



h_da

HOCHSCHULE DARMSTADT
UNIVERSITY OF APPLIED SCIENCES

fbeit

FACHBEREICH ELEKTROTECHNIK
UND INFORMATIONSTECHNIK

MASTERARBEIT

**Thema: Regelkreisentwurf im Frequenzbereich nach Drei-Bode-Plots Verfahren und
Bode-aided Design**

Bearbeiter/in: Julian Buhl

Referent/in: Prof. em., Dr.-Ing. Serge Zacher

Korreferent/in: Prof. Dr.-Ing Karl Kleinmann

Abgabe am: 12.05.2021

Inhaltsverzeichnis

INHALTSVERZEICHNIS	V
ABBILDUNGSVERZEICHNIS	VII
TABELLENVERZEICHNIS	XI
ABKÜRZUNGSVERZEICHNIS	XII
SYMBOLVERZEICHNIS	XIII
1 EINLEITUNG	1
1.1 Stabilitätskriterien für Regelkreise	1
1.2 Zwei-Bode-Plots Verfahren	2
1.3 Aufgabenstellung	3
1.4 Struktur der Arbeit	4
2 GRUNDLAGEN	5
2.1 Drei-Bode-Plots Verfahren	5
2.1.1 Weiterentwicklung des Zwei-Bode-Plots-Verfahren	5
2.1.2 Drei-Bode-Plots Verfahren Stabilitätskriterium.....	6
2.1.3 Vorteile des Drei-Bode-Plots Verfahren	7
2.1.4 Reglereinstellung nach dem Drei-Bode-Plots Verfahren.....	7
2.2 Bode-aided Design	8
2.2.1 Definition Bode-aided Design.....	8
2.2.2 Konzept des BAD.....	8
2.2.3 Vorgehensweise beim BAD	10
2.2.4 BAD nach einem Punkt.....	12
2.2.5 Grenzen des Systems.....	14
3 FINGERPRINT-APPLIKATION	16
3.1 Bestandsapplikation	16
3.2 Anpassen der Bestandsapplikation auf DBV	16
3.3 Erweitern der Benutzeroberfläche	18
3.4 Bode-aided Design	20
3.4.1 P-Regler.....	22
3.4.2 PI-Regler	24

1 Einleitung

1.1 Stabilitätskriterien für Regelkreise

In der Welt der Regelungstechnik gibt es diverse Stabilitätskriterien im Frequenzbereich um die Stabilität eines Regelkreises zu bewerten. Die Stabilitätskriterien unterscheiden sich in der betrachteten Übertragungsfunktion des Systems, siehe Tabelle 1 [1, S. 233].

Beim Stabilitätskriterium nach Mikhailov (1936) ist der Nenner der Übertragungsfunktion eines Regelkreises, $1+G_0(s)$, das charakteristische Polynom welches behandelt wird.

Das Stabilitätskriterium nach Nyquist (1932) betrachtet die Übertragungsfunktion des offenen Systems $G_0(s)$.

Beim Stabilitätskriterium nach Leonhard (1940) findet eine getrennte Betrachtung der Übertragungsfunktion des Reglers $G_R(s)$ und der Übertragungsfunktion der negativ inversen (reziproken) Strecke $G_{S\text{rez}}(s) = -1/G_S(s)$ statt. [1, S. 233]

Tabelle 1: Übersicht über verschiedene Stabilitätskriterien

Bezeichnung des Kriteriums	Mathematische Grundlage im Bildbereich	Stabilitätsmerkmal	Grafische Charakteristik im Frequenzbereich
<i>Mikhailov</i> , 1936	Charakteristisches Polynom des geschlossenen Systems $1+G_0(s)$	Punkt $(0, j0)$	Ortskurve
<i>Nyquist</i> , 1932	Übertragungsfunktion des offenen Systems $G_0(s)$	Punkt $(-1, j0)$	Ortskurve
		Phase am Schnittpunkt der Amplitude mit 0 dB-Linie	
<i>Leonhard</i> , 1940	Übertragungsfunktion des Reglers $G_R(s)$ und der negativ inversen (reziproken) Strecke $G_{R\text{rez}}(s) = -1/G_S(s)$	Lage und Winkel zwischen Zeiger $\varphi_{S\text{rez}}(\omega)$ und $\varphi_R(\omega)$	Zwei Ortskurven
<i>Zacher</i> , 2018	Übertragungsfunktionen der Strecke $G_S(s)$ und des negativ inversen (reziproken) Reglers $G_{R\text{rez}}(s) = -1/G_R(s)$	Abstand zwischen Phasengang des reziproken Reglers $\varphi_{R\text{rez}}(\omega)$ und des Phasengangs der Strecke $\varphi_S(\omega)$ am Schnittpunkt von Amplituden	Zwei Bode-Plots

Quelle: Zacher, 2020, S. 233

Bei der Betrachtung der Stabilitätskriterien wird ein Wandel der verwendeten Übertragungsfunktion ersichtlich. Beginnend von der Übertragungsfunktion des geschlossenen Regelkreises $G_W(s)$ über den offenen Regelkreis $G_0(s)$ bis hin zu den Bestandteilen der Übertragungsfunktion des Reglers $G_R(s)$ und der Übertragungsfunktion der Strecke $G_S(s)$ wird die betrachtete Übertragungsfunktion immer weiter unterteilt und das Stabilitätskriterium vereinfacht.

Das von Hr. Zacher aufgestellte Stabilitätskriterium basiert auf dem Leonard-Stabilitätskriterium und stellt eine Weiterentwicklung desselbigen dar. Anstelle der Ortskurven des Reglers und der negativ inversen Strecke werden beim Stabilitätskriterium von Hr. Zacher die Bode-Diagramme der Strecke $G_S(s)$ und des negativ inversen (reziproken) Reglers $G_{R_{rez}}(s) = -1/G_R(s)$ betrachtet.

[1, S. 233]

1.2 Zwei-Bode-Plots Verfahren

Beim Zwei-Bode-Plots-Verfahren (ZBV) werden die Bode-Diagramme der Strecke $G_S(j\omega)$ und des negativ inversen Reglers mit der Übertragungsfunktion $G_{R_{rez}}(s) = -1/G_R(s)$ betrachtet, siehe Abbildung 1. [1, S. 234]

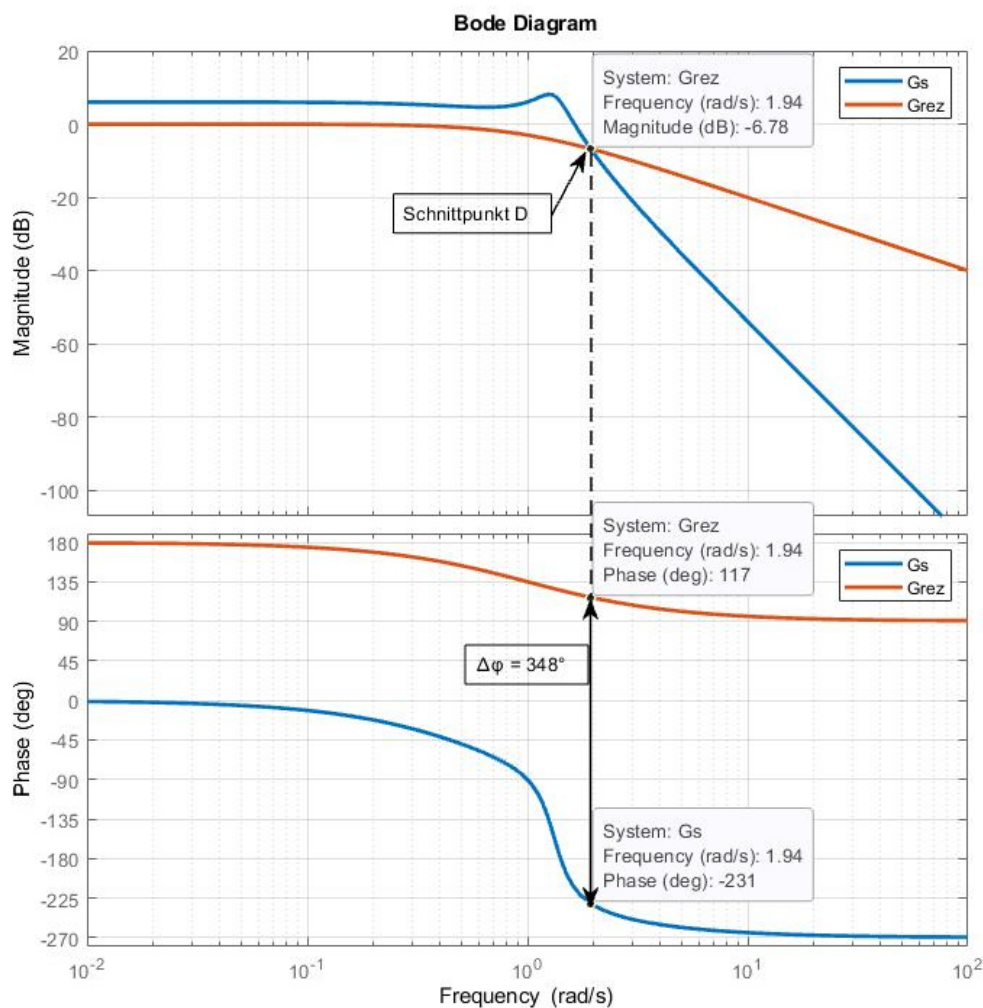


Abbildung 1: Darstellung des Stabilitätskriterium nach ZBV

Quelle: Eigene Darstellung

Das Stabilitätskriterium nach dem ZBV gestaltet sich wie folgt. Damit ein geschlossener Regelkreis stabil wird, muss am Schnittpunkt D der Amplitudengänge der Strecke $|G_S(j\omega)|_{dB}$ und des negativ inversen (reziproken) Reglers $|G_{Rrez}(j\omega)|_{dB}$ die Phasendifferenz $\Delta\varphi$ der Phasengänge der Strecke φ_S und des negativ inversen (reziproken) Reglers φ_{Rrez} kleiner 360° betragen, $\Delta\varphi = \varphi_{Rrez}(\omega D) - \varphi_S(\omega D) < 360^\circ$. Ein geschlossener Regelkreis der diese Anforderung erfüllt, ist stabil, siehe Abbildung 2. [1, S. 234]

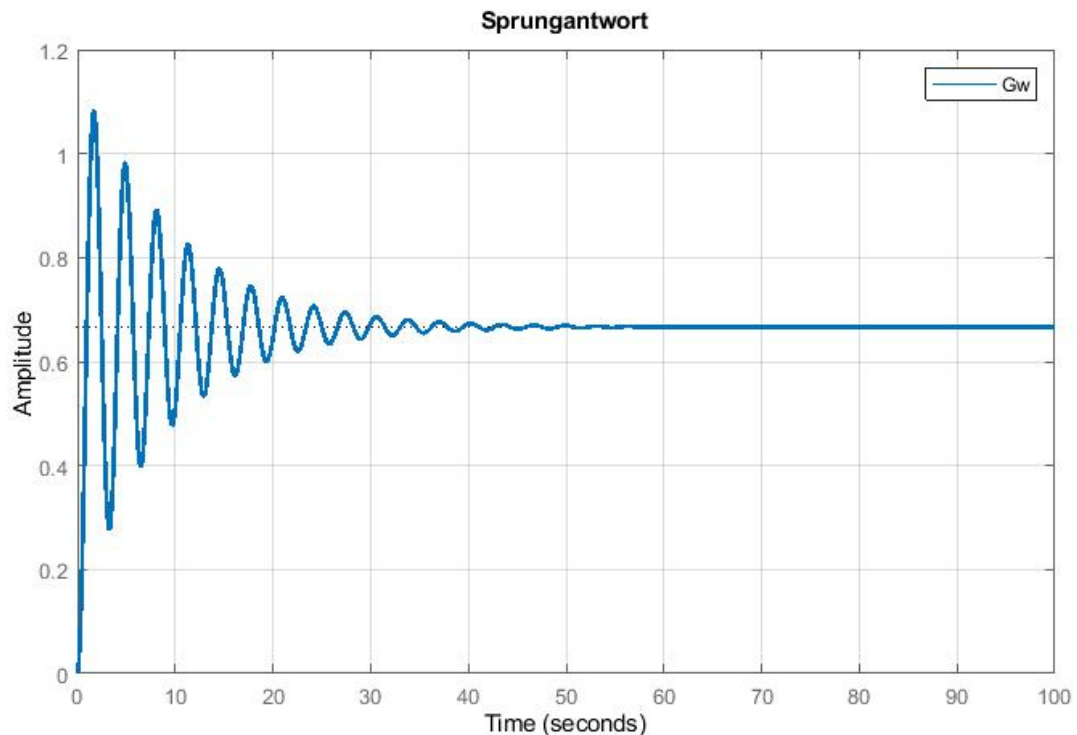


Abbildung 2: Sprungantwort des geschlossenen Regelkreises

Quelle: Eigene Darstellung

Die Vorteile des ZBV bestehen darin, dass dieses Stabilitätskriterium an Matlab® angepasst und leicht zu implementieren ist. Um eine gewünschte Phasenreserve zu erhalten, kann man den Bode-Plot des reziproken Reglers mithilfe der Reglerverstärkung K_{PR} , der Nachhaltezeit T_n , der Vorhaltezeit T_v und der Zeitverzögerungskonstante T_R an die gewünschte Stelle verschieben. [1, S. 235]

1.3 Aufgabenstellung

Ziel dieser Arbeit ist es, die vorhandene Fingerprint-Applikation, welche nach dem ZBV, das durch Prof. Dr. Serge Zacher entwickelt wurde, für das weiterentwickelte Drei-Bode-Plots Verfahren (DBV), ebenfalls von Prof. Dr. Serge Zacher entwickelt, anzupassen und die Applikation um die Funktionalitäten des Bode-aided Design (BAD) zu erweitern.

Das Stabilitätskriterium des ZBV besagt, dass die Phasendifferenz zwischen den Phasengängen der Strecke und des negativ inversen Reglers im Schnittpunkt der Amplitudengänge $\Delta\varphi < 360^\circ$ betragen muss, damit es sich um einen stabilen Regelkreis handelt.

Beim DBV betrachtet man die Amplitudengänge des symmetrischen Reglers und des phasensymmetrischen Reglers mit dem Amplituden- und Phasengang der Strecke, hierbei muss die Phasendifferenz im Schnittpunkt der Amplitudengänge $\Delta\varphi > 0^\circ$ betragen.

Das DBV ermöglicht, wie auch schon das ZBV, mittels BAD die Dimensionierung des Reglers anhand des Bode-Diagramm der Strecke ohne Kenntnisse über die Übertragungsfunktion der Strecke. Mit Hilfe des BAD besteht zusätzlich die Möglichkeit, die Dimensionierung des Reglers nach nur einem Punkt des Bode-Diagramms der Strecke durchzuführen.

Die Applikation soll um diese Funktionalitäten erweitert und anschließend anhand der Modelltafel ET722 als praktisches Beispiel getestet und evaluiert werden.

1.4 Struktur der Arbeit

Kapitel 1 dient als Einleitung der Arbeit. Hier wird der Stand der Technik und die Aufgabenstellung beschrieben. Für den Stand der Technik wird zunächst ein Überblick über einige Stabilitätskriterien für Regelkreise im Frequenzbereich gegeben und auf das Stabilitätskriterium des ZBV näher eingegangen.

In *Kapitel 2* werden die Grundlagen dieser Masterarbeit erläutert. Dabei wird insbesondere auf das DBV und das BAD eingegangen. Des Weiteren wird das Bode-aided Design nach einem Punkt betrachtet.

Kapitel 3 beschreibt die Fingerprint-Applikation und deren Funktionalitäten. In diesem Kapitel werden die notwendigen Schritte zur Weiterentwicklung der bestehenden Fingerprint-Applikation von der Verwendung des ZBV zur Verwendung des DBV und die Implementierung des Reglerentwurfs mittels BAD und dem Reglerentwurf nach einem einzelnen Punkt des Bode-Diagramms der Strecke beschrieben und erläutert.

In *Kapitel 4* werden die Ermittlung des Bode-Diagramm der Modelltafel ET722 und die Einbindung der Ergebnisse in die Fingerprint-Applikation beschrieben. Hierbei wird der grundlegende Aufbau zur Bestimmung des Bode-Diagramms näher beschrieben und das Programm zur Auswertung der Messergebnisse erläutert. Abschließend wird die Einbindung des experimentell bestimmten Bode-Diagramms in die Fingerprint-Applikation dargestellt.

Kapitel 5 wird ein Fazit über die Masterarbeit und die zu lösenden Problemstellungen gezogen. Des Weiteren gibt das Kapitel einen Ausblick über die Weiterentwicklung der Fingerprint-Applikation und deren potentielle Anwendungen.

2 Grundlagen

2.1 Drei-Bode-Plots Verfahren

2.1.1 Weiterentwicklung des Zwei-Bode-Plots-Verfahren

Eine Möglichkeit das ZBV weiter zu entwickeln besteht darin, die benötigte Phasendifferenz von $\Delta\varphi < 360^\circ$ auf $\Delta\varphi < 0^\circ$ zu reduzieren. Diese Vereinfachung entspricht dem Leonhard-Stabilitätskriterium wonach der Phasengang des reziproken Reglers $\varphi_{R\text{rez}}(\omega)$ kleiner als der Phasengang der Strecke $\varphi_S(\omega)$ bei stabilen Systemen sein muss. Durch die Anpassung der Stabilitätsbedingung lässt sich die Phasendifferenz zwischen dem reziproken Regler und der Strecke direkt im Bode-Diagramm erkennen.

Zur Lösung des Problems hat sich Hr. Zacher dazu entschieden, dass beim ZBV verwendete Bode-Diagramm des reziproken Reglers in zwei Frequenzgänge für Amplitude und Phase aufzuteilen. Für die Amplitude wird der Amplitudengang des symmetrischen Reglers $|G_{R\text{symm}}(j\omega)|_{\text{dB}}$ mit der Übertragungsfunktion $|G_{R\text{symm}}(s)|_{\text{dB}} = -|G_R(s)|_{\text{dB}}$ verwendet. Dieser hat eine Symmetrieachse des Amplitudengangs bei $L_G = 0 \text{ dB}$. Für den Phasenverlauf wird der Phasengang des phasensymmetrischen Reglers $\varphi_{R\text{phs}}(\omega)$ verwendet, mit der Übertragungsfunktion $G_{R\text{phs}}(s) = 1 / (s^2 * G_R(s))$. Dieser hat eine Symmetrieachse des Phasengangs bei $L_\varphi = -90^\circ$, siehe Abbildung 3.

Zusammen mit dem Bode-Diagramm, Amplituden- und Phasengang, der Strecke $G_S(j\omega)$ ergeben diese drei Bode-Plots das von Hr. Zacher aufgestellte DBV. [1, S. 235]

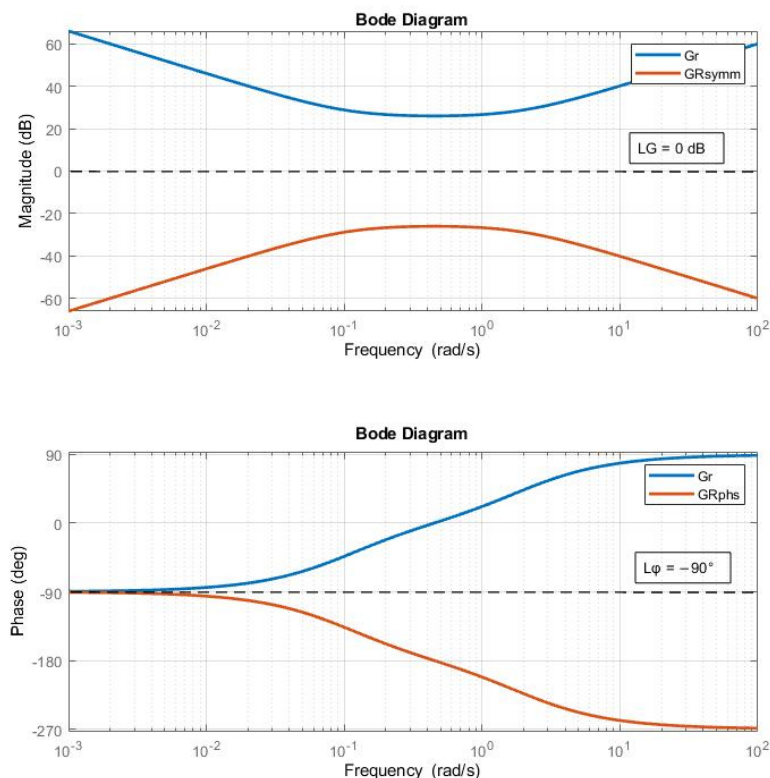


Abbildung 3: Idealer und gespiegelter PID-Regler mit den Symmetrieachsen L_G und L_φ

Quelle: Eigene Darstellung

2.1.2 Drei-Bode-Plots Verfahren Stabilitätskriterium

Nach dem DBV wird ein geschlossener Regelkreis dann stabil, wenn am Schnittpunkt D der Amplitudengänge des symmetrischen Reglers $|G_{Rsymm}(j\omega)|_{dB}$ und der Strecke $|G_S(j\omega)|_{dB}$ der Phasengang der Strecke $\varphi_S(\omega)$ oberhalb des Phasengang des phasen-symmetrischen Reglers $\varphi_{Rphs}(\omega)$ liegt, siehe Abbildung 4. [1, S. 252]

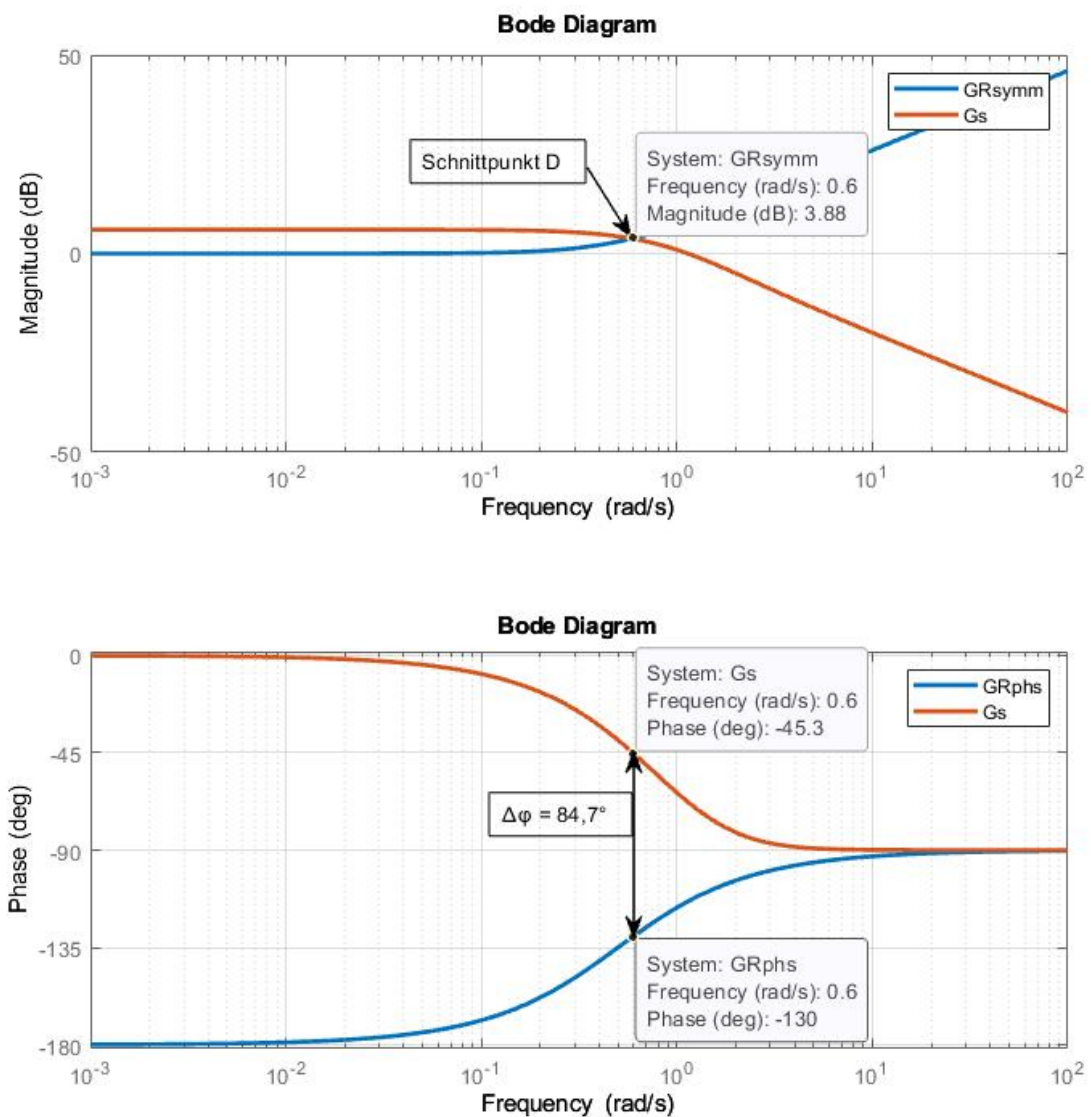


Abbildung 4: Darstellung des Stabilitätskriterium nach DBV

Quelle: Eigene Darstellung

2.1.3 Vorteile des Drei-Bode-Plots Verfahren

Die Vorteile des DBV bestehen darin, dass wie beim ZBV keinerlei Kenntnisse der Übertragungsfunktion, Polstellen, Anzahl der Schnittstellen und I-Gliedern der Strecke vorhanden sein müssen. Auch Kenntnisse über die Übertragungsfunktion und Bode-Plot des offenen Regelkreises sind nicht vonnöten. Lediglich das Bode-Diagramm der Strecke muss vorliegen, um die Stabilität des Systems feststellen zu können. Beim DBV kann man die Stabilitätsbedingung direkt im Bode-Diagramm erkennen, da kein Bestimmen der Phasenabstände notwendig ist. Zur Bestimmung der Stabilität reicht es aus zu überprüfen, ob im Schnittpunkt der Amplitudengänge des symmetrischen Reglers und der Strecke, der Phasengang des phasen-symmetrischen Reglers unterhalb des Phasengangs der Strecke liegt.

Des Weiteren lässt sich bei instabilen Strecken nur anhand des Phasengangs der Strecke erkennen, ob ein Standardregler mit den Kennwerten K_{PR} , T_n , T_v und T_R ohne zusätzliche Korrekturglieder ausreicht um die Strecke stabil regeln zu können. [1, S. 255]

2.1.4 Reglereinstellung nach dem Drei-Bode-Plots Verfahren

Mit Hilfe des DBV kann man, wie auch schon mit dem ZBV, nicht nur die Stabilität eines Regelkreises feststellen, es ermöglicht auch den Entwurf eines Reglers. Dabei müssen wie beim Stabilitätskriterium keinerlei Kenntnisse über die Übertragungsfunktion, Streckenparameter oder Pol-/Nullstellen vorhanden sein. Dazu wird der Amplitudengang des symmetrischen Reglers und der Phasengang des phasen-symmetrischen Reglers bestimmt und zusammen mit dem Bode-Diagramm der Strecke in ein Diagramm eingetragen. Anschließend werden die Bereiche identifiziert, bei denen der Phasengang der Strecke unterhalb des Phasengangs des phasen-symmetrischen Reglers liegt, da nur in diesen Bereichen das Stabilitätskriterium nach DBV erfüllt werden kann. Anschließend wird in Abhängigkeit des gewählten Reglers und der damit zur Verfügung stehenden Kennwerte der Phasengang horizontal in die identifizierten Bereiche verschoben und anschließend wird der Amplitudengang horizontal und vertikal verschoben bis das Stabilitätskriterium nach DBV erfüllt ist und der geschlossene Regelkreis stabil wird.

Die Vorgehensweise zum Entwurf eines Reglers wird im Kapitel 2.2 Bode-aided Design näher erläutert. [1, S. 255]

2.2 Bode-aided Design

2.2.1 Definition Bode-aided Design

Unter dem BAD wird im Sinne dieser Arbeit der Entwurf eines Standardreglers $G_R(s)$ im Frequenzbereich anhand des gegebenen Bode-Diagramms der Strecke $G_S(s)$ mit Hilfe des DBV verstanden. Der Entwurf eines Standardreglers ist auch mit dem ZBV möglich, wird in dieser Arbeit aber nicht behandelt.

Für den Entwurf des BAD wird ein Bode-Diagramm der zu regelnden Strecke benötigt. Bei der Bestimmung des Bode-Diagramms der Strecke werden zwei Fälle unterschieden. Im Falle einer stabilen Strecke kann das Bode-Diagramm experimentell bestimmt und für die Anwendung des Reglerentwurfs mittels BAD benutzt werden.

Für den Fall einer instabilen Strecke ist ein experimentelles Bestimmen des Bode-Diagramms nicht möglich, da ein harmonisches Eingangssignal zu einem Aufschwingen oder starkem Ansteigen des Ausgangssignal führt. Dadurch ist eine Messung von Amplitude und Phasenverschiebung nicht möglich. Zur Bestimmung des Bode-Diagramms einer instabilen Strecke wird ein bekanntes Identifikationsverfahren angewendet, um die Übertragungsfunktion der Strecke $G_S(s)$ zu bestimmen. Mit der Übertragungsfunktion der Strecke wird das Bode-Diagramm erstellt und dann mittels BAD der Regler entworfen. [1, S. 270]

2.2.2 Konzept des BAD

Das Konzept des BAD unterscheidet sich von den bisher bekannten Entwurfsverfahren von Regelkreisen im Frequenzbereich. Anstatt wie bei den bekannten Entwurfsverfahren das Bode-Diagramm des gespiegelten Reglers mit Hilfe der Kennwerte des Reglers zu verschieben, wird beim BAD das Bode-Diagramm des gespiegelten Reglers zusammen mit dem Bode-Diagramm der Strecke direkt in den Bode-Plot eingetragen. Als Grundlage hierfür dient der gespiegelte PID-Regler, der sich aus einem P-Anteil, einem I-Anteil und einem D-Anteil zusammensetzt, siehe Abbildung 5.

Diese Anteile werden durch die drei Kennwerte von Reglern, die Reglerverstärkung K_{PR} und die Zeitkonstanten Nachhaltezeit T_n und Vorhaltezeit T_v , beeinflusst. Der Amplitudengang des gespiegelten Reglers setzt sich aus drei Asymptoten, die jeweils einem Kennwert zugeordnet sind. Der gespiegelte P-Anteil des Reglers erzeugt eine Asymptote A_P mit der Steigung $k_P = 0$ dB/dek, des Weiteren bestimmt die Reglerverstärkung den Abstand des P-Anteils von der 0 dB-Linie mit $\Delta p = 20 \cdot \log(1/K_{pr})$. Die gespiegelten I- und D-Anteile erzeugen gegenläufige Amplituden A_n und A_v mit $k_I = +20$ dB/dek und $k_D = -20$ dB/dek.

Die Knickpunkte der Asymptoten A_n und A_v sind die Kehrwerte der jeweiligen Zeitkonstanten mit $\omega_n = 1/T_n$ und $\omega_v = 1/T_v$ wobei die Differenz der Asymptoten zum realen Verlauf des Amplitudengangs 3 dB im Knickpunkt beträgt. [1, S. 277]

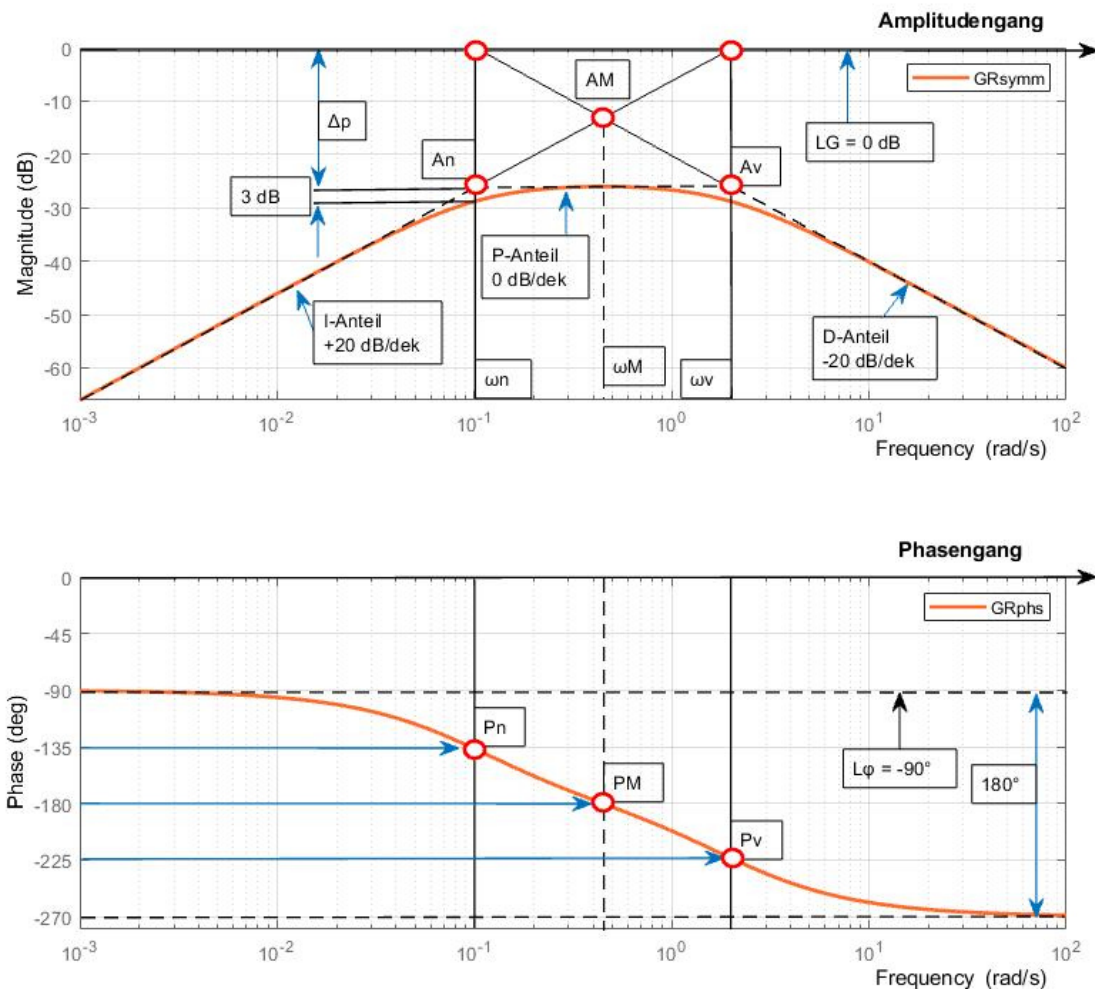


Abbildung 5: Darstellung des gespiegelten PID-Reglers als Grundlage des BAD
 Quelle: In Anlehnung an [1, S. 273]

Der Phasengang des gespiegelten PID-Reglers bewegt sich im Bereich von $\varphi = -90^\circ$ und $\varphi = -270^\circ$ und hat somit eine Breite von 180° . Innerhalb des Phasengangs gibt es drei charakteristische Punkte P_n , P_v und P_M , die durch die Knickfrequenzen A_n und A_v der Asymptoten des Amplitudengangs und deren Schnittpunkt A_M festgelegt sind. Der Punkt P_n befindet sich bei der Knickfrequenz $\omega_n = 1/T_n$ und weist eine Phase von $\varphi_n = -135^\circ$ auf, der Punkt P_v befindet sich bei der Knickfrequenz $\omega_v = 1/T_v$ und weist eine Phase von $\varphi_v = -225^\circ$ auf und der Punkt P_M im Schnittpunkt der Asymptoten besitzt die Phase $\varphi_M = -180^\circ$. [1, S. 273-275]

2.2.3 Vorgehensweise beim BAD

Für den Entwurf eines Reglers wird zunächst ein Gütekriterium des Reglers festgelegt, im Rahme dieser Arbeit wurde eine vorgegebene Phasenreserve φ_{Rd} als Gütekriterium gewählt. Anschließend wird der Phasengang des gespiegelten Reglers in das Bode-Diagramm der Strecke eingetragen. Der Phasengang des gespiegelten Reglers ist so einzutragen, dass dieser unterhalb des Phasengangs der Strecke liegt, um das Stabilitätskriterium nach DBV zu erfüllen.

Angefangen wird damit, dass die Punkte P_n mit einer Phase von $\varphi_n = -135^\circ$ und P_v mit einer Phase von $\varphi_n = -225^\circ$ manuell eingetragen werden und anschließend aus den Knickfrequenzen die Zeitkonstanten berechnet werden, wobei $T_n = 1/\omega_n$ und $T_v = 1/\omega_v$ gilt, siehe Abbildung 6.

[1, S. 277-279]

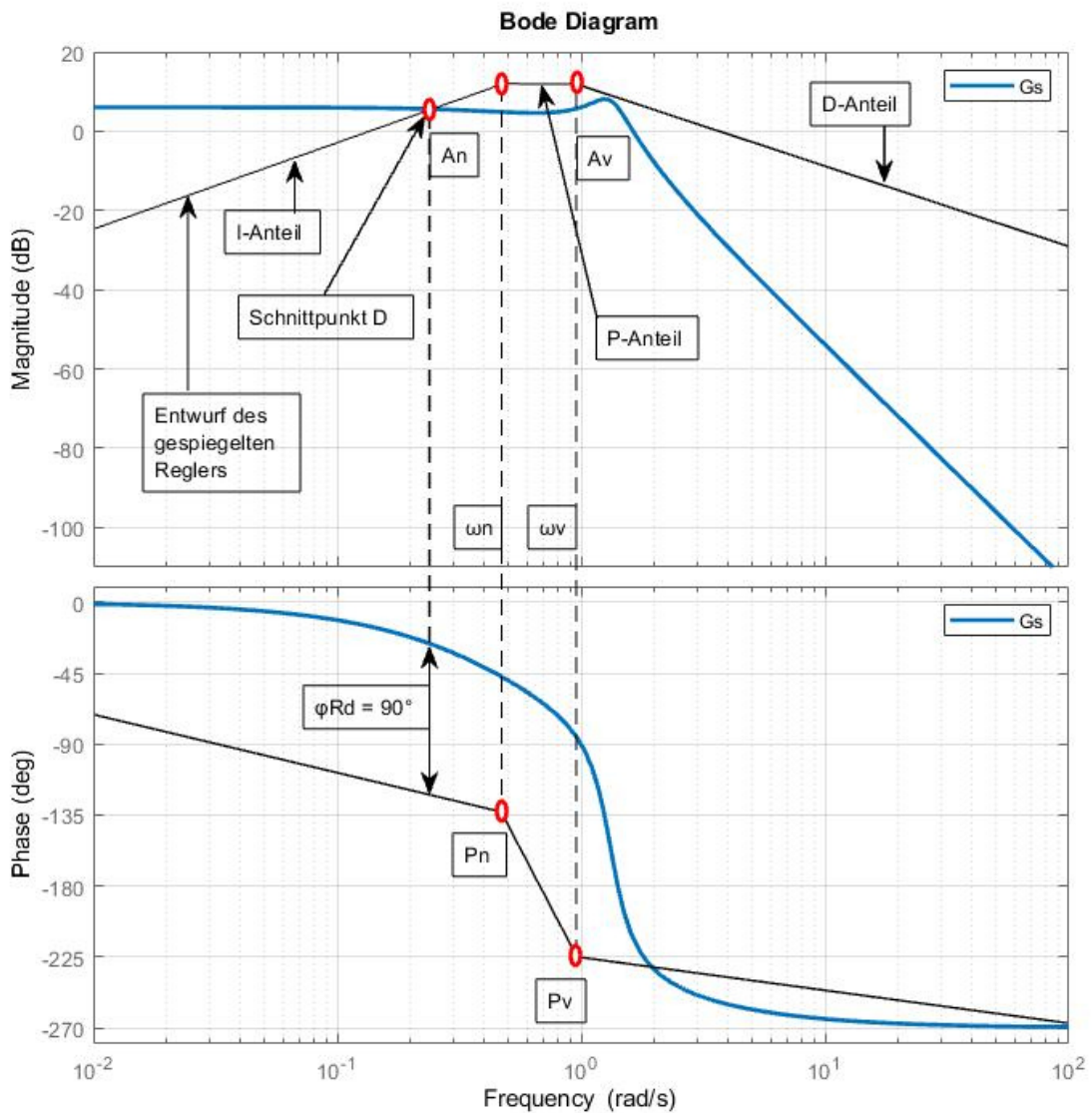


Abbildung 6: Entwurf des gespiegelten PID-Reglers mittels BAD

Quelle: Eigene Darstellung

Im nächsten Schritt wird ein Punkt D gewählt, bei dem sich der Amplitudengang der Strecke und der Amplitudengang des Reglers schneiden und dessen Phasenreserve φ_{Rd} dem gewünschten Gütekriterium, in unserem Beispiel $\varphi_{Rd} = 90^\circ$, beträgt, siehe Abbildung 7. Die Asymptote des I-Anteils mit einer Steigung von +20 dB/dek wird durch den Punkt D gezogen und endet bei der Knickfrequenz ω_n . [1, S. 277-279]

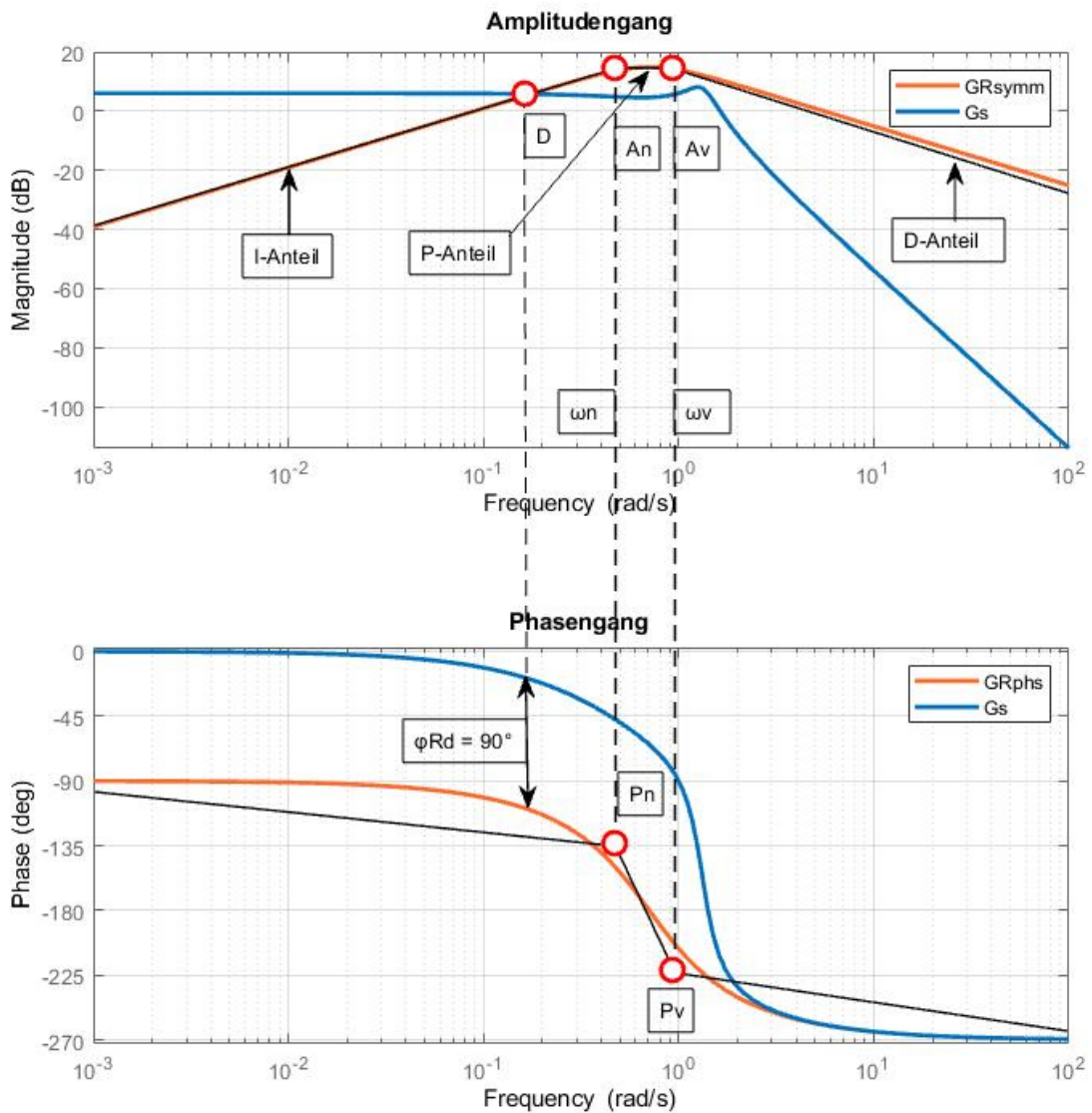


Abbildung 7: Bode-Diagramm des gespiegelten PID-Reglers und der Strecke

Quelle: Eigene Darstellung

Beginnend bei der Knickfrequenz ω_v wird anschließend die Asymptote des D-Anteils mit einer Steigung von -20 dB/Dek in das Bode-Diagramm eingetragen. Die horizontale Verbindung der beiden Asymptoten ist der P-Anteil des Reglers, aus dessen Amplitudendifferenz Δp zur 0-dB-Linie lässt sich die Reglerverstärkung mit $K_{pr} = 10^{(-\Delta p/20)}$ berechnen. [1, S. 277-279]

Der damit erzeugte Regler führt zu einem stabilen Regelkreis, siehe Abbildung 8.

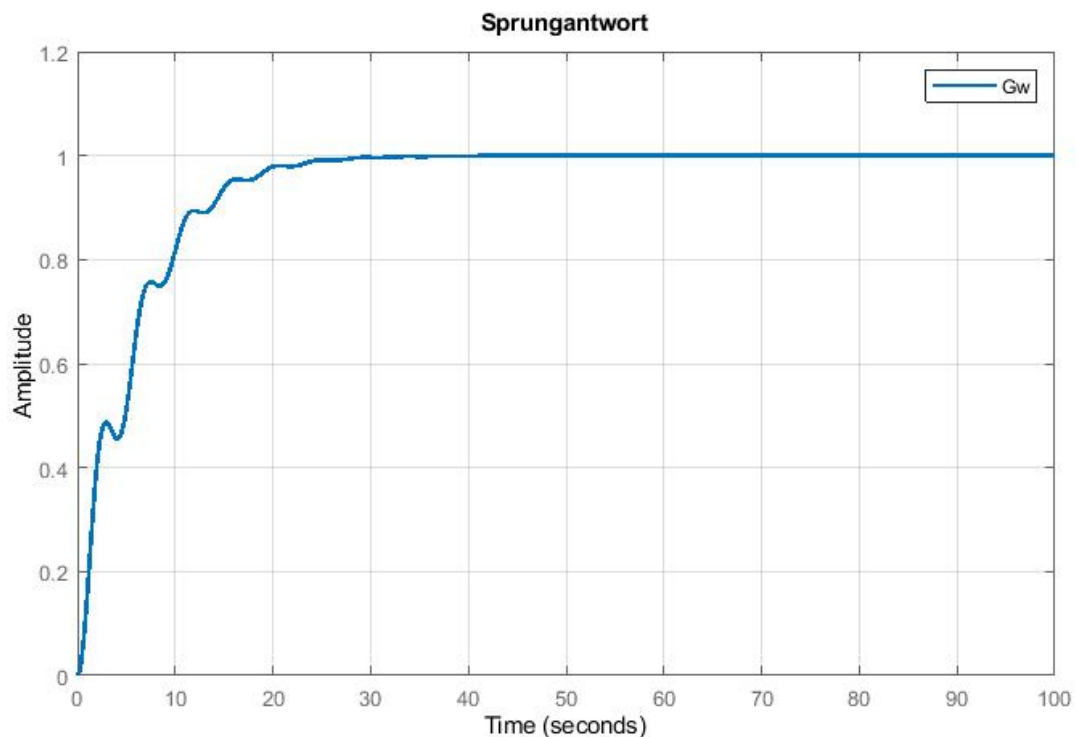


Abbildung 8: Sprungantwort des geschlossenen Regelkreises mit dem entworfenen PID-Regler
Quelle: Eigene Darstellung

2.2.4 BAD nach einem Punkt

Das BAD ermöglicht nicht nur den Entwurf eines Standardreglers nach dem gegebenen Bode-Diagramm der Strecke, es ermöglicht auch den Entwurf eines Standardreglers sofern nur ein Punkt des Bode-Diagramms der Strecke gegeben ist. Durch die Kenntnisse eines Punktes der Strecke lassen sich keine Aussagen über den gesamten Verlauf des Bode-Diagramms der Strecke treffen, dennoch werden neue Wege für den Reglerentwurf durch das Verfahren ermöglicht. Insbesondere bei Strecken mit monoton dynamischem Verlauf, beziehungsweise bei Strecken ohne steile Änderungen im kleinen Frequenzbereich, ist das Verfahren für den Reglerentwurf geeignet.

[1, S. 291]

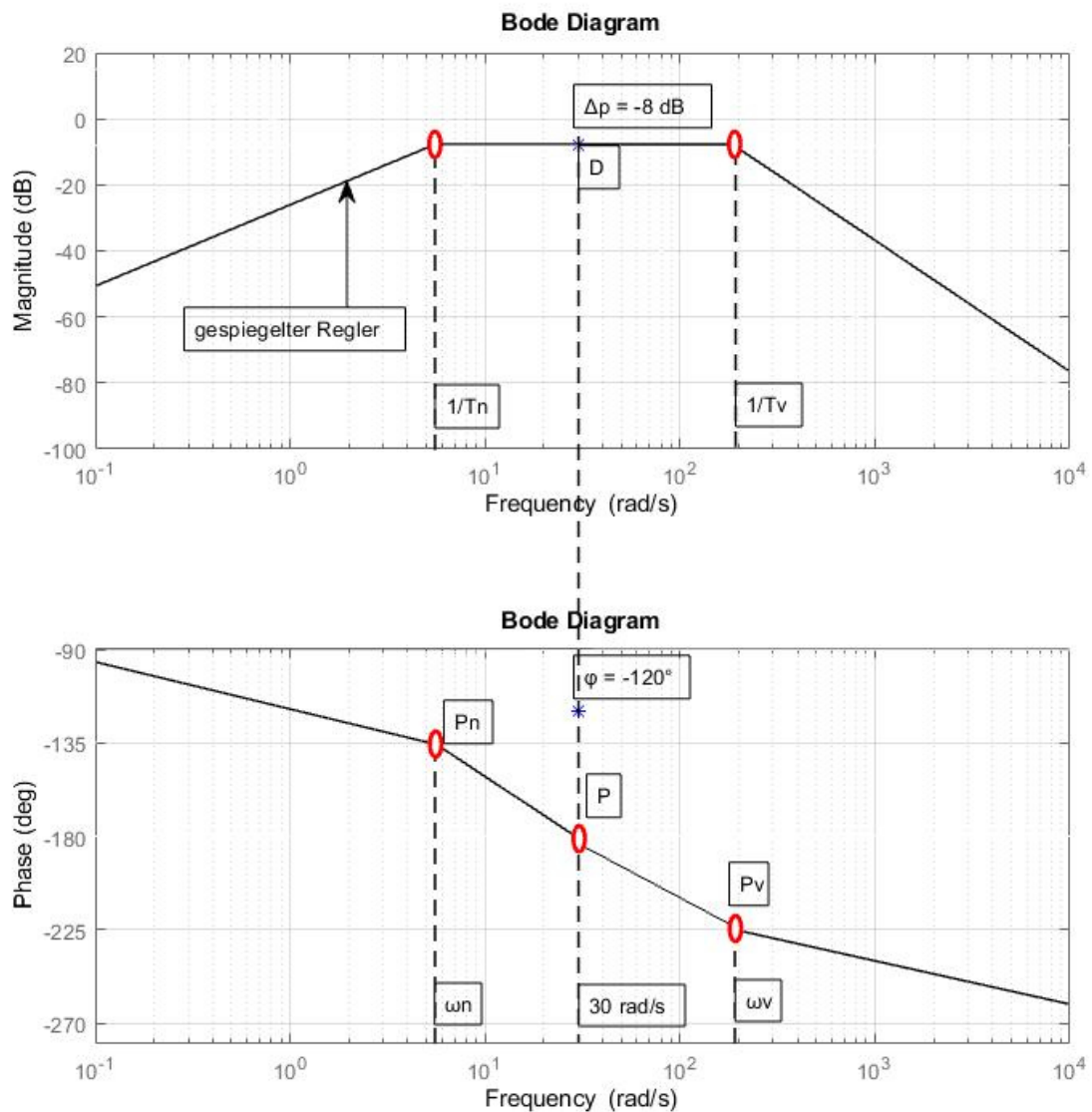


Abbildung 9: Entwurf eines gespiegelten PID-Reglers mittels BAD nach einem Punkt

Quelle: Eigene Darstellung

Die Vorgehensweise beim BAD nach einem Punkt beruht auf dem BAD-Prinzip, jedoch ist der zu betrachtende Punkt der Strecke nicht frei wählbar, sondern durch den einen Punkt vorgegeben, siehe Abbildung 9. Dabei werden die Knickfrequenzen ω_n und ω_v so gewählt, dass der Abstand der Punkte P_n und P_v zum Phasengang des gespiegelten Reglers im vorgegebenen Punkt P möglichst identisch ist. Aus den gewählten Knickfrequenzen werden wieder die Zeitkonstanten Nachhaltezeit T_n und Vorhaltezeit T_v berechnet. Die Reglerverstärkung K_{pr} wird so gewählt, dass der Amplitudengang des gespiegelten Reglers durch den vorgegebenen Punkt P verläuft, siehe Abbildung 10. Dabei entspricht Δp der Amplitudendifferenz des Punktes P zur 0 dB-Linie. [1, S. 293]

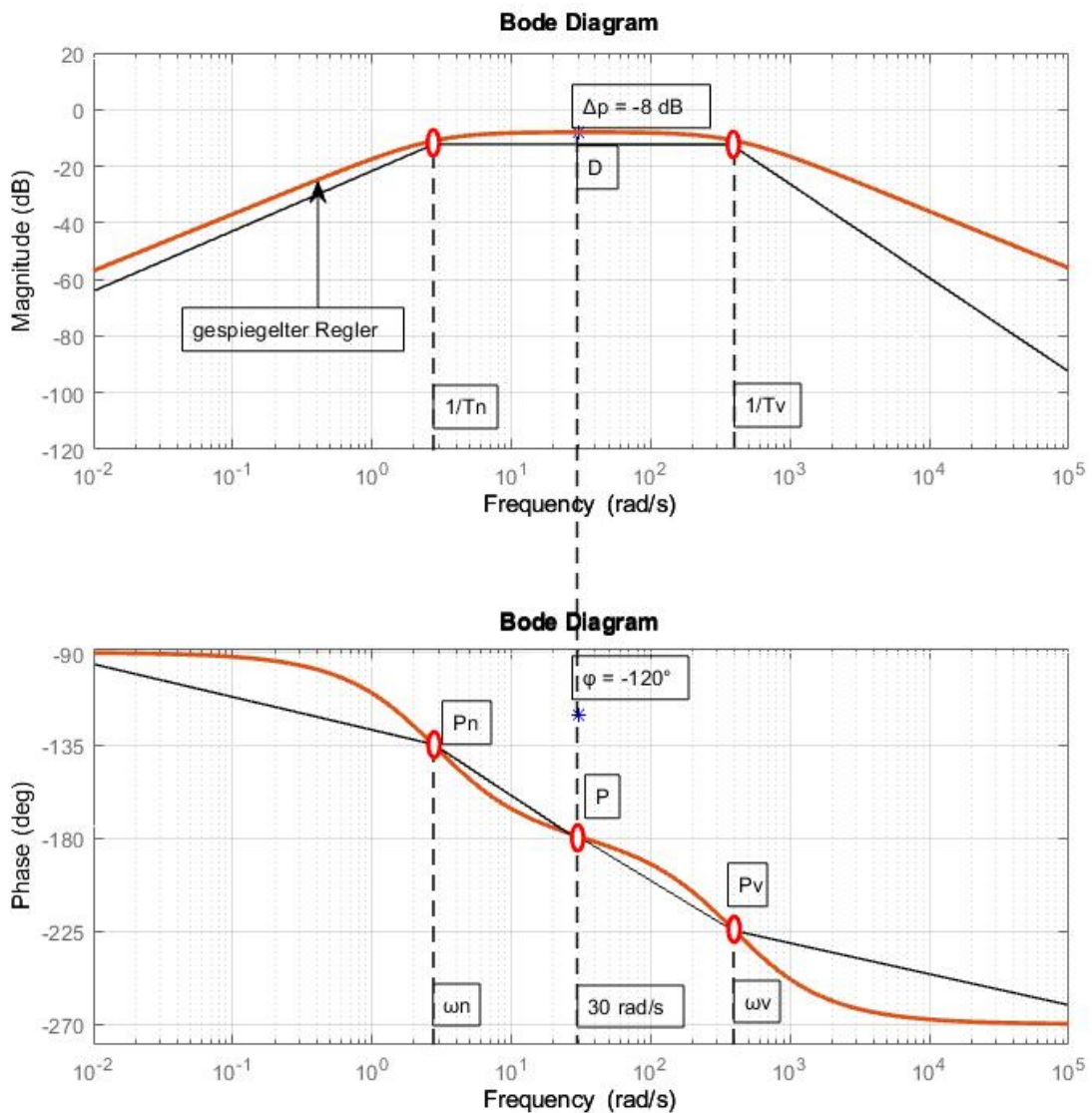


Abbildung 10: Bode-Diagramm des entworfenen gespiegelten PID-Reglers mittels BAD nach einem Punkt
 Quelle: Eigene Darstellung

2.2.5 Grenzen des Systems

Die Möglichkeiten einen Regler mittels BAD zu Entwerfen sind allerdings begrenzt. Dabei wird zwischen zwei Fällen unterschieden, bei der ein Reglerentwurf mittels BAD zu keiner stabilen Lösung führt.

Im ersten Fall verläuft der Phasengang der Strecke $\varphi_S(\omega)$ im gesamten Verlauf des Bode-Diagramms unterhalb des Phasengangs des phasen-symmetrischen Reglers $\varphi_{Rphs}(\omega)$. Durch die Kennwerte des Standardreglers kann der Phasengang des phasen-symmetrischen Reglers $\varphi_{Rphs}(\omega)$ nur horizontal verschoben werden, wodurch das Stabilitätskriterium nicht erfüllt werden kann.

Beim zweiten Fall befindet sich ein Teil des Phasengangs der Strecke $\varphi_S(\omega)$ oberhalb des Phasengangs des Reglers $\varphi_{R\text{phs}}(\omega)$, wodurch ein Stabilitätsbereich vorhanden ist. Verlaufen die Amplitudengänge des symmetrischen Reglers $|G_{R\text{symm}}(j\omega)|_{\text{dB}}$ und der Strecke $|G_S(j\omega)|_{\text{dB}}$ im Bereich des Schnittpunktes der Amplitudengänge jedoch nahezu parallel zueinander, ist nur mit einem präzisen Reglerentwurf eine stabile Regelung möglich, siehe

Dem Reglerentwurf mittels BAD nach einem Punkt sind, wie schon beim BAD, Grenzen gesetzt. Befindet sich der zu betrachtende Punkt der Strecke unterhalb des Phasengangs des gewählten gespiegelten Reglers, beispielsweise bei der Verwendung eines gespiegelten PID-Reglers unterhalb von $\varphi = -270^\circ$, ist ein Erfüllen der Stabilitätsbedingung nicht möglich und der Regelkreis wird instabil. [1, S. 295-296]

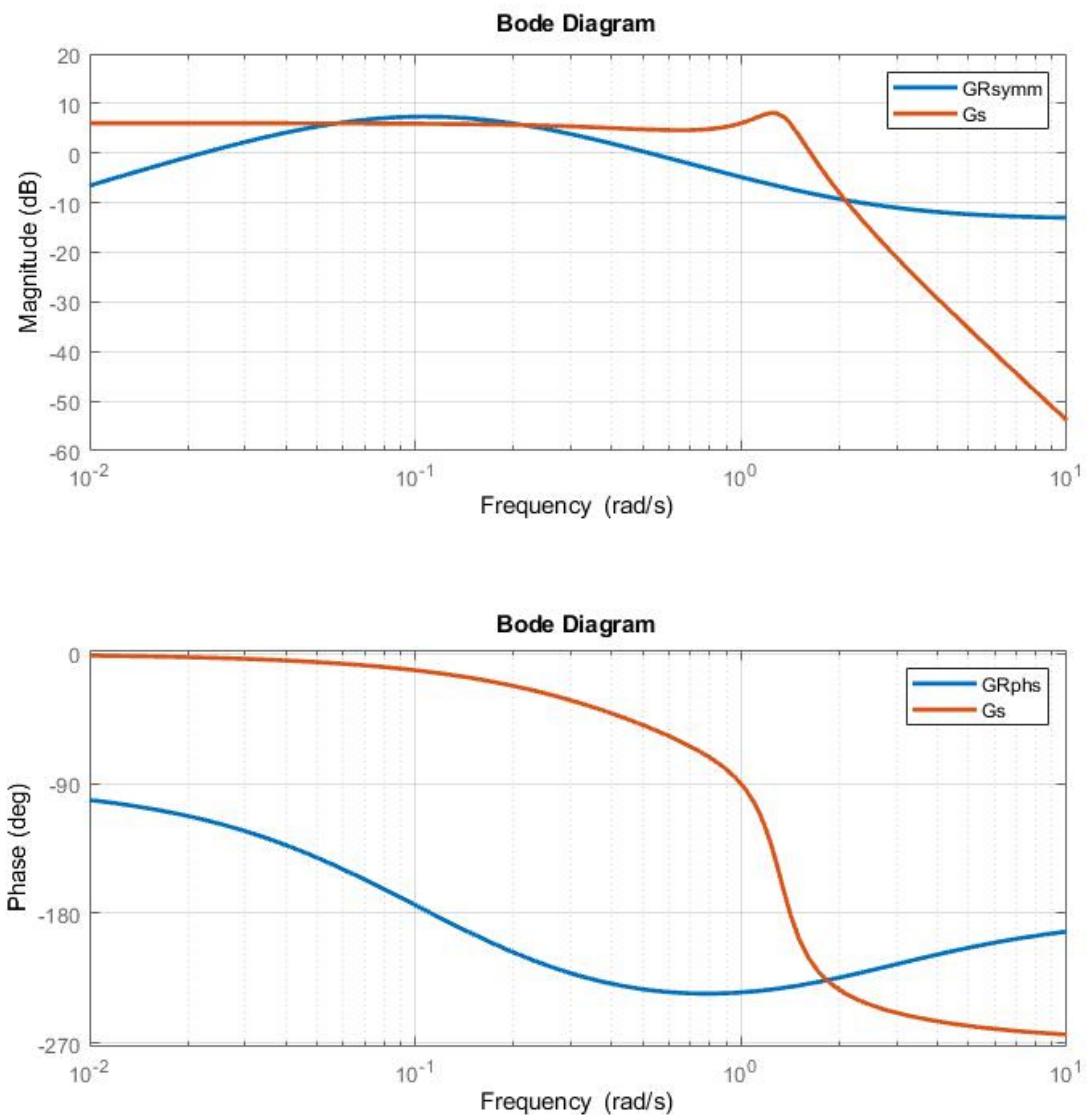


Abbildung 11: Nahezu paralleler Verlauf der Amplitudengänge im Schnittpunktbereich

Quelle: Eigene Darstellung

3.5 BAD nach einem Punkt

Für die Implementierung des BAD nach einem Punkt musste die Möglichkeit geschaffen werden, einen einzelnen Punkt des Bode-Diagramms der Strecke festzulegen, der für den Entwurf verwendet werden soll. Dazu musste die Benutzeroberfläche der Fingerprint-Applikation um Schaltflächen ergänzt und die Funktionalitäten der ergänzten Schaltflächen mussten in der Applikation programmiert und implementiert werden.

3.5.1 Punkte eintragen

Als erste Funktion für die Anwendung des BAD nach einem Punkt wurde die Möglichkeit zum Eintragen eines Punktes in den Amplituden- und Phasengang der gegebenen Strecke geschaffen. Das gesamte Bode-Diagramm der Strecke $G_S(j\omega)$ wird in der Applikation dargestellt aber lediglich der eingetragene Punkt für den Reglerentwurf verwendet. Beim Start der Funktion „Punkte eintragen“ durch Drücken des Buttons „Punkte eintragen“ werden die Bode-Diagramme und die Sprungantwort bereinigt und anschließend die Strecke und der Regler erneut berechnet, um vorgenommene Änderung durch den Benutzer, wie z.B. Änderungen der Strecke, zu berücksichtigen, siehe Abbildung 37.

```
% Button pushed function: PunkteeintragenButton
function PunkteeintragenButtonPushed(app, event)

    % Simulation neu plotten, wenn der Button gedrückt wird
    cla(app.figureBodeMagnitude);
    cla(app.figureBodePhase);
    cla(app.figureStepResponse);

    plot_process(app);
    plot_controller(app);
    legend(app.figureBodeMagnitude, "Gs", "Grsymm");
    legend(app.figureBodePhase, "Gs", "Grphs");
    app.Schnittpunkt.Value = app.wout(app.intersection);
```

Abbildung 37: Quellcode zum Starten der Funktion "Punkte eintragen"

Quelle: Eigene Darstellung

Dazu kann vom Benutzer eine Frequenz ω_{BAD} in das entsprechende Feld eingetragen werden, falls vom Benutzer kein Wert eingetragen wurde, wird die Frequenz auf einen Initialwert ω_{Init} mit $\omega_{\text{Init}} = 1 \text{ rad/s}$ gesetzt, siehe Abbildung 38.

```

%setzen der benötigten Variablen, wenn Frequenz gleich 0
if app.FrequenzinradseingebenEditField.Value == 0
    app.FrequenzinradseingebenEditField.Value = app.init_freq;
end

%Bestimmen der zugehörigen Frequenz nach Vorgabe
app.frequenz_bad = app.FrequenzinradseingebenEditField.Value;

```

Abbildung 38: Quellcode zum Setzen der gewünschten Frequenz

Quelle: Eigene Darstellung

Anschließend wird der Frequenzvektor w_{out} der Bode-Diagramme durchsucht und die Frequenz bestimmt, welcher der vorgegebenen Frequenz ω_{BAD} am nächsten ist, siehe Abbildung 39.

```

%Bestimmen der zugehörigen Frequenz nach Vorgabe
app.temp = abs(app.wout - app.frequenz_bad);
app.intersection_bad = find(app.temp == min(app.temp));

```

Abbildung 39: Quellcode zum Bestimmen der Frequenz

Quelle: Eigene Darstellung

Mit der bestimmten Frequenz werden die Punkte als Rechtecke in den Amplituden- und Phasengang der Strecke eingetragen, siehe Abbildung 40.

```

%Eintragen der Punkte in die Bode-Plots
rectangle(app.figureBodeMagnitude,'Position', ...
    [(app.wout(app.intersection_bad)-0.05*app.wout(app.intersection_bad)) , ...
    (app.mag_process(app.intersection_bad)-2), (app.wout(app.intersection_bad)*0.1), 4], ...
    'FaceColor',[1 0 0], 'EdgeColor',[1 0 0]);
rectangle(app.figureBodePhase,'Position', ...
    [(app.wout(app.intersection_bad)-0.05*app.wout(app.intersection_bad)) , ...
    (app.phase_process(app.intersection_bad)-1), (app.wout(app.intersection_bad)*0.1), 2], ...
    'FaceColor',[1 0 0], 'EdgeColor',[1 0 0]);

end

```

Abbildung 40: Quellcode zum Eintragen der Punkte in das Bode-Diagramm

Quelle: Eigene Darstellung

Die bestimmten Punkte werden in der Applikation dargestellt und der Benutzer kann die Lage der Punkte überprüfen und bei Bedarf eine neue Frequenz zum Eintragen der Punkte in die Applikation eintragen, siehe Abbildung 41.

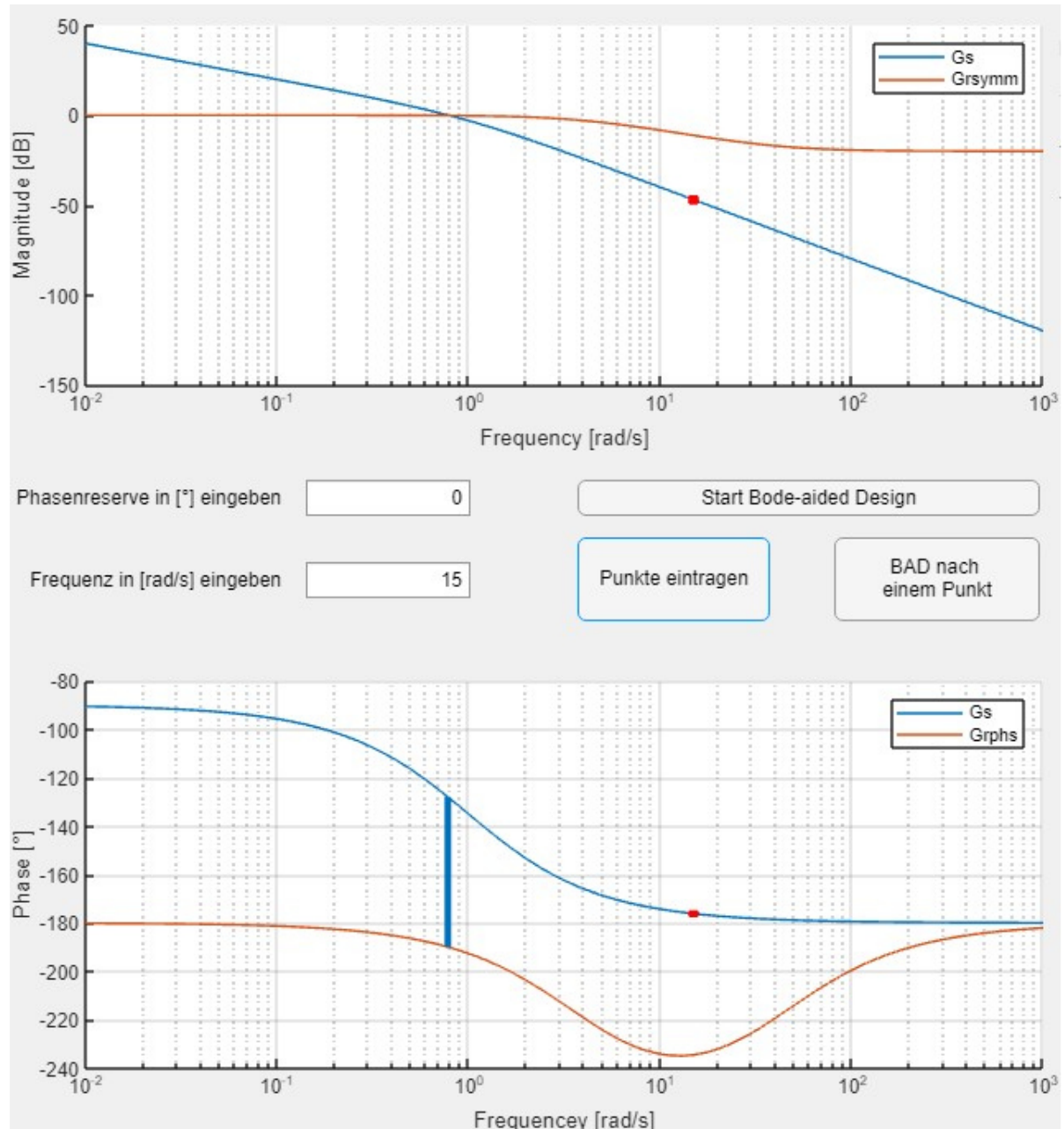


Abbildung 41: Eintragen der bestimmten Punkte in das Bode-Diagramm der „Fingerprint Applikation“

Quelle: Eigene Darstellung

3.5.2 BAD nach einem Punkt Starten

Für die zweite Funktion zur Anwendung des BAD nach einem Punkt musste eine Möglichkeit zum Starten des Reglerentwurfs geschaffen und implementiert werden. Durch Drücken des Buttons „BAD nach einem Punkt“ wird der Reglerentwurf nach einem Punkt des Bode-Diagramms der Strecke $G_s(j\omega)$ gestartet. Zu Beginn der Funktion wird die Funktion „Punkte eintragen“ aufgerufen um die Bode-Diagramme und die Sprungantwort zu bereinigen und anschließend die Strecke und der Regler erneut zu berechnen, um vorgenommene Änderung durch den Benutzer, wie z.B. Änderungen der Strecke, zu berücksichtigen. Zusätzlich wird die vom Benutzer eingetragene Frequenz für den Reglerentwurf bestimmt. Anschließend wird über eine Switch-Case-Anweisung der ausgewählte Reglertyp bestimmt und der entsprechende Quellcode ausgeführt, siehe Abbildung 42.

```
% Button pushed function: BADnacheinemPunktButton
function BADnacheinemPunktButtonPushed(app, event)

    %Aufrufen der "Punkte eintragen" Funktion zum Bestimmen der Frequenz
    PunkteeintragenButtonPushed(app);

    switch app.ReglerauswahlDropDown.Value
        case {"P-Regler" }
```

Abbildung 42: Quellcode zum Start der Funktion „BAD nach einem Punkt“

Quelle: Eigene Darstellung

Nach Abschluss des Reglerentwurfs wird die Funktion „Punkte eintragen“ erneut aufgerufen, um die Bode-Diagramme und die Sprungantwort zu bereinigen. Anschließend werden die Strecke und der entworfene Regler zur Darstellung des Ergebnisses des Reglerentwurfs berechnet und die zuvor bestimmten Punkte eingetragen, siehe Abbildung 43.

```
%Eintragen der Punkte in die Bode-Plots und Berechnung von Regler und Strecke
PunkteeintragenButtonPushed(app);
```

Abbildung 43: Erneutes Aufrufen der Funktion "Punkte eintragen"

Quelle: Eigene Darstellung

Der Amplitudengang des symmetrischen Reglers $|G_{R_{\text{symm}}(s)}|_{\text{dB}}$ verläuft durch den bestimmten Punkt, siehe Abbildung 44.

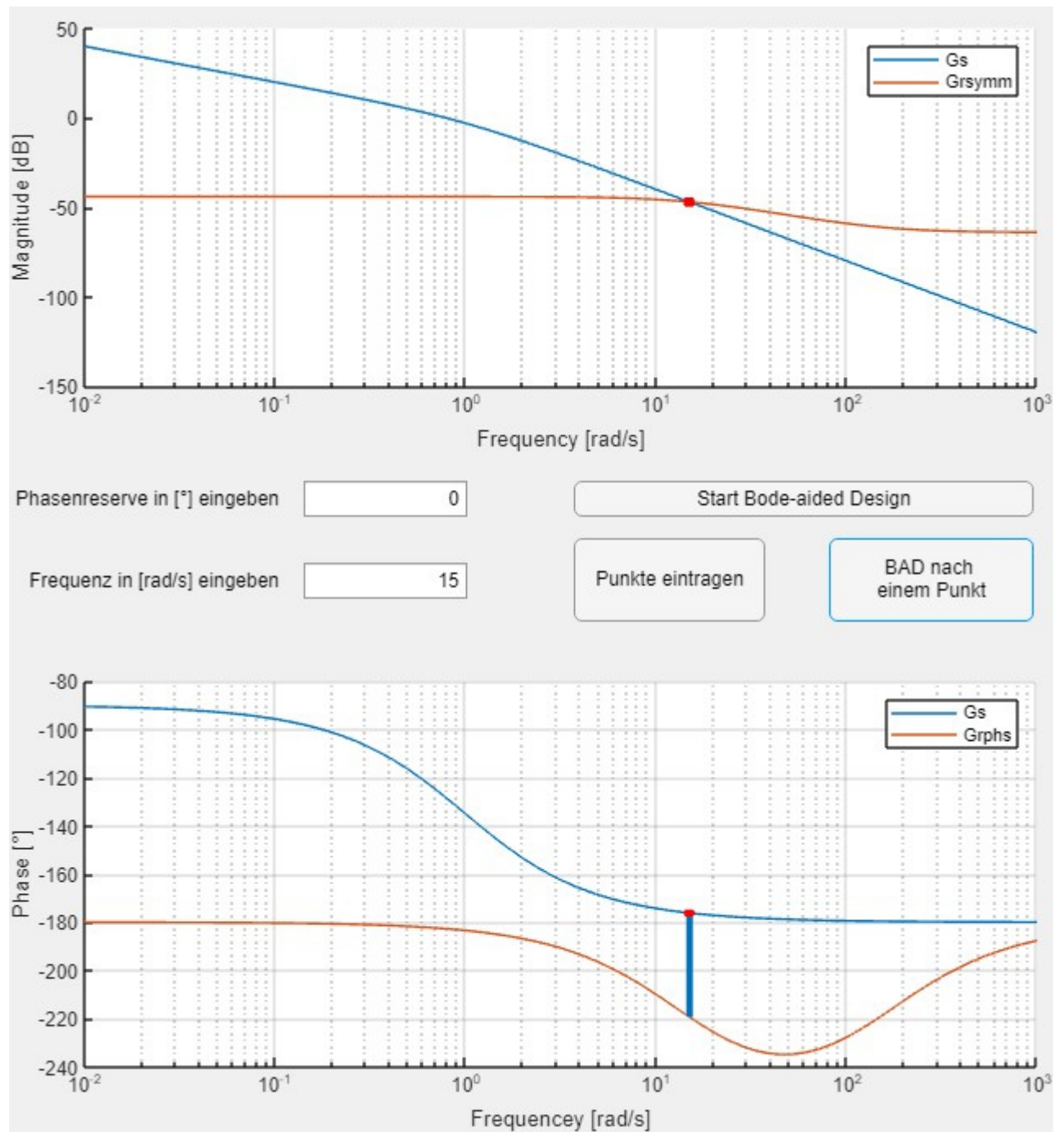


Abbildung 44: Ergebnis des Reglerentwurfs mittels BAD nach einem Punkt

Quelle: Eigene Darstellung

3.5.3 P-Regler

Der Entwurf des symmetrischen P-Reglers erfolgt analog zum Reglerentwurf mittels BAD. Beim BAD nach einem Punkt wird der Amplitudengang des symmetrischen Reglers $|G_{R_{\text{symm}}}(\omega)|_{\text{dB}}$ durch den bestimmten Punkt P gelegt. Dazu wird die Amplitudendifferenz Δp des bestimmtem Punkt P zur 0 dB-Linie ermittelt und zur Berechnung der Reglerverstärkung K_{PR} mit $K_{\text{PR}} = 10^{(-\Delta p/20)}$ verwendet, siehe Abbildung 45. [1, S. 291]

```
switch app.ReglerauswahlDropDown.Value
    case {"P-Regler" }

        %Bestimmen delta_p
        delta_P = app.mag_process(app.intersection_bad);

        %Neuberechnen Reglerverstärkung Kpr
        app.KpREditField.Value = 10^(-delta_P/20);
```

Abbildung 45: Quellcode zum Entwerfen eines symmetrischen P-Reglers mittels BAD nach einem Punkt
Quelle: Eigene Darstellung

3.5.4 PI-Regler

Für den Entwurf des symmetrischen PI-Reglers wurde die bestimmte Frequenz ω_{BAD} als Knickfrequenz für den symmetrischen PI-Regler festgelegt und zur Berechnung der Nachhaltezeit T_n mit $T_n = 1/\omega_{\text{BAD}}$ verwendet. Nach der Neuberechnung des symmetrischen PI-Reglers wird die Amplitudendifferenz Δp zwischen dem Amplitudengang des symmetrische PI-Reglers $|G_{R_{\text{symm}}}(\omega)|_{\text{dB}}$ und der Amplitude des Punktes P bestimmt und die Reglerverstärkung K_{PR} mit $K_{\text{PRneu}} = K_{\text{PR}} * 10^{(-\Delta p/20)}$ neu berechnet, siehe Abbildung 46.

```
case { "PI-Regler" }

    %Neuberechnen Nachstellzeit Tn
    app.TnEditField.Value = 1 / app.wout(app.intersection_bad);

    %Neuberechnen des horizontal-verschobenen Reglers
    plot_controller(app);

    %Bestimmen der Amplitudendifferenz
    delta_P = app.mag_process(app.intersection_bad) - app.mag_controller_symm(app.intersection_bad);

    %Neuberechnen Reglerverstärkung Kpr
    app.KpREditField.Value = app.KpREditField.Value * 10^(-delta_P/20);
```

Abbildung 46: Quellcode zum Entwerfen eines symmetrischen PI-Reglers mittels BAD nach einem Punkt
Quelle: Eigene Darstellung

3.5.5 PID-T1-Regler

Beim Entwurf des symmetrischen PID-T1-Reglers mittels BAD nach einem Punkt wird der Punkt P_R des phasen-symmetrischen PID-T1-Reglers bestimmt, bei dem der Phasengang des phasen-symmetrischen PID-T1-Reglers $\varphi_{R_{\text{phs}}} = -180^\circ$ ist. Danach wird die Amplitudendifferenz Δp zwischen dem Amplitudengang des symmetrischen PID-T1-Reglers $|G_{R_{\text{symm}}}(j\omega)|_{\text{dB}}$ und der Amplitude des Punktes P bestimmt und die Reglerverstärkung K_{PR} mit $K_{PR_{\text{neu}}} = K_{PR} * 10^{(-\Delta p/20)}$ neu berechnet. Mit der Frequenz ω_{BAD} an der sich der Punkt P befindetet, wird über einen Faktor $\Delta\omega$ die Nachhaltezeit T_n des I-Anteils mit $T_n = 1 / (\omega_{\text{BAD}} / \Delta\omega)$ und die Vorhaltezeit T_v des D-Anteils mit $T_v = 1 / (\omega_{\text{BAD}} * \Delta\omega)$ berechnet, die Zeitverzögerungskonstante T_R berechnet sich mit $T_R = 0,1 * T_v$, siehe Abbildung 47.

```

case {"PID-T1-Regler" }

app.delta_omega = 2;

%Finden Punkt P bei dem PID-T1-Regler bei -180°
app.temp = abs(app.phase_controller_phs - (-180));
app.intersection_pid = find(app.temp == min(app.temp));

%Bestimmen der Amplitudendifferenz
delta_P = app.mag_process(app.intersection_bad) - app.mag_controller_symm(app.intersection_pid);

%Neuberechnen Reglerverstärkung Kpr
app.KpREditField.Value = app.KpREditField.Value * 10^(-delta_P/20);

%Neuberechnen Nachstellzeit Tn zur horizontalen Verschiebung
app.TnEditField.Value = 1 / (app.wout(app.intersection_bad) / app.delta_omega);

%Neuberechnen Vorhaltezeit Tv & Zeitverzögerung Tr
app.TvEditField.Value = 1 / (app.wout(app.intersection_bad) * app.delta_omega);
app.TREditField.Value = 0.1 * app.TvEditField.Value;

```

Abbildung 47: Quellcode zum Entwerfen eines symmetrischen PID-T1-Reglers mittels BAD nach einem Punkt
Quelle: Eigene Darstellung

3.5.6 PD-T1-Regler

Für den Entwurf des symmetrischen PD-T1-Reglers wurde die bestimmte Frequenz ω_{BAD} als Knickfrequenz für den symmetrischen PD-T1-Regler festgelegt und zur Berechnung der Vorhaltezeit $T_v = 1/\omega_{\text{BAD}}$ verwendet. Nach der Neuberechnung des symmetrischen PD-T1-Reglers wird die Amplitudendifferenz Δp zwischen dem Amplitudengang des symmetrischen PD-T1-Reglers $|G_{R_{\text{symm}}}(\omega)|_{\text{dB}}$ und der Amplitude des Punktes P bestimmt und die Reglerverstärkung K_{PR} mit $K_{PR_{\text{neu}}} = K_{PR} * 10^{(-\Delta p/20)}$ neu berechnet, siehe Abbildung 48.


```

case {"PD-T1-Regler" }

app.TvEditField.Value = 1 / app.wout(app.intersection_bad);
app.TREditField.Value = 0.1 * app.TvEditField.Value;

%Neuberechnen des horizontal-verschobenen Reglers
plot_controller(app);

%Bestimmen der Amplitudendifferenz
delta_P = app.mag_process(app.intersection_bad) - app.mag_controller_symm(app.intersection_bad);

%Neuberechnen Reglerverstärkung Kpr
app.KpREditField.Value = app.KpREditField.Value * 10^(-delta_P/20);

```

Abbildung 48: Quellcode zum Entwerfen eines symmetrischen PD-T1-Reglers mittels BAD nach einem Punkt
Quelle: Eigene Darstellung

3.5.7 I-Regler

Für den Entwurf des symmetrischen I-Reglers wird die Nachstellezeit T_n berechnet indem die Asymptote A_n durch den bestimmten Punkt P gelegt und der Schnittpunkt der Asymptote mit der 0 dB-Linie berechnet wird. Die Berechnung des Schnittpunkts erfolgt analog zu der Berechnung in Kapitel 0. Daraus berechnet sich die Nachstellzeit T_n mit $T_n = 1/\omega_0 = 10^{(A/20)} / \omega_{BAD}$, siehe Abbildung 49.

```

case {"I-Regler" }

%Bestimmen delta_p
delta_P = app.mag_process(app.intersection_bad);

%Neuberechnen Nachstellzeit Tn
app.TnEditField.Value = (10^(delta_P/20)) / app.wout(app.intersection_bad);

```

Abbildung 49: Quellcode zum Entwerfen eines symmetrischen I-Reglers mittels BAD nach einem Punkt
Quelle: Eigene Darstellung

3.6 Grenzen des Systems

Dem Reglerentwurf mittels BAD in der Fingerprint-Applikation sind Grenzen gesetzt. Wie schon in Kapitel 2.2.5 beschrieben kann keine stabile Lösung gefunden werden, wenn der Phasengang der Strecke $\varphi_S(\omega)$ im gesamten Verlauf des Bode-Diagramms unterhalb des Phasengangs des phasensymmetrischen Reglers $\varphi_{Rphs}(\omega)$ verläuft. Des Weiteren liefert der Reglerentwurf kein stabiles Ergebnis sofern der Phasengang der Strecke $\varphi_S(\omega)$ unterhalb des charakteristischen Punktes φ_{Char} des gewählten Reglers liegt, da das Stabilitätskriterium des DBV nicht erfüllt wird. Beim BAD nach einem Punkt ist dies der Fall, wenn der Phasengang des bestimmten Punktes P unterhalb des Phasengangs des charakteristischen Punktes φ_{Char} des gewählten Reglers liegt.

Bei einem asymptotischen Verlauf des Amplitudengangs des symmetrischen Reglers $|G_{Rsymm}(j\omega)|_{dB}$ und des Amplitudengangs der Strecke $|G_S(j\omega)|_{dB}$ detektiert die Fingerprint-Applikation ein Schnittpunkt der Amplitudengänge, obwohl kein Schnittpunkt vorhanden ist. Die Überprüfung des Stabilitätskriteriums liefert fälschlicherweise ein positives Ergebnis, obwohl die Sprungantwort ein instabiles Verhalten des geschlossenen Regelkreises zeigt, siehe Abbildung 50.

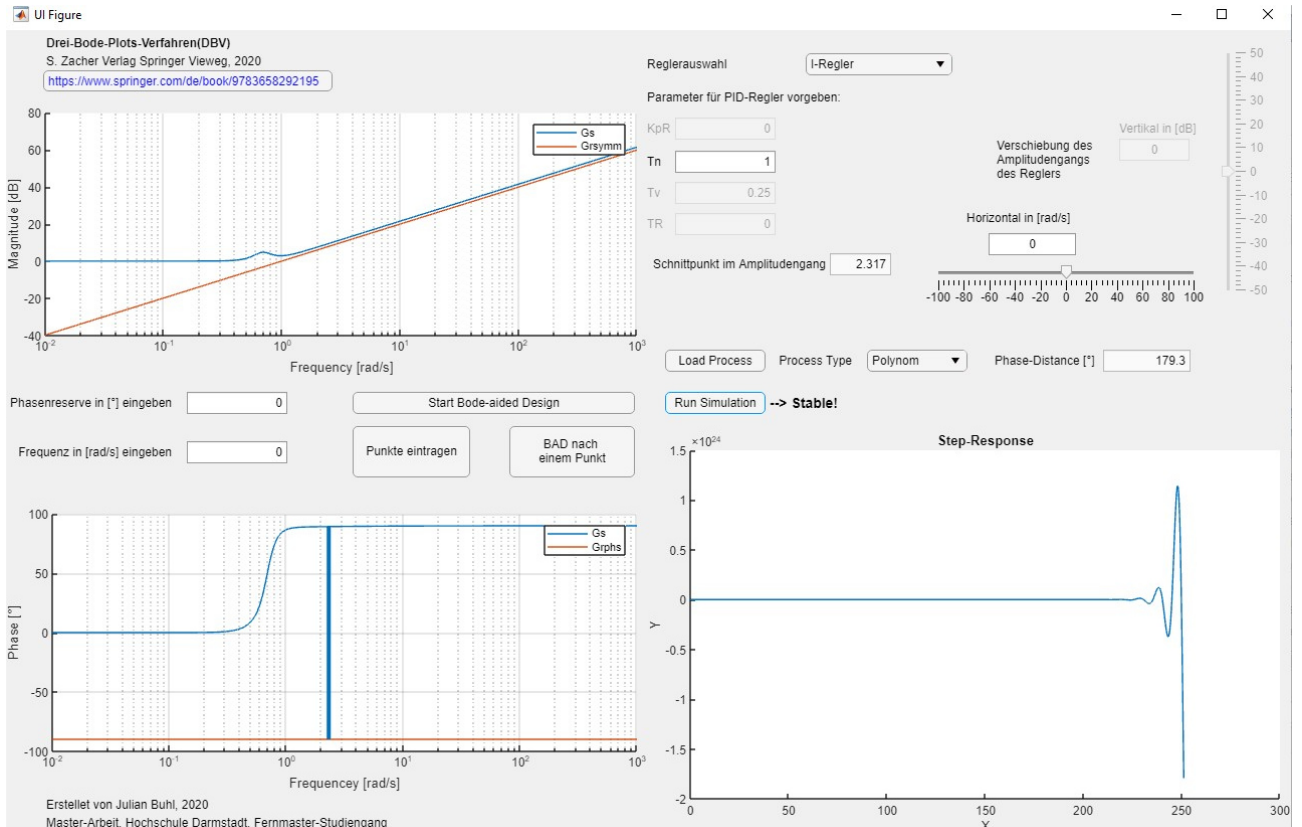


Abbildung 50: Asymptotischer Verlauf der Phasengänge mit positiver Überprüfung des Stabilitätskriterium und gleichzeitiger instabiler Sprungantwort
Quelle: Eigene Darstellung

Der Grund für die fälschliche Erkennung eines Schnittpunktes der Amplitudengänge liegt in der Findung des Schnittpunktes begründet. Da die Bode-Diagramme aus einer endlichen Anzahl an Punkten bestehen, ist die exakte Findung eines Schnittpunktes quasi unmöglich. Stattdessen wird in der Fingerprint-Applikation nach dem Punkt gesucht, bei dem die Amplitudengänge des symmetrischen Reglers und der Strecke die geringste Differenz aufweisen, siehe Abbildung 51.

```
%Findung des Schnittpunktes
app.temp = abs(app.mag_process - app.mag_controller_symm);
app.intersection = find(app.temp == min(app.temp));
```

Abbildung 51: Quellcode zum Finden des Schnittpunkts
Quelle: Eigene Darstellung

4 Bode-Diagramm der Teststrecke

Zur Evaluierung der Funktionalitäten der Fingerprint-Applikation sollte das Bode-Diagramm einer Hardware-Strecke bestimmt und in der Applikation eingelesen werden. Als Hardware-Strecke wurde die Modelltafel ET722 der Firma AEG gewählt. Zur Ansteuerung der Modelltafel ET722 wurde eine speicherprogrammierbare Steuerung (SPS) der Fa. Siemens verwendet und die Vorgabe und Auswertung der Signale wurde über ein Matlab-Skript realisiert.

4.1 Aufbau der Modelltafel ET722

Die Modelltafel ET722 ermöglicht die Simulation von chemikalischen Prozessen eines Rührkesselreaktors. Dazu kann die Befüllung und Entleerung des Kessels mit zwei Flüssigkeiten und die Temperatur des Kessels über einen Kühl- und Heizkreislauf simuliert werden.

Die Modelltafel ET722 simuliert zwei Analogventile AO1 und AO2 zur Regelung des Füllstands und der Temperatur, die mit Signalen von 0 – 10 V arbeiten. Es stehen sechs binäre Ventile O1 - O6 zur Steuerung von Produktzulauf, Kühlmittel und Heizmittel, die eine Spannung > 12 V für den Zustand „Auf“ benötigen, zur Verfügung. Die Sollwerte werden über zwei Potentiometer AI3 und AI4 eingestellt und an den gleichnamigen Buchsen ausgegeben. An den Buchsen AI1 und AI2 werden die Istwerte der Regelkreise ausgegeben. Zusätzlich werden die Soll-, Ist- und Stellwerte über Balkenanzeigen optisch dargestellt, siehe Abbildung 52.

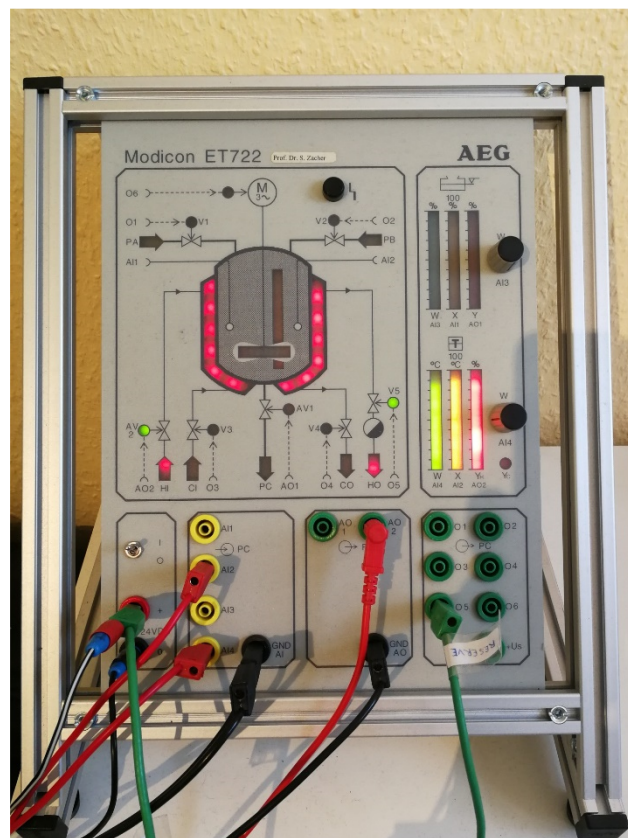


Abbildung 52: Anschluss der Modelltafel ET722 zur experimentellen Bestimmung des Bode-Diagramms
Quelle: Eigene Darstellung

Im Rahmen dieser Arbeit wurde sich auf die Regelung der Temperatur des Kessels beschränkt, der Temperaturbereich der Modelltafel ET722 geht von 0 °C bis 100 °C mit einer linearen Skalierung. Für die Temperaturregelung wurde das binäre Ventil O5 in den Zustand „Auf“ versetzt und der Istwert AI2, der Sollwert AI4 und der Stellwert AO2 des Temperaturkreises mit der SPS verbunden. Der Sollwert AI4 kann über das Potentiometer vorgegeben werden, im Rahmen dieser Arbeit wurde diese Aufgabe von der SPS übernommen. Das Ventil O5 muss angesteuert werden, da ansonsten kein Fluss des Heizmittels simuliert wird und die Regelung nicht funktioniert.

Eine vollständige Auflistung der Prozesseingänge sind in Tabelle 2 und die Prozessausgänge sind in Tabelle 3 dargestellt.

Tabelle 2: Prozesseingänge der Modelltafel ET722

Bezeichnung	Funktion	Signal	Min. Wert	Max. Werte
AO1	AV1, Gefäß-Ablauf	Analog	0 V \triangleq 0 %	10 V \triangleq 100 %
AO2	AV2, Heizmittel-Zulauf	Analog	0 V \triangleq 0 %	10 V \triangleq 100 %
O1	V1, Zulauf Produkt A	Digital	< 5 V \triangleq Zu	> 12 V \triangleq Auf
O2	V2, Zulauf Produkt B	Digital	< 5 V \triangleq Zu	> 12 V \triangleq Auf
O3	V3, Kühlmittelzulauf	Digital	< 5 V \triangleq Zu	> 12 V \triangleq Auf
O4	V4, Kühlmittelablauf	Digital	< 5 V \triangleq Zu	> 12 V \triangleq Auf
O5	V5, Heizmittelablauf	Digital	< 5 V \triangleq Zu	> 12 V \triangleq Auf
O6	Motor Rührwerk	Digital	< 5 V \triangleq Aus	> 12 V \triangleq An

Quelle: Ausgehändigter Bericht des Dozenten

Tabelle 3: Prozessausgänge der Modelltafel ET722

Bezeichnung	Funktion	Signal	Min. Wert	Max. Werte
AI1	Istwert Füllstand	Analog	0 V \triangleq 0 %	10 V \triangleq 100 %
AI2	Istwert Temperatur	Analog	0 V \triangleq 0 %	10 V \triangleq 100 %
AI3	Sollwert Füllstand	Analog	0 V \triangleq 0 %	10 V \triangleq 100 %
AI4	Sollwert Temperatur	Analog	0 V \triangleq 0 %	10 V \triangleq 100 %

Quelle: Ausgehändigter Bericht des Dozenten

4.2 Aufbau der SPS

Für die Ansteuerung der Modelltafel ET722 wurde eine Simatic S7-1200 der Fa. Siemens verwendet. Das System besteht aus einer Spannungsversorgung der Fa. Phoenix Contact, der CPU 1214C DC/DC/RLY und einer analogen Ein-/Ausgabegruppe SM 1234 AI/AQ der Fa. Siemens, siehe Abbildung 53.

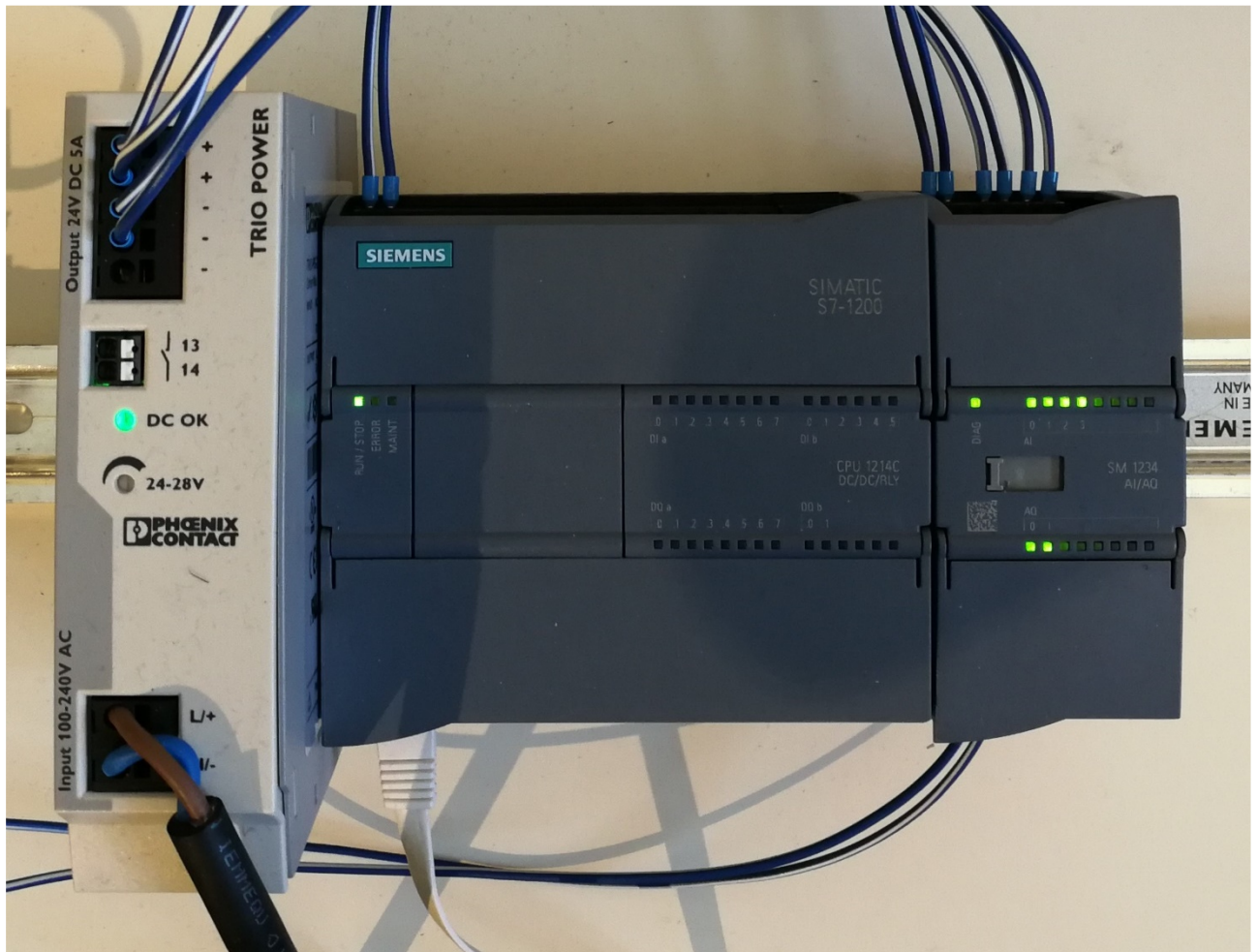


Abbildung 53: Aufbau der SPS-Hardware mit Spannungsversorgung, CPU und analoge Ein-/Ausgabegruppe
Quelle: Eigene Darstellung

Die Spannungsversorgung dient der Bereitstellung der benötigten 24 V DC zur Versorgung der CPU und der analogen Ein-/Ausgabegruppe. In der CPU wird das Programm zur Bestimmung des Bode-Diagramms ausgeführt und über die integrierte Ethernet-Schnittstelle wird die Verbindung zu Matlab hergestellt. Die analoge Ein-/Ausgabegruppe dient dem Lesen und Schreiben der Analogsignale und kann Signale mit ± 10 V DC oder 0-20 mA verarbeiten. Die analogen Eingangssignale werden dabei von der analogen Ein-/Ausgabegruppe durch einen integrierten Analog-Digital-Umsetzer in ein 16 Bit Integer-Wert mit dem Wertebereich von -32.768 bis 32.767 umgewandelt. Die Ausgabe von analogen Signalen erfolgt über einen Digital-Analog-Umsetzer der Integer-Werte von -32.768 bis 32.767 in ein Signal mit ± 10 V DC oder 0-20 mA umwandelt und ausgibt.

Das SPS-Programm besteht aus der Funktion FC1 in der der Sollwert als „Bode_Ein“ aus der Variablen-Tabelle des Datenbausteins DB1 eingelesen und per Analogausgang auf das analoge Ventil AO2 übertragen wird, siehe Abbildung 54.

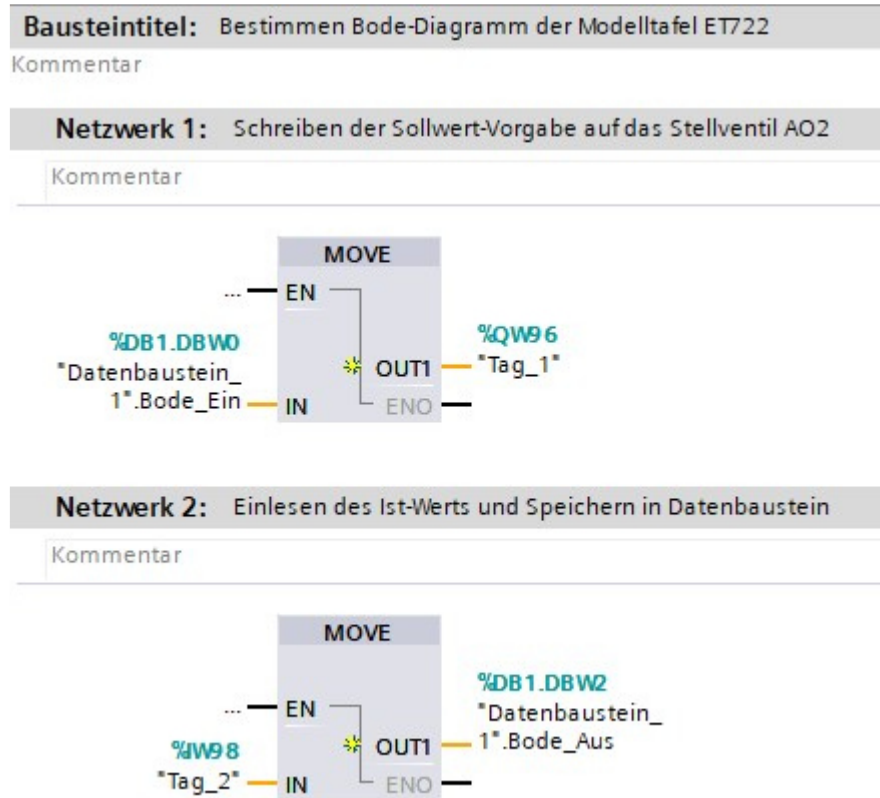


Abbildung 54: Aufbau des SPS-Programms zum Lesen und Schreiben von Soll- und Istwert zur Bestimmung des Bode-Diagramms der Modelltafel ET722

Quelle: Eigene Darstellung

Anschließend wird der Istwert AI2 der Modelltafel ET722 per Analogeingang eingelesen und in der Variablen-Tabelle des Datenbausteins DB1 als „Bode_Aus“ gespeichert. In dem Datenbaustein DB1 werden die Variablen gespeichert, die von Matlab übertragen und an Matlab gesendet werden. siehe Abbildung 55.

Datenbaustein_1		
	Name	Datentyp
1	Static	
2	Bode_Ein	Int
3	Bode_Aus	Int

Abbildung 55: Aufbau des Datenbausteins DB1 für den Austausch der SPS mit dem Matlab-Skript

Quelle: Eigene Darstellung

4.3 Matlab-Skript zur Bestimmung des Bode-Diagramms

In dem Matlab-Skript zur Bestimmung des Bode-Diagramms der Modelltafel ET722 findet der gesamte Prozess zur Bestimmung des Bode-Diagramms von der Erzeugung des Sinus-Signals als Vorgabe für die Modelltafel ET722 über die Verarbeitung des Eingelesenen Istwertes bis zur Erzeugung der Werte für das Bode-Diagramm statt. Die SPS dient im Rahmen dieser Arbeit lediglich zur Übertragung der Werte von der Modelltafel ET722 nach Matlab und umgekehrt.

Für die direkte Kommunikation der SPS mit Matlab wurde die Toolbox „S-7 Toolbox for use with Matlab“ der Fa. Traeger [2] verwendet. Die Toolbox ermöglicht den direkten Zugriff aus Matlab auf die Datenbereiche der SPS wie z.B. Datenbausteine, damit werden die Variablen des Datenbausteins DB1 gelesen und beschrieben. Die Verbindung zwischen Matlab und der SPS erfolgt über die Ethernet Schnittstelle der SPS und des Computers auf dem Matlab ausgeführt wird.

Zur Herstellung der Verbindung muss im Matlab-Skript die IP-Adresse, die belegten Racks und die Modellreihe der verwendeten SPS angegeben werden. Anschließend wird über den Befehl „connection“ die Verbindung erzeugt und mit dem Befehl „connection.Open()“ wird die Verbindung hergestellt, siehe Abbildung 56.

```
device = SiemensDevice(IPDeviceEndPoint('192.168.0.1',0,1)); %Ip-Adresse der SPS definieren
device.Type = SiemensDeviceType.S71200; % Auswahl der verwendeten SPS
connection = device.CreateConnection(); %Verbindung erzeugen
connection.Open(); %Verbindung herstellen
```

Abbildung 56: Quellcode zur Verbindung der SPS mit Matlab mit Hilfe der Traeger Toolbox

Quelle: Eigene Darstellung

Nach dem Aufbau der Verbindung werden in Matlab die Frequenzvektoren $m_0 - m_4$ zur Bestimmung des Bode-Diagramms erzeugt und in dem Frequenzvektor w zusammengefasst. Des Weiteren werden die Arrays mag_dB und $angle_1$, in denen die Werte von Amplitude und Phase des Bode-Diagramms gespeichert werden, und die Laufvariable k für die experimentelle Bestimmung des Bode-Diagramms erzeugt wobei die Länge des Frequenzvektors w den Endpunkt der Laufvariable k vorgibt, siehe Abbildung 57.

```
m0 = 0.01:0.01:0.09; %Frequenzbereich definieren
m1 = 0.1:0.1:0.9; %Frequenzbereich definieren
m2 = 1:1:9; %Frequenzbereich definieren
m3 = 10:10:90; %Frequenzbereich definieren
m4 = 100:100:1000; %Frequenzbereich definieren
w = [m1 m2 m3 m4]; %Frequenzbereiche zusammenführen
mag_dB = zeros(1, length(w), 'double'); %Initialisieren des Amplituden-Arrays
angle1 = zeros(1, length(w), 'double'); %Initialisieren des Phasen-Arrays
```

```
for k=1:1:length(w) %Laufvariable der Arrays
```

Abbildung 57: Quellcode zum Erzeugen der Frequenzvektoren $m_0 - m_4$ und w , der Arrays mag_dB und $angle_1$ und der Laufvariablen k

Quelle: Eigene Darstellung

Im nächsten Schritt wird das Sinussignal, welches an die Modelltafel ET722 geschickt werden soll erzeugt. Dazu wird die aktuelle Frequenz f_0 mit der Laufvariable k aus dem Frequenzvektor w bestimmt und die Anzahl der Perioden anz_per , der Offset der Sinusschwingung dc und die Amplitude der Sinusschwingung a festgelegt.

Der Offset dc ist notwendig da die Modelltafel ET722 von der Umgebungstemperatur beeinflusst wird und unter Verwendung des Heizkreislauf mit dem Ventil O5 der Istwert AI2 nicht unter die Umgebungstemperatur fällt. Aus der Frequenz f_0 wird die Periodendauer T mit $T = 1/f_0$ und mit der Anzahl der Perioden anz_per der Endpunkt der Sinusschwingung T_end mit $T_end = T * anz_per$ berechnet. Aus der Frequenz f_0 wird die Abtastfrequenz fa mit $fa = 10 * f_0$ und daraus die Abtastrate Ta mit $Ta = 1/fa$ berechnet. Mit der Abtastrate wird der Zeitvektor t erzeugt, der von 0 bis $(T_end - Ta)$ läuft.

Aus diesen Werten wird das Sinussignal U_e , das als Eingangssignal für die Modelltafel ET722 dient, erzeugt mit $U_e = dc + a * \sin(w(k) * t * f_0)$ und anschließend vom Datentyp Double in Integer umgewandelt. Die Umwandlung erfolgt, da die analoge Ein-/Ausgabegruppe der SPS mit Integer-Werten arbeitet und somit eine Umwandlung des Signals in der SPS entfallen kann. Das Ausgangssignal U_a der Strecke wird erzeugt und mit Nullen befüllt, das Ausgangssignal ist ebenfalls vom Typ Integer, da somit die gemessenen Werte der Modelltafel ET722 direkt in dem Array gespeichert werden können, siehe Abbildung 58.

```
f0 = w(k)/(2*pi); %Frequenz
anz_per = 100; %Anzahl Perioden
dc = 20000; %Offset
a = 3000; %Amplitude
T = 1/f0; % Periodendauer
T_end = T * anz_per; %Endpunkt
fa = 10 * f0; %Abtastfrequenz
Ta = 1/fa; % Abtastrate
t = 0:Ta:(T_end-Ta); % Zeitvektor
Ue = dc + a*sin(w(k)*t*f0); % Sinussignal
Ue = int16(Ue); %Umwandeln von DOUBLE zu INTEGER

Ua = zeros(1,1000,'int16'); %Ausgangssignal der Strecke
```

Abbildung 58: Quellcode zum Erzeugen der Sinussignale U_e als Eingangssignal und U_a als Ausgangssignal
Quelle: Eigene Darstellung

Im nächsten Schritt wird das erzeugte Sinussignal an die SPS gesendet und im Anschluss der Istwert der Modelltafel ET722 ausgelesen. Dazu wird eine for-Schleife zum Begrenzen der Schreib- und Lesevorgänge und die Variable i als Laufvariable zum Auslesen des Eingangssignal U_e und zum Beschreiben des Ausgangssignal U_a verwendet.

Nach dem Erzeugen der for-Schleife wird mit der Laufvariable i der Wert aus dem Eingangssignal U_e ausgelesen und die Variable die beschrieben werden soll, im Rahmen dieser Arbeit wurde das Wort 0 des Datenbausteins DB1 beschrieben, mit dem Befehl „PlcInt16“ angegeben

und als Variable value gesetzt. Die Variable value wird anschließend mit dem Befehl „connection.WriteValues“ an die SPS gesendet.

Zum Auslesen des Ausgangssignal wird die zu Lesende Variable der SPS, im Rahmen dieser Arbeit wurde das Wort 2 des Datenbausteins DB1 gelesen, mit dem Befehl „PlcInt16“ angegeben und in der Variable value1 gespeichert. Mit dem Befehl „connection.ReadValues“ wird der Wert aus der SPS gelesen und in der Variable Us gespeichert. Der Befehl „connection.ReadValues“ speichert den gelesen Wert als Cell-Array ab, weswegen eine Umwandlung der Variable Us in einen Integer-Wert mit dem Befehl „cell2mat“ erfolgt. Anschließend wird die Variable Us entsprechend der Laufvariable i in dem Array des Ausgangssignal Ua gespeichert.

Nach den Schreib- und Lesevorgängen wird mit dem Befehl „pause“ das weitere Ausführen des Quellcodes verzögert, diese Verzögerung ist nötig da ansonsten die Werte des Eingangssignal Ue unabhängig von der Frequenz f_0 an die SPS gesendet. Als Größe der Verzögerung wurde die Abtastrate T_a gewählt, da somit die Signale entsprechend der gewählten Abtastfrequenz f_a zur SPS übertragen werden, siehe Abbildung 59.

```
for i=1:1:1000 %Begrenzen der Schreib-Vorgänge auf 1000

value = PlcInt16('DB1.DBW 0', Ue(i)); %Schreiben der Eingangsvariable
connection.WriteValues({ value });

value1 = PlcInt16('DB1.DBW 2'); %Lesen der Antwort der Strecke
Us = connection.ReadValues({ value1 });
Ua(i) = cell2mat(Us); %Umwandeln des Cell-Array in INTEGER und speichern

pause(Ta) %Begrenzen der Übertragung auf definierte Frequenz des Signals
end
```

Abbildung 59: Quellcode zum Schreiben des Eingangssignal und Lesen des Ausgangssignal

Quelle: Eigene Darstellung

Die somit gewonnen Signale werden im nächsten Schritt ausgewertet, um die Amplitude und Phase der untersuchten Frequenz f_0 zu bestimmen. Dazu werden im ersten Schritt mögliche Einschwingvorgänge des Ausgangssignals Ua entfernt. Für die Auswertung der Signale ist es notwendig, dass beide Signale die gleiche Größe haben weswegen auch das Eingangssignal Ue begrenzt wird. Dazu werden die ersten 488 Werte der Arrays entfernt, dadurch bleiben 512 Werte der Arrays erhalten.

Anschließend wird über den Befehl „mean()“ der Offset der Arrays ermittelt und von den Werten subtrahiert um den Einfluss des Offsets bei der Berechnung der Amplitude zu eliminieren.

Nach dem Anpassen der Signale wird die Schnelle Fourier-Transformation durch den Befehl „fft()“ der beiden Signale durchgeführt. Die Durchführung der Schnellen Fourier-Transformation ist nur möglich, wenn die Größe der Signale einer Zweierpotenz entsprechen, weswegen die Anzahl der Signale auf 512 Werte reduziert wurde. Die durch die Fourier-Transformation berechneten Signale werden in den Variablen X und Y gespeichert.

Mit dem Befehl „max(abs())“ werden die Maxima der transformierten Signale bestimmt. Dabei werden die Maxima-Werte der Amplituden in den Variablen mag_x und mag_y gespeichert und die jeweiligen Frequenzen in den Index-Variablen idx_x und idx_y gespeichert. Mit den Index-Variablen werden mit dem Befehl „angle()“ die Phasen der transformierten Signale X und Y in rad/s bestimmt und aus deren Differenz die Phasenverschiebung berechnet. Die Phasenverschiebung wird mit der Laufvariable k in dem Array angle1 gespeichert.

Aus den Maxima-Werte der Amplituden mag_x und mag_y wird das Amplitudenverhältnis berechnet und mit $\text{mag_dB}(k) = 20 * \log_{10}(\text{amplitude_ratio})$ wird der Wert des Amplitudenverhältnis in einen Wert in dB umgewandelt und mit der Laufvariable k in dem Array mag_dB gespeichert, siehe Abbildung 60.

Damit wurde ein Punkt des Bode-Diagramms der Modelltafel ET722 bestimmt und der Quellcode wird für die nächste Frequenz des Frequenzvektors w erneut ausgeführt.

```
%Entfernen der Einschwingvorgänge
Ue(1:488) = [];
Ua(1:488) = [];

%Entfernung des Offsets
Ue = Ue - mean(Ue);
Ua = Ua - mean(Ua);

%Fourier-Transformation der Signale
X=fft(Ue);
Y=fft(Ua);

%Feststellen der Maxima
[mag_x, idx_x] = max(abs(X));
[mag_y, idx_y] = max(abs(Y));

%Berechnen der Phasenverschiebung
px = angle(X(idx_x));
py = angle(Y(idx_y));
phase_lag = py - px;
angle1(k) = phase_lag;

%Berechnung des Amplitudenverhältnis
amplitude_ratio = mag_y/mag_x;

%Umrechnung in dB
mag_dB(k) = 20*log10(amplitude_ratio);

end
```

Abbildung 60: Quellcode zur Auswertung der Signale zur Bestimmung von Amplitude und Phase der Modelltafel ET722
Quelle: [3]

Inbesondere bei hohen Frequenzen kam es bei der Bestimmung des Phasengangs zu großen Sprüngen zwischen einzelnen Werten der Phase. Für diese Sprünge kann es mehrere Ursachen geben,

dazu zählen die Übertragungszeit zwischen Matlab und der SPS, die Zykluszeit der SPS, die Umwandlungszeit der analogen Ein-/Ausgabegruppe, Messfehler durch die Verdrahtung, die Zykluszeit des Matlab-Skripts etc.

Um diese Sprünge zu unterdrücken wurde nach dem Abschluss der Messreihe eine Korrektur der Phase vorgenommen. Die Phasenwerte in dem Array `angle1` werden von rad/s mit dem Befehl `rad2deg` in Grad umgewandelt und in der Variable `phase_deg` gespeichert. Beim Plotten des Bode-Diagramms wurden starke Sprünge zwischen Werten des Phasengangs festgestellt, siehe Abbildung 61.

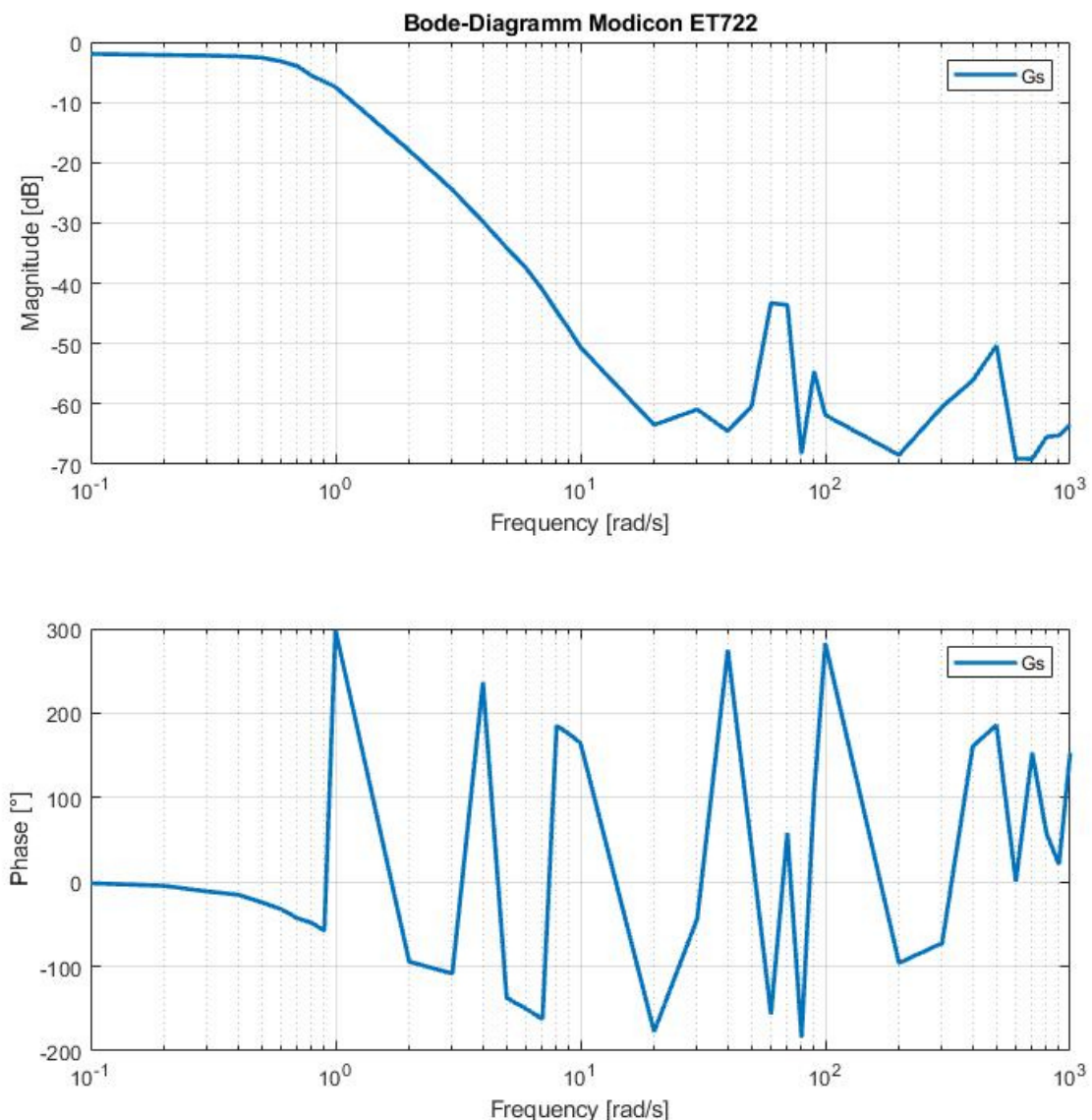


Abbildung 61: Ausgabe des Ergebnisses ohne Phasenkorrektur

Quelle: Eigene Darstellung

Um diese Sprünge zu reduzieren wird mit dem Befehl „unwrap“ eine Korrektur der Phasenwerte vorgenommen. Dabei wird beim Aufrufen des Befehls überprüft, ob die Differenz von aufeinander folgenden Werten des Arrays `phase_deg` kleiner als der vorgegebene Wert des Befehls ist. Liegt eine größere Differenz als der vorgegebene Wert vor, wird die Phasendifferenz auf den vorgegebenen Wert begrenzt. Die korrigierten Werte werden in dem Array `phase` gespeichert, siehe Abbildung 62.

```
%Korrektur der Phasenverschiebung
phase_deg = rad2deg(angle1);
phase = unwrap(phase_deg, 90);
```

Abbildung 62: Quellcode zum Korrigieren der Phase

Quelle: Eigene Darstellung

Nach der Korrektur der Phase werden die bestimmten Werte der Amplitude und der Phase in einem Bode-Plot ausgegeben und zusätzlich in einer Matlab-Datei zur Weiterverwendung in der Fingerprint-Applikation ausgegeben. Dazu werden zwei Subplots, jeweils für die Amplitude und die Phase, erzeugt und mit dem Befehl „semilogx“ werden die Amplitude und Phase mit einer logarithmischen Frequenzachse dargestellt. Anschließend wird eine neue Matlab-Datei mit dem Datum der Messung erzeugt und gespeichert, siehe Abbildung 63.

```
%Plotten der Messwerte
figure;
ax1=subplot(2,1,1);
semilogx(w,mag_dB);
title('Bode-Diagramm Modicon ET722');
xlabel('Frequency [rad/s]');
ylabel('Magnitude [dB]');
grid on

ax2=subplot(2,1,2);
semilogx(w,phase);
xlabel('Frequency [rad/s]');
ylabel('Phase [°]');
grid on
linkaxes([ax1 ax2], 'x')

%Speichern der Messwerte in Matlab-Datei
filename = '2021_03_13_Bestimmung_bode_Tafel.mat';
save(filename)
```

Abbildung 63: Quellcode zum Plotten der Messergebnisse und Erzeugen der Matlab-Datei

Quelle: Eigene Darstellung

Ein mit dem oben beschriebenen Matlab-Skript bestimmtes Bode-Diagramm ist in Abbildung 64 dargestellt, die anderen Bode-Diagramme befinden sich im Anhang, siehe Kapitel 7.15.

Das vollständige Matlab-Skript befindet sich im Anhang als Abbildung 85 und Abbildung 86.

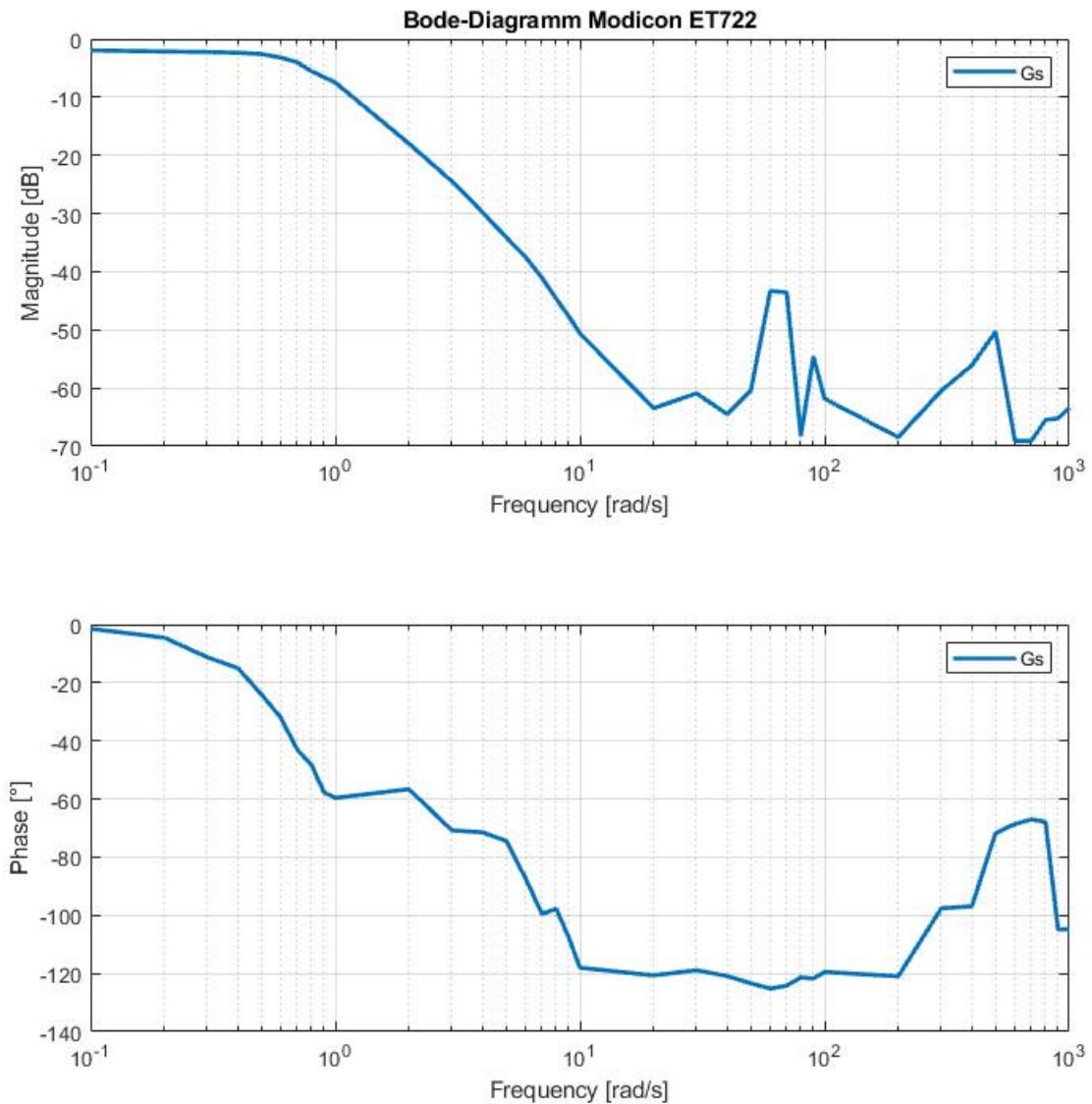


Abbildung 64: Experimentell bestimmtes Bode-Diagramm der Modelltafel ET722 vom 10.03.2021

Quelle: Eigene Darstellung

5 Abschließende Betrachtung

5.1 Fazit

Ziel dieser Masterarbeit war es die bestehende Fingerprint-Applikation, die das ZBV als Stabilitätskriterium verwendet, anzupassen, sodass das DBV als neues Stabilitätskriterium verwendet wird. Es sollten auch weitere Funktionalitäten wie der Reglerentwurf mittels BAD und BAD nach einem Punkt implementiert werden. Zuletzt sollte die Funktion der Fingerprint-Applikation an einem praktischen Beispiel evaluiert werden. Dazu sollte das Bode-Diagramm einer Hardware-Strecke experimentell bestimmt und in die Fingerprint-Applikation eingelesen und verarbeitet werden.

Das Anpassen der Fingerprint-Applikation erfolgte in zwei Schritten. Im ersten Schritt wurden die für das DBV notwendigen Regler, symmetrischer und phasen-symmetrischer Regler, aus der Übertragungsfunktion des Reglers berechnet und anschließend als Bode-Diagramm zusammen mit der Strecke dargestellt. Für die Anpassung des Stabilitätskriteriums musste in einem zweiten Schritt die Abfrage der Phasendifferenz $\Delta\varphi$ im Schnittpunkt der Amplitudengänge des symmetrischen Reglers und der Strecke von $\Delta\varphi > 360^\circ$ zu $\Delta\varphi < 0^\circ$ geändert werden.

Für die Implementierung des Reglerentwurfs mittels BAD mussten Anpassungen an der Benutzeroberfläche der Fingerprint-Applikation und Erweiterungen des Quellcodes vorgenommen werden. Über ein ergänztes Eingabefeld kann eine gewünschte Phasenreserve φ_{Rd} als Gütekriterium für den Reglerentwurf vorgegeben werden, im Falle das keine Phasenreserve eingetragen wird, wird ein Initialwert verwendet. Für den Reglerentwurf wird in Abhängigkeit des gewählten Reglertyps, der Punkt P des Phasengangs der Strecke bestimmt, der vom charakteristischen Punkt P_{Char} des Phasengangs des phasen-symmetrischen Reglers den Abstand φ_{Rd} hat. Die Vorgabe des charakteristischen Punktes P_{Char} erfolgt beim Ausführen des Quellcodes des gewählten Reglertyps. Nach Finden des Punktes P wird der Regler entworfen und zusammen mit der Strecke in einem Bode-Diagramm dargestellt.

Für die Funktion des Reglerentwurfs mittels BAD nach einem Punkt wurde die Möglichkeit geschaffen, über Vorgabe einer Frequenz einen einzelnen Punkt des Bode-Diagramms der Strecke festzulegen, der für den Entwurf verwendet werden soll. Der Entwurf des Reglers erfolgt analog zum Reglerentwurf mittels BAD, wobei beim BAD nach einem Punkt der Punkt P bereits vorgegeben ist und nicht bestimmt werden muss.

Für die Evaluierung der Funktionalität der Fingerprint-Applikation sollte das Bode-Diagramm der Modelltafel ET722 experimentell bestimmt und in die Fingerprint-Applikation eingelesen und verarbeitet werden, dabei wurde sich auf den Temperaturkreis der Modelltafel ET722 begrenzt. Zur Bestimmung des Bode-Diagramms wurde die Modelltafel ET722 an eine SPS S7-1200 der Fa. Siemens angeschlossen, um die Istwerte zu messen und eine Sollwertvorgabe zu realisieren. Die Sollwerte wurden von einem Matlab-Skript vorgegebenen und die Istwerte in dem Skript eingelesen und mit den Werten der Sollwertvorgabe verarbeitet, um ein Bode-Diagramm der Modelltafel ET722 zu bestimmen. Zum Einlesen des bestimmten Bode-Diagramms wurde die Fingerprint-Applikation erneut erweitert, sodass Matlab-Dateien eingelesen und verarbeitet werden können.

Diese Masterarbeit hat gezeigt, dass sich das DBV eignet um die Stabilität eines Regelkreises nur Anhand der Bode-Diagramme der Strecke und des symmetrischen und phasen-symmetrischen Reglers in der Fingerprint-Applikation zu bestimmen. Das DBV ermöglicht im Vergleich zu seinem Vorgänger dem ZBV das Erkennen der Stabilität des geschlossenen Regelkreises anhand der Lage der Phasengänge des phasen-symmetrischen Reglers und der Strecke. Liegt der Phasengang der Strecke im Schnittpunkt der Amplitudengänge der Strecke und des symmetrischen Reglers oberhalb des Phasengangs des phasen-symmetrischen Reglers dann ist der geschlossene Regelkreis stabil.

Die Implementierung des BAD in die Fingerprint-Applikation ermöglicht den Entwurf eines Standardreglers nur durch Kenntnisse über das Bode-Diagramm der Strecke. Dazu müssen vom Benutzer der Fingerprint-Applikation nur in sehr geringem Maße Vorgaben, gewünschter Reglertyp und Phasenreserve, zum Reglerentwurf gemacht werden. Der Regler wird von der Fingerprint-Applikation selbstständig entworfen und das Ergebnis wird von der Applikation selbst auf seine Stabilität geprüft und die Sprungantwort berechnet und dargestellt. Liegt der Phasengang der Strecke unterhalb des charakteristischen Punkts P_{Char} des Phasengangs des phasen-symmetrischen Reglers, kann kein stabiler Regler entworfen werden, da das Stabilitätskriterium nach DBV nicht erfüllt werden kann. Es empfiehlt sich die Kontrolle der Sprungantwort, ob der geschlossene Regelkreis tatsächlich stabil ist. Verlaufen die Amplitudengänge der Strecke und des symmetrischen Reglers im Bereich des Schnittpunktes nahezu parallel zueinander, kann der entworfene Regler ein instabiles Ergebnis liefern obwohl die Überprüfung des Stabilitätskriteriums ein stabiles Ergebnis anzeigt. Um in diesem Fall ein stabiles System zu erzeugen, müsste ein präziser Reglerentwurf vorgenommen werden, was in der Praxis kaum realisierbar ist.

Innerhalb der Fingerprint-Applikation kommt es bei einem asymptotischen Verlauf der Amplitudengänge von Strecke und symmetrischen Reglers zur Detektion eines Schnittpunkts obwohl kein Schnittpunkt vorhanden ist. Das liegt an der Findung des Schnittpunkts in der Fingerprint-Applikation, da die Amplitudengänge des Bode-Diagramms aus einer endlichen Anzahl an Werten bestehen wird der Punkt mit der geringsten Differenz gesucht und als Schnittpunkt deklariert. Dadurch liefert die Überprüfung der Stabilität nach DBV ein stabiles Ergebnis aber die Sprungantwort zeigt das der geschlossene Regelkreis instabil ist.

Die Funktionalität des BAD nach einem Punkt hat zudem gezeigt, dass der Entwurf eines Standardreglers nur Anhand eines Punktes des Bode-Diagramms der Strecke mit der Fingerprint-Applikation möglich ist und ein stabiles Ergebnis liefert. Zur Überprüfung des Ergebnis wird bei der Anwendung des BAD nach einem Punkt ein Punkt des Bode-Diagramms der Strecke vorgegeben und anhand dieses Punktes der Reglerentwurf durchgeführt. Da Kenntnisse der Strecke vorhanden sind, kann zusätzlich zur Überprüfung der Stabilität nach DBV auch anhand der Sprungantwort das Erhalten eines stabilen Regelkreises überprüft werden. Liegt der vorgegebene Punkt der Strecke unterhalb des charakteristischen Punkts P_{Char} des ausgewählten Reglertyps, liefert der Reglerentwurf mittels BAD nach einem Punkt kein stabiles Ergebnis.

Die Evaluierung der Fingerprint-Applikation durch das Einlesen und Verarbeiten eines experimentell bestimmten Bode-Diagramms einer Hardwarestrecke hat gezeigt, dass die Funktionalitäten BAD und BAD nach einem Punkt nicht nur bei bekannten Strecken funktionieren, sondern auch wenn nur das Bode-Diagramm der Strecke gegeben ist.

5.2 Ausblick

Nach der Erstellung dieser Masterarbeit sind noch einige Punkte offengeblieben, die in zukünftigen studentischen Arbeiten betrachtet werden könnten. Diese Punkte sind:

- 1. Erweitern des Reglerentwurfs um Korrekturglieder:** Bei dem Reglerentwurf mittels BAD besteht in der aktuellen Version der Fingerprint-Applikation keine Möglichkeit den Phasengang des phasen-symmetrischen Reglers vertikal zu verschieben. Dadurch kann für eine Strecke deren Phasengang unterhalb von $\varphi = -270^\circ$ verläuft, kein stabiler Regelkreis entworfen werden. Durch die Einführung von Korrekturglieder könnte der Phasengang von Reglern horizontal als auch vertikal verschoben werden und für die oben beschriebenen Strecken könnte ein stabiler Regelkreis entworfen werden.
- 2. Überprüfen ob ein echter Schnittpunkt vorhanden ist:** Bei der aktuellen Schnittpunktfindung wird lediglich der Punkt mit dem geringsten Abstand gesucht, es findet keine Überprüfung statt, ob der gefundene Punkt tatsächlich ein Schnittpunkt ist. Durch die Überprüfung ob ein echter Schnittpunkt gefunden wurde, könnte die fälschliche Feststellung der Stabilität des geschlossenen Regelkreises vermieden werden.
- 3. Verbesserung der experimentellen Bestimmung des Bode-Diagramms:** Die experimentell bestimmten Bode-Diagramm der Modelltafel ET722 zeigen, insbesondere bei hohen Frequenzen, starke Abweichungen zwischen den einzelnen Messreihen und teilweise Sprünge beim Phasengang von einer Frequenz zur nächsten Frequenz. Durch eine Verbesserung der experimentellen Bestimmung könnte die Qualität der bestimmten Bode-Diagramme verbessert werden und die Verarbeitung der Bode-Diagramme in der Fingerprint-Applikation wird dadurch verbessert, was zu einem verbessertem Reglerentwurf führt.
- 4. Möglichkeit zur Identifikation der Strecke in der Fingerprint-Applikation:** Bei der Verwendung von experimentell bestimmten Bode-Diagramme in der Fingerprint-Applikation können keine Aussagen über die Sprungantwort des geschlossenen Regelkreises getroffen werden, da keine Informationen über die Übertragungsfunktion der Hardwarestrecke vorliegen. Durch eine Möglichkeit die Strecke zu identifizieren könnte die Sprungantwort näherungsweise berechnet und eine Überprüfung der Stabilität des geschlossenen Regelkreises vorgenommen werden.

Die aufgezeigten Punkte zeigen, dass es noch Möglichkeiten gibt die Funktionalität der Fingerprint-Applikation durch weitere studentische Projekte zu verbessern und somit das regelungstechnische Verständnis von Studenten zu vertiefen.

6 Literaturverzeichnis

Literatur

- [1] S. Zacher, *Drei-Bode-Plots-Verfahren für Regelungstechnik: Ein universelles Stabilitätskriterium für stabile und instabile Regelstrecken*, 1. Aufl. Wiesbaden: Springer Vieweg, 2020.
- [2] A. Träger, *S7-Toolbox for use with MATLAB | Softwaretreiber Simatic S7 | Softwaretreiber*. [Online]. Verfügbar unter: <https://www.traeger.de/softwaretreiber-simatic-s7/s7-toolbox-for-use-with-matlab.html> (Zugriff am: 17. April 2021).
- [3] *How do I calculate the amplitude ratio and phase lag for two sinusoidal signals in MATLAB? - MATLAB Answers - MATLAB Central*. [Online]. Verfügbar unter: <https://de.mathworks.com/matlabcentral/answers/91647-how-do-i-calculate-the-amplitude-ratio-and-phase-lag-for-two-sinusoidal-signals-in-matlab> (Zugriff am: 17. April 2021).