Protokoll Grundpraktikum II:
E2 - Innenwiderstand von Messgeräten

Sebastian Pfitzner

8. November 2013

Durchführung: Sebastian Pfitzner (553983), Anna Andrle (550727)
Arbeitsplatz: Platz 2
Betreuer: Martin Rothe
Versuchsdatum: 30.10.2013

Abstract

Ziel dieses Versuchs ist die Bestimmung der Innenwiderstände eines gegebenen Strom- und Spannungsmessers. Dies wird durch eine Schaltung erreicht, die für die Spannungsmessung einen Spannungsteiler mit einem variablen Widerstand und für die Strommessung eine Parallelschaltung aus Amperemeter und Regelwiderstand erzeugt. Bei konstanter Strom- bzw. Spannungsversorgung lässt sich dann aus den Spannungs-/Stromveränderungen in Abhängigkeit von der Veränderung des Widerstands der Innenwiderstand berechnen.

Inhaltsverzeichnis

1. Messwerte

1.1. Innenwiderstand des Spannungsmessgeräts

Für die vom Voltmeter angezeigte Spannung gilt bei konstanter Betriebsspannung U_B, einem Regelwiderstand R_X und einem Innenwiderstand von R_V nun

$$\frac{1}{U_V} = \frac{1}{U_B} \cdot \frac{R_X}{R_V} + \frac{1}{U_B} \tag{1}$$

Dieser Zusammenhang ergibt sich aus der Betrachtung des Spannungsmessers mit einem Ersatzschaltbild, sodass er sich aus einem Widerstand und einem idealen Spannungsmesser zusammensetzt. Dadurch ergibt sich ein Spannungsteiler - die gemessene Spannung entspricht dem Spannungsabfall über dem Innenwiderstand.

R_X [kΩ]	0	3	5	8	10	15	20	25	30	35	40	45	50	55	74	100
U_V [V]	20	17,9	16,8	15,1	14,2	12,5	11	10	9	8,2	7,5	7,1	6,6	6,1	5	4

Tab. 1: Messwerte für die Spannung in V in Abhängigkeit vom Regelwiderstand R_X in kΩ

Demzufolge ist $\frac{1}{U_V}$ linear von R_X abhängig. Diese Abhängigkeit lässt sich linear fitten und liefert zwei Parameter a_1 und a_2. Das Inverse von a_1 liefert das Produkt aus der Betriebsspannung und dem Innenwiderstand, während das Inverse von a_2 die Betriebsspannung ergibt. Als Messungenauigkeit wurde dabei von $\Delta U_V = 0{,}7\,\mathrm{V}$ ausgegangen - allerdings geht hier hauptsächlich der Anzeigefehler von $2,5\% \cdot 25\,\mathrm{V} = 0{,}62\,\mathrm{V}$ ein, da der Ablesefehler (auf 0,2 V abgeschätzt) demgegenüber fast zu vernachlässigen ist. Diese Ungenauigkeit muss noch durch Anwendung der gaußschen Fehlerfortpflanzung umgerechnet werden, da $\frac{1}{U_V}(R_X)$ gefittet wird. Es ergibt sich offensichtlich

$$\Delta\tilde{U}_{V_i} = \frac{1}{U_{V_i}} \cdot \Delta U_V \tag{2}$$

Der nachfolgende Fit der Messwerte in Tabelle 1 wurde mit der Matlab-Function `chisqlinfit`[1] erstellt. Aus den Fitparametern ergibt sich mit der oben genannten Beziehung:

$$R_V = \frac{1}{a_1 \cdot U_B} = 24{,}4\,\mathrm{k\Omega}$$

Mithilfe der Gaußschen Fehlerfortpflanzung lässt sich nun auch die Unsicherheit des Innenwiderstands des Voltmeters berechnen:

$$\Delta R_V = \sqrt{\left(-\frac{1}{a_1^2 \cdot U_B}\right)^2 \cdot \Delta a_1^2 + \left(-\frac{1}{a_1 \cdot U_B^2}\right)^2 \cdot \Delta U_B^2}$$

[1] Quellcode auf http://people.physik.hu-berlin.de/~pfitzseb/matlab.html und im Anhang

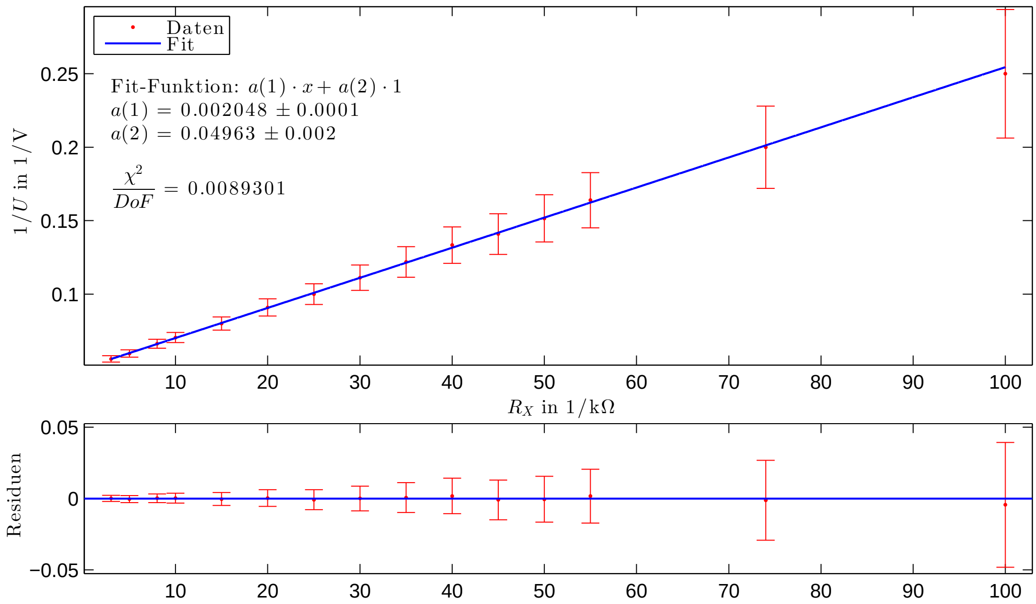

Abb. 1: Linearisierte Darstellung der Spannung in Abhängigkeit vom Widerstand

Als Gesamtergebnis ergibt sich also

$$R_V = (24 \pm 1)\text{k}\Omega \tag{3}$$

Aus dem Inversen von a_2 ergibt sich für die Betriebsspannung $U_{B_{fit}} = (20{,}1 \pm 0{,}6)$V. Dieser Wert stimmt gut mit dem angezeigten Wert von $U_{B_{anz}} = (20{,}0 \pm 0{,}1)$V überein und bestätigt damit also den Fit.

1.2. Innenwiderstand des Strommessgeräts

Die im Skript dargestellte Schaltung sorgt dafür, dass der Strom I_0 in Abhängigkeit vom Verhältnis der Widerstände auf das Amperemeter und den Regelwiderstand aufgeteilt wird, also folgende Beziehung gilt (mit R_X als Regelwiderstand, R_A als Innenwiderstand und I_X und I_A als den zugehörigen Strömen):

$$\frac{1}{I_A} = \frac{1}{I_0} \cdot \frac{R_A}{R_X} + \frac{1}{I_0} \tag{4}$$

Es lässt sich also offensichtlich wiederum ein linearer Fit der Form $\frac{1}{I_A}(\frac{1}{R_X})$ aus den Daten in Tabelle 2 erstellen. Auch hier muss die Unsicherheit der gemessenen Stromstärke ähnlich wie in Gleichung (2) umgerechnet werden:

$$\Delta \tilde{I}_{A_i} = \frac{1}{I_{A_i}} \cdot \Delta I_A \tag{5}$$

R_X [kΩ]	0,1	0,2	0,4	0,5	0,8	1	1,5	2	2,5	3	3,5	4	4,5	6	7	8,4	∞
I_A [µA]	8	15	26	29	40	45	55	61	66	70	72,5	75	76	81	82,5	85	96

Tab. 2: Messwerte für den Strom in µA in Abhängigkeit vom Regelwiderstand R_X in kΩ

ΔI_A wird mit 3 µA abgeschätzt, denn das Messgerät hat eine Anzeigegenauigkeit von $1,5\% \cdot 100$ µA, zu dem noch eine Ableseungenauigkeit in der gleichen Größenordnung dazukommt. Die pythagoreische Addition liefert dann den obengenannten Wert. Der Fit mit `chisqlinfit` liefert nun ebenfalls zwei Paramter b_1 und b_2. Aus b_1 lässt sich nun ohne weiteres der Innenwiderstand R_A berechnen, das Inverse von b_2 ist die Quellstromstärke. Für die Berechnung von R_A und seine Ungenauigkeit

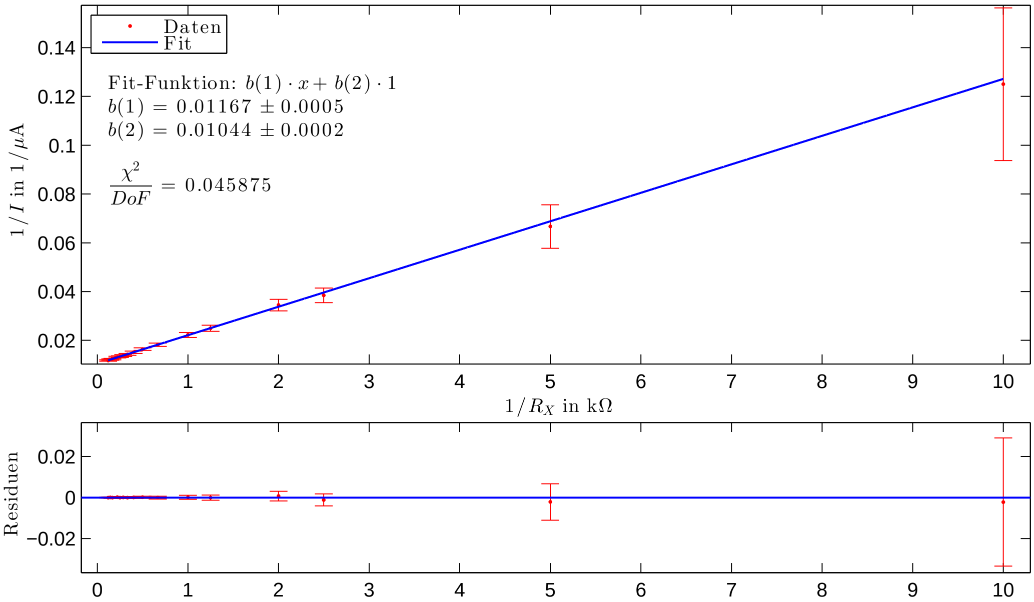

Abb. 2: Linearisierte Darstellung des Stroms in Abhängigkeit vom Regelwiderstand

aus b_2 ergeben sich folgende Beziehungen:

$$R_A = b_1 \cdot I_0$$
$$\Delta R_A = \sqrt{(I_0 \cdot \Delta b_1)^2 + (b_1 \cdot \Delta I_0)^2}$$

Die Quellstromstärke entspricht hier dem in der Tabelle 2 für einen unendlichen Widerstand eingetragenen Strom. Außerdem lässt sie sich auch aus dem Parameter b_2 des Fits ableiten, aus dem sie sich zu (96 ± 2)µA ergibt, was auch diesen Fit bestätigt. ΔI_0 entspricht also der aus dem Fit ermittelten Unsicherheit und wird als konstant angenommen, da der vorgeschaltete Widerstand R_0 die Last so gering hält, dass die Spannungsquelle innerhalb ihrer Spezifikationen betrieben wird und damit als Konstantstromquelle angesehen werden kann. Für den Innenwiderstand und seiner Unsicherheit ergibt sich nach der Rechnung folgender Wert

$$R_A = (1{,}12 \pm 0{,}05)\mathrm{k}\Omega$$

2. Auswertung

Die durchgeführten Fits führen - wie die oben durchgeführten Rechnungen bzgl. des zweiten Parameters bestätigen - auf realistische Werte. Allerdings sind die $\frac{\chi^2}{DoF}$-Werte äußerst klein, denn ein Wert von 1 ist der Erwartungswert dieser mit der Anzahl der Freiheitsgrade normierten Chi-Quadrat-Verteilung. Solch ein kleiner Wert deutet auf zu groß gewählte Unsicherheiten der Messgröße hin. Allerdings müssten diese um eine Größenordnung kleiner sein um ein $\frac{\chi^2}{DoF}$ nahe 1 zu erzeugen, was sich aus dem Experiment aber nicht ableiten lässt.

Die Unsicherheiten des Regelwiderstands R_X wurden als vernachlässigbar klein angenommen, da der Fehler eines Dekadenwiderstands laut blauem Skript bei $\Delta R = 0{,}02\,\Omega + 0,0003 \cdot R$ liegt, was für einen Widerstand von $10\,\mathrm{k}\Omega$ eine Unsicherheit von $3\,\Omega$ zur Folge hat. Die Messgeräte haben andererseits - relativ zum Messwert - eine um zwei Größenordnungen größere Ungenauigkeit, sodass obige Annahme gerechtfertigt erscheint.

Die Korrelation der Fitparameter wurde bei den Fehlerrechnungen nicht berücksichtigt, da ihr Einfluss in der Wurzel der gaußschen Fehlerrechnung in der Größenordnung von 10^{-5} liegt und ist damit mehrere Größenordnungen kleiner als die anderen Terme, die bei 10^{-3} (Innenwiderstand des Amperemeters) bzw. 10^{0} (Innenwiderstand des Voltmeters).

Bei der Strommessung sorgt der vorgeschaltete Widerstand R_0 dafür, dass die Last nicht - im Rahmen des Innenwiderstands der Spannungsquelle - beliebig hoch werden kann und damit ein nahezu konstanter Quellstrom gewährleistet werden kann. Wird dieser Widerstand zu klein gewählt und eine Messung bei einem geringen Widerstand R_X durchgeführt, so ist es durchaus möglich, dass die Stromstärke I_0 sich verändert und der Messwert demzufolge nicht zu gebrauchen ist. Demzufolge ist R_0 mit $200\,\mathrm{k}\Omega$ sehr groß gewählt, so dass die maximalen $\approx 100\,\mathrm{k}\Omega$ die Last nicht mehr stark verändern und der Strom sich somit kaum verändert. Dies erlaubt es, diese Spannungsquelle hier als Konstantstromquelle zu betrachten.

Die im Experiment verwendeten Schaltungen können auch zur Messbereichserweiterung dienen, wenn der Innenwiderstand des jeweiligen Messgeräts bekannt ist. Ist beispielsweise der in Reihe (Spannungsmessung) bzw. parallel (Strommessung) geschaltete Widerstand genauso groß wie der Innenwiderstand des Messgeräts, wird der Messbereich verdoppelt. Bei der Reihenschaltung fällt über beiden Widerständen die gleiche Spannung ab, so dass das doppelte der Eingangsspannung gemessen werden kann, während bei der Strommessung sowohl das Messgerät als auch der parallel geschaltete Widerstand vom gleichen Strom durchflossen werden, was die gleiche Messbereichserweiterung zur Folge hat.

A. chisqlinfit

```
function [ p, err_o, chisqdof, res, kov ] = chisqlinfit( x, y, err, fun )
%CHISQLINFIT Fits function (linear in parameter) to data by minimizing Chi^2
%
%
%   [ p, err_o, chisqdof, res, kov ] = chisqlinfit( x, y, err, fun )
%
%
%   INPUT
%
%   x - vector of independent variable.
%   y - vector of dependent variable.
%   err - vector of error of y. If shorter than x, last known error will be
%         used for all following data.
%   fun - anonymous function ( e.g. @(q,x) q(1)*x+q(2)... ). q will be
%         optimized so that the sum of weighted residues is minimal.
%         Must be linear in q.
%         As long as I don't optimize the recognition: Only use a function
%         of the form <param>*<expr>+<p2>*<expr>.
%         Syntax is important!
%         Examples:
%           DO: @(q,x) q(1)*1+q(2)*(x+1).^2+q(3)*x.^4
%               @(param,x) param(1)*x+param(2)*1
%           DON'T: @(q,x) q(1)*x.^2*q(2)
%                  @(x,q) q(1)/x*q(2)+5
%
%
%  OUTPUT
%
%   p - vector of optimized parameters
%   chisqof - minimized Chi^2/DoF
%   err_o - uncertainty of fitted parameters
%   res - residues, vector of same length as x,y,err
%   kov - covariance-matrix, for determining the correlation of the
%         parameters
%
%
%  EXAMPLE
%
%   fun=@(a,x) a(1)*1-a(2)*(x+2).^2+a(3)*x.^3;  % define anonymous function
%   q=10;
%   x=1:.1:q;                                   % set steps
%   q=length(x);
%   ya=fun([40 5 2 ],x);                        % generate data
%   y=ya+randn(1,q)*5;                          % wiggle a little
%   err=5;                                      % define error
```

```matlab
%   [p, err_o, chi, res, kov]=chisqlinfit(x,y,err,fun);     %fit
%
%
%  AUTHOR
%
%   Sebastian Pfitzner
%   pfitzseb @ physik . hu - berlin . de
%   last updated: 7.11.2013
%
%
%  THANKS TO
%
%   Jannick Weisshaupt, Franziska Flegel

%% ------ erster Input-Check -------------------------------------------
if nargin < 4
    error('Not enough input!')
end

%% ------ x und y berichtigen ------------------------------------------
if length(x)<length(y)
    y=y(1:length(x));
    warning('y-data too long. Discarding additional values.')
elseif length(x)>length(y)
    x=x(1:length(y));
    warning('x-data too long. Discarding additional values.')
end

if length(x)~=length(y)
    error('X and Y are not of equal length!')
end

% x, y, err als Vektor in gleicher Richtung zusammenbauen

[a1, b1]=size(x);
if a1<b1,
   x=x';
end

[a1, b1]=size(y);
if a1<b1,
   y=y';
end

[a1, b1]=size(err);
if a1<b1,
   err=err';
end

%% ------ Fehlerangaben berichtigen ------------------------------------
if length(err)<length(y)
    disp([' [f]-> length(err)<length(y), filling undefined errors with '...
                                            'last known uncertainty.'])
    err(end:length(y))=err(end);
```

```
elseif length(err)>length(y)
    disp(' [f]-> length(err)>length(y), ignoring additional errors.')
    err=err(1:length(y));
end

if any(err==0)
    warning(' [f]-> err_i = 0 found. Replacing with err_i=sqrt(y_i).')
    ind=find(err==0);
    err(ind)=sqrt(y(ind));
end

[a1, b1]=size(err);
if a1<b1,
   err=err';
end

%% ------- Input verarbeiten, DoF berechnen, Funktion splitten --------------
%          ziemlich optimierungsbeduerftig

fstr=func2str(fun);
param=tabulate(regexp(fstr,'\w+','match'));
                                     % nach Parametern und Variablen suchen
mat=tabulate(regexp(fstr,[param{1,1} '\(\d+\)'],'match'));
var=param{2,1};
par=param{1,1};
pc=length(mat(:,1));
fstrcut=fstr(regexp(fstr,')')+1:end);
funmatstr=regexp(fstrcut,[par '\(\d+\)\W'],'split');
                                          % in einzelne Funktionen splitten
funmatstr=funmatstr(2:end);
dof=length(x)-pc-1;          % Freiheitsgrade
[m,~]=size(mat);
funmat=cell(1,m);
for i=1:length(funmatstr)
    funmatstr{i}=regexprep(funmatstr{i}, '+$|-$', '');
                                                     % +/- am Ende entfernen
    funmatstr{i}=regexprep(funmatstr{i}, '+$|-$', '');
    funmatstrinner=regexprep(funmatstr{i},[param{1,1} '\(\d+\)'],'');
                                                       % Parameter entfernen
    if isempty(funmatstrinner)
        funmatstrinner='x./x';       % nicht x-abhaengige Funktion bauen
    end
    funmatstr{i}=['@(' var ')' funmatstrinner];
    funmat{i}=str2func(funmatstr{i});% Liste mit einzelnen Funktionen bauen
end

%% ------- Lineares Fitten -------------------------------------------------

H=ones(m,length(x));         % init
for i=1:m          % Matrix mit Loesungen der Gleichungen fuer alle x fuellen
    H(i,:)=funmat{i}(x);
end
H=H';                        % transponieren

```

```
V=diag(err,0).^2;          % Diagonalmatrix mit Unsicherheiten der y fuellen

U=H'*(V\H);                % Kovarianz-Matrix
U=U\eye(size(U));          % invertieren
p=U*(H'*(V\y));            % Loesen des Gleichungssystems fuer die Parameter
kov=U;

err_o=sqrt(diag(kov));
chisqdof=chi2(x,y,err,fun,p)/dof;    % Chi^2 berechnen

res=y-fun(p,x);             % Residuen berechnen

if chisqdof>5
    errmsg=[' [f]-> Fit failed. Adjust start points or function! Chi^2/DoF='...
                                                                  num2str(chisqdof)];
    warning(errmsg)
elseif chisqdof<0.1
    errmsg=[' [f]-> Adjust errors! Chi^2/DoF=' num2str(chisqdof)];
    warning(errmsg)
end
end

%% ------- Hilfsfunktionen -------------------------------------------------

function [chi] = chi2(x,y,err,fun,p)
    chi=sum( ((fun(p,x)-y)./err).^2 );
end
```