

Projekt B

Entwicklung einer MATLAB Applikation zur Darstellung der Schubmittelpunktberechnung am Beispiel eines Lehrmodells



Stefanie Borz	M111644
Betreuer und Prüfer:	Prof. Dr.-Ing. T. Pawletta, FIW/MVU, FG CEA Prof. Dr.-Ing. Schmidt, FIW/MVU
Abgabetermin:	17.06.2011

Inhaltsverzeichnis

	Inhaltsverzeichnis	II
	Abbildungsverzeichnis.....	III
	Aufgabenstellung	4
1	Problemstellung	5
2	Systemanalyse des Lehrmodells	6
2.1	Konstruktiver Aufbau	6
2.2	Der Schubmittelpunkt	7
3	Mathematische Modellierung des Profiltypen	8
3.1	Biegung am Ständer	8
3.2	Biegung am U-Profil	8
3.3	Schubmittelpunkt der Querkraft.....	9
3.4	Schwerpunkt am U-Profil	10
3.5	Torsion am U-Profil	10
3.6	Verdrehwinkel des U-Profils	11
3.7	Biegung an der Querstange	11
4	Entwurf, Implementierung und Test der MATLAB Applikation.....	13
4.1	Programmstruktur und Implementierungsgrundlagen	13
4.2	Benutzeroberfläche	16
4.3	Test anhand einer Beispielrechnung	17
5	Zusammenfassung und Ausblick	18
	Das Literaturverzeichnis gehört hinter "Zusammenfassung und Ausblick"	
	Anhang.....	19
A	MATLAB Skript Verdrehwinkel zur Berechnung von Verdrehwinkeln	19
B	MATLAB GUI Funktion Schubmittelpunktmodell für die graphische Benutzeroberfläche	20
C	MATLAB Funktion Berechnung	24
D	MATALB Funktion dreiDAnsicht zur graphische Darstellung des U- Profils.....	25
E	MATLAB Skript dreiDAnsicht2 für die graphische Darstellung des U- Profils mit einer Stärke t	26
	Das Literaturverzeichnis muss vor den Anhang!!!	
	Literaturverzeichnis	27
	Eidestattliche Erklärung.....	28

Abbildungsverzeichnis

Abbildung 1: Skizze des Schubmittelpunktmodells nach Projekt A.....	6
Abbildung 2: Beispiele für sternförmige Profile	7
Abbildung 3: Beispiele für einfach symmetrische Profile.....	7
Abbildung 4: Beispiele für doppelt symmetrische Profile	7
Abbildung 5: 3D Darstellung des Ständers mit der wirkenden Kraft.....	8
Abbildung 6: 3D Darstellung des U-Profils in Abhängigkeit von der Kraft.....	9
Abbildung 7: Querkraft und Schubmittelpunkt am U-Profil.....	9
Abbildung 8: Koordinatensystem zur Schwerpunkts Bestimmung.....	10
Abbildung 9: systematische Darstellung des Torsionsmomentes am U-Profil	10
Abbildung 10: Verdrehwinkel des U-Profils in Bezug auf die Gewichtskraft.....	11
Abbildung 11: Krafteinwirkung an der Querstange	11
Abbildung 12: Skizze zur Herleitung der Matrix mit der Dicke t (siehe Anhang E).....	14
Abbildung 13: Skizze zur Herleitung der Matrix (siehe Anhang D).....	15
Abbildung 14: graphische Benutzeroberfläche	16

Aufgabenstellung

Thema:

Entwicklung einer MATLAB Applikation zur Darstellung der Schubmittelpunktberechnung am Beispiel eines Lehrmodells

Am Beispiel eines vorgegebenen Lehrmodells ist eine GUI-gestützte MATLAB Applikation zur Darstellung der Schubmittelpunktberechnung für ausgewählte Profiltypen zu entwickeln.

Schwerpunkte:

- Konstruktive Analyse des gegebenen Lehrmodells
- Mathematische Modellierung ausgewählter Profiltypen
- Entwurf und Implementierung einer GUI-basierten MATLAB Applikation
- Test und Dokumentation der Implementierung

Die Arbeit ist nach den vorgegebenen Richtlinien anzufertigen. Neben der geforderten Anzahl schriftlicher Exemplare ist die Arbeit (Originaldateien & PDF-Dateien) sowie die entwickelte Software auf einem dokumentierten digitalen Datenträger abzugeben.

Betreuer und Prüfer: Prof. Dr.-Ing. Thorsten Pawletta, FIW/MVU, FG CEA
Prof. Dr.-Ing. Ina Schmidt, FIW/MVU
Hochschule Wismar

Abgabetermin: 17.06.2011

1 Problemstellung

Mit diesem Projekt soll eine GUI Benutzeroberfläche mit Hilfe von MATLAB erstellt werden. Ziel soll es sein, durch die Eingabe von Variablen eines U-Profils, ein Ergebnis zum Verdrehwinkel und zum Schubmittelpunkt zu erhalten, ohne sich mit den Hintergründen der Technischen Mechanik befassen zu müssen.

Die ersten Informationen zum konstruktiven Aufbau, zum Schubmittelpunkt und der mathematischen Modellierung des U-Profils gehen aus einem vorlaufenden Projekt hervor. Anschließend wurden mehrere MATLAB Funktionen entwickelt, um die Benutzeroberfläche darzustellen. Zusätzlich wurde eine Abbildung, als ergänzendes Material, zum besseren Verständnis der Eingabeparameter hinzugefügt. Nachdem die Berechnung erfolgt ist, wird das U-Profil durch Länge, Breite und Höhe graphisch dargestellt. Durch eine Testrechnung werden vorherige Schritte überprüft.

2 Systemanalyse des Lehrmodells

Das Schubmittelpunktmodell, welches aus dem Projekt A hervorgeht, ist eine mögliche Variante, die im Folgenden erläutert wird. Anschließend werden grundsätzliche Definitionen erklärt und verschiedenen Profiltypen gezeigt.

2.1 Konstruktiver Aufbau

Dieses Projekt baut auf dem Projekt A „Konstruktion eines Lehrmodells zur Veranschaulichung des Schubmittelpunktes“ auf. Die Anforderungen an das Modell waren zum Beispiel die Transportabilität.

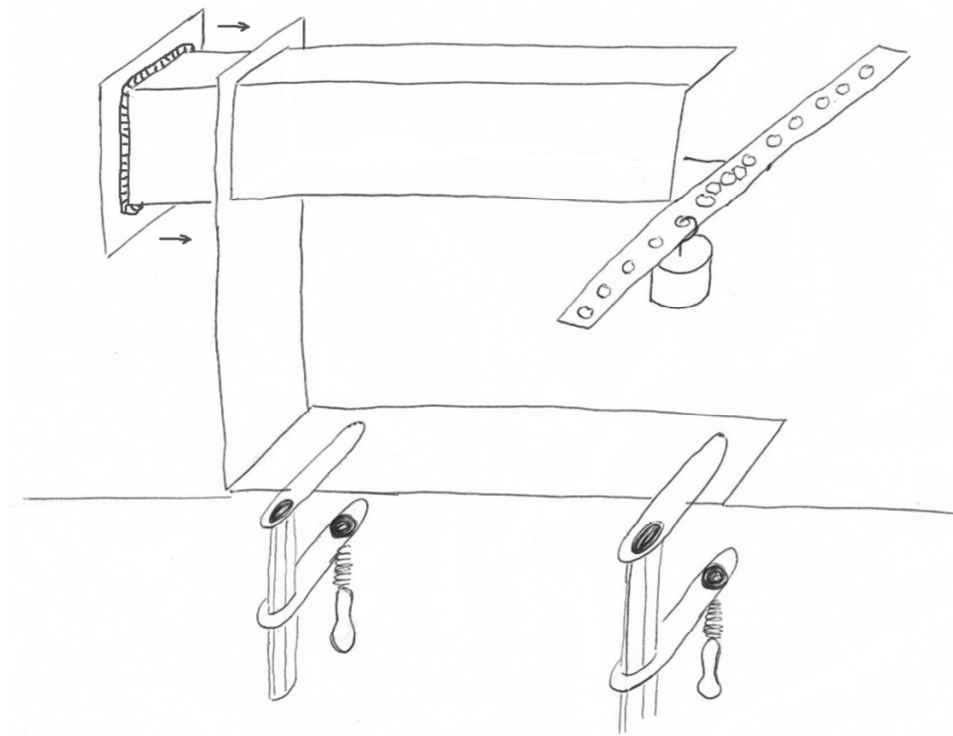


Abbildung 1: Skizze des Schubmittelpunktmodells nach Projekt A

Das Projekt A hat im Ergebnis gezeigt, dass der Verdrehwinkel erst über eine längere Strecke visuell wahrnehmbar ist. Als Beispiel: beträgt die Länge 5m, mit einem entstehenden Verdrehwinkel von 10° . Bei einer Länge von 0,5m ergibt sich ein Verdrehwinkel von 1° . Dieser kleine Verdrehwinkel kann beispielsweise durch eine Lichtprojektion auf eine Hintergrundfläche sichtbar gemacht werden.

2.2 Der Schubmittelpunkt

Der Schubmittelpunkt M , auch Querkraftmittelpunkt oder Drillpunkt genannt, ist der Punkt eines Querschnitts, durch den die Wirkungslinie der Querkräfte Q_z verlaufen muss, sodass die Kraft keine Verdrehung des Profils bewirkt, beziehungsweise damit keine Torsionsbeanspruchung auftritt. [1]

Grundlegend kann der Schubmittelpunkt anhand von geometrischen Gegebenheiten in Gruppen unterteilt werden:

Sternförmige Profile

Bei sternförmigen Querschnitten befindet sich der Schubmittelpunkt im Schnittpunkt der einzelnen geraden Segmente des Profils.

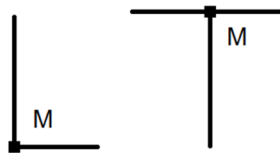


Abbildung 2: Beispiele für sternförmige Profile

Einfach symmetrische Profile

Bei einfach symmetrischen Querschnitten befindet sich der Schubmittelpunkt auf der Symmetrieachse.

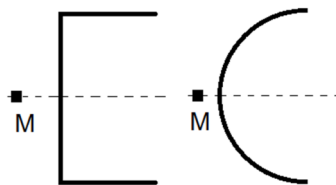


Abbildung 3: Beispiele für einfach symmetrische Profile

Doppelt symmetrische Profile

Bei doppelt symmetrischen Querschnitten befindet sich der Schubmittelpunkt im Schnittpunkt der beiden Symmetrieachsen und damit im Schwerpunkt.

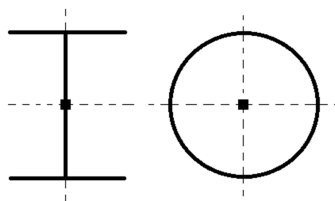


Abbildung 4: Beispiele für doppelt symmetrische Profile

3 Mathematische Modellierung des Profiltypen

Die ausführlichen Herleitungen sind aus der Quelle [1] zu entnehmen. Im Folgenden werden nur die Ergebnisse aufgeführt.

Der Werkstoff ist Edelstahl X2CrNiMo17-12-2, weil dieser nicht rostet und für einfache Maschinenteile gut geeignet ist. Das Material besitzt damit eine Zugfestigkeit R_m von 500 bis 700 N/mm^2 und eine Streckenlast R_e von 200 N/mm^2 bei einer Dicke bis zu 160 mm .

3.1 Biegung am Ständer

Die Biegespannung σ_b darf nicht größer sein als die zulässige Biegespannung σ_{bzul} .

$$\sigma_b < \sigma_{bzul} = 200 N/mm^2 \quad (3.1)$$

Die Formel zur Berechnung der Biegespannung σ_b setzt sich aus der Gewichtskraft F_G , der Länge l des U-Profils, der Dicke t_s und der Breite b des Ständers zusammen.

$$\sigma_b = 6 \cdot F_G \cdot l \cdot t_s / (b \cdot t_s^3) \quad (3.2)$$

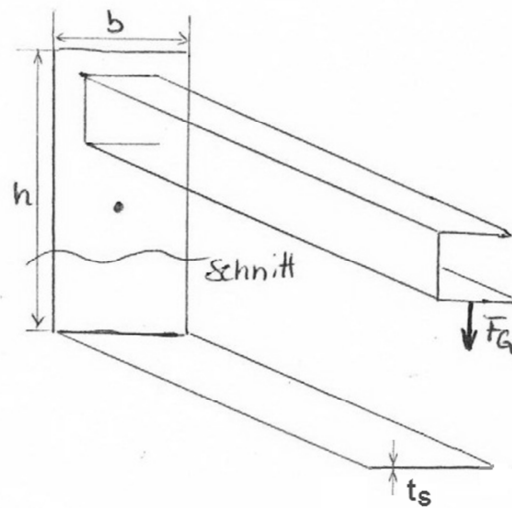


Abbildung 5: 3D Darstellung des Ständers mit der wirkenden Kraft

3.2 Biegung am U-Profil

Für die folgende Berechnung gilt ebenfalls die Formel 3.1, wobei sich die Biegespannung nun aus der Gewichtskraft F_G , dem Flächenträgheitsmoment I_{yy} , der Länge l und der Höhe des U-Profils zusammen setzt.

$$\sigma_b = F \cdot l \cdot H / (2 \cdot I_{yy}) \quad (3.3)$$

Das Flächenträgheitsmoment I_{yy} ergibt sich aus dem Verhältnis der Höhe H , der Breite B und der Dicke t des U-Profils.

$$I_{yy} \approx t \cdot H^3 / 12 + B \cdot t^3 / 6 + H^2 \cdot B \cdot t / 2 \quad (3.4)$$

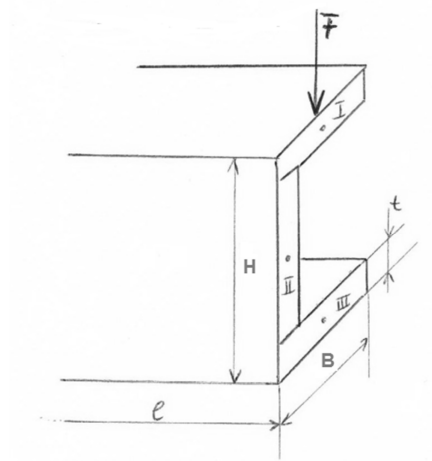


Abbildung 6: 3D Darstellung des U-Profils in Abhängigkeit von der Kraft

3.3 Schubmittelpunkt der Querkraft

Der Schubmittelpunkt y_M setzt sich aus der Höhe H , der Breite B und der Dicke t des U-Profils und dem Flächenträgheitsmoment I_{yy} zusammen.

$$y_M = H^2 \cdot B^2 \cdot t / (4 \cdot I_{yy}) \quad (3.5)$$

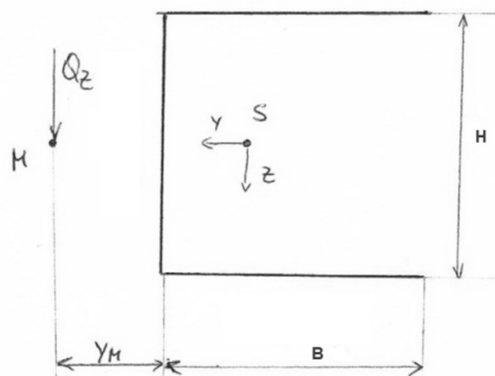


Abbildung 7: Querkraft und Schubmittelpunkt am U-Profil

3.4 Schwerpunkt am U-Profil

Der Schwerpunkt wird im Koordinatensystem in Abbildung 8 durch die Abstände z_s und y_s bestimmt.

$$z_s = H/2 \quad (3.6)$$

$$y_s \approx (B \cdot H + B^2)/(H + 2 \cdot B) \quad (3.7)$$

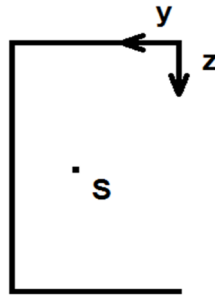


Abbildung 8: Koordinatensystem zur Schwerpunkts Bestimmung

3.5 Torsion am U-Profil

Die Torsion τ_{max} wird durch die Gewichtskraft F_G , dem Abstand c der Querstange und dem Torsionsmoment I_t bestimmt, wobei auch hier die Bedingung aus der Formel 3.1 gültig ist.

$$\tau_{max} = F_G \cdot c \cdot t / I_t \quad (3.8)$$

Das Torsionsmoment I_t setzt sich aus der Höhe H , der Breite B und der Dicke t des U-Profiles zusammen.

$$I_t = (2/3 \cdot B \cdot t^3 + 1/3 \cdot H \cdot t^3) \quad (3.9)$$

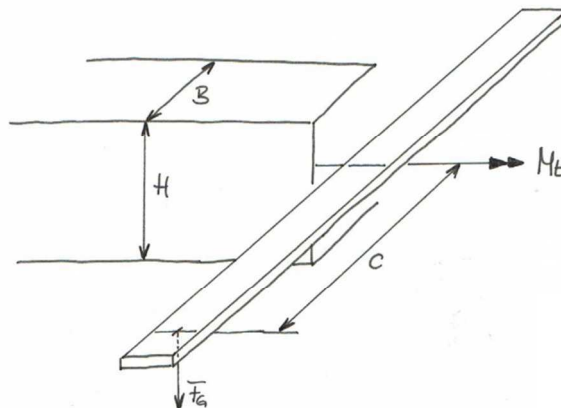


Abbildung 9: systematische Darstellung des Torsionsmomentes am U-Profil

3.6 Verdrehwinkel des U-Profils

Der Verdrehwinkel φ ergibt sich aus der Gewichtskraft F_G , der Länge l des U-Profils, dem Abstand c der Querstange, dem Elastizitätsmodul E und dem Torsionsträgheitsmoment I_t .

$$\varphi = l \cdot F_G \cdot c / (E \cdot I_t) \quad (3.10)$$

Der E-Modul beträgt für den gewählten Werkstoff $2.1 \cdot 10^5 \text{ N/mm}^2$ und das Torsionsmoment I_t wurde bereits in Kapitel 3.5 erläutert.

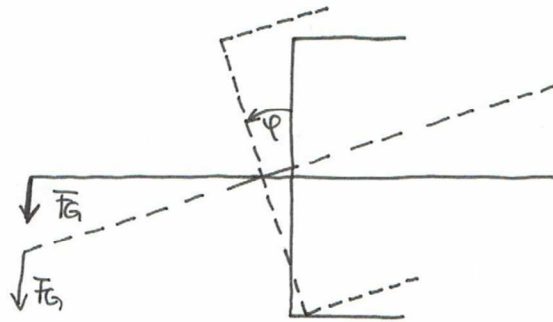


Abbildung 10: Verdrehwinkel des U-Profils in Bezug auf die Gewichtskraft

3.7 Biegung an der Querstange

Das Biegemoment σ_b setzt sich zusammen aus der Gewichtskraft F_G , der Länge c , der Breite d und der Dicke t_Q der Querstange. Für die Berechnung an der Querstange gilt die Formel 3.1 ebenso.

$$\sigma_b = 6 \cdot F_G \cdot c / (d \cdot t_Q^2) \quad (3.11)$$

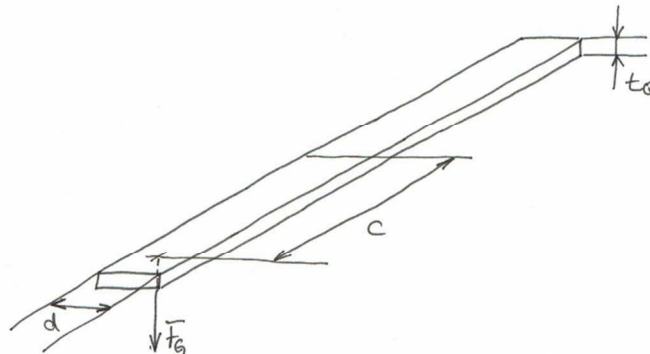


Abbildung 11: Krafteinwirkung an der Querstange

Durch das MATLAB-Programm, siehe Anhang A aus Projekt A, hat sich gezeigt, dass die Spannungen am Ständer und der Querstange nicht die ausschlaggebenden Größen sind, sodass bei der in MATLAB zu entwickelnden Oberfläche nur auf die Spannung am U-Profil Bezug genommen wird. Durch diese Annahme kann die zulässige Spannung aus (3.1) mit der Torsionsspannung aus (3.8) gleichgesetzt werden.

$$200 = \tau_{max} = F_G \cdot c / I_t$$

Wird nun das Torsionsmoment I_t mit der Formel (3.9) ersetzt, ergibt sich

$$200 = F_G \cdot c / (2/3 \cdot B \cdot t^2 + 1/3 \cdot H \cdot t^2).$$

Anschließend wird nach der Breite B umgestellt:

$$B = (F_G \cdot c / 200 - 1/3 \cdot H \cdot t^2) / (2/3 \cdot t^2). \quad (3.12)$$

Nun kann mit der Breite B der Verdrehwinkel φ errechnet werden.

$$\varphi = l \cdot F_G \cdot c / (E \cdot (2/3 \cdot B \cdot t^3 + 1/3 \cdot H \cdot t^3)) \quad (3.13)$$

4 Entwurf, Implementierung und Test der MATLAB Applikation

Das MATLAB-Programm soll die mathematischen Grundlagen des U-Profils einbeziehen und am Ende eine einfache Bedienung für jedermann ermöglichen. Die Benutzeroberfläche muss also veränderbare Eingabeparameter beinhalten. Diese werden zum besseren Verständnis durch eine Abbildung des Schubmittelpunktmodells erläutert. Die Ausgabeparameter werden zusätzlich mittels einer graphischen Darstellung des U-Profils ergänzt. Die Graphik sollte mit den vorgegeben und entstandenen Maßen dargestellt werden.

4.1 Programmstruktur und Implementierungsgrundlagen

Das MATLAB-Programm besteht aus drei Funktionen.

```
function Schubmittelpunktmodell  
function Berechnung(cmd)  
function dreiDAnsicht
```

Die Funktion `Schubmittelpunktmodell` definiert die graphische Benutzeroberfläche, welche in Anhang B zu finden ist. Durch den Term `uicontrol` wird eine Kontrollfläche erzeugt. Diese Fläche kann zum Beispiel durch folgende Begriffe in ihrer Gestalt und Funktionalität beeinflusst werden:

<i>Style</i>	<code>edit</code> / Textfeld welches veränderbar ist <code>text</code> / Textfeld <code>pushbutton</code> / Druckknopf
<i>Position</i>	<code>[x0 y0 Breite Höhe]</code> , interpretiert werden die Werte durch <i>Units</i>
<i>Units</i>	<code>pixel</code> / Standardeinstellung <code>normalized</code> / Position als Anteile der aktuellen Fensterbreite
<i>String</i>	Beschriftung
<i>Callback</i>	Objekt wird eine Aktion zugewiesen

Mit dem Start-Button wird die Funktion zur Berechnung aufgerufen. Welche die Formeln (3.5), (3.12) und (3.13) zur Berechnung des Schubmittelpunktes, der Breite und des Verdrehwinkels verwendet. Durch `switch` bleibt die Möglichkeit offen weitere Fälle hinzuzufügen. (siehe Anhang C)

Zur graphischen Darstellung des U-Profils wird die Funktion `dreiDAnsicht` genutzt. Das Programm ist im Anhang D zu finden. Zu Zwecken der Erklärung wird erst das

Programm aus Anhang E erläutert. Hier wird durch `surf` aus zwei Linien eine Fläche gebildet.

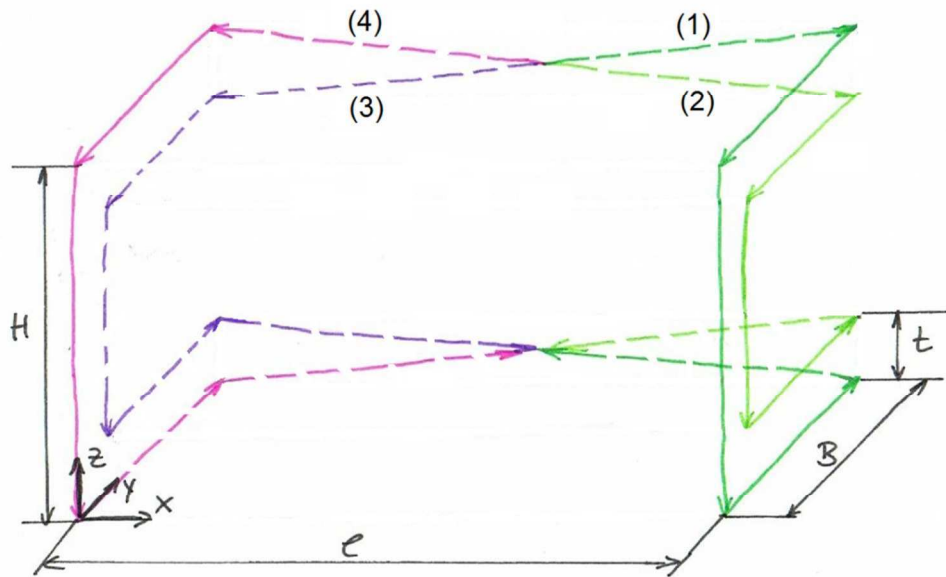


Abbildung 12: Skizze zur Herleitung der Matrix mit der Dicke t (siehe Anhang E)

Jeder Punkt kann im Koordinatensystem durch eine x , y und z Koordinate beschrieben werden. Diese Koordinaten werden in drei Matrizen zusammengefasst. In der ersten Matrix befinden sich in der ersten Zeile die x Koordinaten der Linie (1), in der zweiten die x Koordinaten der Linie (2) usw. In der zweiten Matrix sind die y Koordinaten und in der dritten die z Koordinaten definiert, analog zur ersten Matrix.

```
x=[1/2,1,1,1,1,1,1/2;
   1/2,1,1,1,1,1,1/2;
   1/2,0,0,0,0,0,1/2;
   1/2,0,0,0,0,0,1/2;
   1/2,1,1,1,1,1,1/2];
y=[B,B,0,0,B,B;
   B,B,t,t,B,B;
   B,B,t,t,B,B;
   B,B,0,0,B,B;
   B,B,0,0,B,B];
z=[H-t/2,H,H,0,0,t/2;
   H-t/2,H-t,H-t,t,t,t/2;
   H-t/2,H-t,H-t,t,t,t/2;
   H-t/2,H,H,0,0,t/2;
   H-t/2,H,H,0,0,t/2];
```

Wird das Prinzip nun auf das U-Profil ohne die Dicke t übertragen, ist zu beachten, dass Linie (1) weniger Koordinaten benötigt als Linie (2).

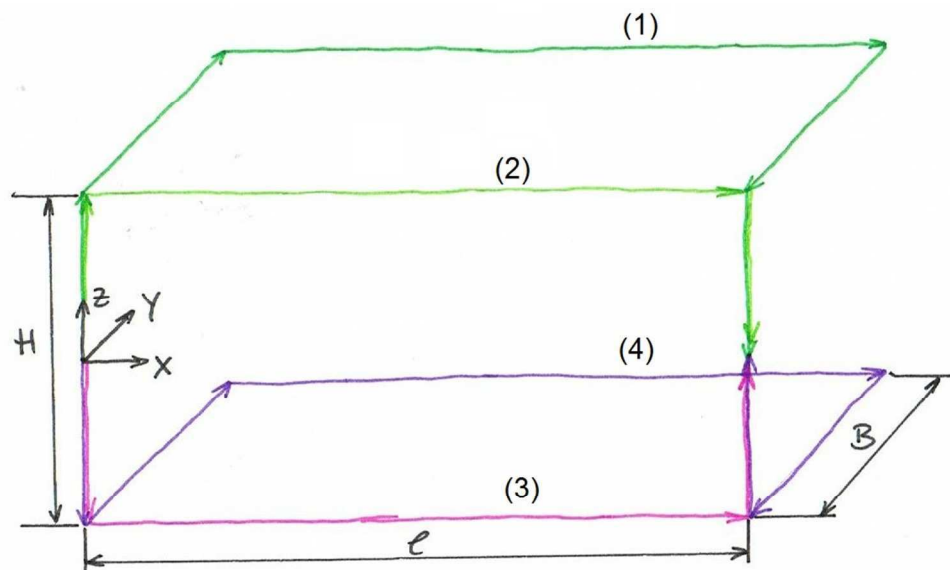


Abbildung 13: Skizze zur Herleitung der Matrix (siehe Anhang D)

Dennoch muss in einer Matrix die Spaltenanzahl stets übereinstimmen. Deshalb wird bei der Beschreibung der Linie (1) jeweils an den Eckpunkten verweilt, somit werden die Koordinaten doppelt aufgeschrieben.

4.2 Benutzeroberfläche

Durch Aufruf der Funktion `Schubmittelpunkt` wird das Programm am MATLAB-Prompt gestartet und das GUI gemäß Abbildung 14 geöffnet.

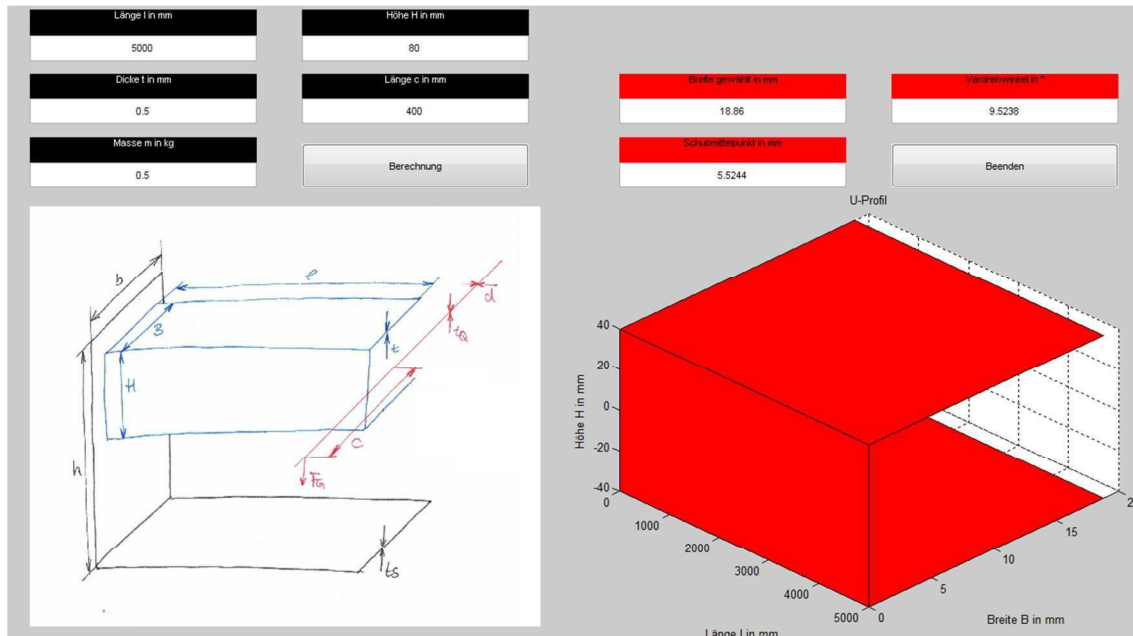


Abbildung 14: graphische Benutzeroberfläche

Das GUI öffnet sich mit einer Initialparametrierung. Die schwarzen Felder stellen Kommentare dar, welche die in den weißen Feldern einzugebenden Parameter erläutern. Zur Orientierung befindet sich unterhalb eine Skizze. Sind die neuen Variablen eingetragen, kann der Berechnungs-Button betätigt werden. Die Ausgabe der Ergebnisse erfolgt auf der rechten Seite der Oberfläche. In den rot/weißen Feldern wird den einzelnen Größen jeweils ein Ergebnis zugeordnet. Weiter unten werden diese Ergebnisse in einer Graphik dargestellt. Mit dem Beenden-Button kann die Applikation beendet werden.

4.3 Test anhand einer Beispielrechnung

Zur Sicherung der Funktionsfähigkeit und als Orientierung wurde ein Rechenbeispiel eingearbeitet. Für das gesamte Schubmittelpunktmodell sind wesentlich mehr Parameter zu bestimmen, als hier gefordert. Das liegt daran, dass der Schwerpunkt auf der Dimensionierung des U-Profils gelegt wird. Die Eingabeparameter des Rechenbeispiels sind:

Länge l	5000 mm
Dicke t	0.5 mm
Masse m	0.5 kg
Höhe H	80 mm
Länge c	400 mm

Diese Parameter werden durch betätigen des Start-Buttons an die Funktion Berechnung übergeben und in die Formel (3.12) eingesetzt.

$$B = \frac{\frac{F_G \cdot c}{200} - \frac{1}{3} \cdot H \cdot t^2}{\frac{2}{3} \cdot t^2} = \frac{\frac{0.5 \text{ kg} \cdot 9.81 \text{ m/s}^2 \cdot 400 \text{ mm}}{200} - \frac{1}{3} \cdot 80 \text{ mm} \cdot (0.5 \text{ mm})^2}{\frac{2}{3} \cdot (0.5 \text{ mm})^2} = 18.86 \text{ mm}$$

Anschließend kann der Verdrehwinkel φ mit der Formel (3.13) errechnet werden.

$$\varphi = \frac{l \cdot F_G \cdot c}{E \left(\frac{2}{3} \cdot B \cdot t^3 + \frac{1}{3} \cdot H \cdot t^3 \right)} = \frac{5000 \text{ mm} \cdot 0.5 \text{ kg} \cdot 9.81 \text{ m/s}^2 \cdot 400 \text{ mm}}{2.1 \cdot 10^5 \text{ N/mm}^2 \left[\frac{2}{3} \cdot 18.86 \text{ mm} \cdot (0.5 \text{ mm})^3 + \frac{1}{3} \cdot 80 \text{ mm} \cdot (0.5 \text{ mm})^3 \right]} = 9.5238^\circ$$

Der letzte Ausgabeparameter ist der Schubmittelpunkt. Mit der Formel (3.5) wird die Verschiebung in negative y-Richtung vom Flansch des U-Profils weg berechnet.

$$y_M = \frac{H^2 \cdot B^2 \cdot t}{4 \cdot I_{yy}} = \frac{H^2 \cdot B^2 \cdot t}{4 \cdot \left[\frac{1}{12} \cdot t \cdot H^3 + \frac{1}{6} \cdot B \cdot t^3 + \frac{1}{2} H^2 \cdot B \cdot t \right]} = \frac{(80 \text{ mm})^2 \cdot (18.86 \text{ mm})^2 \cdot 0.5 \text{ mm}}{4 \cdot \left[\frac{1}{12} \cdot 0.5 \text{ mm} \cdot (80 \text{ mm})^3 + \frac{1}{6} \cdot 18.86 \text{ mm} \cdot (0.5 \text{ mm})^3 + \frac{1}{2} (80 \text{ mm})^2 \cdot 18.86 \text{ mm} \cdot (0.5 \text{ mm}) \right]} = 5.5244 \text{ mm}$$

Zum Schluss werden die Länge l, die Höhe H und die Breite B zur graphischen Darstellung des U-Profils verwendet.

5 Zusammenfassung und Ausblick

Das Projekt zeigt, dass eine benutzerfreundliche Bedienoberfläche entwickelt wurde. Durch die ergänzende Abbildung wird auch Personen, welche nicht mit dem Thema vertraut sind, ermöglicht, sich leicht zurecht zu finden, die gegebenen Werte einzutragen und das gewünschte U-Profil zu berechnen. Durch die entsprechende graphische Darstellung des Profils wird das Thema veranschaulicht. Als Weiterführung im Darstellungsbereich wäre eine Animation möglich, welche sowohl den Verdrehwinkel, als auch das gesamte Schubmittelpunktmodell zeigt. Für eine optimale Lösung, welche sich eventuell durch die MATLAB-Programmierung finden lässt, könnte anschließend eine Werkstattzeichnung angefertigt werden. Somit kann den Studenten zum Thema Schubmittelpunkt nicht nur die Theorie, sondern auch das Lehrmodell zur Veranschaulichung des Schubmittelpunktes gezeigt werden.

Anhang

A MATLAB Skript Verdrehwinkel zur Berechnung von Verdrehwinkeln

```
% Programmname: Verdrehwinkel
l=5000;           %[mm]
B=5:0.1:100;     %[mm]
H=2*B;           %[mm]
t=0.5;           %[mm]
tS=3;            %[mm]
b=B+60;          %[mm]
h=400;           %[mm]
c=400;           %[mm]
FG=0.5*9.81;     %[N=kg*m/s^2]
E=210000;        %[N/mm^2]
Iyy=(t.*H.^3)/12+(B.*t.^3)/6+(H.^2./2).*B*t;
% Biegung am Ständer
SigmaB1=(6*FG*l*tS)./(b*tS^3);
% SigmaB,taumax < SigmaB zul. = 200 N/mm^2
subplot(2,2,1)
plot(b,SigmaB1,'blue')
xlabel('Breite B in mm');ylabel('Biegespannung SigmaB in N/mm^2');
title('Biegung am Ständer');grid on

% Biegung am U-Profil
SigmaB2=(FG*l*H)./(Iyy*2);
subplot(2,2,2)
plot(H,SigmaB2,'blue')
xlabel('Breite B in mm');ylabel('Biegespannung SigmaB in N/mm^2');
title('Biegung am U-Profil');grid on

% Torsion am U-Profil
taumax=(FG*c)./(2/3*B*t.^2+H/3*t.^2);
subplot(2,2,3)
hold on
plot(B,taumax,'blue')
plot(B,200,'red')
hold off
xlabel('Breite B in mm');ylabel('Torsionsspannung Tau_m_a_x in N/mm^2');
title('Torsion am U-Profil');grid on

% Verdrehwinkel des U-Profils
Phi=(3*l*FG*c)./(E*(2*B*t.^3+H*t.^3));
subplot(2,2,4)
plot(B,Phi,'blue')
xlabel('Breite B in mm');ylabel('Verdrehwinkel Phi in Grad');
title('Verdrehwinkel des U-Profils');grid on
```

B MATLAB GUI Funktion Schubmittelpunktmodell für die graphische Benutzeroberfläche

```
function Schubmittelpunktmodell
global SMPM
disp('Dimensionierung eines U-Profiles')
if nargin==0

    % Edit-Feld der Länge l in mm
    SMPM.l= uicontrol(...
        'Style',          'edit',...
        'Units',          'normalized',...
        'Position',       [.02 .91 .2 .04 ],...
        'BackgroundColor', 'white',...
        'ForegroundColor', 'black',...
        'String',         '5000');

    % Text-Feld der Länge l in mm
    SMPM.ll= uicontrol(...
        'Style',          'text',...
        'Units',          'normalized',...
        'Position',       [.02 .95 .2 .04 ],...
        'BackgroundColor', 'black',...
        'ForegroundColor', 'white',...
        'String',         'Länge l in mm');

    % Edit-Feld der Höhe H in mm
    SMPM.H= uicontrol(...
        'Style',          'edit',...
        'Units',          'normalized',...
        'Position',       [.26 .91 .2 .04 ],...
        'BackgroundColor', 'white',...
        'ForegroundColor', 'black',...
        'String',         '80');

    % Text-Feld der Höhe H in mm
    SMPM.HH= uicontrol(...
        'Style',          'text',...
        'Units',          'normalized',...
        'Position',       [.26 .95 .2 .04 ],...
        'BackgroundColor', 'black',...
        'ForegroundColor', 'white',...
        'String',         'Höhe H in mm');

    % Edit-Feld der Dicke t in mm
    SMPM.t= uicontrol(...
        'Style',          'edit',...
        'Units',          'normalized',...
        'Position',       [.02 .81 .2 .04 ],...
        'BackgroundColor', 'white',...
        'ForegroundColor', 'black',...
        'String',         '0.5');
```

```
% Text-Feld der Dicke t in mm
SMPM.tt= uicontrol(...
    'Style',          'text',...
    'Units',          'normalized',...
    'Position',       [.02 .85 .2 .04 ],...
    'BackgroundColor', 'black',...
    'ForegroundColor', 'white',...
    'String',         'Dicke t in mm');

% Edit-Feld der Länge c in mm
SMPM.c= uicontrol(...
    'Style',          'edit',...
    'Units',          'normalized',...
    'Position',       [.26 .81 .2 .04 ],...
    'BackgroundColor', 'white',...
    'ForegroundColor', 'black',...
    'String',         '400');
% Text-Feld der Länge c in mm
SMPM.cc= uicontrol(...
    'Style',          'text',...
    'Units',          'normalized',...
    'Position',       [.26 .85 .2 .04 ],...
    'BackgroundColor', 'black',...
    'ForegroundColor', 'white',...
    'String',         'Länge c in mm');

% Edit-Feld der Masse m in kg
SMPM.m= uicontrol(...
    'Style',          'edit',...
    'Units',          'normalized',...
    'Position',       [.02 .71 .2 .04 ],...
    'BackgroundColor', 'white',...
    'ForegroundColor', 'black',...
    'String',         '0.5');
% Text-Feld der Masse m in kg
SMPM.mm= uicontrol(...
    'Style',          'text',...
    'Units',          'normalized',...
    'Position',       [.02 .75 .2 .04 ],...
    'BackgroundColor', 'black',...
    'ForegroundColor', 'white',...
    'String',         'Masse m in kg');

% Berechnungs-Button zur Berechnung von der Breite B, des Verdreh-
winkel Phi
% und dem Schubmittelpunkt yM
SMPM.start= uicontrol(...
    'Style',          'Pushbutton', ...
    'Units',          'normalized',...
    'Position',       [.26 .71 .2 .07], ...
    'String',         'Berechnung',...
    'Callback',       'Berechnung(''start'')');
```

```
% Beenden-Button zum Beenden des Programmes
SMPM.stop=uicontrol( ...
    'Style',          'pushbutton', ...
    'Units',          'normalized', ...
    'Position',       [.78 .71 .2 .07], ...
    'String',         'Beenden', ...
    'Call',           'close(gcf)');

% Erstellen einer Oberfläche für eine Graphik
SMPM.Bild= axes(...
    'Units',          'normalized',...
    'Position',       [.02 .02 .45 .66]);

%Einfügen der Graphik
picture=imread('Skizze.jpg');
axes(SMPM.Bild);
image(picture);
axis off;

% Text-Feld der Breite B in mm
SMPM.B= uicontrol(...
    'Style',          'text',...
    'Units',          'normalized',...
    'Position',       [.54 .81 .2 .04 ],...
    'BackgroundColor', 'white',...
    'ForegroundColor', 'black',...
    'String',         '18.86');

% Text-Feld der Breite B in mm
SMPM.BB= uicontrol(...
    'Style',          'text',...
    'Units',          'normalized',...
    'Position',       [.54 .85 .2 .04 ],...
    'BackgroundColor', 'red',...
    'ForegroundColor', 'black',...
    'String',         'Breite gewählt in mm');

% Text-Feld des Verdrehwinkels Phi in °
SMPM.Phi= uicontrol(...
    'Style',          'text',...
    'Units',          'normalized',...
    'Position',       [.78 .81 .2 .04 ],...
    'BackgroundColor', 'white',...
    'ForegroundColor', 'black',...
    'String',         '9.5238');

% Text-Feld des Verdrehwinkels Phi in °
SMPM.PhiPhi= uicontrol(...
    'Style',          'text',...
    'Units',          'normalized',...
    'Position',       [.78 .85 .2 .04 ],...
    'BackgroundColor', 'red',...
    'ForegroundColor', 'black',...
    'String',         '9.5238');
```

```
        'ForegroundColor', 'black',...
        'String',          'Verdrehwinkel in °');

% Text-Feld vom Schubmittelpunkt yM in mm
SMPM.yM= uicontrol(...
    'Style',          'text',...
    'Units',          'normalized',...
    'Position',       [.54 .71 .2 .04 ],...
    'BackgroundColor', 'white',...
    'ForegroundColor', 'black',...
    'String',         '5.5244');

% Text-Feld vom Schubmittelpunkt yM in mm
SMPM.yMyM= uicontrol(...
    'Style',          'text',...
    'Units',          'normalized',...
    'Position',       [.54 .75 .2 .04 ],...
    'BackgroundColor', 'red',...
    'ForegroundColor', 'black',...
    'String',         'Schubmittelpunkt in mm');

disp('Der Schubmittelpunkt ist die Verschiebung in negative y-
Richtung vom Flansch des U-Profiles weg.')

% Erstellen einer Oberfläche für eine Graphik
SMPM.Bild2= axes(...
    'Units',          'normalized',...
    'Position',       [.54 .05 .44 .62]);

% Einfügen der Grafik
dreidAnsicht;

end
```

C MATLAB Funktion Berechnung

```
function Berechnung(cmd)
global SMPM

switch cmd
case 'start'
    disp('Berechnung erfolgt.')

    % Die Variablen werden als Zeichenkette eingelesen und in
    % eine Zahl umgewandelt.
    m=get(SMPM.m,'string'); m=str2double(m);
    c=get(SMPM.c,'string'); c=str2double(c);
    H=get(SMPM.H,'string'); H=str2double(H);
    t=get(SMPM.t,'string'); t=str2double(t);
    l=get(SMPM.l,'string'); l=str2double(l);

    FG=m*9.81;
    E=2.1e5;

    %Berechnung der Breite B des U-Profils
    taumax=200;
    B=(FG*c/taumax-H/3*t.^2)/(2/3*t.^2);

    %Berechnung vom Verdrehwinkel Phi des U-Profils
    Phi=(3*l*FG*c)/(E*(2*B*t.^3+H*t.^3));

    %Berechnung des Schubmittelpunktes yM des U-Profils
    Iyy=t*H^3/12+B*t^3/6+H^2*B*t/2;
    yM=H^2*B^2*t/(4*Iyy);

    %Umwandlung der Zahl in eine Zeichenkette und gleichzeitige
    %Übergabe an Edit-Feld.
    set(SMPM.B,'String',num2str(B));
    set(SMPM.Phi,'String',num2str(Phi));
    set(SMPM.yM,'String',num2str(yM));

    %Aktualisierung der 3D-Ansicht des U-Profils
    dreiDAnsicht;

end
```

D MATLAB Funktion dreiDAnsicht zur graphische Darstellung des U-Profils

```
function dreiDAnsicht
global SMPM

% Die Variablen werden als Zeichenkette eingelesen und in eine Zahl
% umgewandelt.
H=get(SMPM.H, 'string'); H=str2double(H);
B=get(SMPM.B, 'string'); B=str2double(B);
l=get(SMPM.l, 'string'); l=str2double(l);

x=[0,0,0,l,l,l;      % Beschreibung der x-Koordinaten der Linie(1)
    0,0,0,l,l,l;      % Beschreibung der x-Koordinaten der Linie(2)
    0,0,0,l,l,l;      % Beschreibung der x-Koordinaten der Linie(3)
    0,0,0,l,l,l];     % Beschreibung der x-Koordinaten der Linie(4)
y=[0,0,B,B,0,0;
    0,0,0,0,0,0;
    0,0,0,0,0,0;
    0,0,B,B,0,0];
z=[0,H/2,H/2,H/2,H/2,0;
    0,H/2,H/2,H/2,H/2,0;
    0,-H/2,-H/2,-H/2,-H/2,0;
    0,-H/2,-H/2,-H/2,-H/2,0];
surf(x,y,z)           % verbindet die Linien zu Flächen
title('U-Profil')
xlabel('Länge l in mm')
ylabel('Breite B in mm')
zlabel('Höhe H in mm')
colormap([1 0 0])
view(45,45)
```

E MATLAB Skript dreiDAnsicht2 für die graphische Darstellung des U-Profils mit einer Stärke t

```
%Programmname: dreiDAnsicht2
```

```
l=5000;  
H=80/2;  
B=20.3692;  
t=5;  
x=[l/2,l,l,l,l,l,l/2;  
    l/2,l,l,l,l,l,l/2;  
    l/2,0,0,0,0,0,l/2;  
    l/2,0,0,0,0,0,l/2;  
    l/2,l,l,l,l,l,l/2];  
y=[B,B,0,0,B,B;  
    B,B,t,t,B,B;  
    B,B,t,t,B,B;  
    B,B,0,0,B,B;  
    B,B,0,0,B,B];  
z=[H-t/2,H,H,0,0,t/2;  
    H-t/2,H-t,H-t,t,t,t/2;  
    H-t/2,H-t,H-t,t,t,t/2;  
    H-t/2,H,H,0,0,t/2;  
    H-t/2,H,H,0,0,t/2];  
surf(x,y,z)  
colormap([0 0 1])  
view(45,45)
```

Literaturverzeichnis

DAS LITERATURVERZEICHNIS MUSS VOR DIE ANHÄNGE. d.h. HINTER
"Zusammenfassung und Ausblick"

- [1] Borz, S.: *Konstruktion eines Lehrmodells zur Veranschaulichung des Schubmittelpunkte*. Projekt A, Hochschule Wismar, 2010
- [2] Schweizer, W.: *MATLAB kompakt*. 2 Aufl. München: Oldenbourg, 2007.
- [3] Hanselman, D.; Littlefield, B.: *Mastering MATLAB 6: A Comprehensive Tutorial and Reference*. Upper Saddle River: Pearson Education, 2001
- [4] Pawletta, T.: *Graphikprogrammierung mit Matlab*. Skriptum zur Vorlesung. Hochschule Wismar, 2009
- [5] The MathWorks: *Matlab: The Language of Technical Computing*. [7.7.471.] Ismaning, 2008
- [6] <http://mo.mathematik.uni-stuttgart.de/kurse/kurs4/seite35.html> Stand vom 26.05.2011

Eidestattliche Erklärung

Ich erkläre hiermit, dass ich diese Diplomarbeit/Dissertation selbstständig ohne Hilfe Dritter und ohne Benutzung anderer als der angegebenen Quellen und Hilfsmittel verfasst habe. Alle den benutzten Quellen wörtlich oder sinngemäß entnommenen Stellen sind als solche einzeln kenntlich gemacht.

Diese Arbeit ist bislang keiner anderen Prüfungsbehörde vorgelegt worden und auch nicht veröffentlicht worden.

Ich bin mir bewusst, dass eine falsche Erklärung rechtliche Folgen haben wird.

Ort, Datum, Unterschrift