLab и да се симулира работата й.



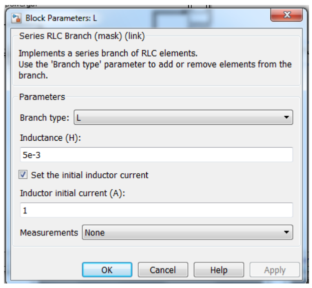
Фиг. 3. *Зависимост на напрежението върху кондензатора от времето в MatLab*

На фиг. 4 е показан моделът на веригата.

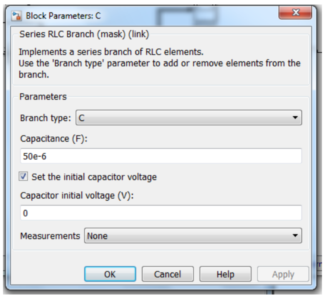


Фиг. 4. *Модел на изследваната верига в SimPowerSystem на MatLab*

При настройката на индуктивния елемент *L* и капацитивния елемент *C* е необходимо да се отбележат независимите начални условия, както е показано на фиг. 5 и фиг. 6.

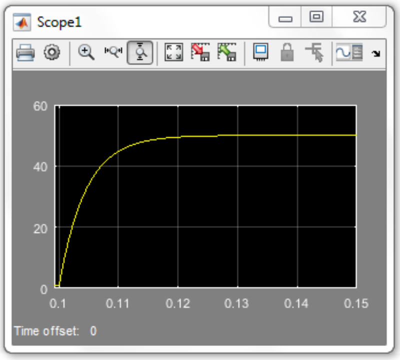


Фиг. 5. *Настройка на параметрите на индуктивния елемент*



Фиг. 6. *Настройка на параметрите на капацитивния елемент*

При симулация на работата на веригата на осцилоскопа се наблюдава изменението на напрежението върху кондензатора (фиг. 7).



Фиг. 7. *Дисплей на осцилоскопа Scope1*

Като се сравнят зависимостите от фиг.3 и фиг. 7 се вижда пълно съвпадение на резултатите.

ЗАКЛЮЧЕНИЕ

Програмният продукт MatLab е универсален, лесен за усвояване и употреба от студентите компютърен пакет. Той може да бъде незаменим помощник при математическите пресмятания и при решаване на преходен процес в сложни електрически вериги по класически метод. Разработената програма *prehoden_proces.m* е универсална и може да се прилага при изследването на произволни схеми. Приложението *Simulink* дава възможност да се моделира електрическата верига и да се изследва преходния процес в нея.

Acknowledgement
The authors would like to acknowledge the support of the "Research & Development" division of UNIVERSITY OF PLOVDIV PAISII HILENDARSKI in the project: СП17-ТК-004/ 16.05.2017.

REFERENCE

[1] Raidovska Velislava, Velichko Manev, Anton Haritev. Solving the transitional process in a complex electrical circuit via the computer product Mathcad. Scientific researches of Union of Scientists of Bulgaria – Smolyan, Volume II, 2016, pp 280-285

[2] Farhi C, S. Papazov. Theoretical Electrical Engineering, Part I, Ed. Technika, Sofia, 1990.

[3] Genov Ludmil. Theoretical Electrical Engineering, Ed. Technika, Sofia, 1991.

[4] Karris Steven T., Circuit Analysis II with MATLAB Computing and Simulink/ SimPowerSystems Modeling, Orchard Publications, Fremont, California, 2009.