

Aktorik Sensorik

Labor 1

Anton Kress (S872899), Jan Abel (S876662)

October 2020

Contents

1	Messung des Stillstands Drehmomentes	5
1.1	Beschreibung	5
1.2	Ausgabe der Lösung	6
1.3	Matlab Code	6
2	Messung des Ankerwiderstand	8
2.1	Beschreibung	8
2.2	Ausgabe der Lösung	9
2.3	Matlab Code	9
3	Messung des Stillstands Drehmomentes	11
3.1	Beschreibung	11
3.2	Ausgabe der Lösung	11
3.3	Matlab Code	11
4	Messung der Kennlinie des Verstärkers	14
4.1	Beschreibung	14
4.2	Ausgabe der Lösung	15
4.3	Matlab Code	15

Aufgabenstellung

1 Termin

In der ersten Laborübung sollen die Werte der Elemente des Ersatzschaltbildes eines permanent erregten Gleichstrommotors bestimmt werden.

Vorbereitung:

1. Um Parameter richtig identifizieren zu können, muss man viele Messungen durchführen. Nun ist die Frage, wie man einen statistisch relevanten Wert für einen Parameter aus vielen Messungen bekommt ...
2. Recherchieren Sie nach der Methode der Kleinsten Quadrate und finden Sie heraus, wie diese Methode funktioniert.
3. Was ist ein Inkrementalgeber? Welche Typen davon gibt es? Wie funktioniert so ein Gerät? Lesen Sie bitte die Wikipedia Seiten dazu: https://en.wikipedia.org/wiki/Incremental_encoder und <https://de.wikipedia.org/wiki/Inkrementalgeber>).

1.1 Messung des Stillstandsrehmomentes

Das vom Motor abgegebene Antriebsmoment ist über die Momentenkonstante k_M mit dem Motorstrom verknüpft. Ziel der ersten Teilaufgabe ist die Bestimmung dieser Konstante k_M . Hierzu wird der Motorstrom mit Hilfe unseres Netztes von 0 A bis 2 A vorgegeben und das Drehmoment mit Hilfe der Federwaage gemessen. Der Motor wird bei dieser Messung im Stillstand betrieben. Die entstehende Kennlinie, die das Drehmoment über dem Motorstrom darstellt, wird in MATLAB gezeichnet und die Steigung dieser Kennlinie stellt die Konstante k_M dar.

1.2 Messung des Ankerwiderstandes R

Ein wesentlicher Teil des Modells der permanent erregten Gleichstrommaschine ist der Ankerwiderstand R . Ziel dieser Teilaufgabe ist es, diesen zu bestimmen. Welcher Teil des technischen Aufbaus des Motors liegt im Stromkreis, wurde aber bisher nicht berücksichtigt? Wenn dieses Bauteil weiter außen vor gelassen wird, welche Konsequenz hat das für die Durchführung der Messung des Ankerwiderstandes R ? Schlagen sie nun eine geeignete Messung vor und bestimmen sie nach dieser den Ankerwiderstand R .

1.3 Messung der Leerlaufkennlinie

Die induzierte Spannung u_i ist proportional zur Winkelgeschwindigkeit des Motors ω . Es gilt $u_i(t) = k_e \cdot \omega(t)$. Ziel der zweiten Teilaufgabe ist die Bestimmung der Konstanten k_e . Wir geben mit Hilfe unseres Netztes die Motorspannung mit 0 V bis 12 V vor und messen die Drehzahl. Hierbei wird der Motor nicht belastet, d. h. er wird im Leerlauf betrieben. Die entstehende Kennlinie wird Leerlaufkennlinie genannt und wird mit Hilfe von MATLAB dargestellt. Die gesuchte Konstante k_e ist die Steigung dieser Kennlinie. Zur Messung der Drehzahl verwenden wir einen Inkrementalencoder. Er wird an Timer 3 des Mikrocontrollers C167, der ein Inkrementalkoder-Interface besitzt, angeschlossen und in Vierfachausswertung betrieben. Wir erhalten so eine Pulszahl von PZ = 2000 Pulsen pro Umdrehung. Die Abtastzeit der Interruptroutine, in der der Inkrementaldecoder ausgewertet wird, beträgt $T = 1$ ms. Was ist der Unterschied zwischen Drehzahl und Winkelgeschwindigkeit? Welche Variablen werden für diese beiden Angaben benutzt?

Beuth-Hochschule für Technik Berlin	Labor für Automatisierungstechnik	Übungsveranstaltung für Aktorik & Sensorik
Prof. Dr.-Ing. FJ Morales		3 von 10

1.4 Messung der Kennlinie des Leistungsverstärkers

Die Eingangsspannung des Leistungsverstärkers wird im Bereich von -12 V bis $+12\text{ V}$ mit Hilfe des Netztesles vorgegeben und die Ausgangsspannung mit Hilfe des Multimeters gemessen. Aus der Verstärkerkennlinie, in der mit MATLAB die Ausgangsspannung über der Eingangsspannung dargestellt wird, erhält man die Verstärkung **A** des Leistungsverstärkers.

Hinweis: Nehmen Sie genug Messpunkte auf, damit Sie mögliche Nichtlinearitäten der Kennlinie feststellen können

Beuth-Hochschule für Technik Berlin	Labor für Automatisierungstechnik	Übungsveranstaltung für Aktorik & Sensorik
Prof. Dr.-Ing. FJ Morales		4 von 10

1 Messung des Stillstandsrehmomentes

1.1 Beschreibung

Im ersten Versuch soll die Momentenkonstante k_m bestimmt werden. Sie hängt folgendermaßen mit dem Drehmoment M_M und dem Motorstrom $i_a(t)$ zusammen.

$$\begin{aligned} M_M(t) &= k_m \cdot i_a(t) \\ k_m &= \frac{M_M(t)}{i_a(t)} \end{aligned} \tag{1}$$

Als Messwerte ist eine Matrix mit den Motorströmen I_a und der Auslenkungskraft F gegeben. Um daraus das Drehmoment M_M zu bestimmen wird der Radius r benötigt, welcher mit 1cm gegeben ist.

$$M_M = f(I_a) = r \cdot F(I_a) \tag{2}$$

Anschließend kann die Momentenkonstante k_m über die Steigung der geploteten Gerade bestimmt werden. Hierfür muss gewährleistet werden, dass der Arbeitspunkt linear ist. Deshalb dürfen die letzten drei Messwerte in der linearen Regression nicht betrachtet werden.

$$k_m \simeq 0.022 \frac{\text{Nm}}{\text{A}} \tag{3}$$

1.2 Ausgabe der Lösung

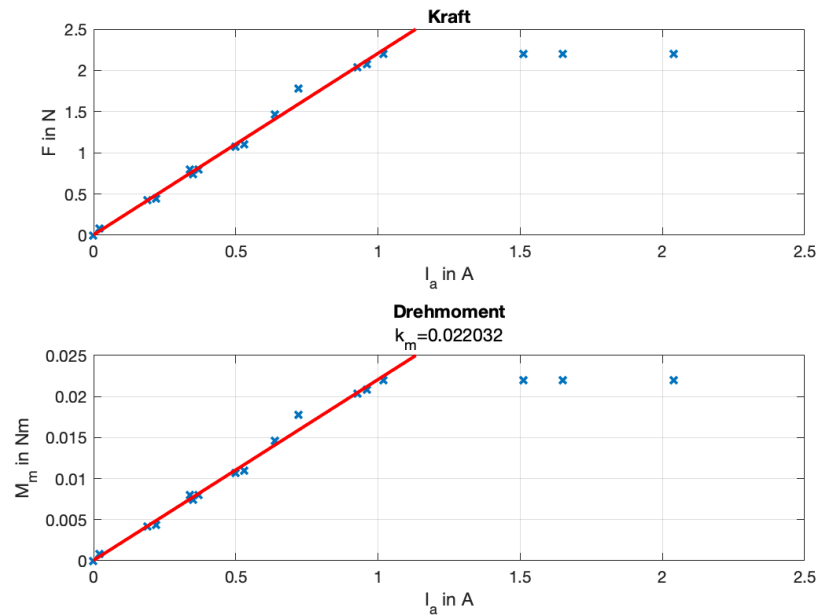


Figure 1: Plot der Aufgabe 1

1.3 Matlab Code

```
1 % Aktorik & Sensorik - WS 2020
2 %
3 % 1.1 Messung des Stillstands Drehmomentes
4 %
5 % Datum: 27.10.2020
6 % Autoren: Anton Kress, S872899
7 %           Jan Abel, S876662
8
9 clear
10 home
11 close all
12
13 FolderName = "./src/";
14 A_Name = "A.mat";
15 B_Name = "B.mat";
16
17 A_test = FolderName + A_Name;
18 A = fullfile(FolderName, A_Name);
19 B = fullfile(FolderName, B_Name);
20
21 % A = B - 3 letzte Elemente - Arbeitsbereich
```

```

22 load(A);
23 load(B);
24 r=0.01;
25 % Auslesen der Ströme und Drehmomente
26 A_plot=sortrows(A,2);
27 B_plot=sortrows(B,2);
28
29 % lineares fitting im Arbeitsbereich - Kraft
30 f1 = polyfit(A_plot(:,2), A_plot(:,3), 1);
31
32 % lineares fitting im Arbeitsbereich - Drehmoment
33 f2 = polyfit(A_plot(:,2), A_plot(:,3)*r, 1);
34 % Momentenkonstante k_m entspricht Steigung der Gerade
35 k_m=f2(1,1)
36
37 % Erzeugung der Ausgleichsgerade
38 x1 = linspace(0, 2.5);
39 y1 = polyval(f1, x1);
40 y2 = polyval(f2, x1);
41
42
43 figure(1);
44 subplot(2,1,1);
45 plot(B_plot(:,2), B_plot(:,3), 'x', x1, y1, 'r', 'linewidth', 2);
46 axis([0 2.5 0 2.5])
47 title('Kraft')
48 xlabel('I_a in A');
49 ylabel('F in N');
50 grid on
51
52 subplot(2,1,2);
53 plot(B_plot(:,2), B_plot(:,3)*r, 'x', x1, y2, 'r', 'linewidth', 2);
54 axis([0 2.5 0 0.025])
55 title('Drehmoment')
56 subtitle(['k_m=' num2str(k_m)])
57 xlabel('I_a in A');
58 ylabel('M_m in Nm');
59 grid on
60
61 % save current plot to img/-folder
62 imagePath = fullfile('..img/', mfilename);
63 print(imagePath, '-dpng');

```

2 Messung des Ankerwiderstand

2.1 Beschreibung

Im zweiten Versuch soll der Ankerwiderstand R bestimmt werden. Der Ankerwiderstand kann über das Ohm'sche Gesetz berechnet werden, dafür ist eine Matrix mit den Messwerten der Spannungen und Ströme gegeben.

Da mit den Messwerten die Ströme über den Spannungen abgebildet werden ist die Steigung nicht der Widerstand sondern der Leitwert. Deshalb muss zur Ermittlung des Ankerwiderstand noch das Reziproke des Leitwerts berechnet werden.

$$R = \frac{U_a}{I_a} \Leftrightarrow G = \frac{1}{R} = \frac{I_a}{U_a} \quad (4)$$
$$R \simeq 3.26\Omega$$

2.2 Ausgabe der Lösung

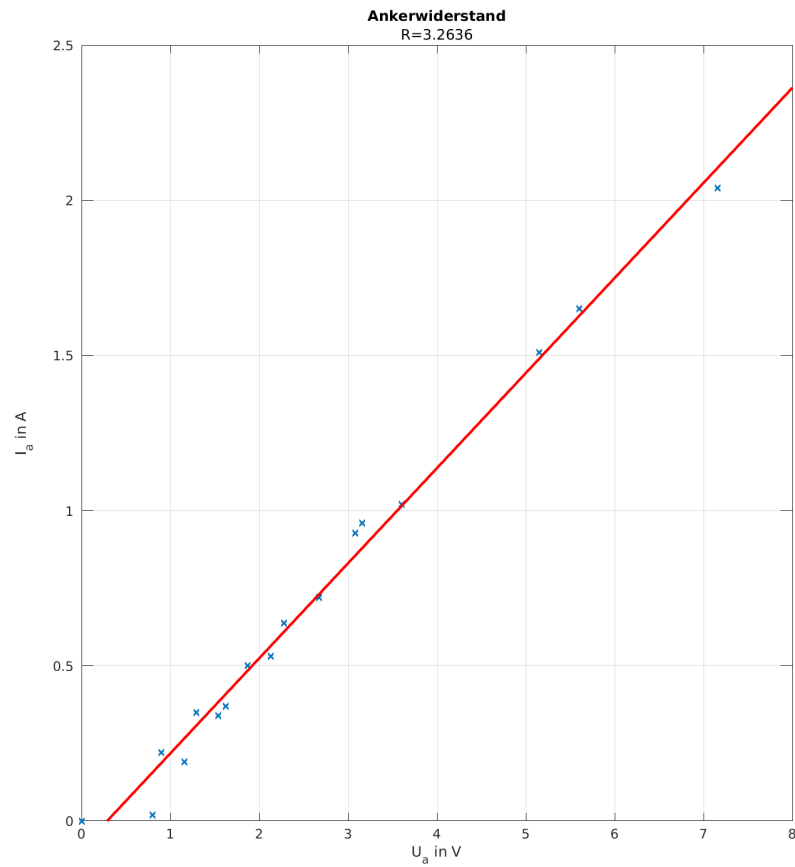


Figure 2: Plot der Aufgabe 2

2.3 Matlab Code

```
1 % Aktorik & Sensorik - WS 2020
2 %
3 % 1.2 Messung des Ankerwiderstandes
4 %
5 % Datum:      27.10.2020
6 % Autoren:    Anton Kress,      S872899
7 %             Jan Abel,         S876662
8
9 clear
10 home
```

```

11 close all
12
13 FolderName = "./src/";
14 B_Name = "B.mat";
15 B = fullfile(FolderName, B_Name);
16
17 % Auslesen der Ströme und Spannungen
18 load(B);
19 B_plot=sortrows(B,1);
20
21 % lineares fitting im Arbeitsbereich
22 f2 = polyfit(B_plot(:,1), B_plot(:,2), 1);
23 % Leitwert G entspricht Steigung der Gerade
24 % R = 1/G
25 R=1/f2(1,1)
26 % Erzeugung der Ausgleichsgerade
27 x1 = linspace(0, 8);
28 y1 = polyval(f2, x1);
29
30 figure(1);
31 plot(B_plot(:,1), B_plot(:,2),'x', x1, y1, 'r','linewidth',2 );
32 axis([0 8 0 2.5])
33 title('Ankerwiderstand')
34 subtitle(['R=' num2str(R)])
35 xlabel('U_a in V');
36 ylabel('I_a in A');
37 grid on
38
39 % save current plot to img/-folder
40 imagePath = fullfile('../img/', mfilename);
41 print(imagePath,'-dpng');

```

3 Messung des Stillstandsrehmomentes

3.1 Beschreibung

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

3.2 Ausgabe der Lösung

3.3 Matlab Code

```
1 % Aktorik & Sensorik - WS 2020
2 %
3 % 1.3 Messung der Leerlaufkennlinie
4 %
5 % Datum:      27.10.2020
6 % Autoren:    Anton Kress,      S872899
7 %             Jan Abel,         S876662
8
9 clear
10 home
11 close all
12
13 FolderName = "./src/";
14 C_Name = "C.mat";
15 C = fullfile(FolderName, C_Name);
16
17 % Auslesen der Winkelgeschwindigkeit und Spannungen
18 load(C);
19 C_plot=sortrows(C,1);
20 % Pulse pro Umdrehung in [pulse/rad]
21 Pz=2000/(2*pi)
22 % Pulse pro Inkrement in [pulse/ink]
23 alpha=4
24 % Faktor in [ink/rad]
25 lambda= Pz/alpha
26 % lineares fitting im Arbeitsbereich
27 f3 = polyfit(C_plot(:,2), C_plot(:,1), 1);
28 % die Steigung hat die Einheit [Ink/V ms]
29 % ke hat die Einheit [Vs/rad]
30 k_e=lambda/(1000*f3(1,1))
```

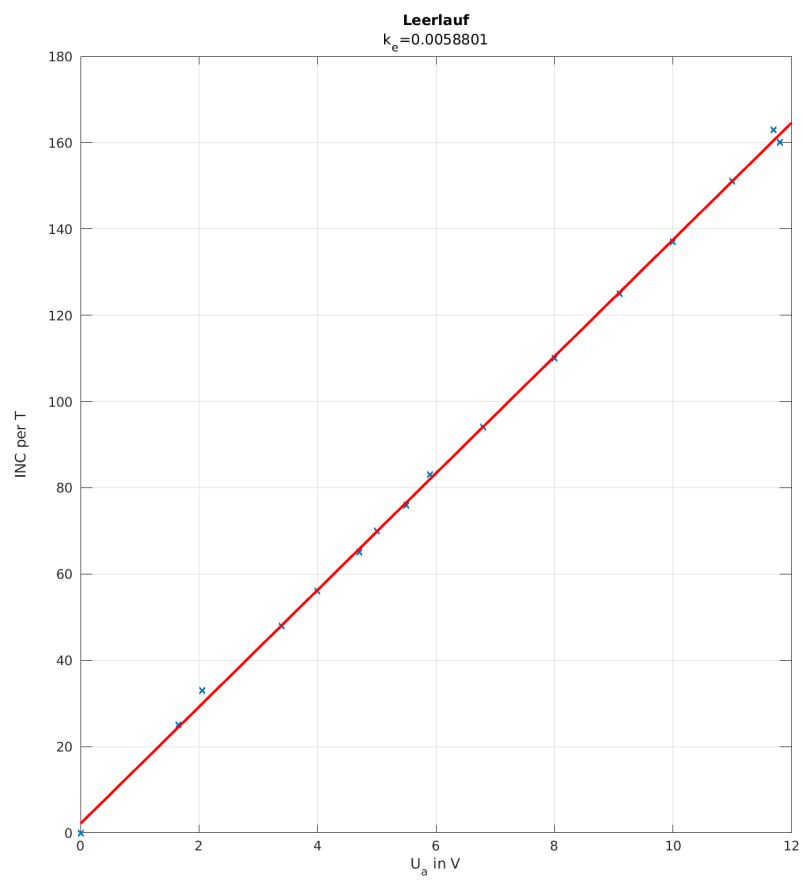


Figure 3: Plot der Aufgabe 3

```

31 % Erzeugung der Ausgleichsgerade
32 x1 = linspace(0, 12);
33 y1 = polyval(f3, x1);
34
35 figure(1);
36 plot(C_plot(:,2), C_plot(:,1), 'x', x1, y1, 'r', 'linewidth', 2);
37 axis([0 12 0 180])
38 title('Leerlauf')
39 subtitle(['k_e=' num2str(k_e)])
40 xlabel('U_a in V');
41 ylabel('INC per T');
42 grid on
43
44 % save current plot to img/-folder
45 imagePath = fullfile('../img/', mfilename);
46 print(imagePath, '-dpng');

```

4 Messung der Kennlinie des Verstärkers

4.1 Beschreibung

In der letzten Messung soll die Kennlinie des Messverstärkers ermittelt werden und damit der Verstärkungsfaktor A bestimmt werden. Dafür liegt eine Matrix mit den Eingangsspannungen U_e und den Ausgangsspannungen U_a vor. Da der Verstärker ausgangsseitig bei etwa $+13.75\text{V}$ und -13.06V in Sättigung geht, werden die jeweils ersten 2 und die letzten beide Messwerte für die Brechnung der Funktion nicht betrachtet.

Der Verstärkungsfaktor A ist der Quotient aus Ausgangs- und Eingangsspannung und somit die Steigung der ermittelten Funktion.

$$A = \frac{U_a}{U_e} \simeq 2\text{V} \quad (5)$$

4.2 Ausgabe der Lösung

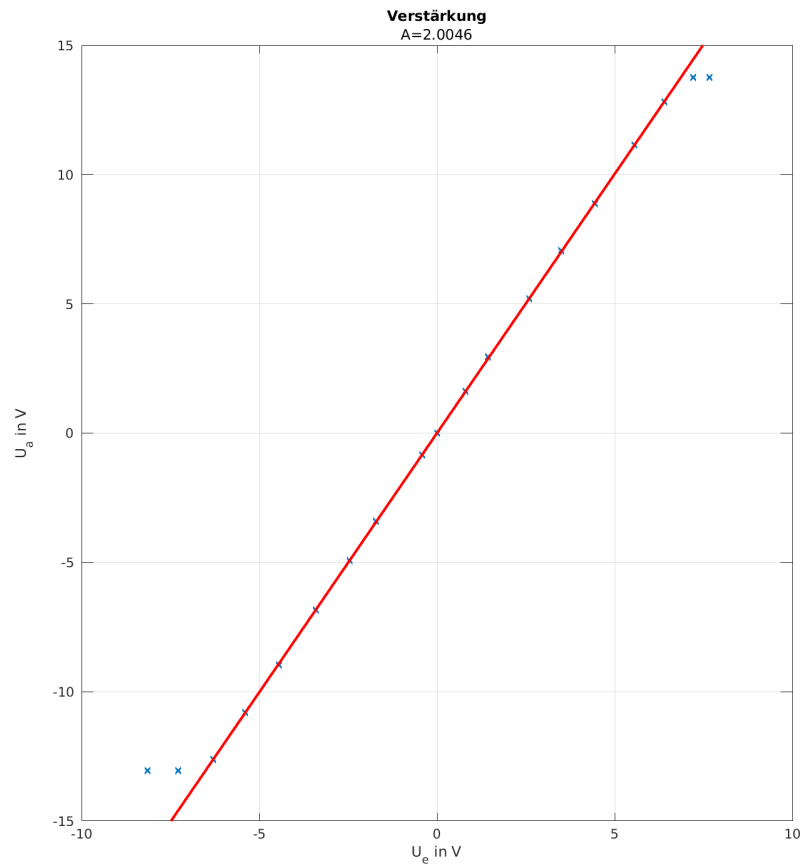


Figure 4: Plot der Aufgabe 4

4.3 Matlab Code

```
1 % Aktorik & Sensorik - WS 2020
2 %
3 % 1.4 Messung der Kennlinie des Leistungsverstärkers
4 %
5 % Datum:      27.10.2020
6 % Autoren:    Anton Kress,      S872899
7 %             Jan Abel,         S876662
8
9 clear
10 home
```

```

11 close all
12
13 FolderName = "./src/";
14 D_Name = "D.mat";
15 D = fullfile(FolderName, D_Name);
16
17 % Auslesen der EIngangs und Ausgangsspannungen
18 load(D);
19 D_plot=sortrows(D,1);
20
21 % lineares fitting im Arbeitsbereich
22 f4 = polyfit(D_plot(3:17,1), D_plot(3:17,2), 1);
23 % Verstärkung A entspricht Steigung der Gerade
24 A=f4(1,1)
25 % Erzeugung der Ausgleichsgerade
26 x1 = linspace(-10, 10);
27 y1 = polyval(f4, x1);
28
29 figure(1);
30 plot(D_plot(:,1), D_plot(:,2),'x', x1, y1, 'r','linewidth',2 );
31 axis([-10 10 -15 15])
32 title('Verstärkung')
33 subtitle(['A=' num2str(A)])
34 xlabel('U_e in V');
35 ylabel('U_a in V');
36 grid on
37
38 % save current plot to img/-folder
39 imagePath = fullfile('../img/', mfilename);
40 print(imagePath,'-dpng');

```