

TECHNISCHE UNIVERSITÄT MÜNCHEN

INGENIEURFAKULTÄT BAU GEO UMWELT
LEHRSTUHL FÜR METALLBAU

**Computergestützte Berechnung des idealen
Biegedrillknickmoments mithilfe des
Ritz-Verfahrens**

BACHELOR'S THESIS
IM STUDIENGANG BAUINGENIEURWESEN

MANUEL MAIER

MÜNCHEN 2019

BACHELOR'S THESIS IM FACH METALLBAU
EINGEREICHT AN DER TECHNISCHEN UNIVERSITÄT MÜNCHEN
INGENIEURFAKULTÄT BAU GEO UMWELT
AM LEHRSTUHL FÜR METALLBAU

THEMA:

**Computergestützte Berechnung des idealen
Biegedrillknickmoments mithilfe des Ritz-Verfahrens**

Verfasser:	Manuel Maier
Matrikelnummer:	03684306
Referent:	Univ.-Prof. Dr.-Ing. Dipl. Wirt.-Ing. Martin Mensinger
Betreuer:	Florian Oberhaidinger, M.Sc.

Begonnen am:	10. Juli 2019
Eingereicht am:	02. September 2019

Erklärung

Hiermit erkläre ich, dass die vorliegende Bachelor's Thesis von mir selbst angefertigt wurde und nur die aufgeführten Quellen und Hilfsmittel Verwendung fanden. Ich erkläre mich damit einverstanden, dass meine Bachelor's Thesis auf unbefristete Zeit zu Hochschulzwecken aufbewahrt werden darf.

München, 02. September 2019

Manuel Maier

Inhaltsverzeichnis

Erklärung	IV
1 Einleitung	1
2 Theoretischer Hintergrund der Energiemethode	2
2.1 Die Energiemethode	2
2.2 Das Potential	2
2.3 Indifferentes Gleichgewicht	3
2.4 Variationsrechnung	6
2.5 Das Ritz-Verfahren	7
3 Numerische Berechnung von M_{cr} mithilfe des Ritz-Verfahrens	8
3.1 Ansatzfunktionen (Hermiteische Interpolationspolynome)	8
3.2 Gleichungssystem	11
3.3 Integralmatrizen	11
3.4 Gleichungssystem in Matrixscheibweise	13
3.5 Vorfaktoren	13
3.6 Potentialanteile aus Einzellasten	14
3.7 Randbedingungen	14
3.8 Übergangsbedingungen und Zusammensetzen des Gesamtsystems	15
3.9 Umsetzung im Falle des Stabilitätsproblems	18
4 Validierung	19
4.1 Einführungsbeispiel	19
4.2 Bestimmung von M_{cr} aus den Diagrammen nach Roik	20
4.3 Vergleich mit den ζ -Diagrammen	21
4.4 Schwachstellen der Programmierung	27
4.5 Zusammenfassung der Validierung	27
5 Auswertung	28
6 Fazit und Ausblick	33
A Anhang: Graphen aus Roik et al. (1972)	34

B	Anhang: Quellcode	39
B.1	Datei integralmatrixGUI.m	39
B.2	Datei EingespanntEinzellastGUI.m	40
B.3	Datei GabellagerEinzellastGUI.m	43
B.4	Datei GabellagerLinienlastGUI.m	45
B.5	Datei GraphGabellagerungEinzellast.m	46
C	Anhang: Näherungsfunktionen	48
	Abbildungsverzeichnis	VIII
	Tabellenverzeichnis	X
	Literaturverzeichnis	XI

1 Einleitung

Das Stabilitätsverhalten von Bauteilen spielt bei der Bemessung von Stahltragwerken oft eine maßgebende Rolle. Um eine wirtschaftliche aber auch ressourcenschonende Planung sicherzustellen, ist es wichtig, Berechnungen auf Grundlage genauer Daten durchzuführen. Zur Nachweisführung eines längs- und/oder querbelastrten Trägers gehört meist auch der Nachweis auf Biegedrillknicken. Dafür wird das sogenannte „ideale Biegedrillknickmoment“ (M_{cr}) benötigt. Es gibt an, unter welcher Momentenbelastung bei einem bestimmten System (Lagerungsbedingungen, Abmessungen und Belastung) Stabilitätsversagen in Form von Biegedrillknicken auftritt. Zur Berechnung von M_{cr} gibt es verschiedene, oft alte, noch nicht digital umgesetzte Berechnungsansätze mit Tabellenwerten, welche allerdings aufgrund von Ablesungenauigkeit und fehlender Werte für bestimmte Systeme oft nur eine grobe Abschätzung zulassen. Auch in Bausoftwarens geschieht die Berechnung des idealen Biegedrillknickmoments meist auf unterschiedliche Art und Weise, was zu sehr unterschiedlichen Ergebnissen führen kann.

In der vorliegenden Arbeit wird eine Berechnungsmethode zur Berechnung von M_{cr} digital umgesetzt. Es wird ein auf der Energiemethode basierendes Verfahren, ähnlich wie Roik es im Buch „Biegetorsionsprobleme gerader dünnwandiger Stäbe“ [ROIK et al., 1972] erklärt, mithilfe des Ritz-Ansatzes gelöst. Das daraus entstehende kritische Moment kann dann entweder direkt als Eingangswert für die Dimensionierung und Nachweisführung verwendet werden oder für verschiedene Systemabmessungen als Graph dargestellt werden.

In den folgenden Kapiteln wird auf den theoretischen Hintergrund des Verfahrens sowie auf die Programmierung selbst eingegangen. Zusätzlich werden eine Validierung des entwickelten Programms durchgeführt und die Ergebnisse dargestellt.

2 Theoretischer Hintergrund der Energiemethode

2.1 Die Energiemethode

Da eine geschlossene Lösung der Differentialgleichungen des Biegedrillknickproblems sehr aufwändig wäre, bedient man sich Methoden, welche solche Probleme näherungsweise lösen. Eine dieser Methoden ist die Energiemethode, mit deren Hilfe man über das Prinzip vom Minimum der freien Energie Näherungslösungen für die Verformung im Stabilitätsfall finden kann. Zur Verwendung dieser Methode müssen Potentiale des Systems aufgestellt werden und ein Minimum derer gefunden werden.

2.2 Das Potential

Bei Belastung eines Systems leisten die inneren Kräfte eine Arbeit W_i . Wird das System in der selben Art entlastet, wie es belastet wurde, verrichten die inneren Kräfte die Arbeit $-W_i$. Diese Arbeit ist also im System gespeichert und kann potentiell frei werden. Man bezeichnet sie auch als Formänderungsenergie oder Potential der inneren Kräfte Π_i [PFLÜGER, 1975].

Die äußeren Kräfte leisten bei Belastung ebenfalls eine Arbeit W_a . Der negative Wert dieser Arbeit wird Potential der äußeren Kräfte Π_a genannt. Das Gesamtpotential Π lässt sich schreiben als:

$$\Pi = \Pi_i + \Pi_a \quad (2.1)$$

Das Gesamtpotential ergibt sich nach ROIK et al. (1972) zu:

$$\begin{aligned}
 \Pi &= \Pi_i + \Pi_a \\
 &= \frac{1}{2} \int_{x=0}^l [EI_z v''^2 + EI_y w''^2 + EI_\omega \vartheta''^2 + GI_T \vartheta'^2 \\
 &\quad + N(2z_M v' \vartheta' - 2y_M w' \vartheta' + v'^2 + w'^2 + i_M^2 \vartheta'^2) \\
 &\quad - M_z(-2w'' \vartheta + r_{M_y} \vartheta'^2) \\
 &\quad + M_y(2v'' \vartheta + r_{M_z} \vartheta'^2) \\
 &\quad + M_x(r_{M_x} \vartheta'^2)] dx \\
 &\quad - \int_{x=0}^l [p_y(v - z_p^M \vartheta - \frac{1}{2} y_p^M \vartheta^2) \\
 &\quad + p_z(w + y_p^M \vartheta - \frac{1}{2} z_p^M \vartheta^2) + m_d^a \vartheta] dx \\
 &\quad - \sum_i [P_{yi}(v_i - z_{Pi}^M \vartheta_i - \frac{1}{2} y_{Pi}^M \vartheta_i^2) \\
 &\quad + P_{zi}(w_i + y_{Pi}^M \vartheta_i - \frac{1}{2} z_{Pi}^M \vartheta_i^2) + M_{Di} \vartheta_i] \\
 &\quad - \sum_r (M_z^r v_r' - M_y^r w_r' - M_x^r \vartheta_r' + M_D^r \vartheta_r) \tag{2.2}
 \end{aligned}$$

2.3 Indifferentes Gleichgewicht

In der Natur nehmen Systeme immer den Zustand ein, in dem der Umwelt am wenigsten freie Energie zur Verfügung steht. Diese Anschauung führt bei statischen Systemen zum Prinzip der minimalen potentiellen Energie, welches besagt, dass sich ein System unter Belastung in demjenigen Zustand in statischer Ruhelage befindet, an dem das Gesamtpotential Π (2.1) ein Minimum ist. Die Gleichgewichtslage eines Systems lässt sich über die drei Gleichgewichtsarten: stabil, labil oder indifferent beschreiben. Beim stabilen Gleichgewicht kehrt das System bei Störung der Ruhelage z.B. durch eine infinitesimal kleine Verschiebung wieder in den Ursprungszustand zurück. Das bedeutet dem System muss Energie zugeführt werden um eine Verschiebung zu erreichen. Dies spiegelt sich in der Erhöhung des Gesamtpotentials wider. Beim labilen Gleichgewicht führt eine kleine Störung zum Versagen des Systems. Es wird

potentielle Energie frei. Das indifferente Gleichgewicht beschreibt den Zustand, in dem eine Störung keine Änderung des Potentials hervorruft.

Veranschaulichen lässt sich das ganze mithilfe des Modells einer Kugel, welche in verschiedenen Umgebungen liegt. Betrachtet man die Kugel in einer Schale, so wird sie auch bei Auslenkung aus der Mitte unter Wirkung der Schwerkraft wieder in die Mitte zurückkehren, was dem stabilen Gleichgewicht entspricht. Dreht man die Schale um und legt die Kugel auf die Spitze, so wird eine winzige Störung dieser Lage das Herunterrollen der Kugel verursachen. Das Gleichgewicht ist labil. Liegt die Kugel auf einer ebenen Fläche, befindet sie sich auch nach einer Störung noch im Gleichgewicht (indifferentes Gleichgewicht).

Für den Fall eines Stabilitätsproblems ist das indifferente Gleichgewicht besonders wichtig. Es ist genau der Zustand, der zwischen stabilem und labilem Gleichgewicht liegt und den Übergangspunkt von der Stabilität zum Versagen abbildet. Befindet sich ein System im indifferenten Gleichgewicht, ist die kritische Last erreicht. Dort wird eine infinitesimal kleine Verschiebung keine Änderung des Potentials hervorrufen. Warum das der Fall ist, lässt sich gut anhand der Potential-/ Verschiebungskurven für den Fall eines an der Spitze durch eine Einzellast F belasteten Euler I Stabs herleiten (siehe Abb. 2.1). Wird der Einfachheit halber angenommen, die Funktion des Potentials sei eine Parabel, so ist diese bei stabilem Gleichgewicht und Belastung, die weit niedriger als die kritische Last ist, sehr steil und konvex. Der Stab wird bei infinitesimaler Verrückung $\delta\vartheta$ immer wieder in seine Ausgangslage zurückkehren. Wird nun die Last gesteigert, so wird die Parabel immer flacher, da für eine Verrückung immer weniger Energie von außen zugeführt werden muss, sich das Potential bei geringer Verrückung also immer weniger steigert. Wird die kritische Last F_{krit} erreicht, so lässt sich der Stab geringfügig verschieben, ohne dass sich das Potential steigert (Bild zu indifferentem Gleichgewicht aus Abb. 2.1). Wird die kritische Last überschritten, kehrt sich die Parabel um und ist somit konkav. Sobald die Parabel konkav ist, besteht ein labiles Gleichgewicht. Das bedeutet, dass die Kurve im Zwischenzustand, also dem Punkt des Umkehrens der Krümmung zwischen stabil und labil, eine Gerade mit der Steigung null sein muss. Dies entspricht der Kurve des indifferenten Gleichgewichts.

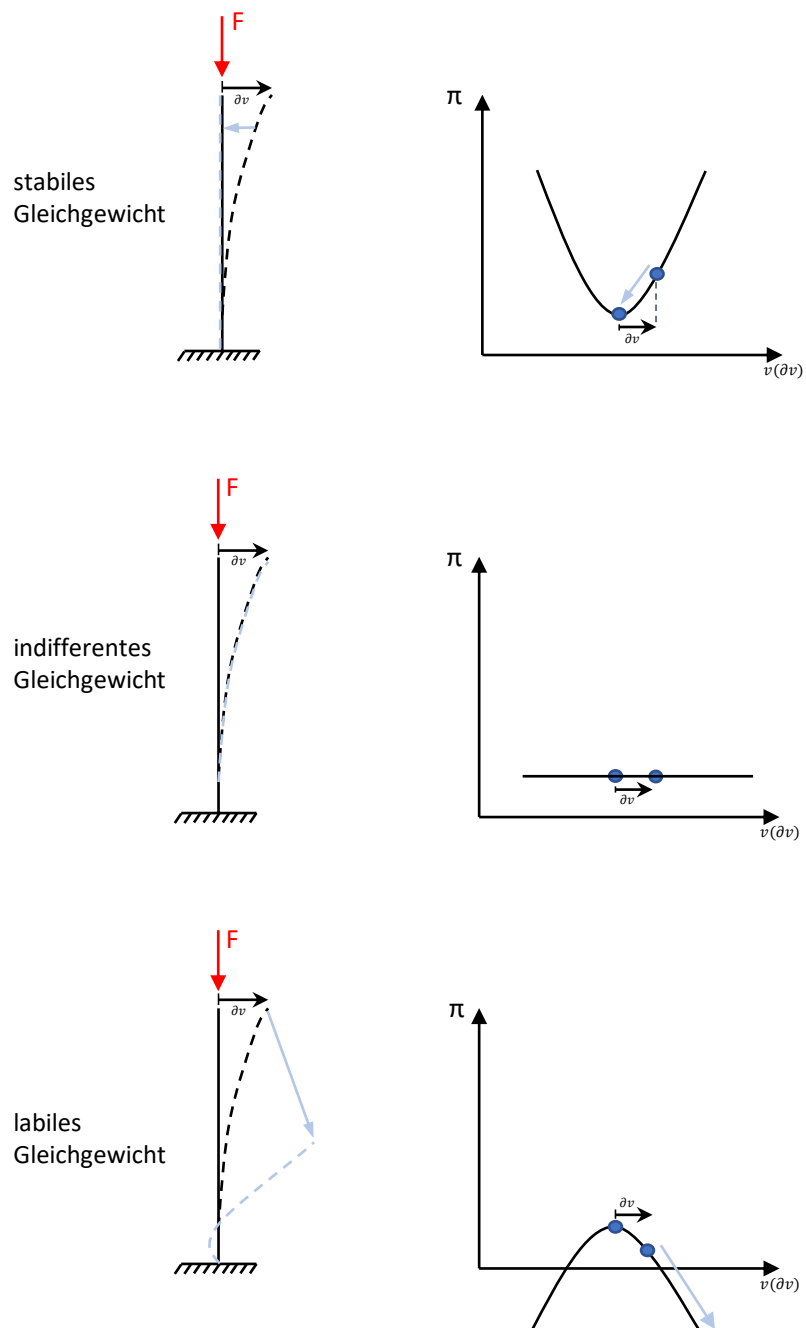


Abbildung 2.1: Typen des Gleichgewichts

2.4 Variationsrechnung

Das Potential ist ein Funktional, also eine Funktion, die wiederum von Funktionen abhängt. Ziel ist es diejenigen Verformungen (Funktionen abhängig von x) zu finden, mit denen das Potential ein Minimum annimmt. Lösungen dieses Problems lassen sich mithilfe der Variationsrechnung finden. Bei der Variationsrechnung werden Funktionen gesucht, mit denen unser Funktional minimal wird.

Unser zu minimierendes Funktional ergibt sich aus folgenden Anschauungen:

Die allgemeine Forderung nach Gleichgewicht lautet $\delta\Pi_0 = 0$, wobei der Index 0 für den Grundzustand steht. Das bedeutet, dass sich das Potential geringer Variation weg vom Grundzustand nicht ändert.

Im Fall eines indifferenten Gleichgewichts muss für mindestens einen Nachbarzustand gelten: $\delta\Pi_1 = 0$. Der Index 1 für den Nachbarzustand steht.

Mithilfe einer Taylorentwicklung kann der Nachbarzustand auch als Variation des Grundzustandes dargestellt werden. Vernachlässigt man Glieder ab der 3. Ordnung, so sieht die Reihe folgendermaßen aus:

$$\Pi_1 = \Pi_0 + \delta\Pi_0 + \frac{1}{2}\delta^2\Pi_0 \quad (2.3)$$

Über die allgemeine Forderung nach Gleichgewicht $\delta\Pi_0 = 0$ entfällt das zweite Glied. Wird nun als Forderung für indifferentes Gleichgewicht die Variation des Nachbarzustandes gebildet, entsteht folgender Ausdruck:

$$\delta\Pi_1 = \delta(\Pi_0 + \frac{1}{2}\delta^2\Pi_0) = \delta\Pi_0 + \delta(\frac{1}{2}\delta^2\Pi_0) = 0 \quad (2.4)$$

Hier entfällt wieder $\delta\Pi_0$. Somit ergibt sich am Ende der Ausdruck

$$\delta(\delta^2\Pi_0) = 0 \quad (2.5)$$

als Forderung für indifferentes Gleichgewicht [PFLÜGER, 1975, S. 67]. Ist diese Bedingung erfüllt, befindet sich das System im Stabilitätsfall bei der kritischen Last.

Das zu minimierende Funktional ist also die zweite Variation des Potentials (2.2) und wird in ROIK et al. (1972, S. 60) folgendermaßen angegeben:

$$\begin{aligned}
 \frac{1}{2}\delta^2\Pi = & \frac{1}{2}\int_l[EI_z\delta v''^2 + EI_y\delta w''^2 + EI_\omega\delta\vartheta''^2 + GI_T\delta\vartheta'^2]dx \\
 & -\frac{1}{2}\int_l[-N(\delta v'^2 + \delta w'^2 + \delta u_0'^2 + i_M^2\delta\vartheta'^2 + 2z_M\delta v''\delta\vartheta - 2y_M\delta w''\delta\vartheta) \\
 & + M_z(-2\delta w''\delta\vartheta + r_{My}\delta\vartheta'^2) \\
 & + M_y(-2\delta v''\delta\vartheta - r_{Mz}\delta\vartheta'^2) \\
 & - M_\omega r_{M\omega}\delta\vartheta'^2]dx \\
 & -\frac{1}{2}\int_l[-p_z z_p^M\delta\vartheta^2 - p_y y_p^M\delta\vartheta^2]dx \\
 & -\frac{1}{2}\sum_i[-P_{zi}z_{pi}^M\delta\vartheta_i^2 - P_{yi}y_{pi}^M\delta\vartheta_i^2] \tag{2.6}
 \end{aligned}$$

2.5 Das Ritz-Verfahren

Das Variationsproblem soll in dieser Arbeit mithilfe eines Computers gelöst werden. Ein geeignetes Verfahren stellt das Näherungsverfahren zur Lösung des Variationsproblems nach Ritz dar. Beim Ritz-Verfahren wird für unbekannte Verformungen der Ansatz $W(x) = \sum_{i=1}^m a_i w_i(x)$ mit den Freiwerten a_i gewählt. Setzt man diesen Verformungsansatz in die Forderung $\delta(\delta^2\Pi_0) = 0$ ein, so geht diese über in $\frac{\delta(\delta^2\Pi_0)}{\delta a_i} = 0$ [ROIK et al., 1972]. Im folgenden wird genauer beschrieben, wie man mithilfe des systematisierten Ritz-Verfahren konkrete Lösungen im Stabilitätsfall ermittelt.

3 Numerische Berechnung von Mcr mithilfe des Ritz-Verfahrens

Abbildung 3.1 zeigt das hier verwendete Koordinatensystem.

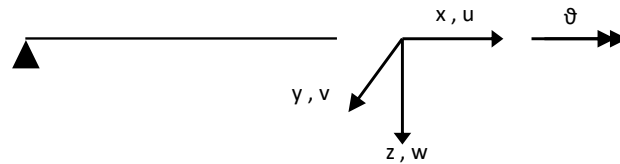


Abbildung 3.1: Verwendetes Koordinatensystem

3.1 Ansatzfunktionen (Hermiteische Interpolationspolynome)

Beim Ritz-Verfahren werden Verformungen gesucht, welche das Funktional minimieren. Dafür muss ein Näherungsansatz für jede unbekannte Verformungskomponente gemacht werden. Im hier aufgeführten Beispiel werden die Verformungen über folgenden Ansatz dargestellt:

$$\varphi(\xi) = \varphi_1 \cdot H_1(\xi) + \varphi_2 \cdot H_2(\xi) + \varphi_3 \cdot H_3(\xi) + \varphi_4 \cdot H_4(\xi) \dots \quad (3.1)$$

$\varphi_{1,2,3,\dots}$ sind Vorfaktoren und unsere Unbekannten, welche die Ansatzpolynome $H_{1,2,3,\dots}(\xi)$ skalieren. Als Ansatzfunktionen werden hier sogenannte Hermiteische Interpolationspolynome gewählt, bei dem jedes Polynom ($H_{1,2,3,\dots}(\xi)$) einer bestimmten Randbedingung zugeordnet ist. Für die Unabhängigkeit von Systemabmessungen oder Abschnittslängen werden die Hermiteischen Interpolationspolynome für die Feldlänge 1 aufgestellt und sind daher von ξ abhängig und nicht von x . Die Höhe des Polynomgrades entscheidet über die Genauigkeit der Näherung sowie über die Anzahl der Randbedingungen, die abgebildet werden können. Mit Hermiteischen Interpolationspolynomen 7. Grades können beispielsweise 8 Randbedingungen

(4 Rechts, 4 Links) befriedigt werden [ROIK et al., 1972, S. 66]. Im Folgenden werden Polynome 7. Grades verwendet.

In (3.2) sind die Vorfaktoren der 8 Polynome in Matrixschreibweise dargestellt:

$$\begin{array}{c|cccccccc}
 & 1 & \xi & \xi^2 & \xi^3 & \xi^4 & \xi^5 & \xi^6 & \xi^7 \\
 H8_1 & 1 & 0 & 0 & 0 & -35 & 84 & -70 & 20 \\
 H8_2 & 0 & 0 & 0 & 0 & 35 & -84 & 70 & -20 \\
 H8_3 & 0 & 1 & 0 & 0 & -20 & 45 & -36 & 10 \\
 H8_4 & 0 & 0 & 0 & 0 & -15 & 39 & -34 & 10 \\
 H8_5 & 0 & 0 & 0,5 & 0 & -5 & 10 & -7,5 & 2 \\
 H8_6 & 0 & 0 & 0 & 0 & 2,5 & -7 & 6,5 & -2 \\
 H8_7 & 0 & 0 & 0 & \frac{1}{6} & -\frac{2}{3} & 1 & -\frac{2}{3} & \frac{1}{6} \\
 H8_8 & 0 & 0 & 0 & 0 & -\frac{1}{6} & 0,5 & -0,5 & \frac{1}{6}
 \end{array} \quad (3.2)$$

Hermite Ansatzpolynome nach FALK(1963, S.154)

Zur Veranschaulichung wird hier das erste Polynom $H8_1$ betrachtet. Es wird also die Funktion $f(\xi) = 1 - 35\xi^4 + 84\xi^5 - 70\xi^6 + 20\xi^7$ auf dem Intervall $[0, 1]$ geplottet (Abb. 3.2).

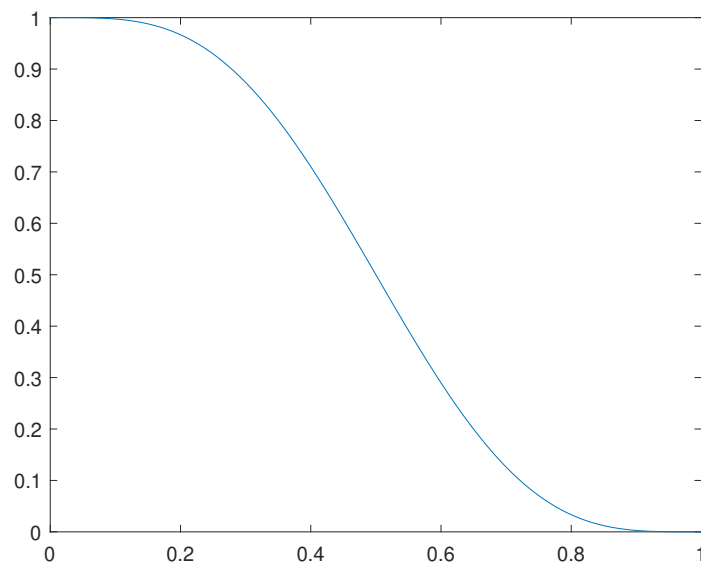


Abbildung 3.2: 1. Hermitesches Interpolationspolynom 7. Grades

Dieser Graph entspricht der Approximation der Biegelinie des Grundelements 1 bei Auslenkung des ersten Freiheitsgrades W_1 um 1 und festhalten der übrigen Freiheitsgrade. H_8 würde der Einheitsverschiebung von W_2 entsprechen (Abb. 3.3). H_3 und H_4 entsprechen den Einheitsverdrehungen links und rechts, also der ersten Ableitung der Verformung. H_5 und H_6 entsprechen der zweiten Ableitung usw..

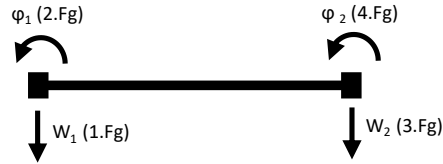


Abbildung 3.3: Grundelement 1 mit Freiheitsgraden

Die Gesamtverformung entsteht also aus der Superposition mehrerer, durch zu suchende Vorfaktoren skalierten, Einheitsverformungen/Verläufen. Da diese Vorfaktoren ($\varphi_{1,2,3,\dots}$) nichts anderes sind als Randbedingungen, entfallen gleich zu Beginn der Rechnung die Glieder aus dem Term der Gesamtverformung, welche vom System her null sind.

Wird nun einmal beispielshalber angenommen, $\varphi(\xi)$ beschreibe die Verschiebung eines auf beiden Seiten gelenkig gelagerten Systems in z -Richtung, so wäre die Verschiebung an den Stellen 0 und 1 gleich 0. Damit wären dann $\varphi_1 = \varphi_2 = 0$. Verdrehungen sind auf beiden Seiten möglich und unbekannt. Daher bleiben φ_3 und φ_4 stehen. Die Momente sind auf beiden Seiten null, d.h. $\varphi''(0) = \varphi''(1) = 0$ und damit $\varphi_5 = \varphi_6 = 0$. Die dritten Ableitungen entsprechen der Querkraft, die beiderseits unbekannt ist. Daher bleiben auch φ_7 und φ_8 stehen. Die Näherung der z -Verschiebung würde also folgendermaßen aussehen:

$$\varphi(\xi) = \varphi_3 \cdot H_3(\xi) + \varphi_4 \cdot H_4(\xi) + \varphi_7 \cdot H_7(\xi) + \varphi_8 \cdot H_8(\xi) \quad (3.3)$$

Dabei wären $\varphi_{3,4,7,8}$ die unbekannten Skalierungsfaktoren, welche mit dem Ritz-Ansatz bestimmt werden könnten.

3.2 Gleichungssystem

In den Unterkapiteln 2.3 und 2.4 wird für Stabilitätsprobleme die Forderung $\delta(\delta^2\Pi_0) = 0$ hergeleitet. Mithilfe von Hermiten Polynomen können wir das Potential und dessen Variationen als Ersatzenergie mit den unbekannten Parametern φ_i darstellen. Aus der Forderung $\delta(\delta^2\Pi_0) = 0$ entsteht für die Ersatzenergie die Forderung $\frac{\delta(\delta^2\Pi_0)}{\delta\varphi_i} = 0$. Umgesetzt in eine numerisch lösbare Form entsteht nach ROIK et al. (1972, S.69) für Stabilitätsprobleme das homogene Gleichungssystem

$$|G + S| \cdot \vec{\varphi} = 0. \quad (3.4)$$

Die $n \times n$ Matrix G enthält die Potentialanteile aus Verformungen und inneren Kräften (Schnittgrößen) und die $n \times n$ Matrix S Potentialanteile aus Einzellasten. Im Vektor $\vec{\varphi}$ mit der Größe $n \times 1$ stehen die unbekannten Freiwerte, welche die Verformung skalieren.

3.3 Integralmatrizen

Das Stabilitätsproblem wird in ROIK et al. (1972) aber auch in dieser Arbeit mithilfe von Matrizen gelöst. Dafür ist es notwendig, die zweite Variation des Potentials (2.6) in Matrixschreibweise darzustellen. Im Potentialausdruck treten Integrale über Verformungen zum Beispiel dieser Art auf: $\int \varphi(x)dx$ oder $\int \varphi^2(x)dx$. Zur vereinfachten Handhabung werden diese Ausdrücke vorab integriert und in sogenannten Integralmatrizen dargestellt [ROIK et al., 1972].

Um anschaulich zu verstehen, was es mit diesen Integralmatrizen auf sich hat, wird dies hier anhand eines Beispiels erklärt.

Folgendes Integral soll als Matrix dargestellt werden: $\int_0^l \vartheta^2 dx$.

Die entstehende Integralmatrix bekommt den Namen G_{00} . Die Indizes "00" stehen für den Grad der Ableitung der miteinander multiplizierten Verformungskomponenten. In diesem Fall steht im Integral $\vartheta \cdot \vartheta$, d.h. die nullte Ableitung wird mit der nullten multipliziert. Da die Verformung hier mit dem Ansatz nach (3.1) angenähert wird, in dem wiederum mehrere einzelne (in diesem Fall 8) Ansatzfunktionen ($H8_i$) stecken, müssen alle Kombinationen von Ansatzfunktionen multipliziert, integriert und als Matrix dargestellt werden. Die Integrale werden wie die Ansatzfunktionen auf die Feldlänge 1 bezogen. Numerisch lässt sich eine solche Matrix mit wenigen einfachen Matrizenoperationen erzeugen. Das genaue Vorgehen ist in ROIK et al. (1972, S.70) beschrieben. In G_{00} stehen also beispielsweise folgende Integrale über Ansatzfunktionen (3.5):

$$\begin{pmatrix} \int_0^1 H_{81} \cdot H_{81} d\xi & \int_0^1 H_{81} \cdot H_{82} d\xi & \cdots & \cdots & \cdots & \cdots & \int_0^1 H_{81} \cdot H_{88} d\xi \\ \int_0^1 H_{82} \cdot H_{81} d\xi & \int_0^1 H_{82} \cdot H_{82} d\xi & \cdots & \cdots & \cdots & \cdots & \vdots \\ \vdots & \vdots & \cdots & \ddots & \int_0^1 H_{83} \cdot H_{84} d\xi & \cdots & \vdots \\ \vdots & \vdots & \cdots & \cdots & \ddots & \cdots & \vdots \\ \vdots & \vdots & \cdots & \cdots & \cdots & \ddots & \vdots \\ \vdots & \vdots & \cdots & \cdots & \cdots & \cdots & \vdots \\ \int_0^1 H_{88} \cdot H_{81} d\xi & \cdots & \cdots & \cdots & \cdots & \cdots & \int_0^1 H_{88} \cdot H_{88} d\xi \end{pmatrix} \quad (3.5)$$

Im Potential sind auch Glieder z.B. der Form $\int M_{\dots} v'' \vartheta dx$, also Schnittkraftverlauf mal Verformung mal Verformung, enthalten. Schnittkraftverläufe werden abschnittsweise mit Verläufen der Form

$$M = M^K + M^\xi \cdot \xi + M^{\xi^2} \cdot \xi^2 \quad (3.6)$$

dargestellt [ROIK et al., 1972, S. 70]. Aus diesem Grund werden auch Integralmatrizen $G_{xx}^\xi, G_{xx}^{\xi^2}$ erstellt. Diese sehen dann Beispielsweise folgendermaßen aus:

$$\begin{aligned} G_{11}^\xi &= \int_0^1 \xi \cdot H_{8'} \cdot H_{8'} d\xi \\ G_{11}^{\xi^2} &= \int_0^1 \xi^2 \cdot H_{8'} \cdot H_{8'} d\xi \end{aligned} \quad (3.7)$$

Betrachtet man z.B. einen gabelgelagerten Einfeldträger mit Einzellast Pz in der Mitte und somit linearem Momentenverlauf M_y in beiden Systemhälften, so lässt sich der Potentialanteil $\int M_y v'' \vartheta$ für das rechte Feld folgendermaßen in die Matrix $G_{v\vartheta}$ einarbeiten:

$$G_{v\vartheta} = \frac{Pz \cdot l}{4} \cdot G_{20} - \frac{\frac{Pz \cdot l}{4}}{1} \cdot G_{20}^\xi \quad (3.8)$$

Wichtig ist auch hier wieder, dass sich der Verlauf auf die Feldlänge 1 bezieht.

Diese Matrizen sind nun die Grundlage zur Darstellung der zweiten Variation in Matrixschreibweise.

3.4 Gleichungssystem in Matrixschreibweise

Die Matrix G der Gleichung (3.4) wird als Gesamtmatrix bezeichnet. Diese setzt sich aus Teilmatrizen zusammen, welche jeweils die Verkopplung der Verformung enthalten. Werden für ein System zum Beispiel drei Verformungsansätze gemacht (z.B. Verdrehung ϑ , Verschiebung in z Richtung w und y Richtung v), erhält man für einen Längenabschnitt folgende Darstellung:

$$\begin{vmatrix} G_{\vartheta\vartheta} & G_{\vartheta v} & G_{\vartheta w} \\ G_{v\vartheta} & G_{vv} & G_{vw} \\ G_{w\vartheta} & G_{wv} & G_{ww} \end{vmatrix} \begin{vmatrix} \varphi_{i\vartheta} \\ \varphi_{iv} \\ \varphi_{iw} \end{vmatrix} = \begin{vmatrix} 0 \\ 0 \\ 0 \end{vmatrix} \quad (3.9)$$

In der Teilmatrix $G_{\vartheta v}$ sind alle Potentialanteile in denen $\vartheta \cdot v$, oder deren Ableitungen wie zum Beispiel $\vartheta \cdot v''$ vorkommen, enthalten. In der folgenden Tabelle 3.1 ist die zweite Variation des Potentials nach Verformungsgrößen aufgegliedert.

	ϑ	v	w
ϑ	$EI_{\omega}\vartheta''^2 + (GI_T + Ni_M^2 - M_z r_{My} + M_y r_{Mz} + M_{\omega} r_{M_{\omega}})\vartheta'^2 + (p_y y_p + p_z z_p)\vartheta^2$	$2Nz_M v''\vartheta + 2M_y v''\vartheta$	$-2Ny_M w''\vartheta$
v		$EI_z v''^2 + Nv'^2$	
w			$EI_y w''^2 + Nw'^2$

Tabelle 3.1: Zusammensetzung der Gesamtmatrix G

3.5 Vorfaktoren

Um alle Komponenten der zweiten Variation in Matrixschreibweise darstellen zu können, fehlen noch die verformungsunabhängigen Vorfaktoren. Alle Teilmatrizen G_{xx} werden aus dem Produkt *Vorfaktor* $F_i \cdot$ *Integralmatrix* gebildet. Bei der Bestimmung der Vorfaktoren ist es wichtig zu beachten, dass die tatsächlichen Verformungen von x abhängen und nicht von ξ . Es müssen daher Transformationsbedingungen eingehalten werden. Die Teilmatrix $G_{\vartheta v}$ hat beispielsweise die Vorfaktoren $F_1 = l_i N$ und $F_2 = \frac{1}{l_i} EI_y$. Die Multiplikation von l_i und $\frac{1}{l_i}$ (l_i ist die Länge des Systemabschnitts) kommt aus den Transformationsbedingungen.

Die Teilmatrix wird dann beispielsweise gebildet durch $G_{\vartheta v} = F_1 \cdot |G_{11}| + F_2 \cdot |G_{22}|$. Weitere Vorfaktoren können [ROIK et al., 1972, S.75] entnommen werden.

3.6 Potentialanteile aus Einzellasten

Die Matrix S aus Gleichung (3.4) berücksichtigt Potentialanteile aus Einzellasten. In der zweiten Variation (2.6) sind nur die Anteile $P_{zi} \cdot z_{Pi} \cdot \delta \vartheta_i^2$ und $P_{yi} \cdot y_{Pi} \cdot \delta \vartheta_i^2$ enthalten. Die Matrix S bringt also nur noch Anteile zur Matrix $G_{\vartheta\vartheta}$. Betrachtet wird wieder der Fall des gabelgelagerten Einfeldträgers mit Einzellast P_z in der Mitte. Die Einzellast greift also am rechten Rand des linken Feldes an. Die Matrix S sieht dann folgendermaßen aus:

$$\begin{array}{c}
 \vartheta_0 \quad \vartheta_1 \quad \vartheta'_0 \quad \vartheta'_1 \quad \vartheta''_0 \quad \vartheta''_1 \quad \vartheta'''_0 \quad \vartheta'''_1 \\
 \left| \begin{array}{cccccccc}
 \vartheta_0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 \vartheta_1 & 0 & P_z z_{P_z} & 0 & 0 & 0 & 0 & 0 \\
 \vartheta'_0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 \vartheta'_1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 \vartheta''_0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 \vartheta''_1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 \vartheta'''_0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 \vartheta'''_1 & 0 & 0 & 0 & 0 & 0 & 0 & 0
 \end{array} \right|
 \end{array} \tag{3.10}$$

ϑ_0, ϑ_1 sind dabei die Randbedingungen/Freiwerte am linken (Index 0) und am rechten Rand (Index 1) des Systemabschnitts. Diese Matrix S muss nun auf die Matrix G addiert werden, um die fertige Abschnittsmatrix zu erhalten [ROIK et al., 1972].

3.7 Randbedingungen

Über die Lagerungs- und Symmetriebedingungen lässt sich das Gleichungssystem stark verkleinern. Sind bestimmte Randwerte gleich null (also Skalierungsfaktoren φ_i des Verformungsansatzes (3.1)), so können sowohl die zugehörige Zeile als auch die Spalte aus der Teilmatrix gestrichen werden. Betrachtet man bei einem gabelgelagerten Einfeldträger der

Länge $2l$ mit Einzellast in der Mitte die linke Systemhälfte, so sind $\vartheta(0) = \vartheta''(0) = \vartheta'(l) = 0$. Damit können die erste, die vierte und die fünfte Zeile und Spalte der Teilmatrizen gestrichen werden. Die Matrix $G_{\vartheta\vartheta}$ vereinfacht sich damit zu:

$$\begin{array}{c|ccccc}
& \vartheta_1 & \vartheta'_0 & \vartheta''_1 & \vartheta'''_0 & \vartheta'''_1 \\
\hline
\vartheta_1 & a_{22} & a_{23} & a_{26} & a_{27} & a_{28} \\
\vartheta'_0 & a_{32} & a_{33} & a_{36} & a_{37} & a_{38} \\
\vartheta''_1 & a_{62} & a_{63} & a_{66} & a_{67} & a_{68} \\
\vartheta'''_0 & a_{72} & a_{73} & a_{76} & a_{77} & a_{78} \\
\vartheta'''_1 & a_{82} & a_{83} & a_{86} & a_{87} & a_{88}
\end{array}
\quad (3.11)$$

In der hier erstellten Programmierung werden Randbedingungen mithilfe einer Matrix 3.12 eingebracht. Bei Hermiten Polynomen 7. Grades hat diese Matrix 8 Spalten. Jede Spalte steht dabei für eine Randbedingung. Ist der Eintrag 0, so werden Zeile und Spalte der jeweiligen Teilmatrix gestrichen. Ist er 1, so ist die Randbedingung unbekannt und Zeile bzw. Spalte bleiben erhalten.

$$\begin{array}{cccccccc}
\vartheta_0 & \vartheta_1 & \vartheta'_0 & \vartheta'_1 & \vartheta''_0 & \vartheta''_1 & \vartheta'''_0 & \vartheta'''_1 \\
\left[\begin{array}{cccccccc}
0 & 1 & 0 & 0 & 1 & 1 & 1 & 1
\end{array} \right]
\end{array}
\quad (3.12)$$

3.8 Übergangsbedingungen und Zusammensetzen des Gesamtsystems

Im Falle eines Systems mit mehreren Systemabschnitten müssen nun die einzelnen Abschnittsmatrizen miteinander gekoppelt werden. Dies geschieht mithilfe von Übergangsbedingungen. Im Fall eines Systems mit zwei Abschnitten liegen zwei Abschnittsmatrizen $G^{I,II}$ vor (G

enthält jetzt auch schon die Matrix S). Für das weitere Vorgehen müssen diese in eine große Systemmatrix eingeordnet werden:

$$\begin{vmatrix} G_{\vartheta\vartheta}^I & G_{\vartheta v}^I & G_{\vartheta w}^I & & & \\ G_{v\vartheta}^I & G_{vv}^I & G_{vw}^I & & 0 & \\ G_{w\vartheta}^I & G_{wv}^I & G_{ww}^I & & & \\ & & & G_{\vartheta\vartheta}^{II} & G_{\vartheta v}^{II} & G_{\vartheta w}^{II} \\ & 0 & & G_{v\vartheta}^{II} & G_{vv}^{II} & G_{vw}^{II} \\ & & & G_{w\vartheta}^{II} & G_{wv}^{II} & G_{ww}^{II} \end{vmatrix} \quad (3.13)$$

Danach müssen Übergangsbedingungen bestimmt werden. Dafür werden die Freiheitsgrade, die nicht schon von den Randbedingungen her null sind, mit Systemabschnitt I beginnend durchnummeriert und wie folgt (Abb. 3.4) dargestellt (hier wieder symmetrisches System, Gabellagerung mit Einzellast in der Mitte):

	I		II	
	↓	↓	↓	↓
ϑ	0	1	11	0
ϑ'	2	0	0	12
ϑ''	0	3	13	0
ϑ'''	4	5	14	15
v	0	6	16	0
v'	7	0	0	17
v''	0	8	18	0
v'''	9	10	19	20

Abbildung 3.4: Nummerierung der Freiheitsgrade in Anlehnung an [Roik et al., 1972, S.84]

Die Freiheitsgrade des Systemabschnitts I und II sind von einander abhängig und können über sogenannte Zusatzbedingungen in Verbindung gebracht werden. In obigem Beispiel ist die Verschiebung v in der Mitte am rechten Rand des linken Systems (Freiheitsgrad 6) natürlich gleich groß wie die Verschiebung am linken Rand des rechten Systems (Freiheitsgrad 16). Dementsprechend folgt $6 \hat{=} 16$. Eine ähnliche Beziehung lässt sich für die Verdrehung, also v' herstellen. Diese ist an beiden Gabellagern aufgrund der Symmetrie entgegengesetzt gleich groß $\Rightarrow 7 \hat{=} -17$ [ROIK et al., 1972]. Ordnet man alle Größen in einer Matrix, so

dass die Freiheitsgarde in der ersten Zeile jeweils den darunter in der zweiten Zeile stehenden entsprechen, sieht das ganze so aus:

$$\begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 \\ 11 & -12 & 13 & -15 & -14 & 16 & -17 & 18 & -20 & -19 \end{bmatrix} \quad (3.14)$$

Diese Darstellung dient der Übersicht und vereinfacht zudem die programmiertechnische Handhabung der Übergangsbedingungen. Im Beispielsystem hat die Systemmatrix die Größe 20x20. Zur Verkoppelung der noch unabhängigen Abschnittsmatrizen in der Systemmatrix werden die Zeilen und Spalten wie folgt addiert:

- 1. Zeile 1. Spalte + 11. Zeile 11. Spalte
- 1. Zeile 2. Spalte - 11. Zeile 12. Spalte
- 1. Zeile 3. Spalte + 11. Zeile 13. Spalte
- ...
- 2. Zeile 1. Spalte + 12. Zeile 12. Spalte
- 2. Zeile 2. Spalte - 12. Zeile 12. Spalte
- 2. Zeile 3. Spalte + 12. Zeile 13. Spalte
- ...

Wichtig ist es zu verstehen, dass in jedem Matrixeintrag der Systemmatrix ein Potentialanteil steht. Die Addition ist dabei das Zusammenfassen zweier, mit zwei unabhängigen Freiwerten a_i skalierten Potentialanteile, zu einem größeren Potentialanteil mit nur noch einem Freiwert. In Abbildung 3.5 wird versucht, dies zu Veranschaulichen.

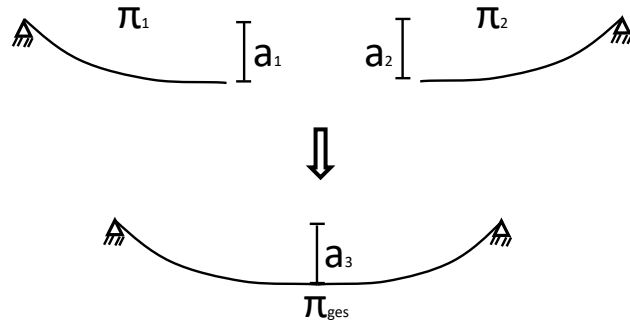


Abbildung 3.5: Skallierungsfunktion der Übergangsbedingungen

Da die Verformung in der Mitte des Balkens gleich groß ist, genügt es, diese nur mit einem Freiwert $a_3 = a_1 = a_2$ zu skalieren. Dieser Freiwert skaliert dann beide Potentialanteile, weshalb diese addiert werden müssen $\Pi_{ges} = \Pi_1 + \Pi_2$.

3.9 Umsetzung im Falle des Stabilitätsproblems

Stabilitätsprobleme führen mathematisch zu einem Eigenwertproblem $(A - \lambda B) \cdot x = 0$. Gesucht ist der Laststeigerungsfaktor λ , bei dem das System versagt. Die nichttriviale Lösung erhält man, indem man $\det(A - \lambda B) = 0$ setzt. Der kleinste Eigenwert λ ist dann der gesuchte Laststeigerungsfaktor. In dem hier vorliegenden Fall sind die Steifigkeitsmatrix A und die Belastungsmatrix B (enthält alle Anteile in denen Schnittgrößen enthalten sind) getrennt aufzustellen [ROIK et al., 1972]. Das Vorgehen zum Aufstellen der Systemmatrizen A und B ist dabei genau gleich wie im vorherigen Teil dieses Kapitels beschrieben. Mit einem Eigenwertlöser lässt sich der kleinste Eigenwert finden. Das ideale Biegedrillknickmoment ergibt sich dann aus der Multiplikation des kleinsten Eigenwerts mit dem maximalen Feldmoment im System.

4 Validierung

Mit dem im Rahmen dieser Arbeit erstellten Programm kann das ideale Biegedrillknickmoment für Einfeldträger mit verschiedenen Lagerungs- und Belastungsarten bestimmt werden. Zur Auswahl stehen eine beidseitig gelenkige Lagerung oder beidseitige Einspannung. Als Belastungsarten wurden eine mittige Einzellast oder eine über die Trägerlänge konstante Linienlast implementiert. Profil-, System- und Materialgrößen können vom Nutzer individuell eingegeben werden. In diesem Kapitel sollen die Ergebnisse des Programms auf ihre Gültigkeit überprüft werden.

4.1 Einführungsbeispiel

Einführend soll anhand der Nachrechnung eines, in ROIK et al. (1972) vorkommenden Beispiels, die Richtigkeit des erstellten Programms für einen konkreten Fall überprüft werden. In ROIK et al. (1972, S.87) wird ein beidseitig gabelgelagerter Einfeldträger mit Einzellast $P_z = 22,6 \text{ Mp}$ in der Mitte als Stabilitätsproblem untersucht. Die Spannweite beträgt 14 Meter und das Profil ist ein IP85.

Roik kommt hier auf einen Eigenwert von $v_{ki} = 1,908$. Das maximale Feldmoment beträgt am gegebenen System $\frac{P_z \cdot L}{4} = 79,1 \text{ Mpm}$. Somit kommt man auf ein M_{cr} von $79,1 \text{ Mpm} \cdot 1,908 = 150,92 \text{ Mpm}$. Das in diesem Rahmen erstellte Programm liefert $M_{cr} = 150,91 \text{ Mpm}$, was einer Abweichung von $\frac{0,01}{150,92} = 0,0066\%$ entspricht. Diese geringe Abweichung kommt wahrscheinlich aus der Rundung des Eigenwerts bei Roik. Dies bestätigt sich, wenn man den Eigenwert des Programms mit $v_{ki} = 1,9078$ betrachtet.

Zusätzlich sei noch erwähnt, dass Roik mit einem E-Modul von $E = 2100 \frac{\text{Mp}}{\text{cm}^2}$ und einem Schubmodul von $G = 810 \frac{\text{Mp}}{\text{cm}^2}$ rechnet, was nicht den normalerweise verwendeten Größen 21000 und 8100 $\frac{\text{kN}}{\text{cm}^2}$ entspricht. Um mit aktuellen Einheiten auf die selben Ergebnisse wie Roik zu kommen, müssen diese Größen also auch umgerechnet werden.

4.2 Bestimmung von M_{cr} aus den Diagrammen nach Roik

Im Folgenden soll das im Rahmen dieser Arbeit in Matlab implementierte Berechnungsverfahren für das ideale Biegedrillknickmoment auf seine Gültigkeit überprüft werden. Hierzu muss zuerst das Vorgehen zur Berechnung von M_{cr} mithilfe der Diagramme nach ROIK et al. (1972) erläutert werden. Grafik 4.1 zeigt dabei die Bedeutung der Größen ψ und z_P .

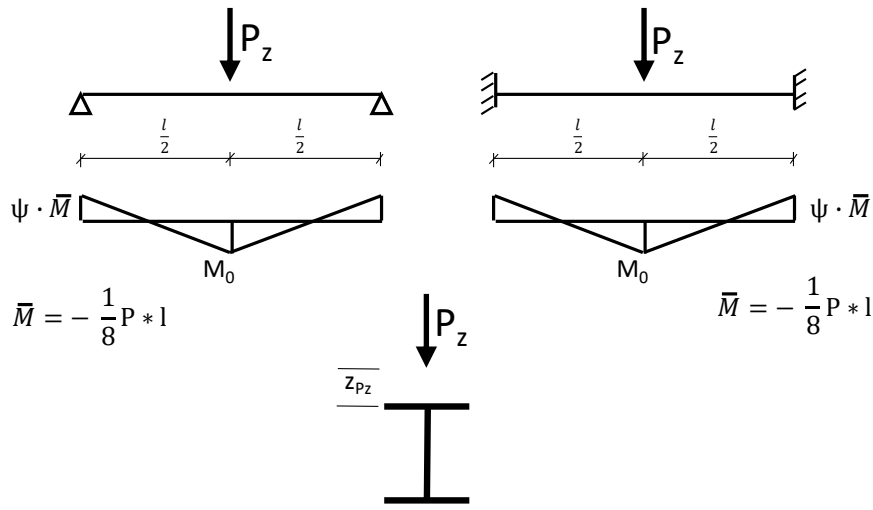


Abbildung 4.1: Bedeutung der Variablen ψ und z_P

Um die Bestimmung von M_{cr} in Zeiten, in denen Computer noch nicht für jeden Ingenieur zugänglich waren zu vereinfachen, hat ROIK et al. (1972) Berechnungen für verschiedene Systeme, nach dem hier im Kapitel 3 beschriebenen Verfahren, durchgeführt und als Diagramme dargestellt. Das ideale Biegedrillknickmoment wird am Ende mit folgender Formel bestimmt:

$$M_{cr} = \zeta \cdot \frac{\pi^2 \cdot E \cdot I_z}{L^2} \cdot [\sqrt{c^2 + 0,2566 \cdot z_p^2} + 0,5066 \cdot z_p] \quad (4.1)$$

z_p ist die Höhe des Lastangriffspunktes ausgehend vom Schubmittelpunkt (negativ am oberen Flansch).

c^2 ist der Drehradius und wird wie folgt bestimmt:

$$c^2 = \frac{I_\omega + 0,039 \cdot L^2 \cdot I_T}{I_z} \quad (4.2)$$

Den ζ -Wert erhält man aus in ROIK et al. (1972) abgedruckten Diagrammen. Eingangsgröße für diese Diagramme ist der längen- und profilabhängige Parameter χ .

$$\chi = \frac{E \cdot I_\omega}{L^2 \cdot G \cdot I_T} \quad (4.3)$$

4.3 Vergleich mit den ζ -Diagrammen

Die im Rahmen dieser Arbeit erstellte Programmierung kann einem Benutzer das ideale Biegedrillknickmoment unter Eingabe beliebiger Profilwerte und Längen für einen beidseitig gabelgelagerten oder eingespannten Träger mit mittiger Einzellast oder konstanter Linienlast ausgeben. Um die Ergebnisse auf ihre Gültigkeit zu überprüfen, werden im Folgenden die Diagramme nach ROIK et al. (1972) mit Plots aus der in diesem Rahmen erstellten Programmierung verglichen.

Grundlage der Plots sind die Querschnittswerte des *IPBI320* Profils, genauso wie bei Roiks Berechnungen. M_{cr} wird für verschiedene, über Umstellung der Formel (4.3) errechnete Systemlängen berechnet. Durch Umstellen der Formel (4.1) nach ζ und einsetzen der berechneten M_{cr} können Werte für ζ in Abhängigkeit von χ ausgegeben werden. Die daraus entstehenden Graphen werden nun mit den Graphen von Roik verglichen.

In den Folgenden Grafiken (Abb. 4.2-4.13) ist jeweils die absolute Abweichung der in diesem Rahmen errechneten ζ -Werte (Abb. 5.1-5.8) von den Werten nach Roik (Abb. A.1-A.8) aufgetragen.

Zur Berechnung der Abweichung wurden die Graphen nach Roik an diskreten Stellen ausgewertet und durch eine Funktion angenähert. Die einzelnen Funktionsauswertungen aus dem hier erstellten Programm wurden ebenfalls durch eine Funktion angenähert. Um die absolute Abweichung zu erhalten wurden beide Funktionen voneinander abgezogen (Anhang C). Geringe Ungenauigkeiten können daher auch aus der Näherung der Funktionen kommen. Da die Graphen nach Roik nicht in digitaler Form vorlagen und deshalb zuerst digitalisiert werden mussten, können grundsätzlich auch geringe Abweichungen aus der Digitalisierung entstanden sein. Es werden für die Verschiedenen Systeme jeweils Extremfälle, also $\psi = 0$ und/oder $\psi = 1$ betrachtet.

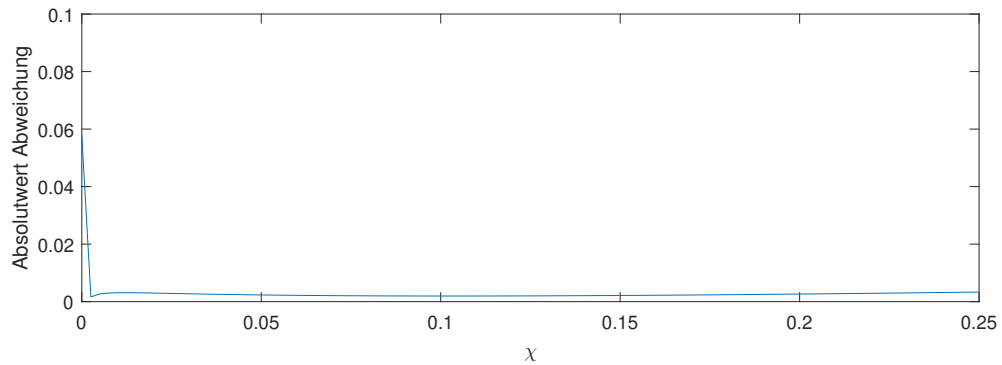


Abbildung 4.2: Absolute Abweichung der ζ -Werte von den Werten nach Roik; Gabellagerung Einzellast in der Mitte $z_p = 0, \psi = 0$

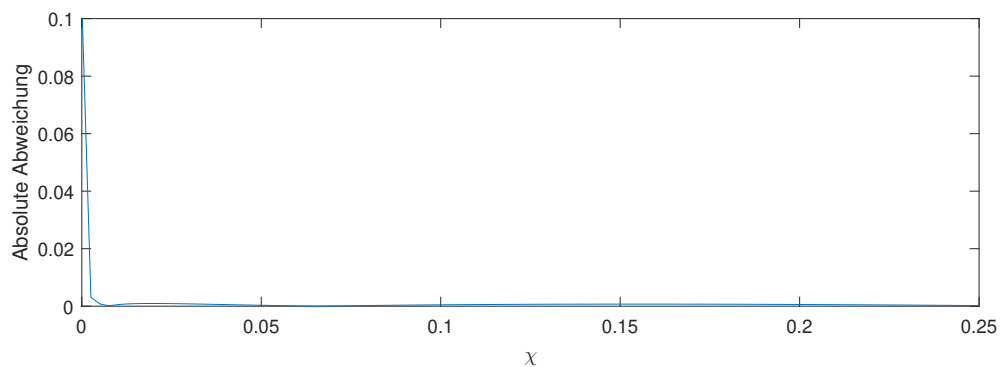


Abbildung 4.3: Absolute Abweichung der ζ -Werte von den Werten nach Roik; Gabellagerung Einzellast in der Mitte $z_p = 0, \psi = 1$

Abbildung 4.2 und 4.3 zeigen die Abweichung von Abbildung A.1. Es lässt sich erkennen, dass die Abweichungen hier zumindest für $\chi \geq 0,01$ extrem gering sind und wahrscheinlich aus der Digitalisierung der Roik-Graphen stammen. Die größer werdende Abweichung für $\chi \rightarrow 0$ lässt sich durch die unrealistisch großen Stützweiten erklären, welche bei der Berechnung für kleine χ -Werte durch Umstellen von (4.3) nach L entstehen.

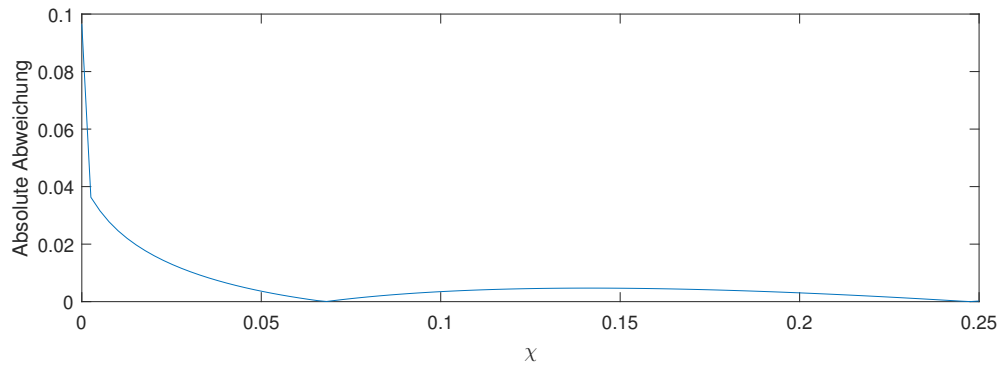


Abbildung 4.4: Absolute Abweichung der ζ -Werte von den Werten nach Roik; Gabellagerung Einzellast in der Mitte $z_p = -\frac{h}{2}$, $\psi = 0$

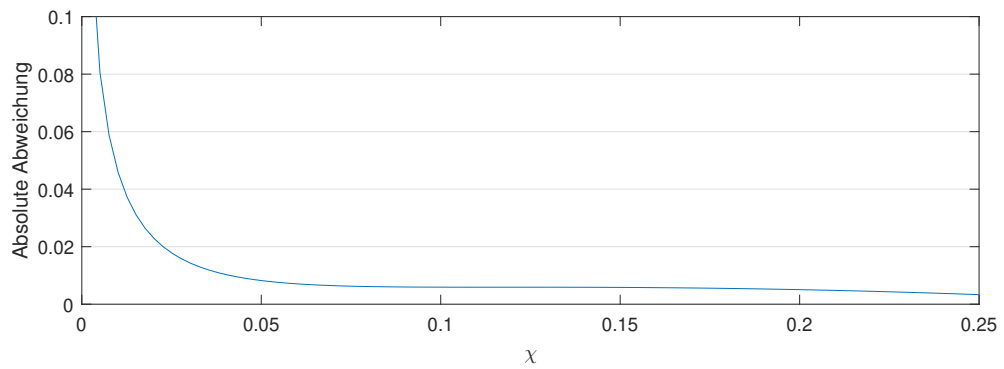


Abbildung 4.5: Absolute Abweichung der ζ -Werte von den Werten nach Roik; Gabellagerung Einzellast in der Mitte $z_p = -\frac{h}{2}$, $\psi = 1$

Für $z_p = -\frac{h}{2}$ zeigen sich ebenfalls keine erheblichen Abweichungen. Die Abweichung für $\chi \rightarrow 0$ stammt wieder aus den unrealistisch groß werdenden Stützweiten.

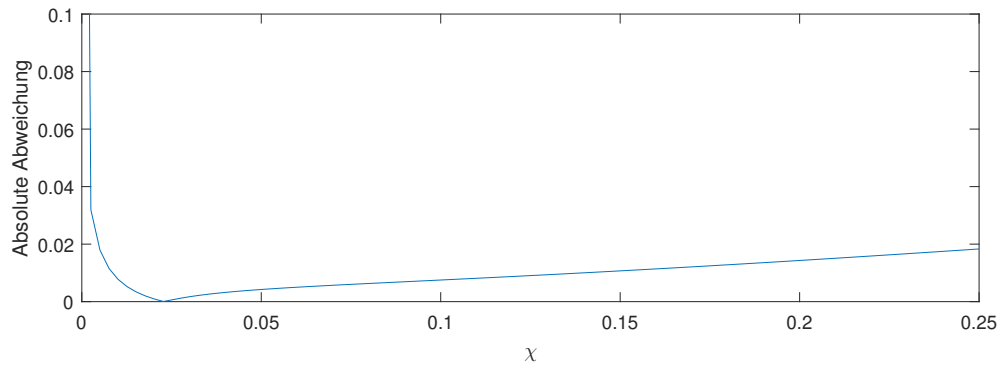


Abbildung 4.6: Absolute Abweichung der ζ -Werte von den Werten nach Roik; Einspannung Einzellast in der Mitte $z_p = 0, \psi = 0$

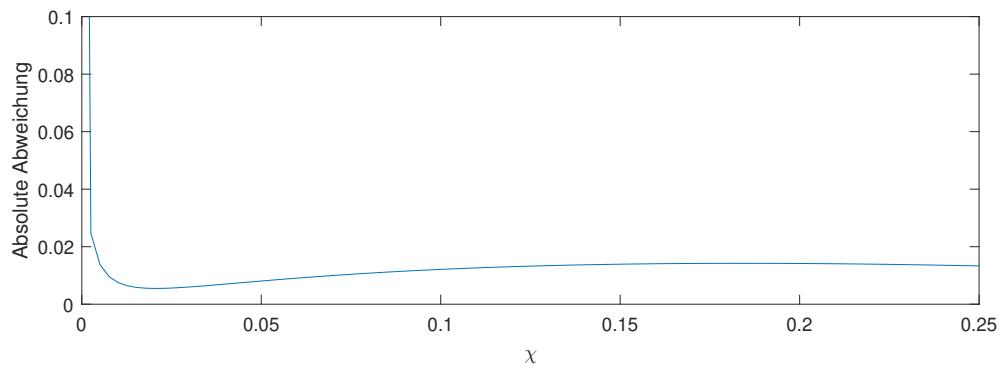


Abbildung 4.7: Absolute Abweichung der ζ -Werte von den Werten nach Roik; Einspannung Einzellast in der Mitte $z_p = 0, \psi = 1$

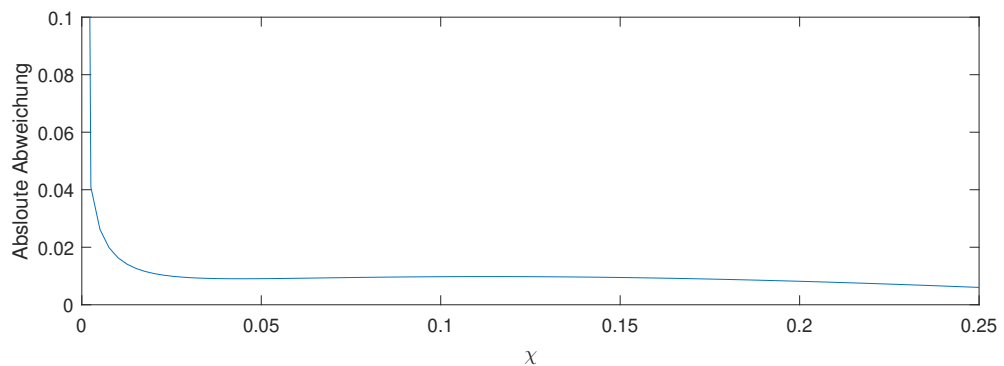


Abbildung 4.8: Absolute Abweichung der ζ -Werte von den Werten nach Roik; Einspannung Einzellast in der Mitte $z_p = -\frac{h}{2}, \psi = 0$

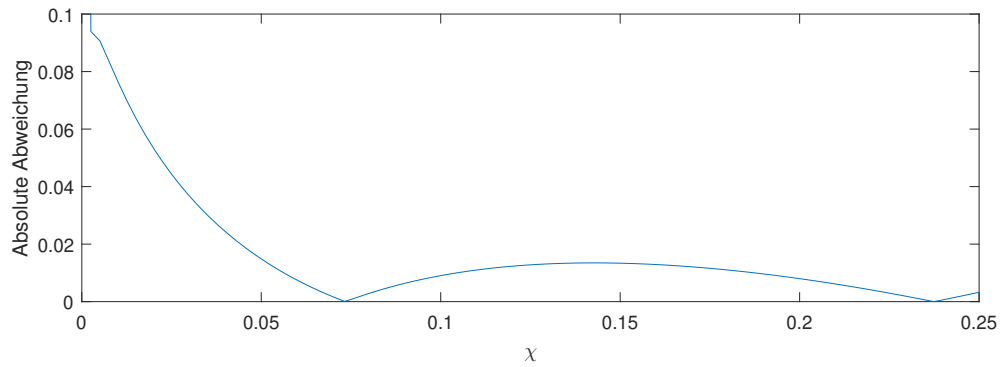


Abbildung 4.9: Absolute Abweichung der ζ -Werte von den Werten nach Roik; Einspannung Einzellast in der Mitte $z_p = -\frac{h}{2}, \psi = 1$

Auch für den beidseitig eingespannten Träger ergeben sich keine bedenklich großen Abweichungen, durch welche auf eine Fehlrechnung des Programms geschlossen werden könnte.

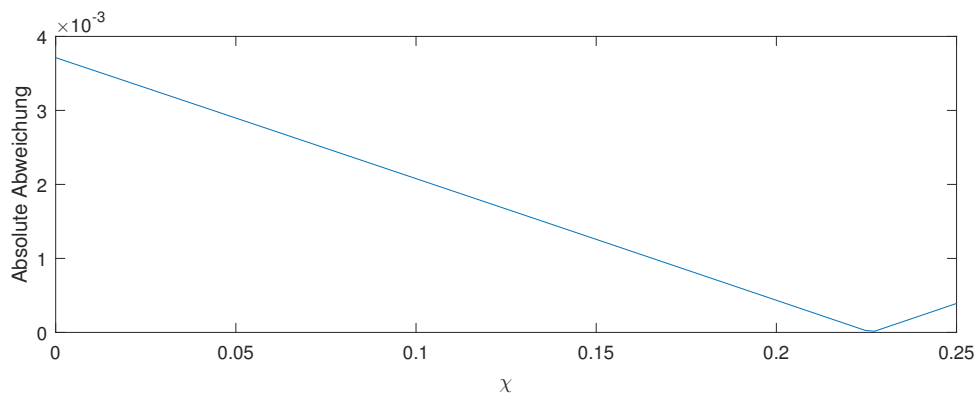


Abbildung 4.10: Absolute Abweichung der ζ -Werte von den Werten nach Roik; Gabellager konstante Linienlast $z_p = 0, \psi = 1$

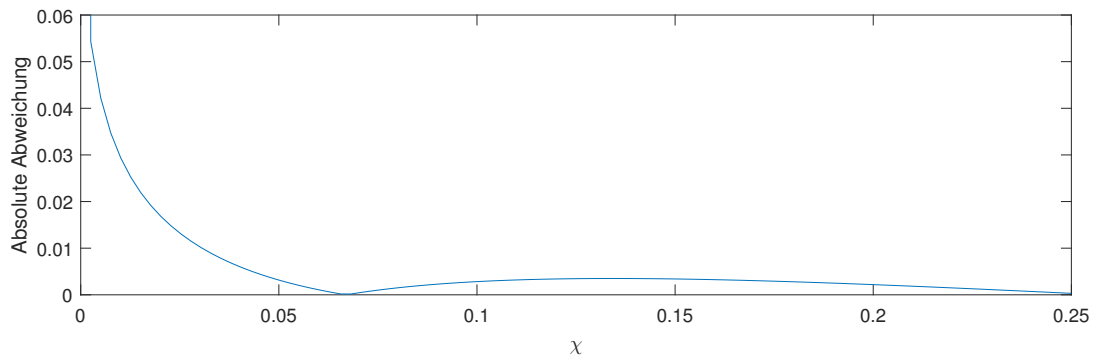


Abbildung 4.11: Absolute Abweichung der ζ -Werte von den Werten nach Roik; Gabellager konstante Linienlast $z_p = -\frac{h}{2}$, $\psi = 1$

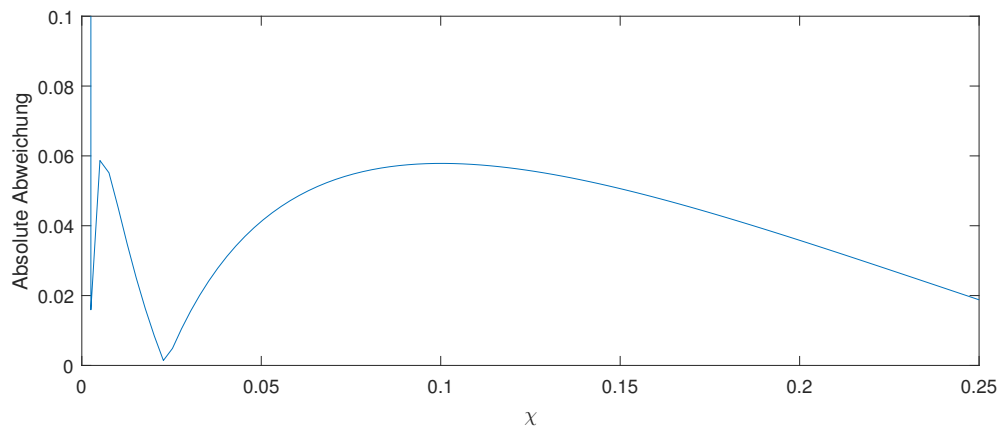


Abbildung 4.12: Absolute Abweichung der ζ -Werte von den Werten nach Roik; Einspannung konstante Linienlast $z_p = 0$, $\psi = 1$

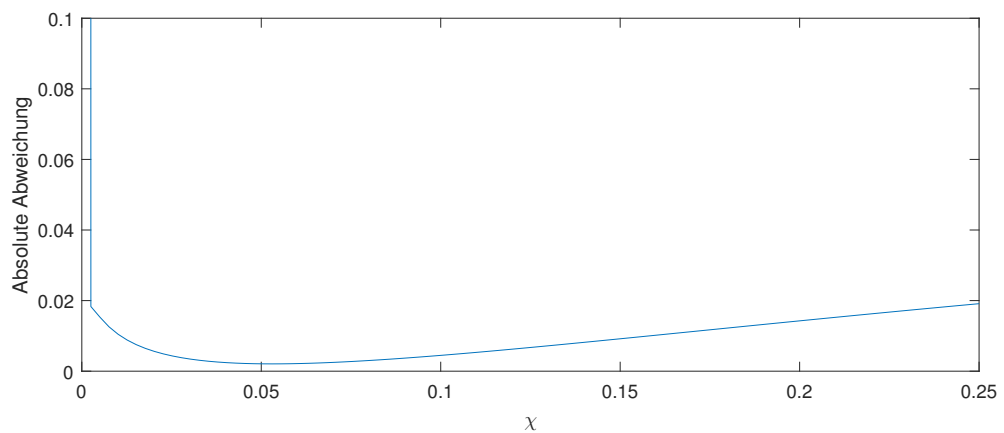


Abbildung 4.13: Absolute Abweichung der ζ -Werte von den Werten nach Roik; Einspannung konstante Linienlast $z_p = -\frac{h}{2}$, $\psi = 1$

Die größte Abweichung taucht beim beidseitig eingespannten Träger mit konstanter Linienlast für $\psi = 1$ und $z_p = 0$ auf. Hier ergibt sich für $\chi = 0,1$ mit einer absoluten Abweichung von 0,06 trotzdem nur eine relative Abweichung von ca. 2%.

4.4 Schwachstellen der Programmierung

Die in ROIK et al. (1972) aufgeführten Graphen (Abb. A.1-A.8) liefern nur Ergebnisse für $z_p = 0$ und $z_p = -\frac{h}{2}$. Entspricht z_p nicht genau einem dieser Fälle, muss aus vorhandenen Werten inter- oder extrapoliert werden. Das in diesem Rahmen geschriebene Programm liefert für $z_p \leq 0$, also Lastangriff am bzw. oberhalb des Schubmittelpunkts plausible Ergebnisse. Für $z_p > 0$ sind die Werte für M_{cr} nicht mehr korrekt. Im begrenzten zeitlichen Rahmen dieser Arbeit konnte bisher keine Lösung für dieses Problem gefunden werden. Ursachen liegen aber wahrscheinlich in der Implementierung der Potentialanteile der zweiten Variation, bei der am Ende eine Änderung des Vorzeichens von z_p lediglich zur Umkehrung des Vorzeichens des Eigenwertes führt. Eigentlich müsste ein $z_p > 0$ zur Vergrößerung des Eigenwerts führen.

4.5 Zusammenfassung der Validierung

Zusammenfassend lässt sich sagen, dass die im Rahmen der Arbeit erstellte Programmierung für die Systeme: Einfeldträger mit beidseitiger Gabellagerung oder beidseitige Einspannung, für die Lastfälle mittige Einzellast oder konstante Linienlast, nur gering von den Ergebnissen aus ROIK et al. (1972) abweicht. Aus den vorhandenen Abweichungen kann nicht auf einen grundsätzlichen Fehler in der in diesem Rahmen erstellten Programmierung geschlossen werden. Für $z_p > 0$ liefert die Programmierung keine gültigen Ergebnisse.

Das Programm kann für in der Praxis vorkommende Systemstützweiten (am besten < 15 m) für $z_p \leq 0$ zur Berechnung des idealen Biegedrillknickmoments verwendet werden. Auch für verschiedene ψ -Werte, also Einspannmomente/ Teileinspannungen am Rand, liefert das Programm valide Ergebnisse.

5 Auswertung

Die im Rahmen der hier umgesetzten Programmierung entstandenen ζ -Wert Graphen sind in Abb. 5.1 bis 5.8 dargestellt und können genauso wie die Diagramme nach ROIK et al. (1972) verwendet werden.

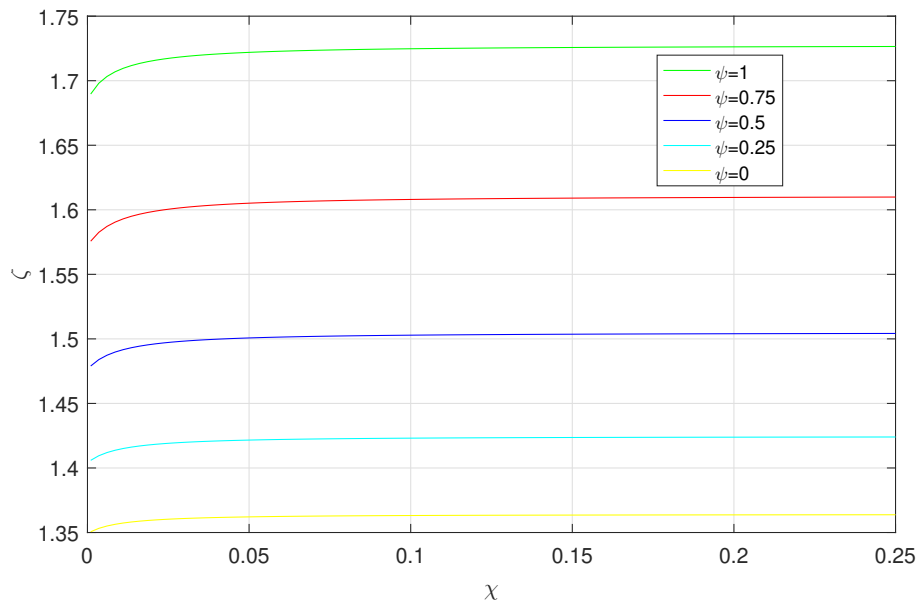


Abbildung 5.1: Plot der ζ -Werte für gabelgelagerten Einfeldträger mit Einzellast in der Mitte und $z_p = 0$

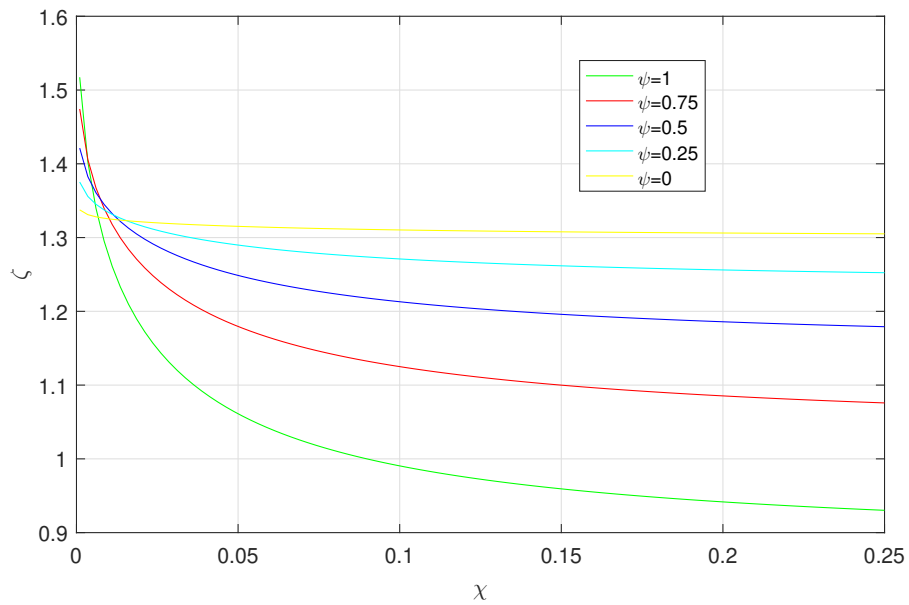


Abbildung 5.2: Plot der ζ -Werte für gabelgelagerten Einfeldträger mit Einzellast in der Mitte und $z_p = -\frac{h}{2}$

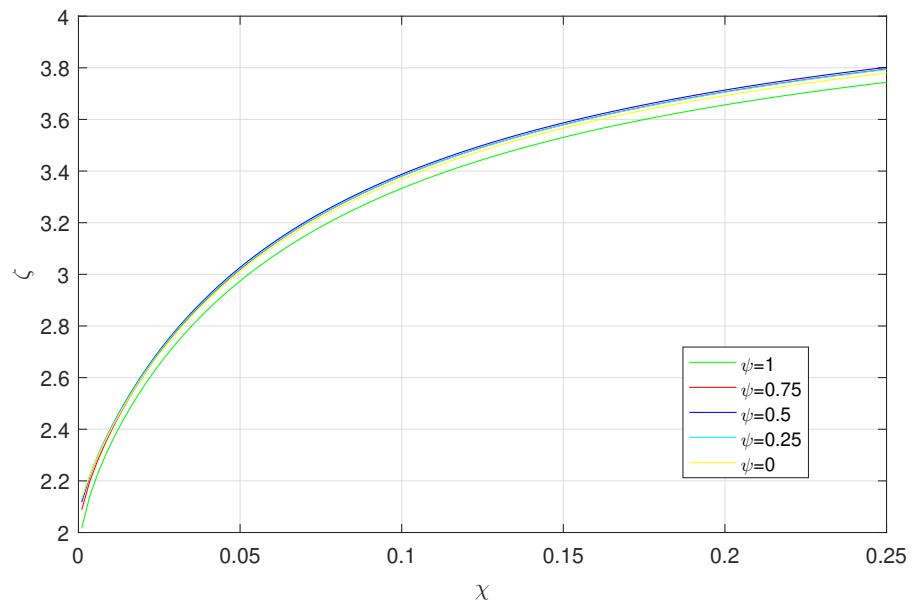


Abbildung 5.3: Plot der ζ -Werte für eingespannten Einfeldträger mit Einzellast in der Mitte und $z_p = 0$

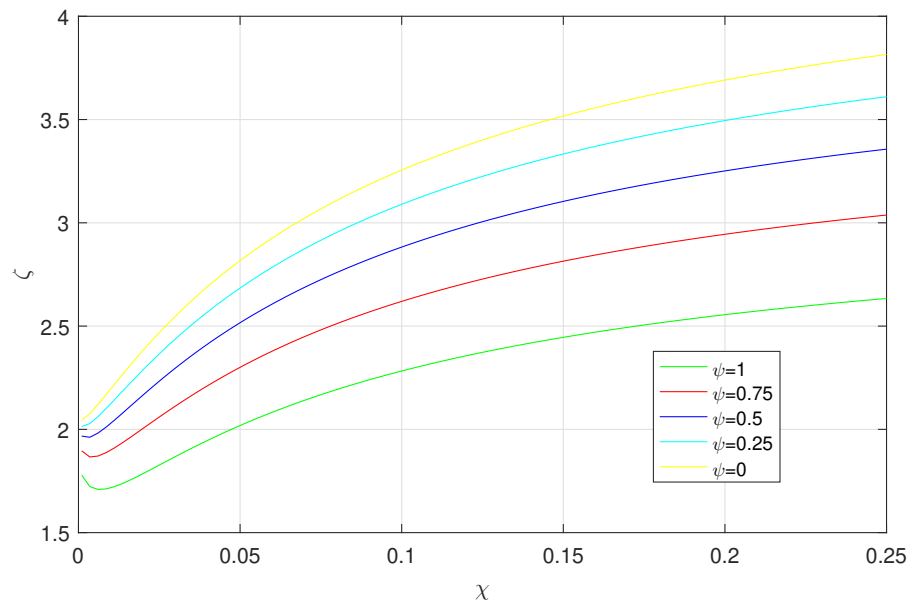


Abbildung 5.4: Plot der ζ -Werte für eingespannten Einfeldträger mit Einzellast in der Mitte und $z_p = -\frac{h}{2}$

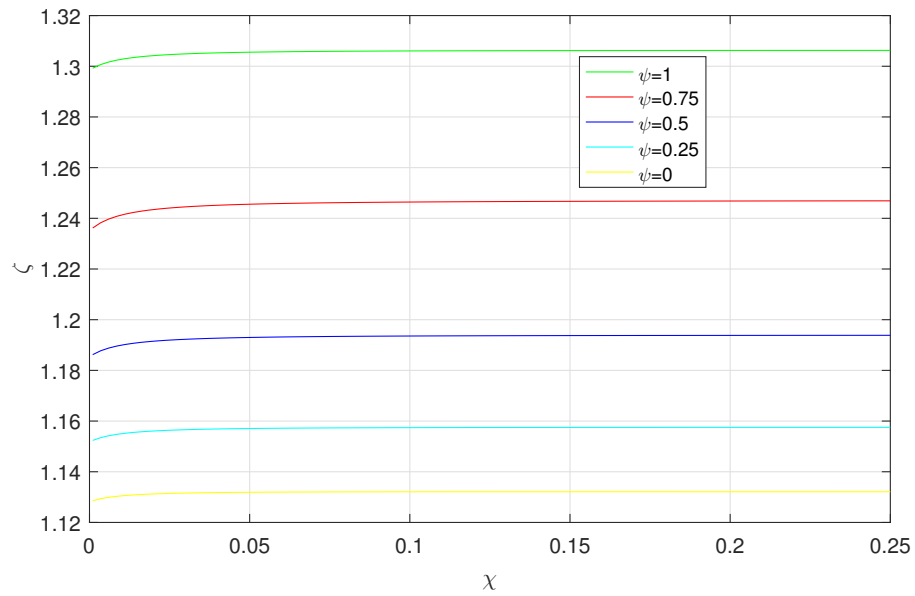


Abbildung 5.5: Plot der ζ -Werte für gabelgelagerten Einfeldträger mit konstanter Linienlast und $z_p = 0$

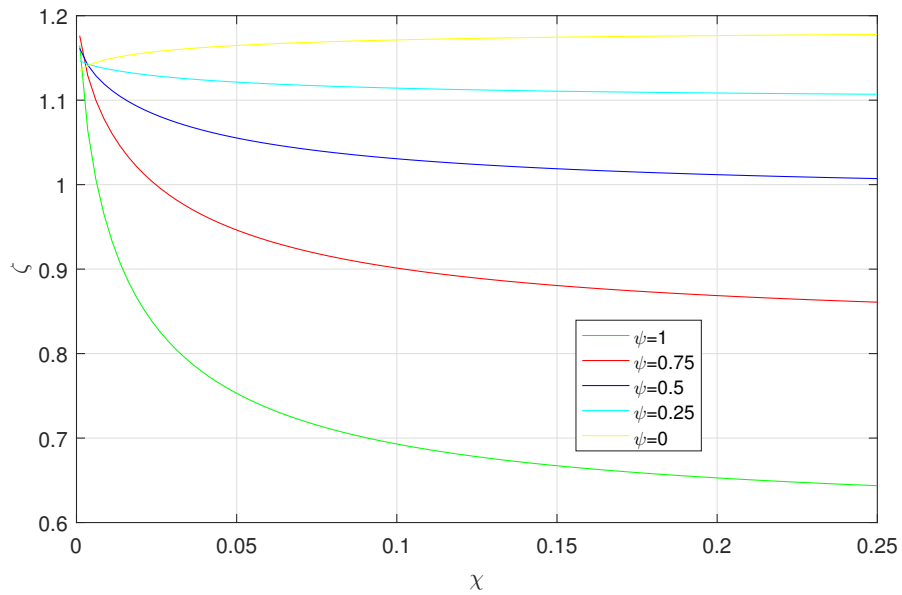


Abbildung 5.6: Plot der ζ -Werte für gabelgelagerten Einfeldträger mit konstanter Linienlast und $z_p = -\frac{h}{2}$

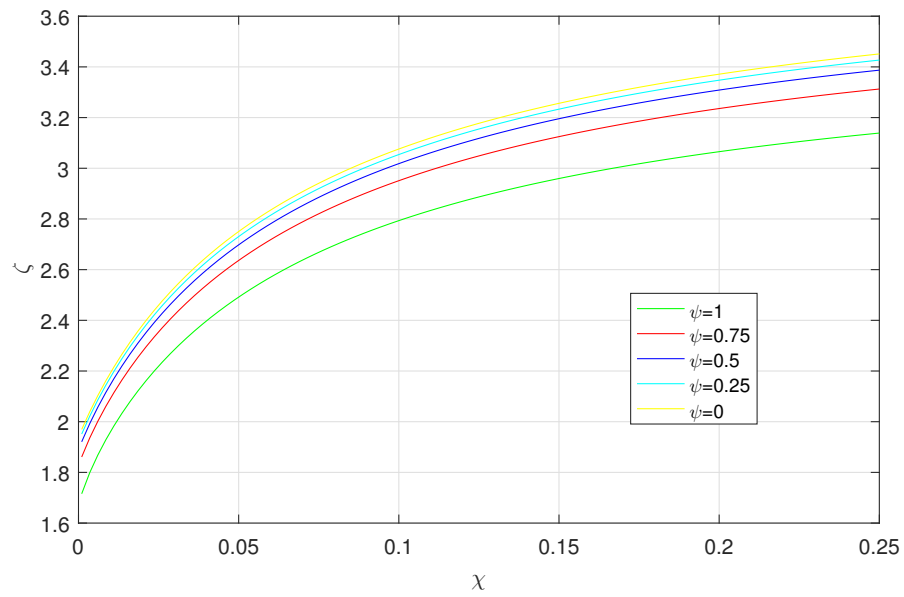


Abbildung 5.7: Plot der ζ -Werte für eingespannten Einfeldträger mit konstanter Linienlast und $z_p = 0$

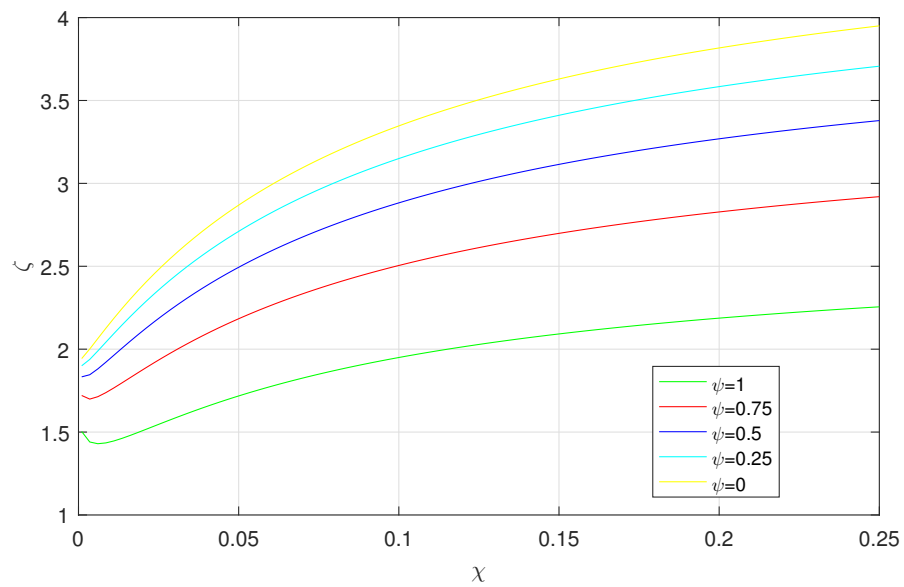


Abbildung 5.8: Plot der ζ -Werte für eingespannten Einfeldträger mit konstanter Linienlast und $z_p = -\frac{h}{2}$

6 Fazit und Ausblick

Die Berechnung des idealen Biegedrillknickmoments stellt in der Praxis häufig ein Problem dar. Bausoftware liefert oft sehr unterschiedliche Ergebnisse und auch im Eurocode gibt es keine einheitliche Regelung zur Bestimmung des kritischen Moments. Aus diesem Grund haben alte, meist nicht in digitaler Form vorliegende Berechnungsverfahren immer noch nicht komplett an Relevanz verloren. Eines dieser manuellen Verfahren ist die Berechnung von M_{cr} mithilfe von ζ -Diagrammen nach ROIK et al. (1972). Damit kann M_{cr} für viele gebrauchshübliche statische Systeme verhältnismäßig einfach über ablesen von ζ -Werten aus Diagrammen und einsetzen in eine Formel ermittelt werden.

Im Rahmen dieser Arbeit wurde versucht, dieses Berechnungsverfahren in Matlab umzusetzen mit dem Ziel ideale Biegedrillknickmomente schnell, flexibel und ohne Ableseungenauigkeiten bestimmen zu können. Das Berechnungsverfahren basiert auf der Energiemethode und wird in der Programmierung hauptsächlich mit Matrizenoperationen umgesetzt. Es wurde versucht, die Basis des Programms möglichst allgemein zu halten, um mögliche Erweiterungen zu vereinfachen. Die Gültigkeit des Programms konnte durch den Abgleich mit den ζ -Diagrammen nach ROIK et al. (1972) für verschiedene fest implementierte Systeme für $z_p \leq 0$ gezeigt werden. Bei der Berechnung von M_{cr} für von den fest implementierten Systemen abweichenden Rand- und Übergangsbedingungen ist bis jetzt noch der manuelle Eingriff des Nutzers in den Code nötig und auch die Validität der Ergebnisse unsicher.

Das entwickelte Programm kann nun als Grundlage für weitere Arbeiten verwendet werden. Eine flexiblere Benutzereingabe des statischen Systems und der Belastung wären ein nächster Schritt. Auch die Implementierung eines Lastangriffs unterhalb des Schubmittelpunktes ($z_p > 0$) wären eine sinnvolle Erweiterung. Weiter würden auch Federhalterungen an verschiedenen Querschnittsstellen oder eine Drehfederbettung der Träger zur Aufwertung beitragen.

A Anhang: Graphen aus Roik et al. (1972)

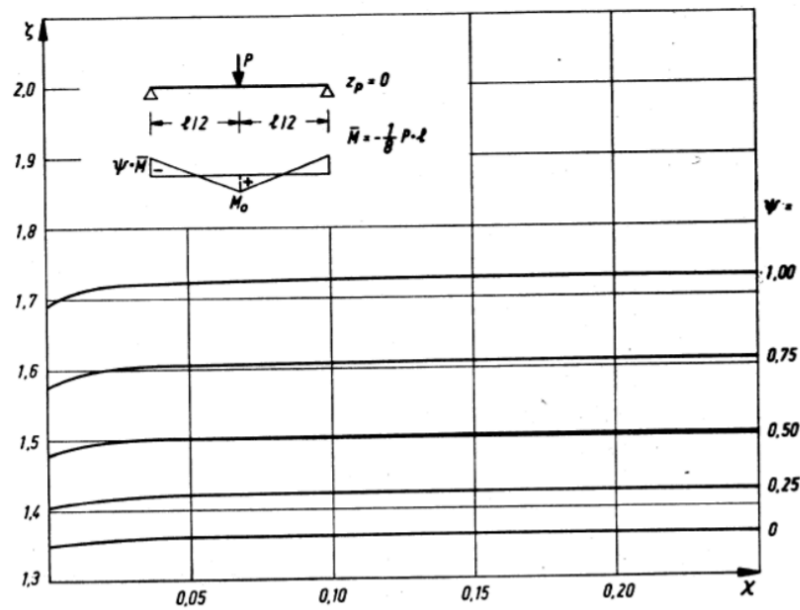


Abbildung A.1: ζ -Werte für gabelgelagerten Einfeldträger mit Einzellast in der Mitte und $z_p = 0$ (aus [Roik et al., 1972])

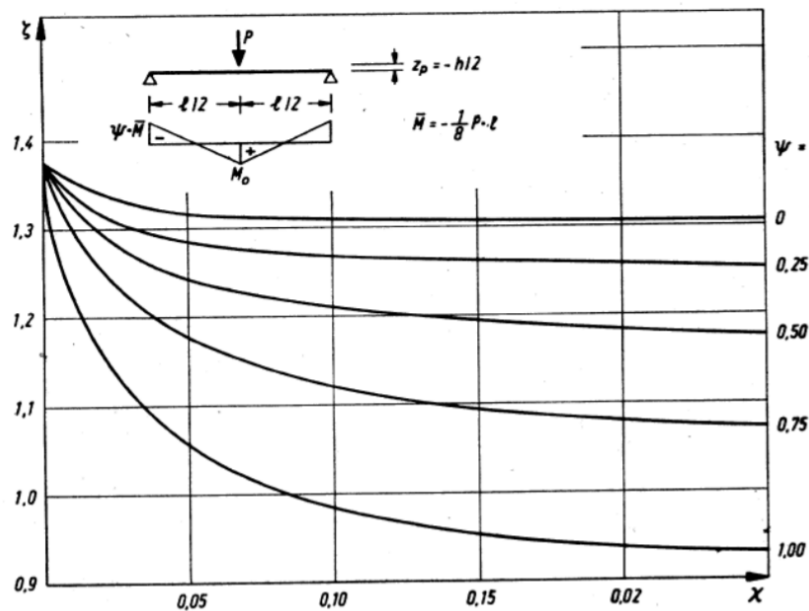


Abbildung A.2: ζ -Werte für gabelgelagerten Einfeldträger mit Einzellast in der Mitte und $z_p = -\frac{h}{2}$ (aus [Roik et al., 1972])

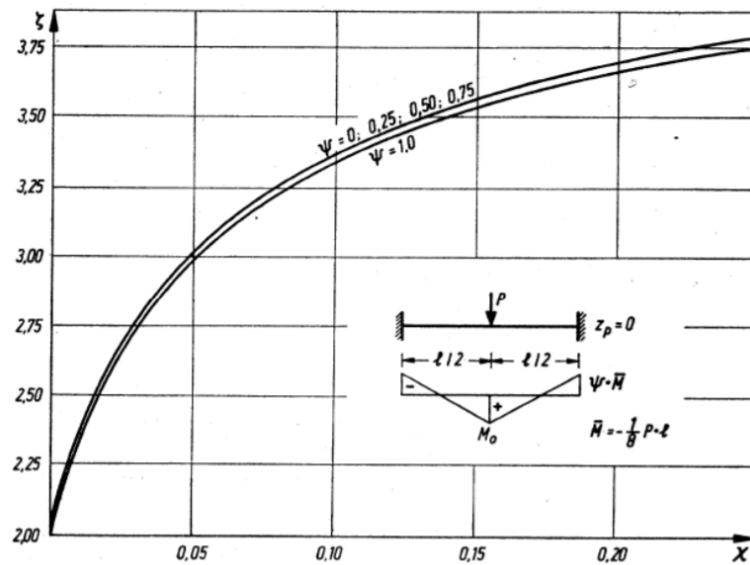


Abbildung A.3: ζ -Werte für eingespannten Einfeldträger mit Einzellast in der Mitte und $z_p = 0$ (aus [Roik et al., 1972])

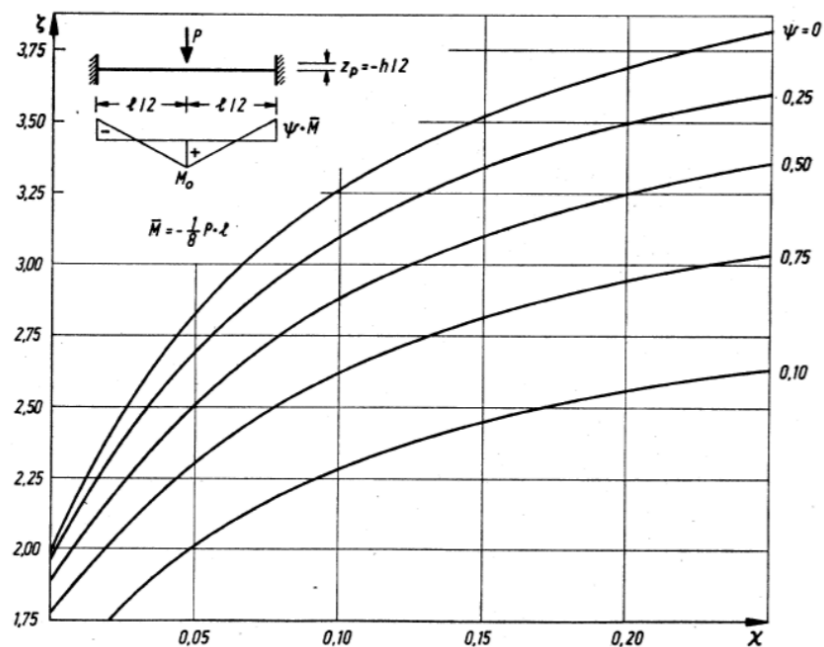


Abbildung A.4: ζ -Werte für eingespannten Einfeldträger mit Einzellast in der Mitte und $z_p = -\frac{h}{2}$ (aus [Roik et al., 1972])

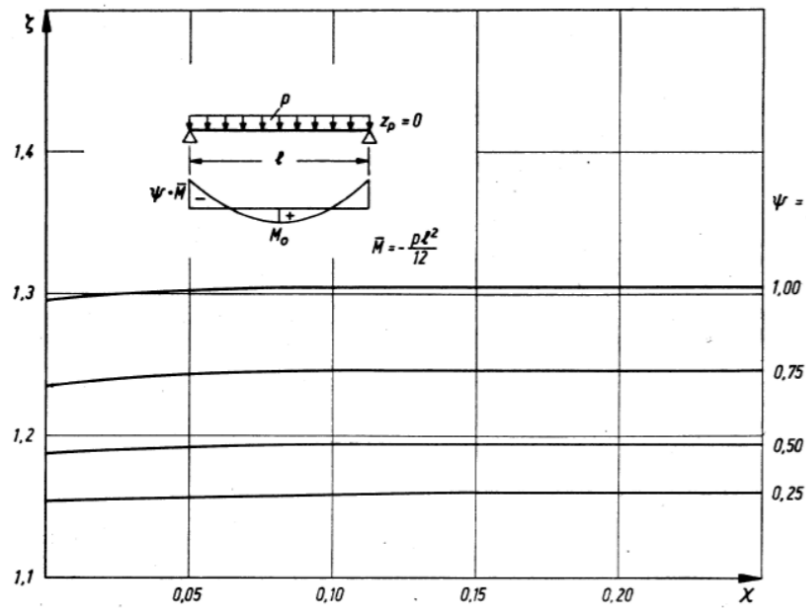


Abbildung A.5: ζ -Werte für gabelgelagerten Einfeldträger mit konstanter Linienlast und $z_p = 0$ (aus [Roik et al., 1972])

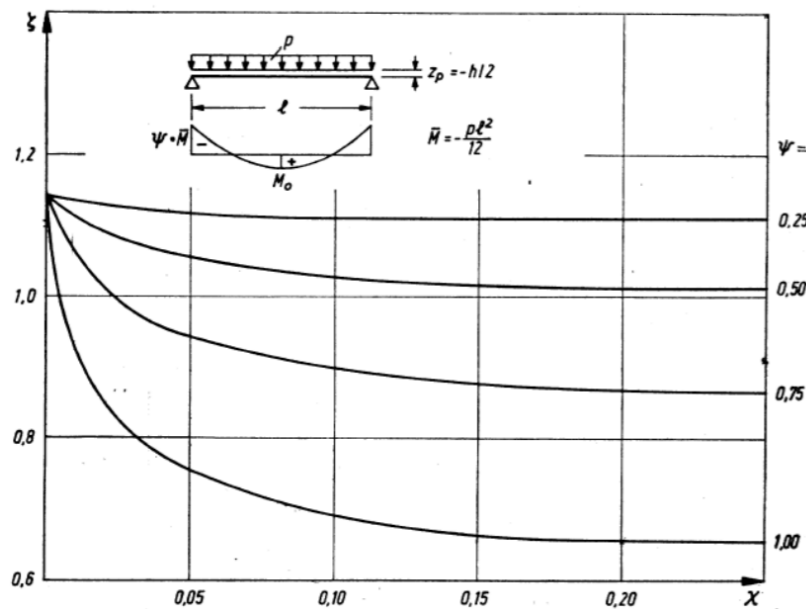


Abbildung A.6: ζ -Werte für gabelgelagerten Einfeldträger mit konstanter Linienlast und $z_p = -\frac{h}{2}$ (aus [Roik et al., 1972])

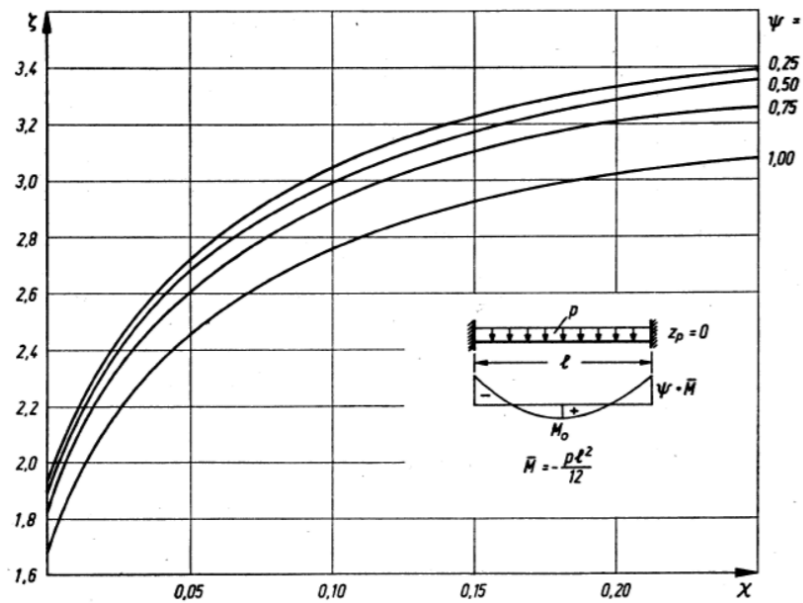


Abbildung A.7: ζ -Werte für eingespannten Einfeldträger mit konstanter Linienlast und $z_p = 0$ (aus [Roik et al., 1972])

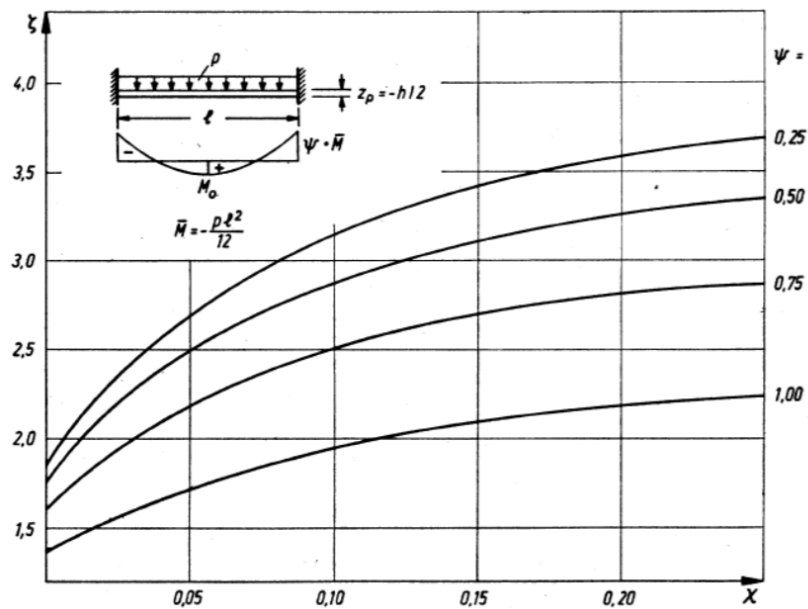


Abbildung A.8: ζ -Werte für eingespannten Einfeldträger mit konstanter Linienlast und $z_p = -\frac{h}{2}$ (aus [Roik et al., 1972])

B Anhang: Quellcode

B.1 Datei integralmatrixGUI.m

Quellcode B.1: Quellcode zur Erstellung der Integralmatrizen als Basis für weitere Matrizenoperationen

```

1 function A=integralmatrixGUI(AbleitungF1,AbleitungF2,Vorfaktor)
2
3 % AbleitungF1: Ableitungsgrad des Ersten Polynoms
4 % AbleitungF2: Ableitungsgrad des zweiten Polynoms
5 % Vorfaktor: Polynomgrad des vorangestellten Ceta ( entweder 0,1 oder 2)
6
7
8 % H=[1,0,0,-10,15,-6;           %H ist die Hemitesche Polynommatrix 5. Grades
9 %   0,0,0,10,-15,6;
10 %   0,1,0,-6,8,-3;
11 %   0,0,0,-4,7,-3;
12 %   0,0,0.5,-1.5,1.5,-0.5;
13 %   0,0,0,0.5,-1,0.5];
14
15
16 H=[1,0,0,0,-35,84,-70,20;      %H ist die Hemitesche Polynommatrix 7. Grades
17   0,0,0,0,35,-84,70,-20;
18   0,1,0,0,-20,45,-36,10;
19   0,0,0,0,-15,39,-34,10;
20   0,0,0.5,0,-5,10,-7.5,2;
21   0,0,0,0,2.5,-7,6.5,-2;
22   0,0,0,1/6,-2/3,1,-2/3,1/6;
23   0,0,0,0,-1/6,0.5,-0.5,1/6];
24 %-----
25 p=size(H,2);
26
27 %Erstellen der Vorfaktormatrizen für die 1. und 2. Ableitung
28 VorfaktorErsteAbl=[0:1:p-1];
29 VorfaktorErsteAblMatrix= repmat(VorfaktorErsteAbl,p,1);
30 VorfaktorZweiteAbl=zeros(1,p);
31 for i=1:p-1
32 VorfaktorZweiteAbl(1,i+1)=VorfaktorErsteAbl(1,i).*VorfaktorErsteAbl(1,i+1);
33 end
34 VorfaktorZweiteAblMatrix= repmat(VorfaktorZweiteAbl,p,1);
35 %-----
36
37 integralmatrix=zeros(p,p);
38 B=zeros(p,2*p-1);
39 t=0;
40
41
42
43 for s=1:p %Spalte in die am Ende eingefügt wird
44 if AbleitungF1==0 %Multiplikation der Vorfaktoren
45     Hvorfaktor=H; %mit der H-Matrix für die verschiedenen
46     %Ableitungen
47 elseif AbleitungF1==1 %mit der die Funktion aufgerufen wird
48     Hvorfaktor=H.*VorfaktorErsteAblMatrix;
49 elseif AbleitungF1==2
50     Hvorfaktor=H.*VorfaktorZweiteAblMatrix;
51 end
52 if AbleitungF2==0

```

```

53     HB=H;
54 elseif AbleitungF2==1
55     HB=H.*VorfaktorErsteAblMatrix;
56 elseif AbleitungF2==2
57     HB=H.*VorfaktorZweiteAblMatrix;
58 end
59
60 if AbleitungF1+AbleitungF2==0
61 Wannbruchstartet=0;
62 elseif AbleitungF1+AbleitungF2==1
63 Wannbruchstartet=1;
64 elseif AbleitungF1+AbleitungF2==2
65 Wannbruchstartet=2;
66 elseif AbleitungF1+AbleitungF2==3
67 Wannbruchstartet=3;
68 elseif AbleitungF1+AbleitungF2==4
69 Wannbruchstartet=4;
70 end
71
72 if Vorfaktor==1
73     Wannbruchstartet=Wannbruchstartet-1;
74 elseif Vorfaktor==2
75     Wannbruchstartet=Wannbruchstartet-2;
76 end
77
78 for i=1:p
79     for j=1:p
80         B(i,j+t)=HB(s,j);
81     end
82     if t<=p
83         t=t+1;
84     else
85         break;
86     end
87 end
88
89 Bruchvektor=zeros(2*p-1,1);
90
91 for k=1:2*p-1-Wannbruchstartet
92     Bruchvektor(k+Wannbruchstartet,1)=1/k;
93 end
94 t=0;
95 A(:,s)=Hvorfaktor*B*Bruchvektor;           %Spalten in Zeilen umwandeln
96 end
97
98 end

```

B.2 Datei EingespanntEinzellastGUI.m

Quellcode B.2: Quellcode zur Berechnung von M_{cr} für einen beidseitig eingespannten Einfeldträger mit Einzellast in der Mitte

```

1 function Mcr=EingespanntEinzellastGUI(L,psi,Iy,Iz,It,Iw,E,G,zPz)
2 format short g
3
4 %Funktion bekommt alle Querschnittswerte,... aus GUI übergeben.
5 %Pz frei wählbar, kritisches Moment lastunabhängig ist.
6 Pz=221.63; %Pz in Feldmitte in [kN]
7
8 %=====
9 %Laden der benötigten Integralmatrizen

```

```

10 G11=integralmatrixGUI(1,1,0);
11 G111=integralmatrixGUI(1,1,1);
12 G112=integralmatrixGUI(1,1,2);
13 G22=integralmatrixGUI(2,2,0);
14 G00=integralmatrixGUI(0,0,0);
15 G021=integralmatrixGUI(0,2,1);
16 G02=integralmatrixGUI(0,2,0);
17 G20=integralmatrixGUI(2,0,0);
18 G201=integralmatrixGUI(2,0,1);
19
20 %Berechnung der Forfaktoren
21 F1=1/(L/2)*(G*It); F2=1/(L/2)^3*(E*Iw); F3=1/(L/2)*(E*Iz); F4=1/(L/2)*(E*Iy); %Forfaktoren für
    %Steifigkeitsmatrix
22
23 %=====
24
25 %Erzeugen der Teilmatrizen:
26
27 %Steifigkeitsmatrix
28 Avv1=F2*G22+F1*G11; %Bei den Matrizen z.B.: Avv1 steht die 1 für linkes und 2 für rechtes
    %Teilsystem
29 Ann1=F3*G22;
30 Avv2=Avv1;
31 Ann2=Ann1;
32
33 %Belastungsmatrix
34 Bvv1=-1*(zeros(8));
35 Bnn1=-1*(zeros(8));
36 Bvn1=-1*(-psi*(Pz*L/8)*G02+(Pz*L/4)*G021); %Momentenverlauf abhängig von gewähltem psi.
37 Bnv1=Bvn1.';
38
39 Bvv2=Bvv1;
40 Bnn2=Bnn1;
41 Bvn2=-1*((Pz*L/4-psi*Pz*L/8)*G02-(Pz*L/4)*G021); %Momentenverlauf abhängig von gewähltem psi.
42 Bnv2=Bvn2.';
43 %=====
44
45 % Einführen der Randbedingungen:
46
47 % Randbedingungen für linken Systemabschnitt
48 % Für beidseitig gelenkig wäre RB=[0,1,1,0,0,1,1,1];
49
50 RB=[0,1,0,0,1,1,1,1]; %Beidseitig Eingespannt
51
52 s=size(RB,2);
53 k=0;
54 for i=1:s
55     if RB(1,i)==0 %Streichen der Zeilen und Spalten
56         Bvv1(i-k,:)=[];
57         Bvv1(:,i-k)=[];
58         Bvn1(i-k,:)=[];
59         Bvn1(:,i-k)=[];
60         Bnv1(i-k,:)=[];
61         Bnv1(:,i-k)=[];
62         Bnn1(i-k,:)=[];
63         Bnn1(:,i-k)=[];
64         Avv1(i-k,:)=[];
65         Avv1(:,i-k)=[];
66         Ann1(i-k,:)=[];
67         Ann1(:,i-k)=[];
68         k=k+1;
69     end
70 end
71
72 % Randbedingungen für rechten Systemabschnitt
73 % Für beidseitig gelenkig wäre RB=[1,0,0,1,1,0,1,1];
74

```



```

75 RB=[1,0,0,0,1,1,1,1]; %Beidseitig Eingespannt
76
77 s=size(RB,2);
78 k=0;
79 for i=1:s
80     if RB(1,i)==0           %Streichen der Zeilen und Spalten
81         Bvv2(i-k,:)=[];
82         Bvv2(:,i-k)=[];
83         Bvn2(i-k,:)=[];
84         Bvn2(:,i-k)=[];
85         Bnv2(i-k,:)=[];
86         Bnv2(:,i-k)=[];
87         Bnn2(i-k,:)=[];
88         Bnn2(:,i-k)=[];
89         Avv2(i-k,:)=[];
90         Avv2(:,i-k)=[];
91         Ann2(i-k,:)=[];
92         Ann2(:,i-k)=[];
93         k=k+1;
94     end
95 end
96 %=====
97
98 %Übergangsbedingungen:
99
100 %Erstellen der Systemmatrizen für die folgende Überlagerung
101 n5=zeros(5);
102 Agross=[Avv1,n5,n5,n5;
103         n5,Ann1,n5,n5;
104         n5,n5,Avv2,n5;
105         n5,n5,n5,Ann2];
106
107 Bgross=[Bvv1,Bvn1,n5,n5;
108         Bnv1,Bnn1,n5,n5;
109         n5,n5,Bvv2,Bvn2;
110         n5,n5,Bnv2,Bnn2];
111
112 Bgross(1,1)=Bgross(1,1)+(zPz*Pz)*-1;           %Addition der Potentialanteile aus Einzellasten
113 % (S-Matrix)
114
115 %Addition/ Subtraktion der Richtigen Zeilen und Spalten
116 %voneinander/aufeinander
117 Z=[1,2,3,4,5,6,7,8,9,10;           %Zuordnungsmatrix Z
118    11,13,12,-15,-14,16,18,17,-20,-19];
119 %Überlagerung der Systemmatrizen (Umsetzung der Übergangsbedingungen)
120 for i=1:10
121     for k=1:10
122         if Z(2,k)<=0
123             V=-1;
124         else
125             V=1;
126         end
127         if Z(2,i)>=0
128             A(i,k)=Agross(Z(1,i),Z(1,k))+V*Agross(abs(Z(2,i)),abs(Z(2,k)));
129             B(i,k)=Bgross(Z(1,i),Z(1,k))+V*Bgross(abs(Z(2,i)),abs(Z(2,k)));
130         else
131             A(i,k)=Agross(Z(1,i),Z(1,k))-V*Agross(abs(Z(2,i)),abs(Z(2,k)));
132             B(i,k)=Bgross(Z(1,i),Z(1,k))-V*Bgross(abs(Z(2,i)),abs(Z(2,k)));
133         end
134     end
135 end
136 %=====
137
138 %Kleinsten Eigenwert Berechnen
139 Vki= abs(eigs(A,B,1,'SM'));
140 %=====

```

```

141 %Berechnung von Mcr unter Berücksichtigung von psi
142 Mcr=(Pz*L/4-psi*Pz*L/8)*Vki;
143 end

```

B.3 Datei GabellagerEinzellastGUI.m

Quellcode B.3: Quellcode zur Berechnung von Mcr für einen beidseitig gabelgelagerten Einfeldträger mit Einzellast in der Mitte

```

1 function Mcr=GabellagerEinzellastGUI(L,psi,Iy,Iz,It,Iw,E,G,zPz)
2 format short g
3
4 %Funktion bekommt alle Querschnittswerte,... aus GUI übergeben.
5 %Pz frei wählbar, kritisches Moment lastunabhängig ist.
6 Pz=221.63; %Pz in Feldmitte in [kN]
7
8 %=====
9 %Laden der benötigten Integralmatrizen
10 G11=integralmatrixGUI(1,1,0);
11 G111=integralmatrixGUI(1,1,1);
12 G112=integralmatrixGUI(1,1,2);
13 G22=integralmatrixGUI(2,2,0);
14 G00=integralmatrixGUI(0,0,0);
15 G021=integralmatrixGUI(0,2,1);
16 G02=integralmatrixGUI(0,2,0);
17 G20=integralmatrixGUI(2,0,0);
18 G201=integralmatrixGUI(2,0,1);
19
20 %Berechnung der Forfaktoren
21 F1=1/(L/2)*(G*It); F2=1/(L/2)^3*(E*Iw); F3=1/(L/2)*(E*Iz); F4=1/(L/2)*(E*Iy); %Forfaktoren für
    %Steifigkeitsmatrix
22
23 %=====
24
25 %Erzeugen der Teilmatrizen:
26
27 %Steifigkeitsmatrix
28 Avv1=F2*G22+F1*G11; %Bei den Matrizen z.B.: Avv1 steht die 1 für linkes und 2 für rechtes
    %Teilsystem
29 Ann1=F3*G22;
30 Avv2=Avv1;
31 Ann2=Ann1;
32
33 %Belastungsmatrix
34 Bvv1=-1*(zeros(8));
35 Bnn1=-1*(zeros(8));
36 Bvn1=-1*((Pz*L/4)*G021-psi*(Pz*L/8)*G02); %Momentenverlauf abhängig von gewähltem psi.
37 Bnv1=Bvn1.';
38
39 Bvv2=Bvv1;
40 Bnn2=Bnn1;
41 Bvn2=-1*((Pz*L/4-psi*(Pz*L/8))*G02-(Pz*L/4)*G021); %Momentenverlauf abhängig von gewähltem psi.
42 Bnv2=Bvn2.';
43 %=====
44
45 % Einführen der Randbedingungen:
46
47 % Randbedingungen für linken Systemabschnitt
48 % Für beidseitig Eingespannt wäre RB=[0,1,0,0,1,1,1,1];
49 RB=[0,1,1,0,0,1,1,1]; %Gelenkig
50 s=size(RB,2);

```

```

51 k=0;
52 for i=1:s
53     if RB(1,i)==0                                %Streichen der Zeilen und Spalten
54         Bvv1(i-k,:)=[];
55         Bvv1(:,i-k)=[];
56         Bvn1(i-k,:)=[];
57         Bvn1(:,i-k)=[];
58         Bnv1(i-k,:)=[];
59         Bnv1(:,i-k)=[];
60         Bnn1(i-k,:)=[];
61         Bnn1(:,i-k)=[];
62         Avv1(i-k,:)=[];
63         Avv1(:,i-k)=[];
64         Ann1(i-k,:)=[];
65         Ann1(:,i-k)=[];
66         k=k+1;
67     end
68 end
69
70 % Randbedingungen für rechten Systemabschnitt
71 % Für beidseitig Eingespannt wäre RB=[1,0,0,0,1,1,1,1];
72 RB=[1,0,0,1,1,0,1,1]; %Gelenkig
73 s=size(RB,2);
74 k=0;
75 for i=1:s
76     if RB(1,i)==0                                %Streichen der Zeilen und Spalten
77         Bvv2(i-k,:)=[];
78         Bvv2(:,i-k)=[];
79         Bvn2(i-k,:)=[];
80         Bvn2(:,i-k)=[];
81         Bnv2(i-k,:)=[];
82         Bnv2(:,i-k)=[];
83         Bnn2(i-k,:)=[];
84         Bnn2(:,i-k)=[];
85         Avv2(i-k,:)=[];
86         Avv2(:,i-k)=[];
87         Ann2(i-k,:)=[];
88         Ann2(:,i-k)=[];
89         k=k+1;
90     end
91 end
92 %=====
93
94 %Übergangsbedingungen:
95
96 %Erstellen der Systemmatrizen für die folgende Überlagerung
97 n5=zeros(5);
98 Agross=[Avv1,n5,n5,n5;
99         n5,Ann1,n5,n5;
100         n5,n5,Avv2,n5;
101         n5,n5,n5,Ann2];
102
103 Bgross=[Bvv1,Bvn1,n5,n5;
104         Bnv1,Bnn1,n5,n5;
105         n5,n5,Bvv2,Bvn2;
106         n5,n5,Bnv2,Bnn2];
107
108 Bgross(1,1)=Bgross(1,1)+(zPz*Pz)*-1; %Addition der S-Matrix
109
110 %Addition/ Subtraktion der Richtigen Zeilen und Spalten
111 %voneinander/aufeinander
112
113 Z=[1,2,3,4,5,6,7,8,9,10;                                %Zuordnungsmatrix Z
114    11,-12,13,-15,-14,16,-17,18,-20,-19];
115
116 for i=1:10
117     for k=1:10

```

```

118         if Z(2,k)<=0
119             V=-1;
120         else
121             V=1;
122         end
123         if Z(2,i)>=0
124             A(i,k)=Agross(Z(1,i),Z(1,k))+V*Agross(abs(Z(2,i)),abs(Z(2,k)));
125             B(i,k)=Bgross(Z(1,i),Z(1,k))+V*Bgross(abs(Z(2,i)),abs(Z(2,k)));
126         else
127             A(i,k)=Agross(Z(1,i),Z(1,k))-V*Agross(abs(Z(2,i)),abs(Z(2,k)));
128             B(i,k)=Bgross(Z(1,i),Z(1,k))-V*Bgross(abs(Z(2,i)),abs(Z(2,k)));
129         end
130     end
131 end
132
133 %=====
134 %Kleinsten Eigenwert Berechnen
135 Vki= abs(eigs(A,B,1,'SM'));
136 %=====
137 %Berechnung von Mcr
138 Mcr=((Pz*L)/4-psi*(Pz*L)/8)*Vki;
139 end

```

B.4 Datei GabellagerLinienlastGUI.m

Im Folgenden ist der Quellcode von GabellagerLinienlastGUI.m dargestellt. Im Quellcode für EingespanntLinienlastGUI.m würden sich nur die Randbedingungen (Zeile 39, 40) ändern (im Quelltext als Kommentar enthalten). Aus diesem Grund ist hier nur ein Quellcode aufgeführt.

Quellcode B.4: Quellcode zur Berechnung von Mcr für einen beidseitig gabelgelagerten Einfeldträger mit konstanter Linienlast

```

1 function Mcr=GabellagerLinienlastGUI(L,psi,Iy,Iz,It,Iw,E,G,zpz)
2 format short g
3
4 %Funktion bekommt alle Querschnittswerte,... aus GUI übergeben.
5 %pz frei wählbar, kritisches Moment lastunabhängig ist.
6 pz=50; %konstante Linienlast pz über gesamtes Feld in [kN/cm]
7
8 %=====
9
10 G11=integralmatrixGUI(1,1,0);
11 G111=integralmatrixGUI(1,1,1);
12 G112=integralmatrixGUI(1,1,2);
13 G22=integralmatrixGUI(2,2,0);
14 G00=integralmatrixGUI(0,0,0);
15 G021=integralmatrixGUI(0,2,1);
16 G022=integralmatrixGUI(0,2,2);
17 G02=integralmatrixGUI(0,2,0);
18 G20=integralmatrixGUI(2,0,0);
19 G201=integralmatrixGUI(2,0,1);
20
21 %Forfaktoren
22 F1=1/(L)*(G*It); F2=1/(L)^3*(E*Iw); F3=1/(L)*(E*Iz); F4=1/(L)*(E*Iy); %Forfaktoren für
    %Steifigkeitsmatrix

```

```

23 F5=L*(pz*zpz);
24 %=====
25
26 %Teilmatrizen
27 Avv1=F2*G22+F1*G11;
28 Ann1=F3*G22;
29 Anv1=zeros(8);
30 Avn1=zeros(8);
31
32 Bvv1=-1*(F5*G00);
33 Bnn1=-1*(zeros(8));
34 Bvn1=-1*(-(pz*L^2/2)*G022+(pz*L^2/2)*G021-psi*(pz*L^2/12)*G02);
35 Bnv1=Bvn1.';
36
37 %Randbedingungen
38
39 RB=[0,0,1,1,0,0,1,1]; %Beidseitig Gelenkig
40 %RB=[0,0,0,0,1,1,1,1]; %Beidseitig Eingespannt
41 s=size(RB,2);
42 k=0;
43 for i=1:s
44     if RB(1,i)==0
45         Bvv1(i-k,:)=[];
46         Bvv1(:,i-k)=[];
47         Bvn1(i-k,:)=[];
48         Bvn1(:,i-k)=[];
49         Bnv1(i-k,:)=[];
50         Bnv1(:,i-k)=[];
51         Bnn1(i-k,:)=[];
52         Bnn1(:,i-k)=[];
53         Avv1(i-k,:)=[];
54         Avv1(:,i-k)=[];
55         Ann1(i-k,:)=[];
56         Ann1(:,i-k)=[];
57         Anv1(i-k,:)=[];
58         Anv1(:,i-k)=[];
59         Avn1(i-k,:)=[];
60         Avn1(:,i-k)=[];
61         k=k+1;
62     end
63 end
64 A=[Avv1,Avn1;
65     Anv1,Ann1];
66 B=[Bvv1,Bvn1;
67     Bnv1,Bnn1];
68
69 %Kleinsten Eigenwert Berechnen
70 Vki= abs(eigs(A,B,1,'SM'));
71 %=====
72 Mcr=((pz*L^2)/8-psi*(pz*L^2)/12)*Vki;
73 end

```

B.5 Datei GraphGabellagerungEinzellast.m

Im Folgenden ist der Quellcode von GraphGabellagerungEinzellast.m dargestellt. Im Quellcode für andere Systeme würde sich nur der Funktionsaufruf in Zeile 5 ändern. Aus diesem Grund ist nur ein Quellcode dazu aufgeführt.

Quellcode B.5: Quellcode zur Berechnung der ζ -Graphen für die beidseitige Gabellagerung

```

1 function GraphGabellagerungEinzellast
2 %Psi-Wert gibt die Größe des Randmoments an (Psi*(p*L^2)/8)
3 %Querschnittswerte und die Höhe des Lastangriffs (zPz) können für die
4 %Graphenerstellung in der Funktion "GabellagerungEinzellast" geändert werden
5 [~,Iw,E,G,It,Iz,zPz]=GabellagerungEinzellast(1000,1);
6 n=100;
7 k=linspace(0.001,0.25,n);
8 for i=1:n
9     J(i)=sqrt((E*Iw)/(k(i)*G*It)); %Erstellen eines Chi-Vektors für die Stützweiten K
10 end
11 %Erstellen der Einzelnen Graphen mit verschiedenen psi-werten
12 psi=1;
13 for j=1:n
14     psi1(j)=GabellagerungEinzellast(J(j),psi)/((E*Iz*pi^2)/(J(j)^2)*
15     (sqrt((Iw+0.039*J(j)^2*It)/Iz+0.2566*zPz^2)+0.5066*zPz));
16 end
17 psi=0.75;
18 for j=1:n
19     psi075(j)=GabellagerungEinzellast(J(j),psi)/((E*Iz*pi^2)/(J(j)^2)*
20     (sqrt((Iw+0.039*J(j)^2*It)/Iz+0.2566*zPz^2)+0.5066*zPz));
21 end
22 psi=0.5;
23 for j=1:n
24     psi05(j)=GabellagerungEinzellast(J(j),psi)/((E*Iz*pi^2)/(J(j)^2)*
25     (sqrt((Iw+0.039*J(j)^2*It)/Iz+0.2566*zPz^2)+0.5066*zPz));
26 end
27 psi=0.25;
28 for j=1:n
29     psi025(j)=GabellagerungEinzellast(J(j),psi)/((E*Iz*pi^2)/(J(j)^2)*
30     (sqrt((Iw+0.039*J(j)^2*It)/Iz+0.2566*zPz^2)+0.5066*zPz));
31 end
32 psi=0;
33 for j=1:n
34     psi0(j)=GabellagerungEinzellast(J(j),psi)/((E*Iz*pi^2)/(J(j)^2)*
35     (sqrt((Iw+0.039*J(j)^2*It)/Iz+0.2566*zPz^2)+0.5066*zPz));
36 end
37 plot(k,abs(psi1),'g',k,abs(psi075),'r',k,abs(psi05),'b',k,abs(psi025),'c',k,abs(psi0),'y')
38 grid on;
39 legend({'\psi=1','\psi=0.75','\psi=0.5','\psi=0.25','\psi=0'},'Location','EastOutside')
40
41 end

```

C Anhang: Näherungsfunktionen

Quellcode C.1: Näherungsfunktionen für Roik- und Matlabgraphen

```

1  Gabellagerung Einzellast
2
3  zp=-h/2
4
5  Psi1
6  Roik:
7  r(x)=-0.2895361795*(ln(x))+0.0212364734*(ln(x)^2)+0.328858536468*(cos(x))+1.02762*(atan(x))
8  Matlab:
9  m(x)= -0,207638*( ln(x) ) -0,0080181*( ln(x)^2 ) + 0,474013*( cos(x) ) + 0,8268706477*( atan(x) )
10
11 Psi0
12 Roik:
13 r(x)=-0,0785688*( ln(x) ) -0,005867122*( ln(x)^2 ) + 1,10964*( cos(x) ) + 0,53124955*( atan(x) )
14 Matlab:
15 m(x)= -0,0389107098*( ln(x) ) + -0,0030107252*( ln(x)^2 ) + 1,20884683*( cos(x) ) + 0,34002138*(
    atan(x) )
16
17
18 zp=0
19
20 Psi1
21 Roik:
22 r(x)= -0,04159523*( ln(x) ) -0,00484668*( ln(x)^2 ) + 1,6196190*( cos(x) ) + 0,4288240*( atan(x) )
23 Matlab:
24 m(x)= -0,0318211016*( ln(x) ) -0,003689024*( ln(x)^2 ) +1,639855*( cos(x) ) + 0,395872424*(
    atan(x) )
25
26 Psi0
27 Roik:
28 r(x)=-0,02001835365*( ln(x) ) + -0,00197280220*( ln(x)^2 ) + 1,303115502*( cos(x) ) +
    0,290580218*( atan(x) )
29 Matlab:
30 m(x)=-0,0268877303*( ln(x) ) + -0,0027031257*( ln(x)^2 ) + 1,28973407416*( cos(x) ) +
    0,32397703150*( atan(x) )
31
32
33
34 Eingespannt Einzellast
35
36 zp=-h/2
37
38 Psi1
39 Roik:
40 r(x)=0,8708038412*(ln(x))+0,080179426642*(ln(x)^2)+3,933914282*(cos(x))-0,4932365149*(atan(x))
41
42 Matlab:
43 m(x)= 0,713735228*( ln(x) ) +0,0668881327*( ln(x)^2 ) + 3,5621187471*( cos(x) ) + 0,206000631*(
    atan(x) )
44
45 Psi0
46 Roik:
47 r(x)=1,19253573*(ln(x))+0,10253237*(ln(x)^2)+5,5089103791*(cos(x))-0,222352804*(atan(x))
48
49 Matlab:
50 m(x)= 1,153308806*( ln(x) ) +0,0972160570*( ln(x)^2 ) + 5,4232353985*( cos(x) )
    -0,0884023204758*( atan(x) )
51
52 zp=0
53

```

```

54 Psi1
55 Roik:
56 r(x)=1,00541257*(ln(x))+0,077129025996*(ln(x)^2)+5,3481165896*(cos(x))-0,75649023875*(atan(x))
57
58 Matlab:
59 m(x)= 0,96357678621*( ln(x) ) +0,0720679695*( ln(x)^2 ) + 5,2539870526*( cos(x) ) -
    -0,6355505568*( atan(x) )
60
61 Psi0
62 Roik:
63 r(x)=0,95966341968*(ln(x))+0,0723185049*(ln(x)^2)+5,27572832*(cos(x))-0,523402595*(atan(x))
64
65 Matlab:
66 m(x)= 0,988562783*( ln(x) ) +0,076623846*( ln(x)^2 ) + 5,32685381*( cos(x) ) -0,670638379*(
    atan(x) )
67
68
69
70 Gabellagerung Linienlast
71
72 zp=0
73
74 Psi=1
75 Roik:
76 r(x) = 1.300652478/ (1 + ((1.300652478-1)/-14.633712) * x)
77
78 Matlab:
79 m(x) = 1.30436543 / (1 + ((1.3043654-1)/-38.094126662) * x)
80
81
82 zp=-h/2
83
84 Psi=1
85 Roik:
86 r(x) = -0.000102537*( x^-1 ) + -0.122597*( ln(x) ) + -0.6975935*( exp(-x) ) + 1.05052133*(
    cos(x) )
87
88 Matlab:
89 m(x) = -0.000115586*( x^-1 ) + -0.14452149*( ln(x) ) + -0.904127809*( exp(-x) ) + 1.1848903*(
    cos(x) )
90
91
92 Eingespannt Linienlast
93
94 zp=0
95
96 Psi=1
97 Roik:
98 r(x) = 0.000032356923*( x^-1 ) + 0.2214777*( ln(x) ) + -2.837868012*( exp(-x) ) + 5.829319395*(
    cos(x) )
99
100 Matlab:
101 m(x) = 0.0006322988*( x^-1 ) + 0.3357797*( ln(x) ) + -1.54830859*( exp(-x) ) + 4.97324127*(
    cos(x) )
102
103
104 zp=-h/2
105
106 Psi=1
107 Roik:
108 r(x) = 0.00055764055*( x^-1 ) + 0.1681823*( ln(x) ) + -2.28966175*( exp(-x) ) + 4.406384004*(
    cos(x) )
109
110 Matlab:
111 m(x) = 0.000533107*( x^-1 ) + 0.1559388*( ln(x) ) + -2.5350007*( exp(-x) ) + 4.60590700560*(
    cos(x) )

```


Abbildungsverzeichnis

2.1	Typen des Gleichgewichts	5
3.1	Verwendetes Koordinatensystem	8
3.2	1. Hermitesches Interpolationspolynom 7. Grades	9
3.3	Grundelement 1 mit Freiheitsgraden	10
3.4	Nummerierung der Freiheitsgrade in Anlehnung an [ROIK et al., 1972, S.84] .	16
3.5	Skallierungsfunktion der Übergangsbedingungen	18
4.1	Bedeutung der Variablen ψ und z_P	20
4.2	Absolute Abweichung der ζ -Werte von den Werten nach Roik; Gabellagerung Einzellast in der Mitte $z_p = 0, \psi = 0$	22
4.3	Absolute Abweichung der ζ -Werte von den Werten nach Roik; Gabellagerung Einzellast in der Mitte $z_p = 0, \psi = 1$	22
4.4	Absolute Abweichung der ζ -Werte von den Werten nach Roik; Gabellagerung Einzellast in der Mitte $z_p = -\frac{h}{2}, \psi = 0$	23
4.5	Absolute Abweichung der ζ -Werte von den Werten nach Roik; Gabellagerung Einzellast in der Mitte $z_p = -\frac{h}{2}, \psi = 1$	23
4.6	Absolute Abweichung der ζ -Werte von den Werten nach Roik; Einspannung Einzellast in der Mitte $z_p = 0, \psi = 0$	24
4.7	Absolute Abweichung der ζ -Werte von den Werten nach Roik; Einspannung Einzellast in der Mitte $z_p = 0, \psi = 1$	24
4.8	Absolute Abweichung der ζ -Werte von den Werten nach Roik; Einspannung Einzellast in der Mitte $z_p = -\frac{h}{2}, \psi = 0$	24
4.9	Absolute Abweichung der ζ -Werte von den Werten nach Roik; Einspannung Einzellast in der Mitte $z_p = -\frac{h}{2}, \psi = 1$	25
4.10	Absolute Abweichung der ζ -Werte von den Werten nach Roik; Gabellager konstante Linienlast $z_p = 0, \psi = 1$	25
4.11	Absolute Abweichung der ζ -Werte von den Werten nach Roik; Gabellager konstante Linienlast $z_p = -\frac{h}{2}, \psi = 1$	26
4.12	Absolute Abweichung der ζ -Werte von den Werten nach Roik; Einspannung konstante Linienlast $z_p = 0, \psi = 1$	26
4.13	Absolute Abweichung der ζ -Werte von den Werten nach Roik; Einspannung konstante Linienlast $z_p = -\frac{h}{2}, \psi = 1$	26

5.1	Plot der ζ -Werte für gabelgelagerten Einfeldträger mit Einzellast in der Mitte und $z_p = 0$	29
5.2	Plot der ζ -Werte für gabelgelagerten Einfeldträger mit Einzellast in der Mitte und $z_p = -\frac{h}{2}$	29
5.3	Plot der ζ -Werte für eingespannten Einfeldträger mit Einzellast in der Mitte und $z_p = 0$	30
5.4	Plot der ζ -Werte für eingespannten Einfeldträger mit Einzellast in der Mitte und $z_p = -\frac{h}{2}$	30
5.5	Plot der ζ -Werte für gabelgelagerten Einfeldträger mit konstanter Linienlast und $z_p = 0$	31
5.6	Plot der ζ -Werte für gabelgelagerten Einfeldträger mit konstanter Linienlast und $z_p = -\frac{h}{2}$	31
5.7	Plot der ζ -Werte für eingespannten Einfeldträger mit konstanter Linienlast und $z_p = 0$	32
5.8	Plot der ζ -Werte für eingespannten Einfeldträger mit konstanter Linienlast und $z_p = -\frac{h}{2}$	32
A.1	ζ -Werte für gabelgelagerten Einfeldträger mit Einzellast in der Mitte und $z_p = 0$ (aus [ROIK et al., 1972])	35
A.2	ζ -Werte für gabelgelagerten Einfeldträger mit Einzellast in der Mitte und $z_p = -\frac{h}{2}$ (aus [ROIK et al., 1972])	35
A.3	ζ -Werte für eingespannten Einfeldträger mit Einzellast in der Mitte und $z_p = 0$ (aus [ROIK et al., 1972])	36
A.4	ζ -Werte für eingespannten Einfeldträger mit Einzellast in der Mitte und $z_p = -\frac{h}{2}$ (aus [ROIK et al., 1972])	36
A.5	ζ -Werte für gabelgelagerten Einfeldträger mit konstanter Linienlast und $z_p = 0$ (aus [ROIK et al., 1972])	37
A.6	ζ -Werte für gabelgelagerten Einfeldträger mit konstanter Linienlast und $z_p = -\frac{h}{2}$ (aus [ROIK et al., 1972])	37
A.7	ζ -Werte für eingespannten Einfeldträger mit konstanter Linienlast und $z_p = 0$ (aus [ROIK et al., 1972])	38
A.8	ζ -Werte für eingespannten Einfeldträger mit konstanter Linienlast und $z_p = -\frac{h}{2}$ (aus [ROIK et al., 1972])	38

Tabellenverzeichnis

3.1	Zusammensetzung der Gesamtmatrix G	13
-----	--	----

Literaturverzeichnis

- FALK, S. (1963): *Das Verfahren von Rayleigh-Ritz mit hermiteschen Interpolationspolynomen*. Bd. 43. 4-5. WILEY-VCH Verlag (siehe S. 9).
- PFLÜGER, A. (1975): *Stabilitätsprobleme der Elastostatik*. Dritte neubearbeitete Auflage. Springer Berlin Heidelberg (siehe S. 2, 6).
- ROIK, K., CARL, J. und LINDNER, J. (1972): *Biegetorsionsprobleme gerader dünnwandiger Stäbe*. Verlag von Wilhelm Ernst & Sohn (siehe S. 1, 2, 7, 9, 11, 12, 14, 16, 18–21, 27, 28, 33–38).