



FRIEDRICH-SCHILLER- UNIVERSITÄT JENA

Ein vereinfachtes Modell der Blutdruckregulation des
Menschen als Regelkreis

PROJEKTARBEIT

Friedrich-Schiller-Universität Jena
Fakultät für Mathematik und Informatik
Matthias-Schleiden-Institut

eingereicht von Lucas Dietrich

betreut von Dr.-Ing. habil. Thomas Hinze

Jena, 6. Juli 2024

Zusammenfassung

Diese Projektarbeit, eingereicht von Lucas Dietrich an der Friedrich-Schiller-Universität Jena, untersucht die Blutdruckregulation des menschlichen Körpers unter Verwendung eines Regelkreismodells. Ziel der Arbeit ist es, die wesentlichen Parameter zu identifizieren, die die Druckbildung in den Blutgefäßen beeinflussen, und darauf basierend einen Regelkreis zur Blutdruckregulation zu entwickeln.

Die Arbeit beginnt mit einer Einführung in die physiologischen Grundlagen des menschlichen Gefäßsystems und des Herzens. Der Blutdruck, seine Messmethoden und Einflussfaktoren werden detailliert beschrieben. Besonders hervorzuheben sind der Radius der Gefäße, der Widerstand und die Viskosität des Blutes, welche maßgeblich den Blutdruck beeinflussen.

Anschließend wird die Regelungstechnik als theoretischer Rahmen vorgestellt. Ein Regelkreis wird definiert als ein System, das durch äußere Einflüsse an einen Soll-Zustand angepasst wird. Dies wird mit der Blutdruckregulation verglichen, bei der der aktuelle Blutdruck (Ist-Zustand) kontinuierlich mit einem gewünschten Wert (Soll-Zustand) verglichen und entsprechend angepasst wird.

Die Zielsetzung der Arbeit ist es, ein Modell zu entwickeln, das die wesentlichen Komponenten des Blutsystems und deren Interaktionen abbildet. Dieses Modell soll es ermöglichen, Simulationen durchzuführen, bei denen der Ist-Zustand des Blutdrucks auf den Soll-Zustand eingestellt wird. Hierfür werden verschiedene physiologische und pathologische Szenarien simuliert, darunter kurzzeitiger Stress, Schocksituationen, Gefäßverengungen und Niereninsuffizienz.

Die Arbeit schließt mit einer Diskussion der Simulationsergebnisse und einem Ausblick auf zukünftige Forschungsarbeiten, die auf diesem Modell aufbauen könnten. Die Ergebnisse zeigen, dass das entwickelte Modell effektiv verschiedene Blutdruckregulationsszenarien abbilden kann und somit ein nützliches Werkzeug für weitere Studien und Anwendungen in der Medizin und Biomedizintechnik darstellt.

Diese Projektarbeit trägt damit wesentlich zum Verständnis der Blutdruckregulation und zur Entwicklung neuer Ansätze in der Regelungstechnik bei, die auf biologischen Systemen basieren.

Inhaltsverzeichnis

Zusammenfassung	2
1 Einführung	5
1.1 Hintergrund	5
1.2 Zielsetzung	6
1.3 Struktur der Arbeit	6
2 Physiologische Grundlagen	7
2.1 Aufbau und Funktion des menschlichen Gefäßsystems	7
2.2 Aufbau und Funktion des Herzens	8
2.3 Der Blutdruck	9
2.3.1 Methoden zur Ermittlung des Blutdrucks	10
2.3.2 Die Blutdruckkurve und ihre Eigenschaften	11
2.4 Einflussfaktoren auf den Blutdruck	12
2.4.1 Der Radius und Widerstand als Einflussfaktoren	12
2.4.2 Die Viskosität als Einflussfaktor	13
2.4.3 Weitere Einflussfaktoren	13
3 Grundlagen der Regelungstechnik	14
3.1 Begriffe der Regelungstechnik	14
3.2 Allgemeiner Aufbau eines Regelkreises	15
3.2.1 Eingrößensystem	15
3.2.2 Mehrgrößensystem	16
4 Regelkreis zur Blutdruckregulation	16
4.1 Konzept des Regelkreises	16
4.2 Implementierung des Regelkreises	17
4.2.1 Eine erste Blutdruckkurve	17
4.2.2 Das Herz	19
4.2.3 Das Gefäßsystem	22
4.2.4 Der Sensor	27
4.2.5 Der Regelkreis	29
4.2.6 Verknüpfungen der Komponenten	31

4.3	Anwendung als Nutzer	31
4.4	Anwendungsbeispiele	33
4.4.1	Mit konstanten Parametern	34
4.4.2	Mit variablen Parametern	35
4.4.3	Als Regelkreis	37
5	Simulationsstudien	38
5.1	Kurzzeitiger Stress	38
5.2	Schocksituation	41
5.3	Verengung der Blutgefäße	42
5.4	Niereninsuffizienz	45
6	Diskussion und Ausblick	46
6.1	Diskussion der Ergebnisse	46
6.2	Schlusswort und Ausblick	49
	Literaturverzeichnis	52
	Abkürzungsverzeichnis	53
	Abbildungsverzeichnis	54
	Tabellenverzeichnis	55
	Quelltextverzeichnis	56
	Anlagen	57
	Eigenständigkeitserklärung	66
	Sonstiges	67

1 Einführung

1.1 Hintergrund

Selbst ein lebender Organismus wie der Mensch besteht aus zahlreichen Komponenten, die zusammenarbeiten müssen, damit Funktionen ausgeführt werden können und um den Organismus insgesamt am Leben zu erhalten. Eine besonders bedeutende Rolle spielt hierbei das Gefäßsystem, auch bekannt als Blutsystem, in einem biologischen Körper.

Das Blutsystem des Körpers dient dazu, die Kommunikation zwischen verschiedenen Organen zu gewährleisten und eine Vielzahl von Botenstoffen von einem Ort zum anderen zu transportieren. Mit anderen Worten fungiert das Blutsystem als eine Art Nachrichtensystem, vergleichbar mit dem Rohrpostsystem in Krankenhäusern, um eine effiziente Kommunikation zu ermöglichen.

Ein wichtiger Bestandteil, der es dem Blut ermöglicht, in den Gefäßen zu zirkulieren, ist der Druck, der innerhalb der Gefäße aufgebaut wird und durch verschiedene Parameter beeinflusst werden kann.

Der Mensch kann als ein System betrachtet werden, das aus verschiedenen Komponenten besteht, welche das Gesamtsystem am Laufen halten, wobei das Gefäßsystem eine zentrale Rolle spielt. Das bedeutet zum Beispiel, dass der Druck in den Blutgefäßen sich an unterschiedlichste Situationen anpassen muss, um die Versorgung der Organe mit benötigten Nährstoffen, Sauerstoff und anderen Transportstoffen zu gewährleisten.

Die Regelungstechnik hat die Aufgabe, einen zeitlich veränderlichen Prozess so zu beeinflussen, dass er in einer bestimmten Weise abläuft. Ein gutes Beispiel aus [15] ist die Regelung der Raumtemperatur. Dabei soll sich die Temperatur in einem Raum den gegebenen Umständen anpassen: Bei höherer Außentemperatur soll die Raumtemperatur gesenkt werden, und bei niedriger Außentemperatur entsprechend erhöht werden. Es handelt sich also um ein System, das durch äußere Einflüsse an den Soll-Zustand angepasst werden soll.

Genau dieses Prinzip verbirgt sich auch hinter der Regulation des Blutdrucks. Der Blutdruck ist ein zeitlich veränderlicher Prozess, der unter anderem durch äußere Einflüsse beeinflusst wird. Um den Blutdruck anzupassen, wird der Ist-Zustand mit dem Soll-Zustand des Systems verglichen und entsprechend eingestellt.

1.2 Zielsetzung

Das Ziel ist zunächst, die Parameter zu identifizieren, die einen maßgeblichen Einfluss auf die Druckbildung in den Blutgefäßen haben. Diese identifizierten Parameter sollen verwendet werden, um einen Regelkreis zur Regulation des Blutdrucks zu entwickeln. Am Ende sollen die wesentlichen Komponenten des Blutsystems, die für die Blutdruckregulation erforderlich sind, in dem Modell enthalten und manipulierbar sein.

Dem Modell sollen dann Ist- und Soll-Parameter übergeben werden, damit Simulationen durchgeführt werden können, bei denen sich der Ist-Zustand auf den Soll-Zustand einstellen soll.

1.3 Struktur der Arbeit

Die vorliegende Arbeit beschäftigt sich zunächst mit den grundlegenden Aspekten, einschließlich der physiologischen Grundlagen des Blutsystems, seiner Struktur, der wichtigsten Komponenten und ihrer Funktionsweise. Es werden auch wichtige Punkte genannt, die für den weiteren Verlauf der Arbeit von Bedeutung sind.

Des Weiteren wird ein Einblick in die Grundlagen der Regelungstechnik gegeben. Es wird erläutert, wie ein Regelkreis aufgebaut ist und welche die wichtigsten Komponenten sowie deren Aufgaben in diesem Gesamtsystem sind.

Nach der Erläuterung der Grundlagen der Regelungstechnik wird auf die Implementierung des Regelkreises eingegangen und erklärt, wie diese zusammengesetzt ist und funktioniert. Dies umfasst auch eine kurze Bedienungsanleitung, die beschreibt, wie ein Nutzer mit dem Modell interagieren kann.

Anschließend werden mit dem implementierten Regelkreis mehrere Simulationen durchgeführt, die verschiedene physiologische Situationen abbilden. Abschließend werden diese Simulationen und Ergebnisse diskutiert, begleitet von einem kleinen Fazit und einem Ausblick in die Zukunft.

2 Physiologische Grundlagen

2.1 Aufbau und Funktion des menschlichen Gefäßsystems

Das Gefäßsystem lässt sich grundsätzlich in zwei Kreisläufe unterteilen: den Lungenkreislauf, auch kleiner Kreislauf genannt, und den Körperkreislauf, auch bekannt als großer Kreislauf [2, S. 224]. Die Aufgabe des Lungenkreislaufs besteht darin, sauerstoffarmes Blut aufzunehmen und es zu sauerstoffreichem Blut umzuwandeln, das anschließend in den Körperkreislauf zurückgeführt wird [14, S. 662].

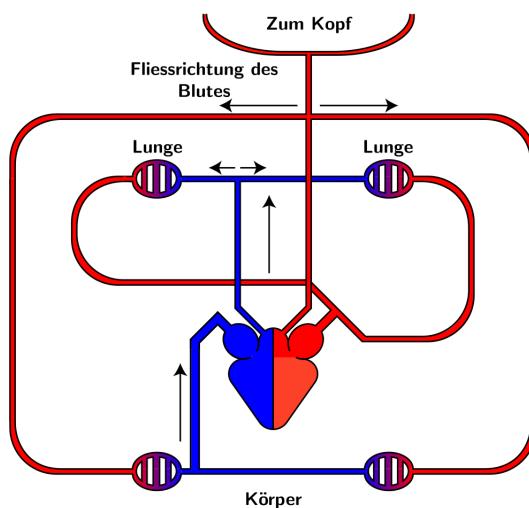


Abbildung 1: Schematische Darstellung des Gefäßsystems und seiner zwei Kreisläufe

Quelle: [1]

Anhand von Abbildung 1 lässt sich der generelle Aufbau des Gefäßsystems erkennen sowie wichtige Bestandteile, die über das Gefäßsystem verbunden sind, wie die bereits erwähnte Lunge und das Herz. Die roten Bereiche stellen dabei das sauerstoffreiche Blut dar, während die blauen Bereiche den Abschnitt kennzeichnen, in dem das Blut sauerstoffarm ist. Bekanntermaßen spielt das Herz eine zentrale Rolle im Blutsystem, indem es sauerstoffarmes Blut zur Lunge und sauerstoffreiches Blut zurück in den Körperkreislauf pumpt [2, 14]. Somit kann das Herz auch als Schnittstelle zwischen dem kleinen und dem großen Blutkreislauf betrachtet werden.

Ebenso wichtig ist die Information, dass das Gefäßsystem aus verschiedenen Arten von Gefäßen besteht. Diese Gefäßarten unterscheiden sich in ihrer Größe, Länge, Wanddicke sowie in ihrer Verzweigung und Anzahl [2, S. 224].

	Aorta	Arterie	Arteriole	Kapillare	Venole	Vene	V. cava
Wanddicke	2000	1000	30	1	2	500	1500
Lumen	20000	4000	20	8	20	5000	30000

Tabelle 1: Gefäßarten und ihre Kenngrößen in μm . Der Begriff Lumen steht für die lichte Weite eines Hohlraums bzw. den Innendurchmesser der jeweiligen Blutgefäße.

Entsprechend Tabelle 1 sind die Gefäße hintereinandergeschaltet. Die Aorta mündet in die Arterien, die Arterien in die Arteriolen, die Arteriolen in die Kapillaren, die Kapillaren in die Venolen, die Venolen in die Venen und die Venen schließlich in die V. cava, auch bekannt als Hohlvene [2, S. 166].

Die Aorta hingegen ist noch mit dem linken Herzen verbunden, von wo aus das sauerstoffreiche Blut gepumpt wird, während das sauerstoffarme Blut aus der V. cava in das rechte Herz fließt [2, 14].

2.2 Aufbau und Funktion des Herzens

Das Herz ist ein großer Muskel, der zunächst in eine rechte und eine linke Seite unterteilt ist. Jede Herzseite besteht dabei aus zwei Kammern: einem Vorhof und einer Herzkammer, was insgesamt vier Kammern ergibt [2, S. 116].

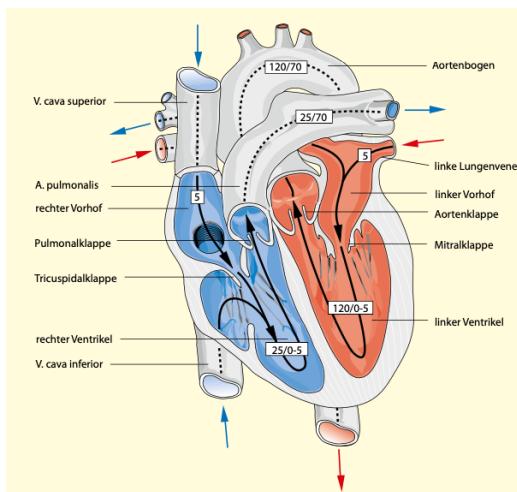


Abbildung 2: Schematische Darstellung des Herzens

Quelle: [14]

Das sauerstoffarme Blut fließt zunächst in den rechten Vorhof, von dort gelangt es in die rechte Herzkammer. Von der rechten Herzkammer wird das Blut zur Lunge gepumpt, wo es mit Sauerstoff angereichert wird, bevor es in den linken Vorhof zurückfließt. Nachdem das Blut den linken Vorhof passiert hat, gelangt es in die linke Herzkammer, von wo aus es wieder ins Gefäßsystem gepumpt wird.

Damit das Blut ordnungsgemäß zirkulieren kann, durchläuft der Herzzyklus vier Phasen.[2, 14].

Insgesamt besteht der Herzzyklus aus folgenden Phasen, die in beiden Seiten des Herzens stattfinden, wobei sich der systolische Druck in den Herzkammern wesentlich unterscheidet [2, 14].

1. **Füllungsphase (Diastole):** In dieser Phase sind die Herzkammern entspannt und werden mit Blut aus den Vorhöfen gefüllt.
2. **Anspannungsphase (Systole):** Die Kammern beginnen sich zu kontrahieren, aber die Klappen sind noch geschlossen, sodass kein Blut ausfließt.
3. **Austreibungsphase (Systole):** Die Kammern kontrahieren vollständig, die Klappen öffnen sich und das Blut wird in die Gefäße gepumpt.
4. **Entspannungsphase (Diastole):** Die Kammern entspannen sich wieder und bereiten sich auf die nächste Füllungsphase vor.

2.3 Der Blutdruck

Der Blutdruck ist einer der wichtigsten Parameter, der bei gesundheitlichen Beschwerden in der Regel als erstes überprüft wird. Dieser Wert gibt einen guten Hinweis darauf, ob die Beschwerden auf Herz-Kreislauf-Probleme oder andere ernsthafte Gesundheitszustände hinweisen könnten.

Mit Hilfe dieses Wertes kann auch überprüft werden, ob angewandte Therapien anschlagen oder nicht, ohne hohe Kosten zu verursachen.

Der Blutdruck kann jedoch von Person zu Person leicht variieren und hängt von verschiedenen Umständen wie dem Alter, Geschlecht, körperlichem Zustand und der Ernährung ab. Daher ist seine Interpretation nicht einfach und sollte unter Berücksichtigung dieser Faktoren erfolgen [2, 17].

2.3.1 Methoden zur Ermittlung des Blutdrucks

Um den Blutdruck zu ermitteln, gibt es unterschiedliche Methoden, wobei zwischen direkten und indirekten Methoden unterschieden wird.

Bei der **direkten Methode** wird der Blutdruck über einen direkten Zugang im Gefäß gemessen, indem eine Kanüle oder ein Sensor eingeführt wird, der mit einem Manometer verbunden ist [2, 4]. Mit anderen Worten besteht eine direkte Verbindung zum Blut. Diese Methode wird besonders auf Intensivstationen oder während Operationen verwendet, da sie als sehr präzise gilt und den Druck direkt auf einem angeschlossenen Gerät abbildet [2, 4].

Das Prinzip der **indirekten Methode** funktioniert ganz anders als bei der direkten Methode. Hierbei wird eine Manschette am Oberarm angelegt, die die Arterie abdrückt. Der Druck in der Manschette wird solange erhöht, bis keine Strömungsgeräusche mehr über das Stethoskop hörbar sind [4].

Von diesem Punkt aus wird nun langsam der Druck der Manschette reduziert, wodurch die Arterie wieder langsam geöffnet wird. Sobald der Druck in der Manschette den arteriellen Druck unterschreitet, entstehen Strömungsgeräusche, die über das Stethoskop hörbar sind. Dieser Punkt, an dem der Manschettendruck den Druck in der Arterie unterschreitet, wird auch als systolischer Druck bezeichnet [2], siehe Abbildung 3.

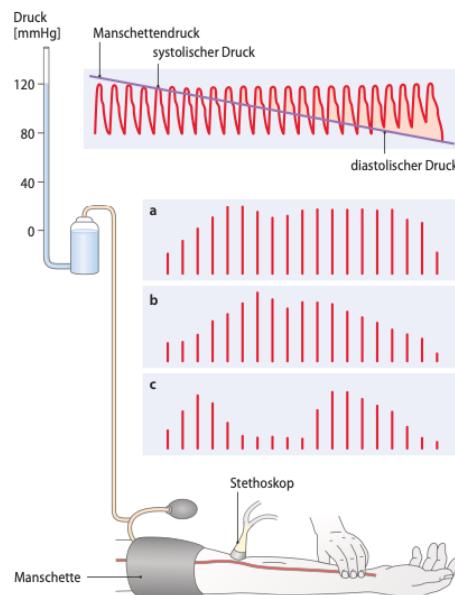


Abbildung 3: Prinzip der indirekten Methode nach Riva-Rocci

Quelle: [2]

Nach diesem Punkt wird der Druck der Manschette weiter reduziert, wodurch sich die Strömungsgeräusche weiter verändern können, bis sie plötzlich dunkler und leiser werden [2]. Der Druck, bei dem diese plötzliche Veränderung der Geräusche auftritt, wird auch als diastolischer Druck bezeichnet, siehe Abbildung 3.

2.3.2 Die Blutdruckkurve und ihre Eigenschaften

Bei genauerer Betrachtung der Druckkurve mit dem sich reduzierenden Manschettendruck in Abbildung 3 fällt auf, dass die Druckkurve einen spezifischen und auffälligen Verlauf aufweist, der genauer betrachtet werden sollte.

Eine Druckkurve weist bestimmte Kenngrößen auf, zu denen der systolische Druck, der diastolische Druck und die Druckamplitude zählen.

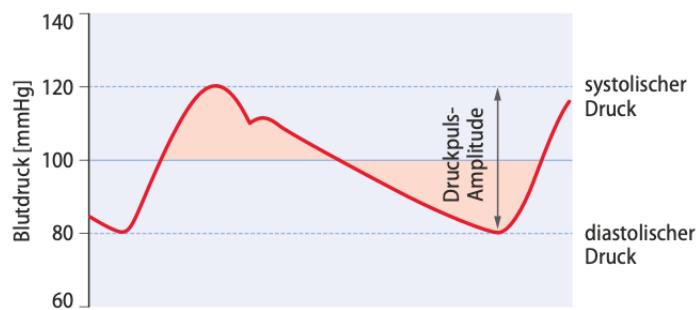


Abbildung 4: Typische Blutdruckkurve mit wesentlichen Eigenschaften

Quelle: [2, S. 234]

Aus Abbildung 4 lässt sich sehr gut erkennen, um welche Kenngrößen es sich handelt und was sie beschreiben.

Der systolische Druck beschreibt den maximalen Druck in einem Gefäß, während der diastolische Druck den minimalen Druck beschreibt.

Die Druckamplitude ist der Abstand zwischen Maximum und Minimum bzw. zwischen systolischem und diastolischem Druck. In diesem Fall beträgt die Amplitude 40 mmHg. Dieser Wert ist jedoch nicht festgelegt, da die Amplitude durch verschiedene Einflussgrößen beeinflusst werden kann und entsprechend kleiner oder größer ausfallen kann.

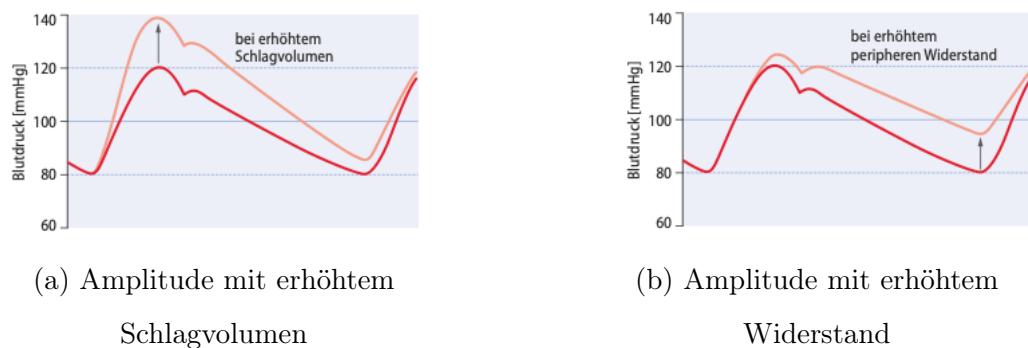


Abbildung 5: Blutdruckkurve mit veränderten Amplituden

Quelle: [2]

Auch nach der Klärung der Kenngrößen bleibt die Frage offen, warum die Blutdruckkurve eine so spezielle Form hat. Um dies zu verstehen, muss die Abfolge der Ereignisse betrachtet werden.

Das Blut wird vom Herzen in die Aorta gepumpt, bis am Ende der Systole, also während der Austreibungsphase des Herzzyklus, die Aortenklappe schließt. Das Schließen der Klappe führt zu einem leichten Rückfluss von Blut in Richtung der Klappe, was einen leichten Druckanstieg und die sogenannte Inzisur verursacht [2, S. 231]. Nach dem Schließen der Klappe sinkt der Druck weiter bis zum diastolischen Wert, bevor der Zyklus von Neuem beginnt und der Druck wieder bis zum systolischen Wert ansteigt.

2.4 Einflussfaktoren auf den Blutdruck

Damit die Versorgung des Körpers und seiner Bestandteile in jeder Situation gewährleistet werden kann, muss der Blutdruck entsprechend angepasst werden. Diese Anpassung erfolgt durch die Einstellung verschiedenster Einflussfaktoren, um den erforderlichen Blutdruck zu erreichen.

2.4.1 Der Radius und Widerstand als Einflussfaktoren

Ein wichtiger Einflussfaktor ist der Radius der Gefäße und der damit verbundene Widerstand. Gemäß Tabelle 1 haben die verschiedenen Arten von Gefäßen unterschiedliche Durchmesser.

Dadurch, dass der Radius der Gefäße bis hin zu den Kapillaren immer weiter reduziert wird, wird auch der Widerstand entsprechend beeinflusst [2].

Das lässt sich wie folgt vorstellen: Eine bestimmte Menge Blut wird vom Herzen in die Aorta gepumpt. Da diese Menge dann durch kleinere Arterien fließen muss, erhöht sich der Widerstand. Diese Situation wiederholt sich bis zu den Kapillaren, wo der Durchmesser am kleinsten und somit der Widerstand am höchsten ist.

Ein weiterer Einfluss auf den Widerstand ist die grundlegende Anatomie der Gefäße. Dies umfasst auch die Gefäßverzweigungen und die Längen der einzelnen Gefäße, die einen spürbaren Einfluss auf den Widerstand und damit indirekt auf den Blutdruck haben können.

2.4.2 Die Viskosität als Einflussfaktor

Das Blut ist keine homogene Flüssigkeit, sondern transportiert verschiedene Moleküle von Punkt A nach Punkt B. Daher kann auch die Fließeigenschaft oder Viskosität je nach Situation unterschiedlich sein.

Wenn das Blut dickflüssiger ist, erhöht dies den Widerstand im Blutfluss. Die Anpassung der Blutviskosität kann jedoch auch ein Schutzmechanismus sein, insbesondere bei Verletzungen mit Blutverlust, wo die erhöhte Viskosität als Schutzmechanismus fungieren kann. Auch eine niedrige Viskosität kann entscheidend sein, zum Beispiel bei Bedarf an schneller Durchblutung, wie bei intensiver sportlicher Aktivität. In solchen Fällen ist es hilfreich, wenn das Blut eine geringe Viskosität aufweist, damit die Sauerstoffversorgung der verschiedenen Gewebetypen gewährleistet werden kann.

2.4.3 Weitere Einflussfaktoren

Weitere Einflussfaktoren, die den Blutdruck beeinflussen, sind das Gesamtvolumen des Blutes, die Herzfrequenz und das Schlagvolumen.

Das Gesamtvolumen des Blutes hat einen direkten Einfluss auf den Blutdruck. Ein erhöhtes Blutvolumen kann zu einem erhöhten Füllungsdruck in den Gefäßen und im Herzen führen, was das Schlagvolumen und das Herzzeitvolumen erhöht. Dies resultiert in einem höheren Blutdruck. Im Gegensatz dazu führt ein verminderter Blutvolumen, etwa durch Dehydratation oder Blutverlust, zu einem reduzierten venösen Rückstrom zum Herzen, was das Schlagvolumen und das Herzzeitvolumen verringert und somit den Blutdruck senkt [6].

Die Herzfrequenz, also die Anzahl der Herzschläge pro Minute, hat einen direkten Einfluss

auf das Herzzeitvolumen und damit auf den Blutdruck. Eine erhöhte Herzfrequenz führt zu einem gesteigerten Herzzeitvolumen, da mehr Blut pro Minute durch den Kreislauf gepumpt wird. Dies kann den Blutdruck erhöhen, insbesondere bei körperlicher Aktivität oder unter Stress. Eine niedrigere Herzfrequenz hingegen kann das Herzzeitvolumen und den Blutdruck verringern, was oft während Ruhephasen oder im Schlaf beobachtet wird [2].

Das Schlagvolumen ist die Menge Blut, die das Herz mit jedem Schlag auswirft. Ein höheres Schlagvolumen führt zu einem höheren Herzzeitvolumen und damit zu einem höheren Blutdruck. Umgekehrt führt ein geringeres Schlagvolumen zu einem niedrigeren Herzzeitvolumen und Blutdruck. Das Schlagvolumen kann durch Training, Herzkrankheiten, den Flüssigkeitsvolumenstatus und andere Faktoren beeinflusst werden [2].

3 Grundlagen der Regelungstechnik

Wie bereits erwähnt, beschäftigt sich die Regelungstechnik damit, Prozesse gezielt zu manipulieren, damit sie einen bestimmten Verlauf über die Zeit nehmen.

Für ein besseres Verständnis sollen in diesem Kapitel die verschiedenen Begriffe und Komponenten eines Regelkreises erläutert werden sowie deren Verbindung zueinander.

3.1 Begriffe der Regelungstechnik

In der Regelungstechnik ist ein Regelkreis ein dynamisches System, das zielgerichtet gesteuert wird [15].

Formal betrachtet besteht ein dynamisches System aus einem Tripel (T, X, Φ) , wobei T die Zeitmenge, X die Zustandsmenge und Φ der Phasenraum ist. Der Phasenraum Φ umfasst alle möglichen Zustände, die das System einnehmen kann [12].

Mit anderen Worten ist ein dynamisches System eine Abbildung $\Phi : X \times T \rightarrow X$, die einen Zustand zu einem Zeitpunkt auf einen anderen Zustand abbildet. Dabei hängen zukünftige Zustände immer vom aktuellen Zustand ab und sind nicht zufällig [12].

Allerdings ist nicht jede Abbildung mit dem Tripel (T, X, Φ) automatisch ein dynamisches System; es müssen zusätzlich die *Identitätseigenschaft* $\Phi(0, X) = X$ und die *Halbgruppen-eigenschaft* $\Phi(s + t, X) = \Phi(s, \Phi(t, X))$ erfüllt sein [9].

Ein System besitzt einen **Sensor**, der den Ist-Wert bzw. die Regelgröße im System iden-

tifiziert [15].

Dann gibt es einen **Regler**, der die Diskrepanz zwischen Ist- und Soll-Wert erkennt und diese Information an das Stellglied weiterleitet, das daraufhin die Stellgröße anpasst [15]. Das **Stellglied** kümmert sich darum, dass die Regelgröße vom Ist-Wert auf den geforderten Soll-Wert eingestellt wird. Die Regelgröße wird jedoch nicht nur vom Stellglied beeinflusst, sondern auch von Störgrößen, die den Ist-Wert der Regelgröße beeinflussen können [15].

3.2 Allgemeiner Aufbau eines Regelkreises

3.2.1 Eingrößensystem

Ein Eingrößensystem ist ein Regelkreis, bei dem es nur eine Regelgröße gibt, die von einer einzigen Stellgröße gezielt beeinflusst wird.

Allgemein ist ein Regelkreis so aufgebaut, dass die Führungsgröße oder der Soll-Wert von außen in das System eingeführt wird. Dieser wird dann mit dem Ist-Wert der Regelgröße verglichen. Falls sich Soll- und Ist-Wert unterscheiden, kommt das Stellglied zum Einsatz, um die Regelgröße auf den Soll-Wert einzustellen.

Dann wird der Ist-Wert der Regelgröße von Sensoren oder dem Messglied gemessen und an den Regler weitergegeben, woraufhin der Zyklus von Neuem beginnt.

Abbildung 6 zeigt erneut den beschriebenen Kontext eines Regelkreises und wie die verschiedenen Komponenten miteinander interagieren.

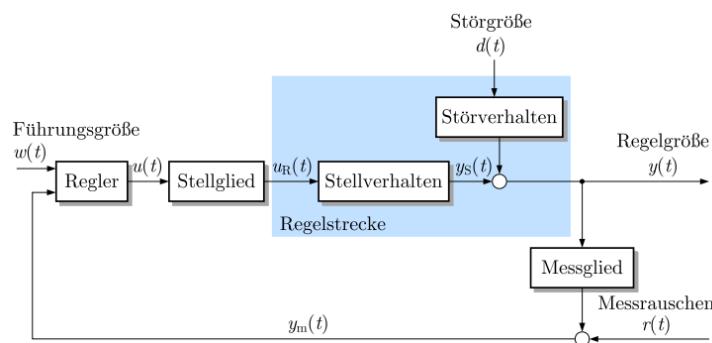


Abbildung 6: Allgemeine Darstellung eines Regelkreises

Quelle: [15, S. 5]

3.2.2 Mehrgrößensystem

Weiterhin gibt es auch Systeme, in denen mehrere miteinander verknüpfte Stellgrößen existieren, die eine oder auch mehrere Regelgrößen beeinflussen können [16].

Ein solcher Regelkreis wird als Mehrgrößensystem bezeichnet. Innerhalb des Systems gibt es Verzweigungen, die die Stellgrößen miteinander verknüpfen [16].

Für ein besseres Verständnis kann folgende Darstellung mit mehreren Stellgrößen und Regelgrößen betrachtet werden.

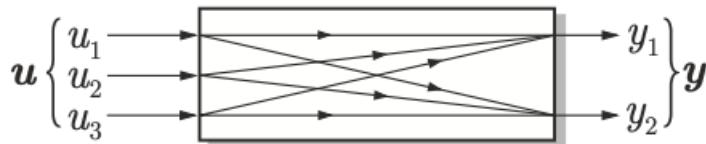


Abbildung 7: Prinzip des Stellglieds eines Mehrgrößensystems

Quelle: [16]

Die Abbildung zeigt, wie drei gekoppelte Stellgrößen u_1 , u_2 und u_3 zwei gekoppelte Regelgrößen y_1 und y_2 beeinflussen. Im Kontext eines ganzen Regelkreises fungiert das Stellglied als Bindeglied, das die Stellgrößen miteinander verknüpft, um einen gebündelten Einfluss auf die Regelgrößen auszuüben.

4 Regelkreis zur Blutdruckregulation

4.1 Konzept des Regelkreises

Der Regelkreis zur Blutdruckregulation vergleicht den Ist-Wert $p(t)$ mit dem Soll-Wert $s(t)$ in dem Quadrat Regelabweichung, siehe Abbildung 8. Diese Regelabweichung $e(t)$ wird berechnet, indem der Ist-Wert vom Soll-Wert subtrahiert wird.

$$e(t) = s(t) - p(t) \quad (1)$$

Nachdem die Regelabweichung bestimmt ist, wird dieser Wert dem Untersystem zugeführt. Das Untersystem umfasst die wesentlichen Komponenten, die den Blutdruck maßgeblich beeinflussen, und nimmt den neuen Soll-Wert entgegen.

Im Untersystem sind anpassbare Größen vorhanden, die zusammen den Gesamteinfluss

auf den Blutdruck repräsentieren. Diese Regelgrößen werden vom Stellglied entsprechend modifiziert und an die Regelstrecke weitergeleitet, wo eine Simulation mit den aktualisierten Werten durchgeführt wird.

Der Sensor erfasst die Ist-Werte aus der Simulation der Regelstrecke und gibt sie aus, um sie mit einem neuen Soll-Wert zu vergleichen, womit der Zyklus von Neuem beginnt.

Der eben beschriebene Vorgang ist nochmals visuell in der folgenden Abbildung 8 dargestellt.

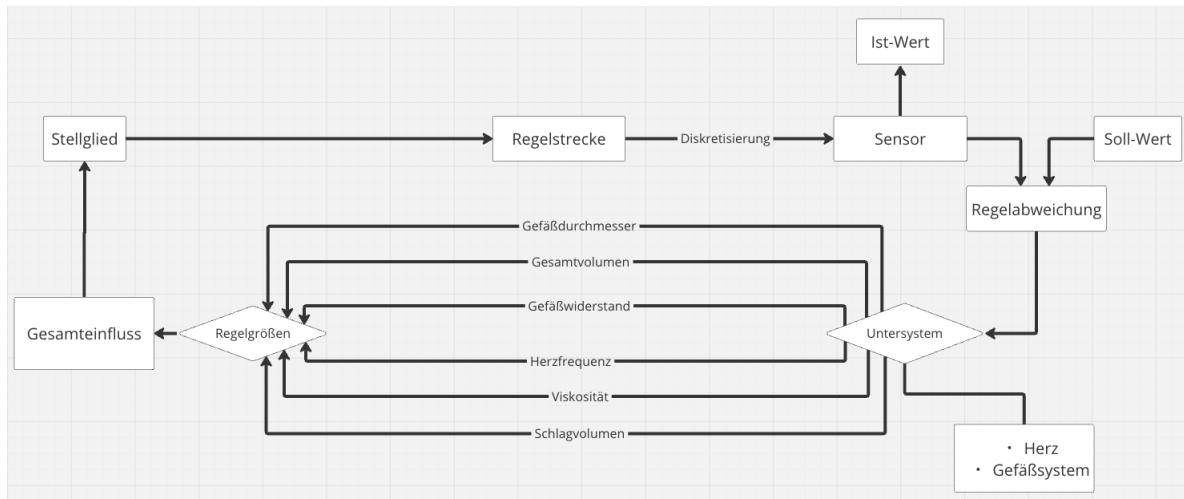


Abbildung 8: Konzept des Regelkreises zur Blutdruckregulation

4.2 Implementierung des Regelkreises

Für die Implementierung wurde Python in der Version 3.12.3 verwendet. Zusätzlich wurden die importierbaren Module NumPy [8] und Scipy.signal [21] für wichtige Berechnungen sowie Matplotlib [10] zum Generieren von Plots verwendet.

Die Idee besteht darin, alle notwendigen Komponenten des Regelkreises jeweils in separate Python-Programme zu implementieren und miteinander zu verbinden.

Um eine Grundlinie der Blutdruckkurve zu erstellen, ist zunächst eine einfache Sinuskurve ein guter Ansatz.

4.2.1 Eine erste Blutdruckkurve

Um eine möglichst realistische Blutdruckkurve zu implementieren, ist eine Fourier-Transformation erforderlich, die zwei einfache Sinuswellen mit unterschiedlichen Parametern addiert [13].

$$\begin{aligned} p_1 &= A \cdot \sin \left(2 \cdot \pi \cdot \frac{\text{Herzfrequenz}}{60} \cdot t \right) \\ p_2 &= A \cdot 0.63 \cdot \sin \left(4 \cdot \pi \cdot \frac{\text{Herzfrequenz}}{60} \cdot t + \frac{2}{\pi} \right) \end{aligned} \quad (2)$$

Hierbei repräsentiert A die Amplitude, die später den systolischen Blutdruck darstellt, während der Parameter *Herzfrequenz* angibt, wie oft das Herz pro Minute schlägt. Die Variable t symbolisiert die Zeit.

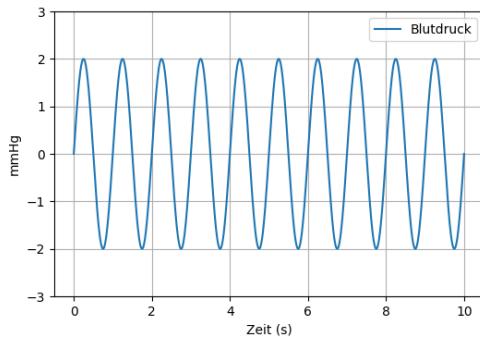
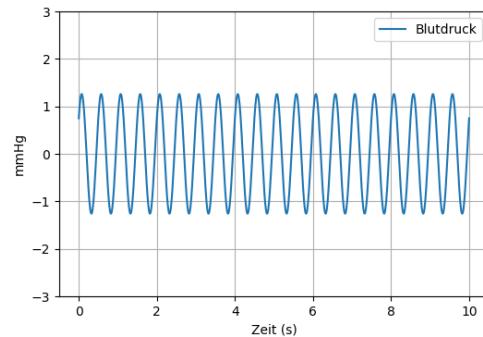
(a) Erste Sinuswelle p_1 (b) Zweite Sinuswelle p_2

Abbildung 9: Zwei separate Sinuswellen für die Blutdruckkurve

Wenn die beiden Sinuswellen addiert und für eine korrekte Darstellung von systolischem und diastolischem Druck normalisiert werden, sieht das Ergebnis wie folgt aus:

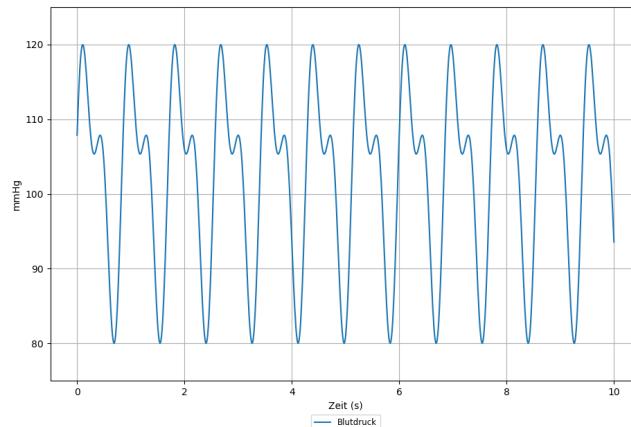


Abbildung 10: Erste Blutdruckkurve auf 120/80 mmHg normalisiert bei einer unveränderten Herzfrequenz (Puls) von 70 Schlägen pro Minute über 10 Sekunden

Diese Implementierung einer einfachen Sinuswelle, die ohne viele Einstellungen auskommt, zeigt annähernd die charakteristischen Eigenschaften einer Blutdruckkurve. Basierend auf dieser einfachen mathematischen Darstellung des Blutdrucks können die anderen Komponenten implementiert werden.

4.2.2 Das Herz

Um einen Blutdruck zu simulieren, wurde die physiologische Arbeitsweise und der Aufbau des Herzens im Programm `heart.py` implementiert.

In der Implementierung wurde das Herz in die rechte und linke Seite unterteilt, wobei jeweils die Herzkammer als eine eigene Funktion implementiert wurde.

Als erstes wurde die linke Herzkammer in der Funktion `leftVentricle` implementiert, um diese Herzkammer zu simulieren. In der Simulation werden im ersten Schritt die verschiedenen Effekte wie Viskosität und Gesamtvolumen definiert, die den Blutdruck beeinflussen. Diese Effekte werden skaliert, um einen realistischen Einfluss zu gewährleisten. Dann wird über eine einfache Schleife, die über die Zeit läuft, Schritt für Schritt der Blutdruck zu jedem Zeitpunkt bestimmt. Dafür werden zunächst zwei Hilfsfunktionen aufgerufen, die zum Einsatz kommen, wenn die Werte für enddiastolisches und endsystolisches Volumen dynamisch über die Zeit angepasst werden sollen. Hierbei werden optionale Parameter als Arrays zugewiesen, und entsprechend den Zeittabellen `ctEDV` und `ctESV` werden die Werte für enddiastolisches und endsystolisches Volumen angepasst. Danach wird die Elastizität der Kammer basierend auf der Herzfrequenz und dem aktuellen Zeitpunkt bestimmt.

Im nächsten Schritt wird das Blutvolumen auf das enddiastolische Volumen (kurz `edv`) gesetzt. Anschließend wird die Änderung des Blutvolumens im linken Ventrikel basierend auf dem Schlagvolumen (`strokeVolume`), der Elastizität und dem vorherigen Blutvolumen bestimmt.

Nach der Berechnung des Blutvolumens kann dann der Blutdruck basierend auf der Elastizität der Herzkammer, dem Blutvolumen in der Herzkammer, dem Gesamtvolumen als `volumeEffect` und der Viskosität als `viscosityEffect` bestimmt werden.

```

1 def leftVentricle(self, ctEDV=[], newEDV=[], ctESV=[], newESV=[], shift=0):
2
3     viskosityEffect = self.viscosity / 100
4     volumePressureConstant = 0.001
5     volumeEffect = volumePressureConstant * self.totalVolume
6     currentEDV = self.edv
7     currentESV = self.esv
8     currSV = self.strokeVolume
9
10    for i in range(0, len(self.time)):
11        t = self.time[i]
12
13        currentEDV = self.updateParameter(t, ctEDV, newEDV, currentEDV)
14        currentESV = self.updateParameter(t, ctESV, newESV, currentESV)
15        currSV = currentEDV - currentESV
16        elasticity = 1 + np.sin(2 * np.pi * self.heartRate * (t - shift) / 60)
17
18        if i == 0:
19            self.bloodVolume_LV[i] = currentEDV
20        else:
21            dVdt = currSV - elasticity * (self.bloodVolume_LV[i-1] - currentESV)
22            self.bloodVolume_LV[i] = self.bloodVolume_LV[i-1] + dVdt * self.dt
23
24        self.bloodPressure_LV[i] = elasticity * (self.bloodVolume_LV[i] - currentESV) +
volumeEffect + viskosityEffect

```

Quelltext 1: Implementierte Funktion für die linke Herzkammer

Die Funktion für die rechte Herzkammer unterscheidet sich in drei Zeilen leicht. Zum einen wird der Parameter ‘shift=-0.5‘ verwendet, um den Druckverlauf zu verschieben. Dies geschieht, weil die rechte Herzkammer früher als die linke Kammer kontrahiert. Die weiteren Unterschiede betreffen die Namen der Arrays, in denen die Werte für das Blutvolumen in der Kammer und die Blutdruckwerte gespeichert werden. Nähere Details dazu finden sich im Quelltext 1 und 6¹.

Um das Herz nun zu simulieren, müssen die entsprechenden Parameter definiert und der Herzklasse übergeben werden.

Die zu definierenden Parameter umfassen die Viskosität, die Herzfrequenz, das Schlagvolumen, das endsystolische Volumen, das enddiastolische Volumen, das Gesamtvolumen und die maximale Zeit.

Das Schlagvolumen wird durch die Differenz $edv - esv$ festgelegt [3].

¹siehe Anhang, Quelltext 6

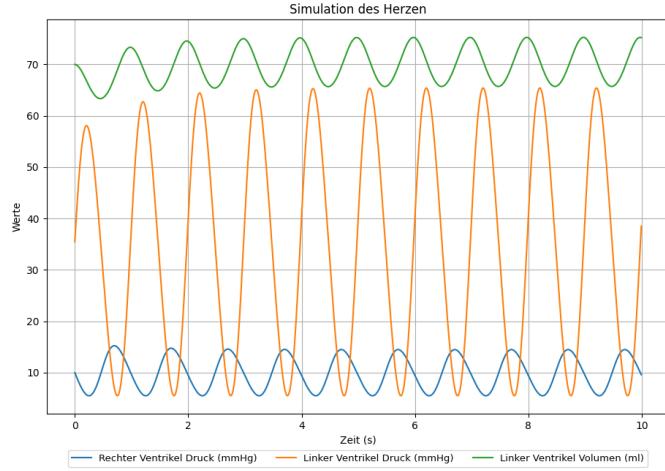


Abbildung 11: Simulation des Herzens mit Viskosität: 50; Herzfrequenz: 60;
edv: 70; esv: 40; Schlagvolumen: $edv - esv$; Gesamtvolumen: 5000

Man kann beobachten, wie sich der Druck im Herzen zwischen seinem Maximum und Minimum bewegt. Diese Oszillationen entsprechen der Systole und Diastole des Herzzyklus. In der Diastole entspannt sich die Herzkammer, wodurch der Druck abnimmt. Sobald jedoch die Systole einsetzt, beginnt die Herzkammer sich zu kontrahieren. Während der Kontraktion steigt der Druck schnell an und übersteigt schließlich den Druck im angeschlossenen Gefäß, typischerweise der Aorta. Mit anderen Worten, wenn der Druck in der linken Herzkammer höher ist als der in der Aorta, öffnet sich die Klappe und das Blut wird in die Aorta gepumpt.

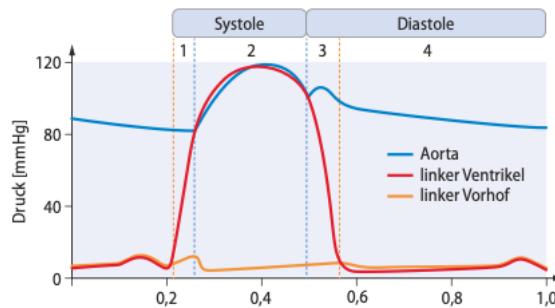


Abbildung 12: Druck in der Systole und Diastole

Quelle: [2]

Abbildung 12 zeigt den Druckverlauf in der linken Herzkammer. Die Phasen sind für die rechte Herzkammer identisch, jedoch ist der Druck dort generell nicht so hoch [2, S. 167].

4.2.3 Das Gefäßsystem

Die nächste wichtige Komponente für den Regelkreis ist das Gefäßsystem mit seinen physiologischen Eigenschaften, die implementiert und mit dem Herzen verknüpft wurden.

Um ein Gefäß zu modellieren, wurde als erster Schritt eine allgemeine Funktion namens `vessel` definiert. Diese Funktion erhält Parameter für den Radius, die Wanddicke und einen Skalierungsfaktor für das Lumen. Sie kann nun mit den entsprechenden Parametern für die jeweiligen Gefäßarten aufgerufen werden, wie sie in Tabelle 1 beschrieben sind.

Der Parameter `lumFactor` kann einen Wert von 0,1 bis 1 zugewiesen bekommen. Dieser Wert wird verwendet, um den Radius zu skalieren, wobei 1 100% und 0,1 10% des ursprünglichen Radius entsprechen.

Die anderen Parameter für Radius und Wanddicke werden der Funktion in Mikrometer (μm) übergeben und innerhalb der Funktion in Millimeter (mm) umgerechnet.

Die allgemeine Funktion `vessel` wird anschließend in einer neuen Funktion für die jeweilige Gefäßart aufgerufen, um die Parameter für die Modellierung zu generieren. Für die Modellierung der Aorta wurde eine Funktion `aorta` definiert, die wie folgt implementiert wurde.

```

1 def aorta(self, wallThickness=2000, lumFactor=1):
2     x, y, x1, y1, radius, wallThickness = self.vessel(self.aor_rad, wallThickness,
3             lumFactor)
4     return x, y, x1, y1, radius, wallThickness

```

Quelltext 2: Funktion zur Modellierung einer Aorta

Wie bereits beschrieben, werden der Funktion die entsprechenden Parameter übergeben, um die allgemeine Funktion für ein Gefäß aufzurufen und das spezifische Gefäß zu modellieren. Die Funktionen für die anderen Gefäße sind analog zur Implementierung im Quelltext 2 umgesetzt.

Nachdem alle Funktionen für jede Gefäßart implementiert wurden, wurde auch eine Hilfsfunktion namens `vesselPlotter` angelegt. Diese Funktion nimmt zwei Parameter an: ein Array bestehend aus sieben Skalierungsfaktoren für die Radien und ein Array mit zwei Zahlen, um die Achsen der Plots für den Querschnitt der Gefäße anzupassen.

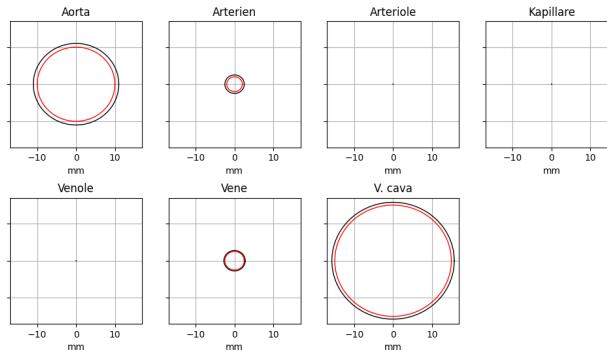


Abbildung 13: Querschnitte der verschiedenen Gefäße in mm

Es ist leicht zu erkennen, dass die Radien der Gefäße stark variieren. Daher steht das fließende Blut vor unterschiedlichen Herausforderungen, insbesondere in Bezug auf den variierenden Strömungswiderstand.

Beim Strömungswiderstand unterscheidet man zwischen zwei Arten: dem parallelen Widerstand und dem Reihenwiderstand.

Der parallele Widerstand ist im Prinzip der Gesamtwiderstand von Gefäßen, durch die Blut fließt, die einen gemeinsamen Start- und Endpunkt haben. Im Gegensatz dazu ist der Reihenwiderstand der Gesamtwiderstand von Gefäßen, die nacheinander angeordnet sind und an einem Punkt miteinander verbunden sind.

Beide Arten von Widerständen lassen sich formal berechnen, wobei die anzuwendenden Formeln unterschiedlich sind. Diese Widerstände wurden jeweils in separaten Funktionen implementiert, um sie als Einflussfaktoren auf den Blutdruck zu berücksichtigen.

Bei beiden Formeln handelt es sich um Summen, jedoch unterscheiden sie sich in der Art der Berechnung. Der Reihenwiderstand ergibt sich aus der Summe der Einzelwiderstände, während der parallele Widerstand durch die Summe der Kehrwerte der parallelen Einzelwiderstände bestimmt wird [2].

$$\begin{aligned}
 \text{Reihe: } R_r &= R_1 + R_2 + R_3 + \cdots + R_n = \sum_{i=1}^n R_i \\
 \text{Parallel: } \frac{1}{R_p} &= \frac{1}{R_1} + \frac{1}{R_2} + \frac{1}{R_3} + \cdots + \frac{1}{R_n} \iff R_p = \left(\sum_{i=1}^n \frac{1}{R_i} \right)^{-1} \quad (3)
 \end{aligned}$$

Jetzt stellt sich die Frage, wie die Einzelwiderstände ermittelt werden können. Um diese Frage zu beantworten, sind weitere Hintergrundinformationen erforderlich.

Ein Blutgefäß kann als eine Art Zylinder oder Rohr betrachtet werden. Das Hagen-

Poiseuille-Gesetz beschreibt die Strömung einer Flüssigkeit durch ein Rohr unter Berücksichtigung der Flüssigkeitseigenschaften [2]. Es ist wichtig anzumerken, dass dieses Gesetz eigentlich für starre Zylinder und konstante Flüssigkeitseigenschaften konzipiert ist [4]. Daher ist es nicht optimal für die genaue Berechnung des Strömungswiderstands in Blutgefäßen geeignet, da die Eigenschaften von Gefäß und Blut nicht konstant sind. Dennoch liefert das Gesetz eine gute Annäherung.

Wenn das Hagen-Poiseuille-Gesetz mit dem Ohm'schen Gesetz $\Delta P = I \cdot R$ kombiniert wird, kann die Formel des Hagen-Poiseuille-Gesetzes umgestellt werden, um den Strömungswiderstand zu ermitteln [2, 4]. Im ersten Schritt wird das Hagen-Poiseuille-Gesetz nach ΔP umgestellt. Anschließend kann das Ohm'sche Gesetz angewendet werden, um den Strömungswiderstand R zu berechnen.

$$\begin{aligned}
 I &= \frac{r_i^4 \cdot \pi \cdot \Delta P}{8 \cdot \eta \cdot l} && / \text{ Umstellen nach } \Delta P \\
 \Leftrightarrow \Delta P &= \frac{8 \cdot \eta \cdot l \cdot I}{r_i^4 \cdot \pi} && / \text{ Ohm'sches Gesetz einsetzen} \\
 \Leftrightarrow R \cdot I &= \frac{8 \cdot \eta \cdot l \cdot I}{r_i^4 \cdot \pi} && / \cdot \frac{1}{I} \\
 \Leftrightarrow R &= \frac{8 \cdot \eta \cdot l}{r_i^4 \cdot \pi}
 \end{aligned} \tag{4}$$

Wobei I die Strömungsstärke, r der Radius, ΔP die treibende Druckdifferenz, l die Länge des Gefäßes und η die Viskosität der Flüssigkeit darstellen.

Unter Verwendung der Formel für den Strömungswiderstand R wurde eine Hilfsfunktion namens `resistance` erstellt. Diese Funktion akzeptiert ein Array von Gefäßlängen und berechnet die Widerstände gemäß der Formel. Anschließend speichert die Funktion die berechneten Widerstände in einem Array namens `res` für spätere Berechnungen.

```

1 def resistance(self, lens):
2     res = []
3     radius = np.copy(self.radi) * 0.001
4
5     for i in range(0, len(radius)):
6         radius[i] *= self.lumFactor[i]
7         res.append((8 * self.viscosity * lens) / (radius[i]**4 * np.pi))
8
9     return res

```

Quelltext 3: Funktion zum Bestimmen des Widerstands R

Im Gefäßsystem gibt es Verzweigungen zwischen verschiedenen Gefäßen. Daher war der nächste Schritt, die Einzelwiderstände entsprechend der Anzahl der Gefäße einer Art zu

bestimmen, also die parallelen Widerstände.

Für diesen Zweck wurde eine neue Funktion namens `vesselResistances` erstellt, die zwei Arrays als Parameter akzeptiert. Das Array `lens` enthält die Längen der verschiedenen Gefäßarten in Millimetern, während das Array `nums` die jeweilige Anzahl der Gefäße jeder Art angibt. Innerhalb der Funktion wird mittels einer For-Schleife über das Array der verschiedenen Gefäße iteriert, um für jedes Gefäß die Einzelwiderstände in Abhängigkeit von der angegebenen Anzahl an Gefäßen zu berechnen.

Mit anderen Worten, je mehr Gefäße einer Art vorhanden sind, desto mehr Einzelwiderstände werden berechnet.

Nach der Bestimmung der Einzelwiderstände müssen die parallelen Widerstände von jeder Gefäßart ermittelt werden, wofür die Funktion `parallelResistance` aufgerufen wird. Abschließend wird ein Array zurückgegeben, das die Gesamtwiderstände jeder Gefäßart speichert.

Der folgende Codeausschnitt zeigt, wie die Gesamtwiderstände aller Gefäße basierend auf ihrer Anzahl bestimmt und in einem Array gespeichert werden.

```

1 def vesselResistances(self, lens, nums):
2     ...
3     for type in types:
4         if type == 'aorta':
5             for _ in range(0, nums[0]):
6                 aortaRes = self.resistance(lens[0])[0]
7                 aortaArr.append(aortaRes)
8         elif type == 'arteries':
9             for _ in range(0, nums[1]):
10                 arteriesRes = self.resistance(lens[1])[1]
11                 arteriesArr.append(arteriesRes)
12         ...
13         aortaCom = self.parallelResistance(aortaArr)
14         arteriesCom = self.parallelResistance(arteriesArr)
15         ...
16         res = [aortaCom, arteriesCom, arteriolesCom, capillariesCom, venulesCom,
17                veinsCom, venaCavaCom]
18     return res

```

Quelltext 4: Ausschnitt zur Bestimmung der Gesamtwiderstände aller Gefäßarten

An dieser Stelle sind alle Einflussfaktoren, die zur Bestimmung des Blutdrucks notwendig sind, vorhanden, so dass wir jetzt zur Blutdruckberechnung übergehen können.

Um den Blutdruck zu bestimmen, wurde für jede Gefäßart eine Funktion erstellt, die schrittweise den Blutdruck für jeden Zeitpunkt basierend auf den Einflussfaktoren be-

rechnet. Dabei greifen diese Funktionen jeweils auf die Werte der vorherigen Gefäßart zu. Mit anderen Worten: Die Aorta greift auf die Blutdruckwerte der linken Herzkammer zu, die Arterien greifen auf die Werte der Aorta zu, und so weiter bis zur V. cava, die auf die Werte der Venen zugreift.

Prinzipiell wird über alle Zeitpunkte iteriert, um den Blutdruckwert zu berechnen. Dabei wird jedoch geprüft, ob der Druck im vorherigen Gefäß bzw. im Fall der Aorta in der linken Herzkammer größer ist als der Druck im folgenden Gefäß. Falls dies zutrifft, wird der Druck auf den Wert des vorherigen Gefäßes gesetzt. Dies dient dazu, den Zeitpunkt zu modellieren, an dem die Klappe zwischen den Gefäßen geöffnet wird.

Ein konkretes Beispiel ist die Aortaklappe: Solange der Druck in der linken Herzkammer größer ist als in der Aorta, bleibt die Klappe zur Aorta geschlossen. Sobald jedoch der Druck in der Aorta größer wird, öffnet sich die Klappe. Ab diesem Zeitpunkt wird mithilfe der Fourier-Transformation und unter Berücksichtigung der Einflussfaktoren der Blutdruck für jeden Zeitpunkt bestimmt und in einem Array gespeichert.

Jede Funktion modelliert diese Situation für ein Gefäß abhängig vom vorherigen Gefäß. Am Ende der Simulation gibt es für jedes Gefäß ein Array mit Blutdruckwerten für jeden Zeitpunkt innerhalb des betrachteten Zeitraums.

Jede Funktion dieser Art besitzt obligatorische und optionale Parameter. Die obligatorischen Parameter sind zwei Arrays, die die Anzahl und Längen jeder Gefäßart enthalten. Die optionalen Parameter sind standardmäßig auf ein leeres Array gesetzt, so dass sie keinen Einfluss haben, wenn sie bei der Ausführung der Simulation nicht angegeben werden. Wenn jedoch optionale Parameter definiert sind, können Zeitpunkte und neue Werte für verschiedene Einflussgrößen innerhalb eines einzigen Simulationsdurchlaufs dynamisch angepasst werden. Zum Beispiel könnte der Herzschlag zur Hälfte der Zeit verdoppelt werden. Weitere optionale Parameter könnten auch die Viskosität, den Radius und das Gesamtvolumen des Blutes dynamisch anpassen. Falls keine optionalen Parameter angegeben werden, erfolgt der Simulationsdurchlauf mit den ursprünglich definierten Parametern.

Weiterhin wurden in der Datei `bodySystem.py` weitere Hilfsfunktionen definiert. Zum einen gibt es die Funktion `vesselSimulator`, die alle Funktionen aufruft, um die Blutdruckwerte für die verschiedenen Gefäße zu erstellen. Darüber hinaus wurde die Funktion `vpPlotter` erstellt, um die erzeugten Blutdruckkurven grafisch in einem Plot darzustellen. Zusammengefasst existiert nach der Ausführung der Implementierungen in `bodySystem.py`

nun für jede Art von Gefäß ein Array mit Blutdruckwerten. Diese Werte hängen von Einflussfaktoren wie Radius, Herzfrequenz, Gesamtvolumen, Widerstand und Viskosität ab.

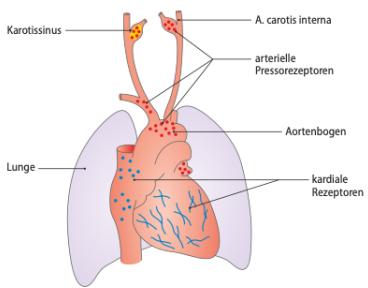
4.2.4 Der Sensor

Damit der Körper den Ist-Blutdruck an den Soll-Blutdruck anpassen kann, benötigt er an entsprechenden Stellen Sensoren, die den Blutdruck messen und die Messwerte weiterleiten.

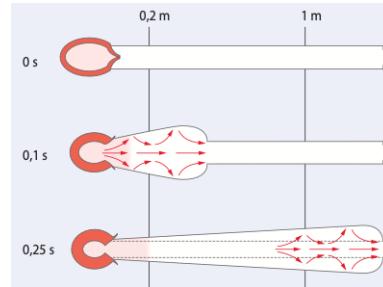
Damit der Körper die Blutdruckwerte messen kann, besitzt er sogenannte Pressorezeptoren oder auch Barorezeptoren genannt [2].

Diese Sensoren messen den Ist-Wert des Blutdrucks und übermitteln diese Information, wie bereits erwähnt, an andere Komponenten des Systems. Dort wird der Ist-Wert mit dem Soll-Wert verglichen, um die Regelgröße entsprechend anzupassen. Im vorliegenden Regelkreis wird der Ist-Wert an die Komponente weitergeleitet, die die Regelabweichung bestimmt, wie in Abbildung 8 dargestellt.

Aus physiologischer Sicht messen die Sensoren den Blutdruck und senden diese Information an das Medulla oblongata im Gehirn, um den Ist-Wert mit dem Soll-Wert vergleichen zu können [2].



(a) Lokalisation der Pressorezeptoren



(b) Ausbreitung der Druckwelle

Abbildung 14: Lokalisierung von Sensoren und Ausbreitung von Druckwellen

Quelle: [2]

Damit die Barorezeptoren den Blutdruck messen können, sind sie im Körper an bestimmten Punkten lokalisiert.

Durch die Bewegung des Blutes als Druckwelle durch die Gefäße, ausgelöst durch die Pumpfunktion des Herzens, können die Rezeptoren darauf reagieren. Genauer gesagt befinden sich die Barorezeptoren in der Gefäßwand. Wenn eine Druckwelle das Gefäß erreicht

und es sich ausdehnt, werden die Rezeptoren durch diese Dehnung aktiviert, wodurch sie den Blutdruck messen können [2].

Um den Blutdruck aus dem generierten Array mit Blutdruckwerten messen zu können, wurde in `sensor.py` ein Sensor implementiert.

Die aktuelle Situation sieht so aus, dass für jedes Gefäß ein Array existiert, das den Blutdruck für jeden Zeitpunkt t enthält. Diese Arrays können geplottet werden, um sie visuell darzustellen.

Da das Ziel darin besteht, den Blutdruck für jede Gefäßart zu erhalten, müssen die Daten aus den erzeugten Arrays noch weiterverarbeitet bzw. Berechnungen darauf angewendet werden, um den eigentlichen Blutdruck zu ermitteln.

Zur Erinnerung, beim Blutdruckmessen gibt es zwei Werte: den systolischen und den diastolischen Wert. Aus diesen beiden Druckwerten kann der mittlere Blutdruck bestimmt werden, indem über die Zeit integriert wird [2, S. 231].

Es existiert aber auch eine Möglichkeit den mittleren Blutdruck über eine Formel anzunähern, die wie folgt lautet:

$$MAP = min + \frac{1}{3} \cdot (max - min) \quad (5)$$

Hierbei entspricht min dem diastolischen und max dem systolischen Druck.

Um den Blutdruck zu messen, wurde diese Formel in einer Funktion `calculatePressure` implementiert. Die Funktion erhält drei Parameter: einen Datensatz mit Blutdruckwerten für jede Gefäßart in einem Array (ein Array von Arrays), sowie zwei Zahlen, die dem systolischen und dem diastolischen Druck entsprechen. Innerhalb der Funktion wird das arithmetische Mittel über alle systolischen und diastolischen Drücke gebildet. Anschließend gibt die Funktion den systolischen und den diastolischen Druck zurück, die dann zur Berechnung des mittleren Blutdrucks verwendet werden können.

```

1 def calculatePressure(self, data, sPeaks, dPeaks):
2     meanSys = np.mean(data[sPeaks])
3     meanDia = np.mean(data[dPeaks])
4
5     map = meanDia + (1/3) * (meanSys - meanDia)
6     return map

```

Quelltext 5: Funktion zur Bestimmung des mittleren Blutdrucks

Die systolischen und diastolischen Drücke über die Zeit wurden mithilfe der Hilfsfunktion

`findPeak`² ermittelt. Diese Funktion erhält als Parameter ein Array von Arrays, das den Datensatz enthält, und identifiziert die gesuchten Druckwerte.

Aufgrund der speziellen Kurvenform müssen die gefundenen Blutdruckwerte gefiltert werden, um sicherzustellen, dass nur der maximale und minimale Druckwert für die Berechnung des mittleren Blutdrucks verwendet werden.

Zur besseren Darstellung gibt es auch eine Funktion, die die Blutdruckkurven der verschiedenen Gefäße zeigt, einschließlich einer Markierung der Maxima und Minima ³.

Wenn alle Maxima und Minima ermittelt wurden, werden sie in der `calculatePressure`-Funktion als zweiter und dritter Parameter verwendet.

Schlussendlich wird über eine letzte Funktion, `presPrinter` ⁴, der mittlere Blutdruck für jede Gefäßart bestimmt und ausgegeben, wobei der bereits erwähnte Datensatz aus Blutdruckwerten als Array übergeben wird.

Das Ganze funktioniert so, dass über die Arrays im Funktionsparameter iteriert wird und in jeder Iteration die Peaks bestimmt werden. Anschließend werden diese verwendet, um den mittleren Druck des aktuell betrachteten Gefäßes zu bestimmen. Dabei werden die oben erklärten Funktionen in jeder Iteration aufgerufen.

4.2.5 Der Regelkreis

Bis zu dieser Stelle kann eine Simulation der Blutdruckkurve von den verschiedenen Gefäßen erstellt werden. Zu diesem Zeitpunkt erfolgt jedoch noch kein Vergleich zwischen dem Ist- und dem Soll-Wert. Da dieser Vergleich fehlt, kann keine Regelabweichung bestimmt werden, weshalb der eigentliche Charakter eines Regelkreises in der bisher vorgestellten Implementierung noch fehlt.

Der Regelkreis wurde in der Datei `controlSystem.py` implementiert. Zur Realisierung des Regelkreises wurde die Funktion `controlSystem` erstellt, die mehrere erforderliche Parameter akzeptiert.

Der erste Parameter `currVals` ist ein Array, das Arrays als Elemente enthält. Diese enthalten Zeitpunkte und Werte zu den Einflussgrößen und werden verwendet, um die dynamischen Parameter in einem Simulationsdurchlauf festzulegen.

²siehe Anhang, Quelltext 7

³siehe Anhang, Abbildung 23

⁴siehe Anhang, Quelltext 8

Parameter zwei bis sechs sind die Sollwerte in einem Array für die Herzfrequenz, die Lumengröße, die Viskosität, das Gesamtvolumen sowie das enddiastolische und endsystolische Volumen. In jeder Iteration des Regelkreises wird das entsprechende Element aus diesen Arrays verwendet, um die Regelabweichung zu bestimmen.

Es ist wichtig zu erwähnen, dass die Länge der Arrays für die Parameter zwei bis sechs mit dem Wert des siebten Parameters `runs` übereinstimmen muss. Dieser Parameter definiert die Anzahl der Iterationen und damit auch die Anzahl der Vergleiche zwischen Ist- und Sollwert des Blutdrucks. Wenn die Längen der Arrays für die Einflussgrößen nicht mit dem Parameter `runs` übereinstimmen, kann dies zu Fehlern oder zur Vernachlässigung von Parametern führen.

Die letzten drei Parameter sind ebenfalls Arrays. Das erste Array definiert die Längen der verschiedenen Gefäße, das zweite Array gibt die jeweilige Anzahl der Gefäßarten an, und das dritte Array enthält die Zeitpunkte, zu denen das System an die neuen Soll-Parameter angepasst wird.

Innerhalb der Funktion werden im ersten Schritt die Objekte für das Herz und das Gefäßsystem mit den initialen Parametern erzeugt, die dem Ist-Zustand des Systems entsprechen.

Es wird eine For-Schleife geöffnet, innerhalb derer Objekte für das Herz und das Gefäßsystem erneut erzeugt werden, jedoch mit den Soll-Parametern für die Simulation. Anschließend wird die Regelabweichung der beiden Blutdruckwerte zu den Zeitpunkten, die im Parameter `ctSim` angegeben sind, bestimmt, indem die Differenz zwischen Soll- und Ist-Blutdruck berechnet wird, entsprechend dem Regelkreiskonzept aus dem vorangegangenen Kapitel. Um sicherzustellen, dass nicht weiter mit dem alten Ist-Wert gerechnet wird, wird der neue Ist-Zustand auf die gerade berechnete Regelabweichung gesetzt. Diese Regelabweichung dient dann als Eingabe für die nächste Iteration in der For-Schleife⁵. Sobald die For-Schleife alle Iterationen durchlaufen hat, wird ein Plot erzeugt, der den Druckverlauf mit Anpassung an die Soll-Parameter darstellt. Zusätzlich werden in der Konsole die systolischen, diastolischen und mittleren Blutdrücke ausgegeben.

⁵siehe Anhang Funktionsausschnitt in Quelltext 9

4.2.6 Verknüpfungen der Komponenten

Die gesamte Implementierung wurde objektorientiert organisiert. Jede Datei im Modell repräsentiert eine Komponente und ist gleichzeitig eine Klasse. Um die Kommunikation zwischen den verschiedenen Klassen zu ermöglichen, wurden sie mittels des `import`-Befehls miteinander verknüpft.

Um einen Startpunkt festzulegen, wo die Verknüpfung beginnt, könnte man `Heart.py` verwenden, da diese Datei keine anderen Klassen importiert. In `bodySystem.py` werden zwei Klassen importiert: zunächst die Klasse des Herzens, um die Verbindung zwischen Herz und Aorta zu realisieren, und zweitens die Klasse `bloodPressure` aus `bloodPressure.py`, um die Grundstruktur der Blutdruckkurve in den anderen Klassen nutzen zu können.

Die Klasse `Regelkreis` in `controlSystem.py` bildet im Prinzip die Schnittstelle zwischen allen anderen Klassen und bildet das Gesamtsystem. Dabei wurden alle anderen Klassen importiert, um eine Verbindung herzustellen.

Die Datei `Main.py` dient dem Nutzer als Eingabe- und Ausführungsdatei. In dieser Datei sind alle wichtigen Bestandteile festgehalten, so dass ein Nutzer nicht jede Datei einzeln ausführen muss. Hier können mit einem Klick die definierten Parameter, Funktionen sowie konstante und dynamische Simulationen durchgeführt werden.

4.3 Anwendung als Nutzer

Wie bereits erwähnt, kann ein Nutzer zur Kommunikation mit den implementierten Komponenten `Main.py` nutzen, indem er die entsprechenden Funktionen aufruft.

Dafür müssen zu Beginn im entsprechend markierten Bereich mit `Initialen Parametern` die Ist-Werte für das Modell festgelegt werden. Die initialen Parameter sind essenziell; ohne sie festzulegen, kann keine Simulation stattfinden.

Wichtige initiale Parameter, die festgelegt werden müssen, sind die Radien der Gefäße als Array, die Viskosität des Blutes als Zahl zwischen 1 und 100, wobei eine niedrige Zahl für dünnflüssiges und eine hohe Zahl für dickflüssiges Blut steht, die Herzfrequenz als ganze Zahl, das enddiastolische und das endsystolische Volumen als ganze Zahlen, die Simulationsdauer und das Gesamtvolumen des Blutes. Zudem gibt es zwei Parameter, die als Arrays angegeben werden müssen: eines für die Anzahl der Gefäße jeder Art und eines für die jeweilige Länge dieser Gefäße.

Des Weiteren gibt es zusätzliche Parameter, die den Hilfsplot für die Radianen der Gefäße

betreffen. Der Parameter `lims` bestimmt die Spannweite der Plotachsen. Wenn das Array leer gelassen wird, passen sich die Achsen automatisch an die Radiengrößen an. Ein Nachteil dabei ist, dass die Größenverhältnisse möglicherweise nicht gut erkennbar sind. Wenn jedoch die Achsengrenzen definiert sind, lassen sich die Größenverhältnisse in dem erzeugten Plot besser visualisieren, wie es in Abbildung 13 dargestellt ist.

Der zweite zusätzliche Parameter `lumFactor` ist ein Array mit Skalierungsfaktoren für die Radien, um sie anzupassen. Die Werte dieser Faktoren können zwischen 0,1 und 1 liegen. Im Abschnitt **Dynamische Parameter** sind optionale Parameter aufgeführt. Diese sind nur relevant, wenn der Nutzer während eines Simulationsdurchlaufs die Parameter dynamisch anpassen möchte, ohne die Regelabweichung zu berücksichtigen. Wenn die Parameter nicht dynamisch angepasst werden sollen, können die entsprechenden Arrays auskommentiert werden. Die Zeile mit den leeren Arrays sollte jedoch unberührt bleiben, da sie sicherstellt, dass die Simulation mit den konstanten Parametern durchgeführt wird, wenn die dynamische Anpassung nicht gewünscht ist.

Die dynamischen Parameter bestehen grundsätzlich immer aus einem Tupel von Arrays: einem Array mit Zeitpunkten und einem Array mit den neuen Parametern, die zu den entsprechenden Zeitpunkten im Zeitarray wirksam werden. Wichtig ist dabei, dass beide Arrays stets die gleiche Anzahl an Elementen haben sollten.

Innerhalb der dynamischen Parameter ist die Anwendung des enddiastolischen und endsystolischen Volumens besonders wichtig, da beide Parameter zur Berechnung des Schlagvolumens verwendet werden. Bei der dynamischen Anpassung müssen sie daher entweder beide aktiviert oder beide deaktiviert sein, je nachdem, ob sie genutzt werden sollen oder nicht.

Der weitere Aufbau der Datei ist immer gleich. Dem Nutzer wird mit einem sichtbaren Kommentar angezeigt, welcher Bereich beginnt, gefolgt von der Erzeugung eines Objekts. Nach der Erzeugung werden wichtige Funktionen aufgelistet, die der Nutzer verwenden kann, um seine vorher definierten Modelleinstellungen anzuwenden. Dabei müssen die Parameter der Funktionen nicht mehr verändert werden; sie können so, wie sie angegeben sind, verwendet werden.

Im ersten Block `Herz` wird das Herz definiert, einschließlich seiner Funktion, die den Druckverlauf in den beiden Herzkammern darstellt, wie in Abbildung 11 gezeigt.

Der zweite Block `Körpersystem` definiert den Körperblutkreislauf mit drei Funktionen.

Die erste Funktion `vesselPlotter` stellt die Radien der verschiedenen Gefäße in mehreren Plots dar. Die zweite Funktion `resisPrinter` gibt die berechneten Widerstände basierend auf den Parametern in der Konsole aus. Die letzte Funktion `vpPlotter` erzeugt einen Plot, in dem alle Druckverläufe der Gefäße dargestellt werden.

Der dritte Block `Sensor` definiert die Sensoren zur Blutdruckmessung. Dieser Block enthält zwei zusätzliche Parameter, die nicht verändert werden sollen, da weitere Berechnungen auf diesen Parametern basieren. Es werden drei Funktionen definiert: `ppPlotter`, die einen Plot erzeugt, ähnlich wie in Abbildung 23, um ein besseres Verständnis zu ermöglichen, und `presPrinter`, die wichtiger ist, da sie die berechneten Drücke in der Konsole ausgibt.

Die letzte Funktion `printPressureTimePoint` ist wieder eine kleine Spielerei, da der Nutzer damit den Druck zu einem bestimmten Zeitpunkt ausgeben lassen kann.

Der vierte und wichtigste Block ist der `Regelkreis`, in dem der eigentliche Regelkreis simuliert wird.

Als ersten Schritt werden die Soll-Parameter in einem Array definiert, wo jedes Element in einer Iteration des Regelkreises verwendet wird. Der einzige Parameter, der kein Array ist, ist der Wert, der die Anzahl an Iterationen definiert. Darüber hinaus gibt es den Parameter `soLF`, der die Skalierungsfaktoren für die Radien beinhaltet und Arrays als Elemente enthält. Mit anderen Worten, für jede Iteration muss ein Array mit entsprechenden Werten definiert werden.

Nachdem die Soll-Werte definiert sind, wird ein Objekt mit den initialen Parametern erzeugt. Anschließend wird der Regelkreis mit der Funktion `controlSystem` ausgeführt, bei dem die Regelabweichung bestimmt und das System auf die Soll-Parameter eingestellt wird.

4.4 Anwendungsbeispiele

Um ein besseres Verständnis für die Anwendung der Software zu erhalten, ist es immer hilfreich, ein Anwendungsbeispiel zu betrachten, das die Funktionsweise verdeutlicht. Hier sollen drei mögliche Anwendungsszenarien demonstriert werden.

Einflussgröße	Werte	Einheit
Radien	[20000, 4000, 20, 8, 20, 5000, 30000]	μm
Viskosität	50	%
Herzfrequenz	60	bpm
edv	100	ml
esv	40	ml
Schlagvolumen	edv - esv = 60	ml
Zeit	10	s
Gesamtvolumen	5000	ml
Gefäßanzahl	[1, 2, 4, 16, 4, 2, 1]	#
Gefäßlängen	[200, 150, 100, 50, 100, 150, 300]	mm
Radiusfaktor	[1, 1, 1, 1, 1, 1, 1]	

Tabelle 2: Initiale Parameter für die Anwendungsbeispiele

4.4.1 Mit konstanten Parametern

Mit den Parametern aus Tabelle 2 lässt sich nun eine Simulation mit konstanten Parametern starten. Beim Ausführen der Funktion `vpPlotter` wird ein Plot mit den Druckverläufen der verschiedenen Gefäße erzeugt. Im Hintergrund werden alle notwendigen Berechnungen durchgeführt, einschließlich der Berechnung der einzelnen Gefäßwiderstände basierend auf Anzahl und Länge der Gefäße.

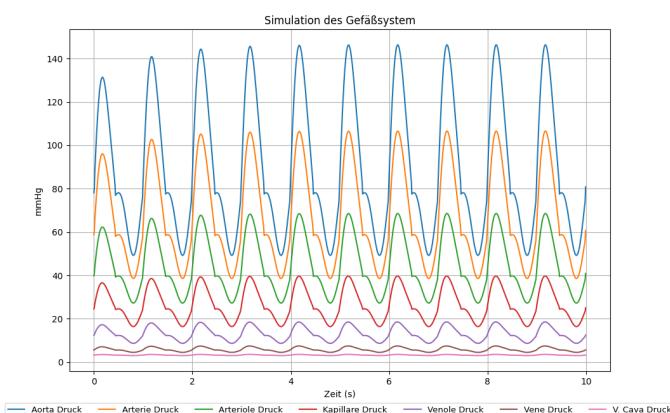


Abbildung 15: Blutdruckkurven unter Verwendung der Parameter aus Tabelle 2

Hinter jeder Kurve befindet sich ein Array, das mit der Sensor-Klasse weiterverarbeitet werden kann, um die Drücke zu ermitteln. Dafür wird die Funktion `presPrinter` verwendet. Nach der Anwendung können die Drücke aus der Konsole entnommen werden. Zusätzlich werden auch die Drücke der Herzkammern ausgegeben, die jedoch in einem eigenen Plot dargestellt werden, wie in Abbildung 11 zu sehen ist. Für jede Kurve gibt es drei Werte: den systolischen und diastolischen Druck sowie den daraus berechneten mittleren Druck.

```
##### Rechter Ventrikel #####
Systolischer Druck: 23.821947192008707 mmHg
Diastolischer Druck: 5.5 mmHg
Mittlerer Druck 11.60731573066957 mmHg

##### Linker Ventrikel #####
Systolischer Druck: 121.96198139264428 mmHg
Diastolischer Druck: 5.5 mmHg
Mittlerer Druck 44.320660464214754 mmHg

##### Aorta #####
Systolischer Druck: 143.97658570613868 mmHg
Diastolischer Druck: 49.22700447234922 mmHg
Mittlerer Druck 80.81019821694571 mmHg

##### Arterie #####
Systolischer Druck: 104.89731563359854 mmHg
Diastolischer Druck: 38.57260876892941 mmHg
Mittlerer Druck 60.68084439048579 mmHg

##### Arteriole #####
Systolischer Druck: 67.57595775798212 mmHg
Diastolischer Druck: 27.237776832490483 mmHg
Mittlerer Druck 40.683836607654364 mmHg

##### Kapillare #####
Systolischer Druck: 39.219647301046635 mmHg
Diastolischer Druck: 16.395384203274546 mmHg
Mittlerer Druck 24.003471902531906 mmHg

##### Venole #####
Systolischer Druck: 18.325848965283843 mmHg
Diastolischer Druck: 8.65236445244265 mmHg
Mittlerer Druck 11.87685929005638 mmHg

##### Vene #####
Systolischer Druck: 7.38831677732406 mmHg
Diastolischer Druck: 4.486271423055369 mmHg
Mittlerer Druck 5.453619874478266 mmHg

##### V. Cava #####
Systolischer Druck: 3.5764659738026565 mmHg
Diastolischer Druck: 2.9960569029489186 mmHg
Mittlerer Druck 3.1895265932334977 mmHg
```

Abbildung 16: Ermittelte Drücke verschiedener Gefäße

4.4.2 Mit variablen Parametern

Als Erweiterung zu den konstanten Parametern können auch variable Parameter übergeben werden, die zusätzlich zu den konstanten oder initialen Parametern aktiviert werden können. Dabei besteht die Möglichkeit, alle oder auch nur einzelne dynamische Parameter zu aktivieren.

Zur besseren Demonstration werden zusätzlich zu den oben definierten konstanten Parametern auch dynamische Parameter für die Herzfrequenz und das Gesamtvolumen übergeben, entsprechend der Abbildung 25 bzw. Tabelle 3.

Einflussgröße	Wertbezeichnung	Werte	Einheit
Herzfrequenz	Zeitpunkte	[2, 4, 6, 9]	s
	Neue Werte	[40, 90, 80, 50]	bpm
Gesamtvolumen	Zeitpunkte	[3, 5, 7, 8]	s
	Neue Werte	[5050, 6200, 7450, 5000]	ml

Tabelle 3: Zusätzliche dynamische Parameter

An dieser Stelle sollte der zentrale Plot deutlich von der Simulation mit konstanten Parametern abweichen, da die Unterschiede zwischen den konstanten und dynamischen Parametern erheblich sind. Dies wird auch in der folgenden Abbildung verdeutlicht:

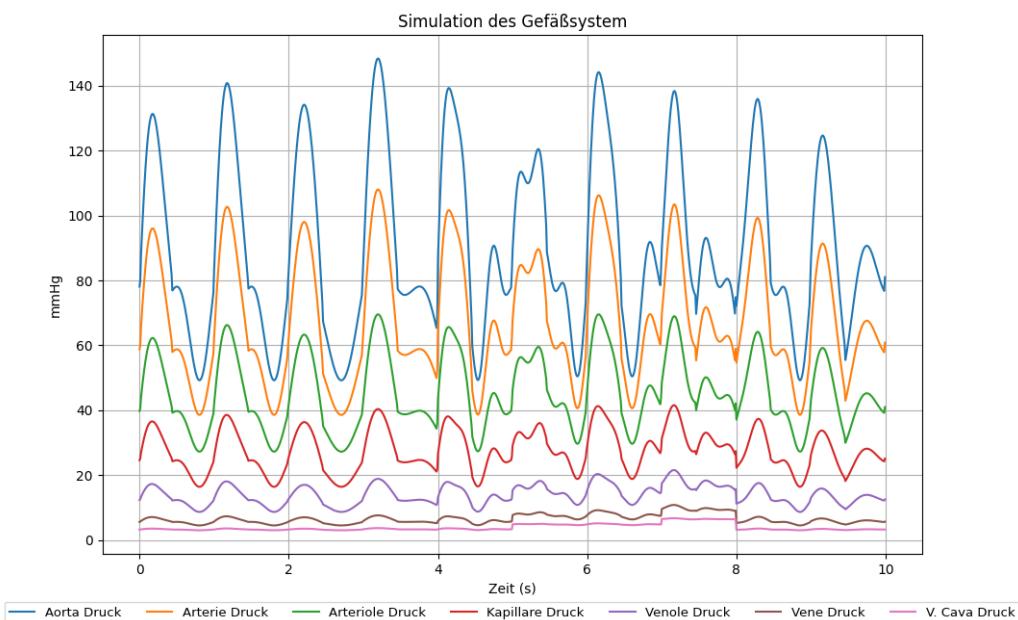


Abbildung 17: Blutdruckkurven mit dynamischen Anpassungen

Es ist ersichtlich, wann die initialen Parameter verwendet werden und ab wann die dynamischen Parameter greifen. Im Fall der Herzfrequenz erfolgen Änderungen zu den Zeitpunkten 2, 4, 6 und 9. Bezüglich des Gesamtvolumens sind Veränderungen deutlich zu den Zeitpunkten 3, 5, 7 und 8 erkennbar, insbesondere im Druckverlauf der V. cava.

4.4.3 Als Regelkreis

Aufbauend auf den Beispielen mit den konstanten und dynamischen Parametern folgt nun ein Beispiel für einen Regelkreis. Es sei angemerkt, dass die dynamischen Parameter in diesem Fall optional sind; der Regelkreis würde auch ohne sie funktionieren. Jedoch sollten sie aktiviert werden, um Dynamik in den Simulationsdurchlauf zu integrieren.

Damit ein Regelkreis simuliert werden kann, sollten neben den obligatorischen initialen Parametern auch die Soll-Werte definiert werden.

Aufbauend auf den beiden obigen Beispielen werden an dieser Stelle neben den bereits definierten initialen Parametern und den dynamischen Parametern noch folgende Soll-Werte definiert:

Einflussgröße	Werte	Einheit
Soll Herzfrequenz	[90, 147, 79, 60]	bpm
Soll Viskosität	[50, 55, 50, 50]	%
Soll Faktor (Radius)	[[1, 1, 1, 1, 1, 1, 1]; [1, 1, 1, 1, 1, 1, 1]; [1, 1, 1, 1, 1, 1, 1]; [1, 1, 1, 1, 1, 1, 1]]	
Soll Gesamtvolumen	[5000, 5000, 5000, 5000]	ml
Soll edv	[110, 120, 125, 110]	ml
Soll esv	[60, 50, 55, 60]	ml
Durchläufe	4	#
ctSim	[2, 5, 7, 9]	s

Tabelle 4: Soll-Parameter für den Regelkreis

Um den Regelkreis zu demonstrieren, werden hier keine ausgefallenen Parameter definiert. Es werden zwei dynamische Parameter sowie die initialen Parameter aktiviert. Bei den Soll-Werten gibt es keine extremen Anpassungen, sondern nur kleine Änderungen in jeder Iteration, um die Dynamik im nachfolgenden Plot zu zeigen.

Da die Simulation auf den dynamischen Parametern basiert, könnte bei Bedarf auch ein Plot angezeigt werden, der den Input für den Regelkreis demonstriert. Da dieser Plot jedoch an dieser Stelle identisch mit Abbildung 25 wäre, wird er hier nicht erneut aufgeführt.

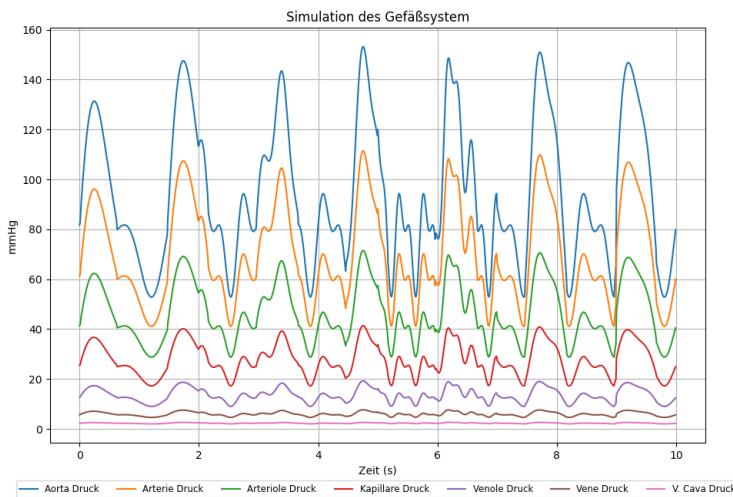


Abbildung 18: Ergebnis der Regelkreis-Simualtion

Nach der Ausführung des Regelkreises erhält der Nutzer einen Plot, wie in Abbildung 18 zu sehen, der klar zeigt, zu welchen Zeitpunkten es zu einer Anpassung der Parameter gekommen ist.

Zu sehen ist, dass zum Zeitpunkt 5 eine Anpassung der Herzfrequenz stattgefunden hat, wodurch die Blutdruckwerte etwas unruhiger sind, bis zum Zeitpunkt 7, wo sich der Herzschlag wieder beruhigt. Dies zeigt sich auch beim Vergleich des Inputplots mit dem Outputplot. Die Blutdruckwerte passen sich den übergebenen Soll-Werten an, da der Input sich deutlich vom Output unterscheidet.

Diese drei kleinen Beispiele bezüglich konstanter und dynamischer Parameter sowie der Anwendung des Regelkreises dienen lediglich als Demonstration der Implementierung, um dem Nutzer die Handhabung näherzubringen. Um das Modell gründlich zu testen, sollten realistische Parameter verwendet werden. Dadurch können Simulationen für reale Situationen durchgeführt werden, um aussagekräftige Ergebnisse zu erzielen.

5 Simulationsstudien

5.1 Kurzzeitiger Stress

Stress kann für jeden Menschen eine unangenehme Situation darstellen, die jedoch von Person zu Person unterschiedlich empfunden wird. Ein gutes Beispiel hierfür sind Situationen, die ein Mensch als belastend empfindet. Dieses Empfinden kann von Person zu

Person variieren: Die eine Person empfindet es als stressig, wenn sie vor einem Publikum Vorträge halten muss, während eine andere Person Prüfungsangst hat, aber gut vor einem Publikum sprechen kann.

Eine Stresssituation ist sehr individuell und nie gleich. Der Körper jedoch unterscheidet nicht, was die Stresssituation auslöst, und passt seine physiologischen Reaktionen an, um in der bedrohlichen Situation das Überleben zu sichern [18].

Abhängig von der Person und der Situation kann der Körper unter anderem mit erhöhter Muskelspannung, erhöhtem Blutdruck, schnellerem Herzschlag oder Schwitzen reagieren. Wie bereits erwähnt, hat all dies das Ziel, den Körper möglichst unbeschadet durch die Situation zu führen [18].

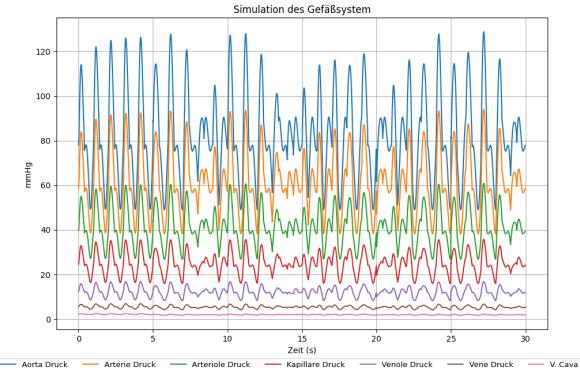
Um eine solche Situation simulieren zu können, bietet es sich an, die Herzfrequenz über einen kurzen Zeitraum schnell ansteigen zu lassen und gleichzeitig die Gefäßradien aufgrund von Adrenalin zu reduzieren.

Dafür wird das Modell zunächst mit initialen Parametern gestartet, wobei diese Parameter über die Zeit leicht schwanken, um die Simulation realistischer zu gestalten. Die initialen und dynamischen Parameter können den Tabellen 9 und 10 entnommen werden. Die Parameter für eine Stresssituation können wie folgt aussehen, wobei leichte Schwankungen in den anderen Parametern berücksichtigt sind.

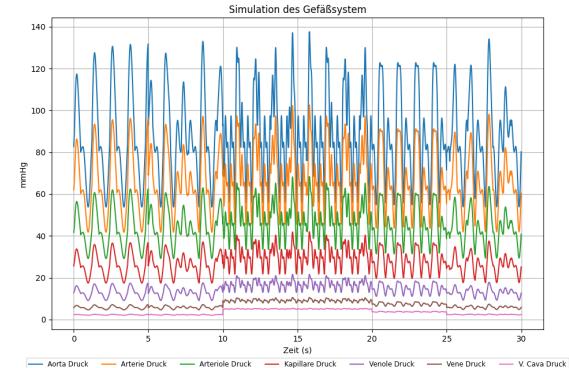
Einflussgröße	Werte	Einheit
Soll Herzfrequenz	[70, 160, 100, 80]	bpm
Soll Viskosität	[50, 65, 60, 60]	%
Soll Faktor (Radius)	[[1, 1, 1, 1, 1, 1, 1]; [0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1]; [0.5, 0.5, 0.5, 0.5, 0.5, 0.5, 0.5]; [1, 1, 1, 1, 1, 1, 1]]	
Soll Gesamtvolumen	[5000, 7000, 6000, 5000]	ml
Soll edv	[130, 130, 130, 130]	ml
Soll esv	[40, 40, 40, 40]	ml
Durchläufe	4	#
ctSim	[5, 10, 20, 25]	s

Tabelle 5: Soll-Parameter für die Stresssituation

Mit diesen Parametern lässt sich schon erahnen, dass die Herzfrequenz sehr schnell und plötzlich steigt, um anschließend wieder zu sinken. Demzufolge sollte im Ergebnisplot ein Bereich existieren, der die hohe Herzfrequenz repräsentiert. Auch in der numerischen Ausgabe sollte eine deutliche Änderung sichtbar sein.



(a) Regelkreis Input als Plot



(b) Regelkreis Output als Plot

```
##### Rechter Ventrikel #####
Systolischer Druck: 18.39979866416475 mmHg
Diastolischer Druck: 5.5 mmHg
Mittlerer Druck 9.799932888054915 mmHg

##### Linker Ventrikel #####
Systolischer Druck: 90.0582282145397 mmHg
Diastolischer Druck: 8.535696578827368 mmHg
Mittlerer Druck 35.992405326369564 mmHg

##### Aorta #####
Systolischer Druck: 97.19666308676774 mmHg
Diastolischer Druck: 65.52661558906904 mmHg
Mittlerer Druck 76.08329808830194 mmHg

##### Arterie #####
Systolischer Druck: 71.99980729985663 mmHg
Diastolischer Druck: 49.83233651053864 mmHg
Mittlerer Druck 57.22149346732145 mmHg

##### Arteriole #####
Systolischer Druck: 47.58885306573423 mmHg
Diastolischer Druck: 34.068204725544994 mmHg
Mittlerer Druck 38.575087505608074 mmHg

##### Kapillare #####
Systolischer Druck: 29.503710408188702 mmHg
Diastolischer Druck: 19.971038007976304 mmHg
Mittlerer Druck 23.14859547471377 mmHg

##### Venole #####
Systolischer Druck: 14.204047735452685 mmHg
Diastolischer Druck: 10.117942483631019 mmHg
Mittlerer Druck 11.479977567571574 mmHg

##### Vene #####
Systolischer Druck: 6.000213908312745 mmHg
Diastolischer Druck: 4.774382332554959 mmHg
Mittlerer Druck 5.182992857807554 mmHg

##### V. Cava #####
Systolischer Druck: 2.1523427806293336 mmHg
Diastolischer Druck: 1.9021165848488362 mmHg
Mittlerer Druck 1.9855253167756686 mmHg
```

(c) Regelkreis Input als Zahlen

```
##### Rechter Ventrikel #####
Systolischer Druck: 25.94373469749448 mmHg
Diastolischer Druck: 1.4528652677539071 mmHg
Mittlerer Druck 9.616488411000764 mmHg

##### Linker Ventrikel #####
Systolischer Druck: 165.5271454415993 mmHg
Diastolischer Druck: 7.892361075653568 mmHg
Mittlerer Druck 60.43728919763548 mmHg

##### Aorta #####
Systolischer Druck: 102.24158886212453 mmHg
Diastolischer Druck: 79.62313681979627 mmHg
Mittlerer Druck 87.16262083390569 mmHg

##### Arterie #####
Systolischer Druck: 76.93753482377316 mmHg
Diastolischer Druck: 61.10650518618558 mmHg
Mittlerer Druck 66.38351506538145 mmHg

##### Arteriole #####
Systolischer Druck: 51.976122552351015 mmHg
Diastolischer Druck: 42.25939116020689 mmHg
Mittlerer Druck 45.49830162425493 mmHg

##### Kapillare #####
Systolischer Druck: 32.614588405857845 mmHg
Diastolischer Druck: 25.91522203329725 mmHg
Mittlerer Druck 28.148344157484114 mmHg

##### Venole #####
Systolischer Druck: 17.29120310634451 mmHg
Diastolischer Druck: 12.590478178814001 mmHg
Mittlerer Druck 14.157386487990838 mmHg

##### Vene #####
Systolischer Druck: 8.33070079741482 mmHg
Diastolischer Druck: 6.944120635376282 mmHg
Mittlerer Druck 7.40631402272461 mmHg

##### V. Cava #####
Systolischer Druck: 4.017720555598586 mmHg
Diastolischer Druck: 3.7640418397527227 mmHg
Mittlerer Druck 3.848601411781344 mmHg
```

(d) Regelkreis Output als Zahlen

Abbildung 19: Input und Output des Regelkreises für die Stresssituation

5.2 Schocksituation

Eine weitere spannende Situation könnte eine Schocksituation darstellen, die unerwartet und plötzlich auftritt. Genauso wie bei der Stresssituation passieren im Körper Ereignisse, die sicherstellen, dass die Versorgung auch in einer solchen Schocksituation gewährleistet wird.

Eine sehr bekannte Schocksituation kann eine Verletzung mit Blutverlust darstellen, die auch als hypovolämischer Schock bekannt ist [11].

Um nun eine Schocksituation zu simulieren, macht es Sinn, an dieser Stelle kurz zu betrachten, was innerhalb des Körpers bzw. im Blutsystem und seinen Komponenten passiert.

Durch den Blutverlust hat das Herz weniger zu pumpen, wodurch der Blutdruck abnimmt. Gleichzeitig erhöht sich die Herzfrequenz, um den Blutverlust zu kompensieren und die Versorgung wichtiger Organe zu gewährleisten [20].

Die Parameter für eine Schocksituation könnten zum Beispiel wie in folgender Tabelle 6 aussehen.

Einflussgröße	Werte	Einheit
Soll Herzfrequenz	[50, 80, 90, 100]	bpm
Soll Faktor (Radius)	[[1, 1, 1, 1, 1, 1, 1]; [1, 1, 1, 1, 1, 1, 1]; [1, 1, 1, 1, 1, 1, 1]; [1, 1, 1, 1, 1, 1, 1]]	
Soll Viskosität	[75, 75, 75, 75]	%
Soll Gesamtvolumen	[5000, 4500, 4000, 3500]	ml
Soll edv	[100, 100, 100, 100]	ml
Soll esv	[60, 71, 72, 73]	ml
Durchläufe	4	#
ctSim	[10, 20, 30, 35]	s

Tabelle 6: Soll-Parameter für die Schocksituation

Zur Initialisierung des Modells werden die initialen und dynamischen Parameter aus den Tabellen 11 und 12 verwendet, um die Simulation der Schocksituation durchzuführen.

In der folgenden Abbildung sind sowohl der Input als auch der Output inklusive der Druckwerte als Zahlen zu finden.

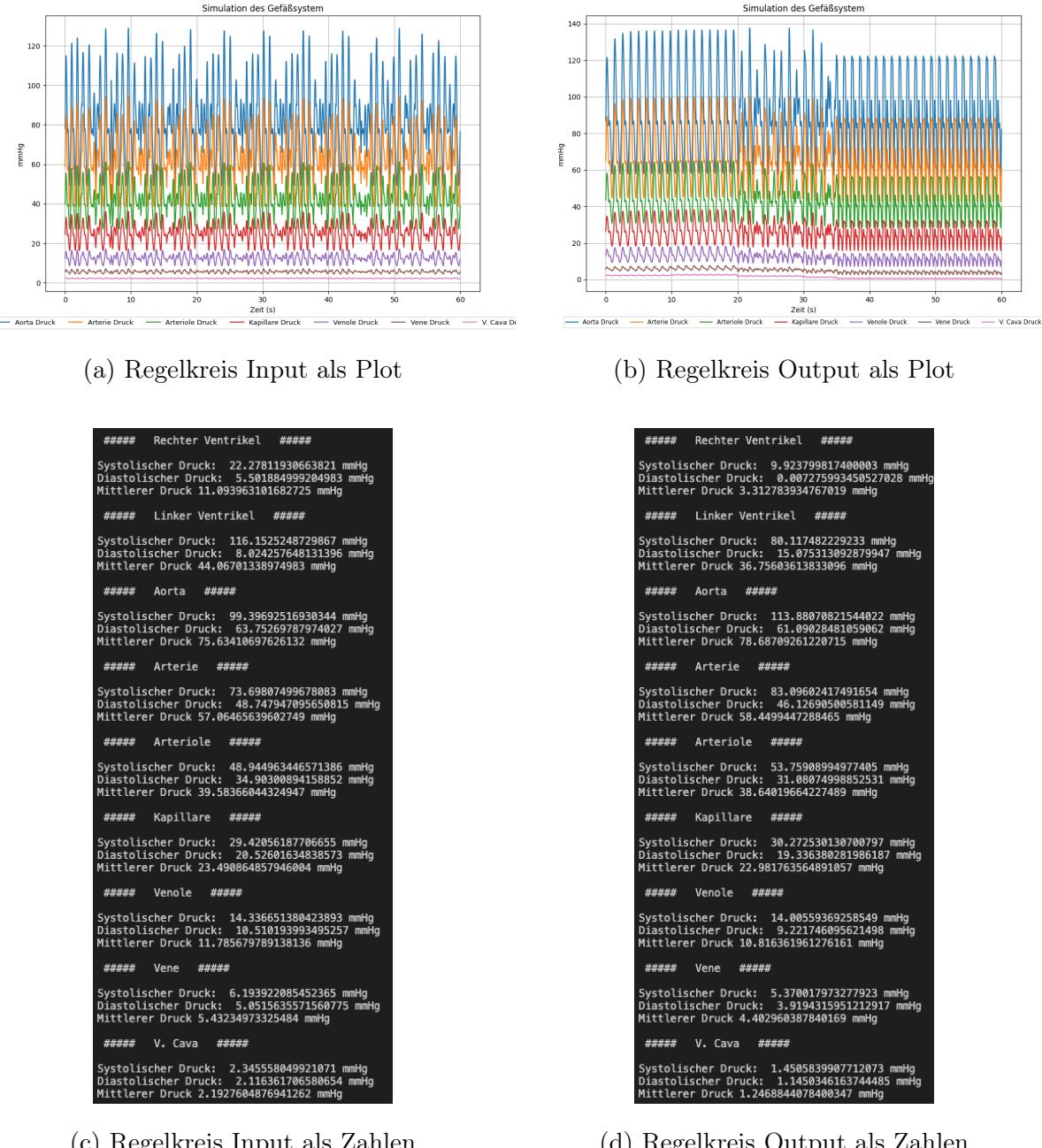


Abbildung 20: Input und Output des Regelkreises für die Schocksituation

5.3 Verengung der Blutgefäße

Eine weitere interessante Situation könnte das gezielte Verengen der Gefäße darstellen. Hierbei werden die Gefäße über den Skalierungsfaktor `lumfactor` immer weiter verengt. Dadurch könnte man den Blutdruck simulieren, der durch zunehmende Verkalkung der Gefäße entsteht.

Bei Verkalkung spricht man zunächst von einer Erkrankung des Gefäßsystems, bei der sich Ablagerungen an den Innenwänden der Gefäße ansammeln und dadurch den Gefäßradius verkleinern [4]. Je nach Lokalisation der Ablagerungen können unterschiedliche Erkrankungen entstehen, wodurch sich die Ablagerungen im späteren Verlauf in Bezug auf die Krankheitsbilder unterscheiden können [4, 19].

Durch die Verkalkungen, die den Gefäßradius reduzieren, erhöht sich der Widerstand, den das Blut überwinden muss. Dies führt dazu, dass der Blutdruck mit jeder Reduktion des Radius erhöht wird.

Um die Simulation für eine Verkalkung der Gefäße zu starten, wurden die initialen und dynamischen Parameter gemäß Tabelle 13 und 14 gewählt.

Einflussgröße	Werte	Einheit
Soll Herzfrequenz	[60, 50, 55, 62]	bpm
Soll Viskosität	[75, 75, 75, 75]	%
Soll Faktor (Radius)	[[0.9, 0.9, 0.9, 0.9, 0.9, 0.9, 0.9, 0.9]; [0.8, 0.8, 0.8, 0.8, 0.8, 0.8, 0.8, 0.8]; [0.4, 0.4, 0.4, 0.4, 0.4, 0.4, 0.4, 0.4]; [0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1]]	
Soll Gesamtvolumen	[5000, 5000, 5000, 5000]	ml
Soll edv	[110, 110, 110, 110]	ml
Soll esv	[50, 50, 50, 50]	ml
Durchläufe	4	#
ctSim	[40, 140, 210, 280]	s

Tabelle 7: Soll-Parameter für die Radiusverengung

Mit diesen Parametern wird eine Simulation gestartet, wobei das Modell folgende Ergebnisse liefert, die in Abbildung 21 gezeigt werden.

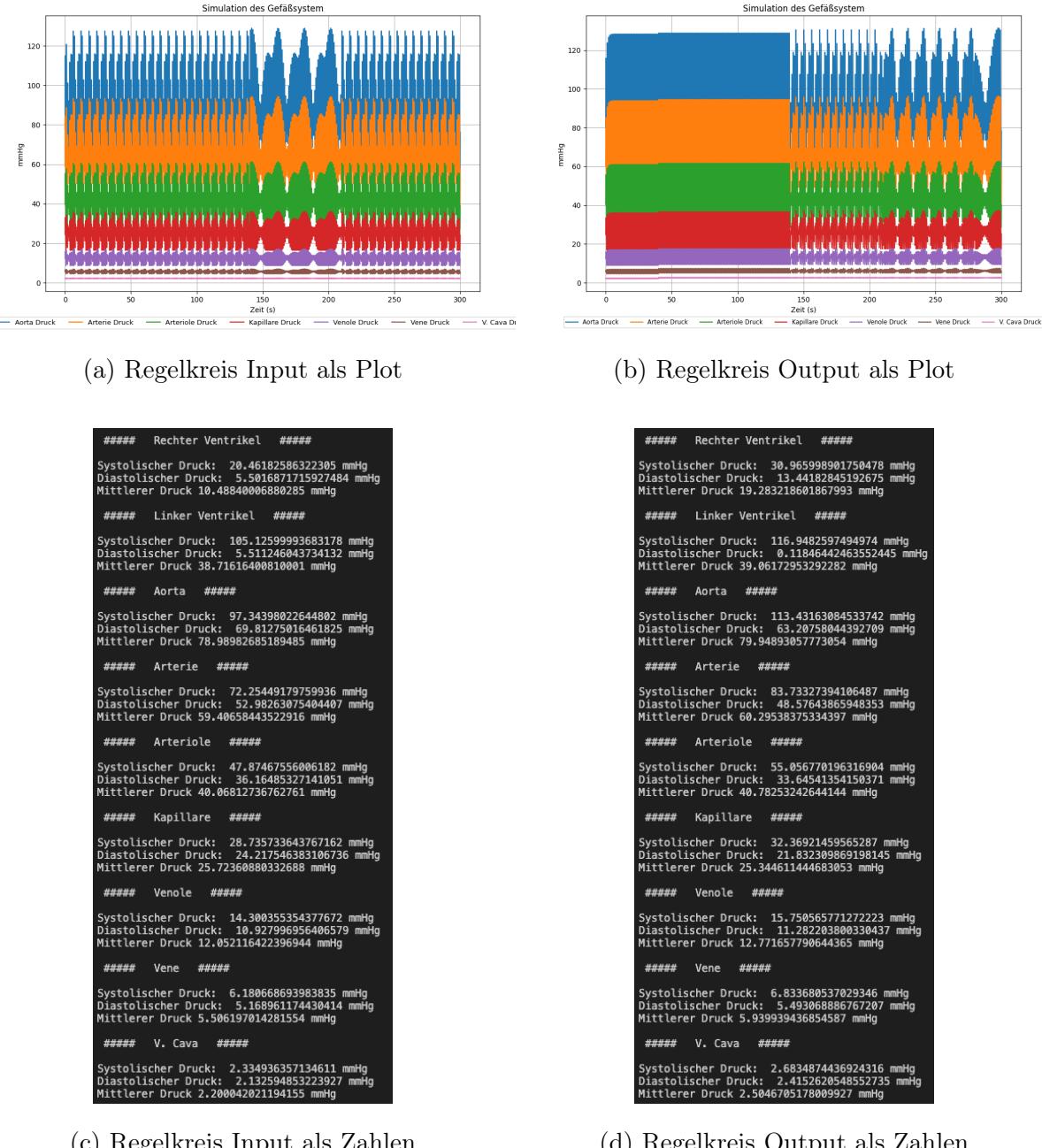


Abbildung 21: Input und Output des Regelkreises für die Radiusverengung

Innerhalb der Plots ist es aufgrund der Zeitspanne möglicherweise schwer, eine Erhöhung des Blutdrucks zu erkennen. Jedoch ist in den berechneten Blutdruckwerten ein deutlicher Anstieg zwischen Input und Output zu sehen.

5.4 Niereninsuffizienz

Hinter einer Niereninsuffizienz verbirgt sich eine Unterfunktion der Nieren, sodass diese nicht mehr ordnungsgemäß arbeiten können. Deshalb spricht man umgangssprachlich auch von einer Nierenunterfunktion [7].

Zu den Funktionen der Nieren gehört unter anderem die Regulation des Elektrolyt-, Wasser- und Mineralstoffhaushalts [2]. Dadurch sind die Nieren ein maßgeblicher Faktor für die Viskosität des Blutes und beeinflussen somit auch den Blutdruck.

Durch die eingeschränkte Funktion der Niere wird beispielsweise weniger Wasser ausgeschieden [7], was wiederum die Regulation des Blutdrucks beeinflusst [5]. Darüber hinaus führt die verringerte Ausscheidung von Flüssigkeit, kombiniert mit der Zufuhr neuer Flüssigkeit, zu einer Zunahme des Gesamtvolumens, was ebenfalls den Blutdruck beeinflusst [7].

Um den Blutdruck bei einer Nierenunterfunktion zu simulieren, können die initialen und dynamischen Parameter gemäß den Tabellen 15 und 16 gewählt werden.

Einflussgröße	Werte	Einheit
Soll Herzfrequenz	[65, 70, 90, 80]	bpm
Soll Viskosität	[50, 40, 35, 30]	%
Soll Faktor (Radius)	[[1, 1, 1, 1, 1, 1, 1]; [1, 1, 1, 1, 1, 1, 1]; [1, 1, 1, 1, 1, 1, 1]; [1, 1, 1, 1, 1, 1, 1]]	
Soll Gesamtvolumen	[5000, 7000, 9000, 11000]	ml
Soll edv	[70, 75, 80, 90]	ml
Soll esv	[10, 10, 15, 20]	ml
Durchläufe	4	#
ctSim	[2, 15, 20, 40]	s

Tabelle 8: Soll-Parameter bei der Niereninsuffizienz

In den Parametern der obigen Tabelle 8 ist ersichtlich, dass das Volumen über die Zeit zunimmt und die Viskosität abnimmt, wodurch das Blut immer flüssiger wird. Zusätzlich wurden die Parameter für die Herzfrequenz sowie das enddiastolische und endsystolische Volumen so gewählt, dass sie ebenfalls ansteigen.

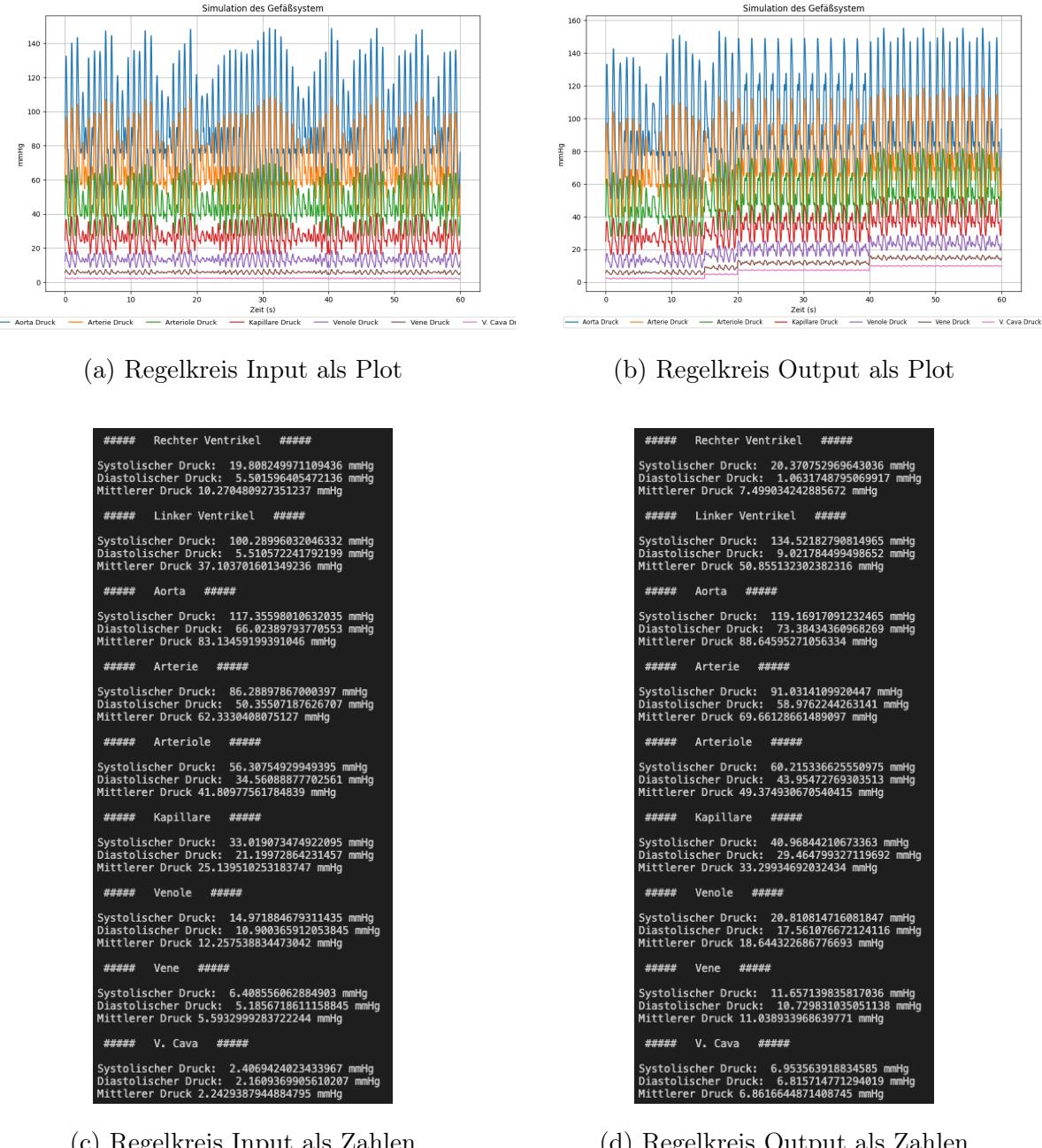


Abbildung 22: Input und Output des Regelkreises bei der Niereninsuffizienz

6 Diskussion und Ausblick

6.1 Diskussion der Ergebnisse

Nach Anwendung und Demonstration des Modells lassen sich aus den Diagrammen und den ermittelten Druckwerten einige Aussagen bezüglich der Anwendbarkeit des Modells

treffen.

Vorab bewegen sich die verschiedenen Drücke innerhalb individueller Spannweiten. Im rechten Ventrikel liegen die Drücke im Bereich von 0 bis 20 mmHg, im linken Ventrikel zwischen 5 und 140 mmHg. In der Aorta bewegen sich die Drücke im Bereich von 80 bis 130 mmHg. In den Arterien liegen die Drücke zwischen 60 und 120 mmHg. In den Arteriolen betragen die Drücke 40 bis 80 mmHg. In den Kapillaren bewegen sich die Drücke im Bereich von 20 bis 30 mmHg. In den Venolen liegen die Drücke zwischen 10 und 20 mmHg. In den Venen betragen die Drücke 5 bis 15 mmHg. In der V. cava bewegen sich die Drücke im Bereich von 0 bis 10 mmHg [2]. Diese Werte können je nach individuellen physiologischen Bedingungen und Messmethoden variieren.

Diese Werte stimmen jedoch ungefähr mit den in Abbildung 16 dargestellten Druckwerten überein, obwohl gewisse Abweichungen erkennbar sind. Diese Abweichungen könnten darauf zurückzuführen sein, dass nicht exakt die gleichen Werte für einen potenziellen Normalzustand gewählt wurden. Es ist praktisch unmöglich, die genauen Werte eines Normalzustands zu rekonstruieren, da diese variieren können. Dennoch nähern sich die simulierten Blutdruckwerte den oben erwähnten normalen Blutdruckwerten sehr gut an. Aus den Ergebnissen zum kurzzeitigen Stress lässt sich schnell anhand der Plots erkennen, wo eine Anpassung stattgefunden hat. Dabei ist auch ersichtlich, ob diese Anpassung nach oben oder nach unten erfolgt ist, also ob der Blutdruck erhöht oder reduziert wurde. Eine Korrektur nach unten bedeutet, dass innerhalb eines bestimmten Zeitraums weniger Peaks vorhanden sind, die zudem einen niedrigeren systolischen Druck aufweisen. Wurde der Blutdruck hingegen nach oben korrigiert, treten in derselben Zeit mehr Peaks mit einem höheren systolischen Wert auf als bei einer Reduktion des Blutdrucks.

Genauso wie bei dem erzeugten Plot zum kurzzeitigen Stress zeigen die ermittelten Blutdruckwerte zu den verschiedenen Gefäßarten in Abbildung 19 vom Normalzustand zur Stresssituation eine erkennbare Blutdruckerhöhung. Genau das war auch erwartet und zeigt, dass das Modell gute Ergebnisse liefert. Des Weiteren ist in den Blutdruckwerten zu erkennen, dass die Blutdruckerhöhung umso kleiner wird, je weiter das Gefäß vom Herzen entfernt ist. In der linken Herzkammer und der Aorta sind die Differenzen zwischen Normal- und Stresszustand sehr eindeutig und nicht zu leugnen. Dahingegen ist die Blutdruckerhöhung in den Venen und der V. cava zwar auch vorhanden, aber im Vergleich zur Herzkammer und Aorta deutlich kleiner.

Dies zeigt aber auch, dass das Modell funktioniert und realistische Ergebnisse liefert, da dies genau so in der Natur vorkommt. Am Herzen wird der Druck aufgebaut und ist dementsprechend höher als an dem Punkt, wo das Blut nur noch zurück in die rechte Herzkammer fließt, wie es bei der V. cava der Fall ist.

Die Simulation der Schocksituation liefert ebenfalls klare Ergebnisse. Wie bereits erwähnt, ist eine der vielen möglichen Schocksituationen der Blutverlust. Dies wurde auch in den Soll-Parametern so festgelegt, dass das Gesamtvolumen im Verlauf der Zeit abnimmt.

Aufgrund des Blutverlusts ist eine Reduktion des Blutdrucks zu erwarten. Dies wird auch im Plot, der vom Regelkreis ausgegeben wird, deutlich. Die berechneten Druckwerte zeigen ebenfalls, dass sich der Blutdruck in den verschiedenen Gefäßarten im Vergleich zum Input des Regelkreises verringert hat.

In der Simulationsstudie zur Verengung der Gefäßradien wurde versucht, Arteriosklerose zu modellieren. Dabei wurden die Gefäßdurchmesser im Laufe der Zeit kontinuierlich verringert. Da sich diese Krankheit über einen längeren Zeitraum entwickelt, wurde die Simulationsdauer entsprechend länger gewählt als bei anderen Simulationen.

Im Plot, der vom Regelkreis ausgegeben wird, ist zu erkennen, dass die Blutdruckwerte im Laufe der Zeit ansteigen. Dies liegt daran, dass die Gefäßdurchmesser sich verkleinern, wodurch der Widerstand, den das Blut überwinden muss, zunimmt.

Auch die ermittelten Blutdruckwerte zeigen vielversprechende Ergebnisse, da die Drücke des Inputs wesentlich niedriger sind als die im Output dargestellten Werte.

Demzufolge liefert das Modell an dieser Stelle die erwarteten Ergebnisse, wenn diese Situation logisch durchdacht wird.

Bei der Simulation zur Niereninsuffizienz war das Ziel, den Blutdruck bei einer schlecht funktionierenden Niere zu simulieren. Wie bereits erwähnt, ist eine Funktion der Niere unter anderem die Regulation des Wasserhaushalts im Körper. Wenn sie jedoch nicht ordnungsgemäß funktioniert, steigt der Wassergehalt im Körper im Laufe der Zeit an. Aus diesem Grund muss das Gesamtvolumen in der Simulation über die Zeit erhöht werden. Die Werte für das Gesamtvolumen wurden jedoch im Vergleich zu den tatsächlichen Werten extrem hoch angesetzt. Dies geschah, weil kleine Veränderungen im Gesamtvolumen zu gering erschienen und einen kaum wahrnehmbaren Effekt auf das Gesamtsystem nach sich ziehen. Möglicherweise liegt dies an einem nicht optimal gewählten Skalierungsfaktor, der erforderlich ist, um komplexe physiologische Abläufe angemessen zu modellieren.

Trotz der übertriebenen Werte zeigt das Modell das erwartete Verhalten: Der Druck steigt mit der Zunahme des Gesamtvolumens, wie erwartet.

6.2 Schlusswort und Ausblick

Schlussendlich lässt sich sagen, dass das Modell in den simulierten Situationen gute Ergebnisse liefert, die erwartet wurden. Jedoch liefert es in anderen Fällen auch Ergebnisse, die zwar in die richtige Richtung gehen, aber kleiner ausfallen als erwartet. Ein Beispiel hierfür ist die Niereninsuffizienz, bei der das Modell nur bei unrealistisch hohen Werten die erwarteten Ergebnisse erzielt hat.

Wie bereits erwähnt, könnten die zu geringen Veränderungen auf eine nicht optimal gewählte mathematische Modellierung der physiologischen Abläufe zurückzuführen sein. Daraus könnte das Modell in potenziellen zukünftigen Optimierungen in dieser Hinsicht weiter verfeinert werden, um eine noch genauere Abbildung der Realität zu ermöglichen.

Ein weiteres Vorgehen könnte sein, diese Ergebnisse mit einer Fachperson zu konsultieren, um sie zu validieren. Dadurch könnte überprüft werden, ob die simulierten Ereignisse in ähnlicher Form auch in der Realität vorkommen oder ob Anpassungen erforderlich sind. Weiterhin könnte die Nutzerkommunikation mit dem Modell verbessert werden, beispielsweise durch eine Interaktion über die Kommandozeile oder noch intuitiver über eine grafische Benutzeroberfläche. Dies würde die Nutzung des Modells weiter vereinfachen, sodass kein explizites Vorwissen zur Programmierung erforderlich ist. Derzeit ist noch ein gewisses Grundwissen über die Anpassung der Parameter in der Datei `Main.py` notwendig, um die Modellparameter entsprechend den Nutzerwünschen einzustellen.

Nichtsdestotrotz liefert das Modell trotz der noch bestehenden Optimierungsmöglichkeiten gute Ergebnisse. Es könnte somit einen Beitrag dazu leisten, die Blutdruckregulation besser zu verstehen und neue Erkenntnisse über den Blutdruck und seine physiologischen Eigenschaften zu liefern.

Literaturverzeichnis

- [1] Vereinfachte schematische Darstellung eines doppelten Blutkreislaufs von gleichwarmen Wirbeltieren.
https://commons.wikimedia.org/wiki/File:Blutkreislauf_Gleichwarme.svg.
- [2] Ralf Brandes, Florian Lang, and Robert F. Schmidt, editors. *Physiologie des Menschen*. Springer eBooks : Medicine. Springer, Berlin ; Heidelberg, 32 edition, 2019.
- [3] Zachary S. Bruss and Avais Raja. *Physiology, Stroke Volume*. StatPearls Publishing, Treasure Island (FL), 2023.
- [4] Christian Fellner. Analyse und Simulation des CNSystems CNAP VERIFI Controllers. Master's thesis, Technische Universität Graz (90000), 2019.
- [5] National Kidney Foundation. Bluthochdruck und chronische Nierenerkrankung.
https://www.kidney.org/sites/default/files/11-10-1402_DAI_PatBro_HBPandCKD_1-4_Pharmenet_German_Apr08_LR.pdf.
- [6] John Edward Hall. *Pocket Companion to Guyton and Hall Textbook of Medical Physiology*. Elsevier, Amsterdam, 2016.
- [7] H. Haller and B. E. Strauer. Niereninsuffizienz. *Der Internist*, 53(7):789–790, Jul 2012.
- [8] Charles R. Harris, K. Jarrod Millman, Stéfan J. van der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, Julian Taylor, Sebastian Berg, Nathaniel J. Smith, Robert Kern, Matti Picus, Stephan Hoyer, Marten H. van Kerkwijk, Matthew Brett, Allan Haldane, Jaime Fernández del Río, Mark Wiebe, Pearu Peterson, Pierre Gérard-Marchant, Kevin Sheppard, Tyler Reddy, Warren Weckesser, Hameer Abbasi, Christoph Gohlke, and Travis E. Oliphant. Array programming with NumPy. *Nature*, 585(7825):357–362, September 2020.
- [9] Morris W. Hirsch, Stephen Smale, and Robert L. Devaney. *Differential Equations, Dynamical Systems, and an Introduction to Chaos*. Academic Press, Amsterdam, Boston, 2013.

-
- [10] J. D. Hunter. Matplotlib: A 2D graphics environment. *Computing in Science & Engineering*, 9(3):90–95, 2007.
 - [11] U. Janssens and J. Graf. Was ist „Schock“? *Der Internist*, 45(3):258–266, Mar 2004.
 - [12] Gottfried Jetschke. *Mathematik der Selbstorganisation - Qualitative Theorie nichtlinearer dynamischer Systeme und gleichgewichtsferner Strukturen in Physik, Chemie und Biologie*. Springer-Verlag, Berlin Heidelberg New York, 2013.
 - [13] Alain Kalmar and K. Matthys. *Evaluation of arterial tonometry in a clinical setting*. PhD thesis, 01 2003.
 - [14] Reinhard Larsen. *Anästhesie und Intensivmedizin für die Fachpflege*. Springer eBooks : Medicine. Berlin ; Heidelberg, 9 edition, 2016.
 - [15] Jan Lunze. *Regelungstechnik 1: Systemtheoretische Grundlagen, Analyse und Entwurf einschleifiger Regelungen*. Lehrbuch. Springer Vieweg, Berlin, 11 edition, 2016.
 - [16] Jan Lunze. *Regelungstechnik 2 - Mehrgrößensysteme, Digitale Regelung*. Springer-Verlag, Berlin Heidelberg New York, 2016.
 - [17] Sheldon Magder. The meaning of blood pressure. *Critical Care*, 22:257, 10 2018.
 - [18] Karin Struhs-Wehr. *Stress*, pages 29–58. Springer Fachmedien Wiesbaden, Wiesbaden, 2017.
 - [19] Christoph Sucker and Peter Lanzer. Arteriosklerose und Mediasklerose Eine Gegenüberstellung zweier kalzifizierender Gefäßerkrankungen. *Medizinische Klinik*, 95, 05 2000.
 - [20] Sharven Taghavi, Aussama k Nassar, and Reza Askari. Hypovolemic Shock. In *StatPearls*. StatPearls Publishing, Treasure Island (FL), January 2024.
 - [21] Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, C J Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R. Harris,
-

Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1.0 Contributors. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272, 2020.

Abkürzungsverzeichnis

mmHg	Millimeter Quecksilbersäule
esv	Endsystolisches Volumen
edv	Enddiastolisches Volumen
T	Zeitmenge
X	Zustandsmenge
Φ	Phasenraum
$p(t)$	Ist-Wert zum Zeitpunkt t
$s(t)$	Soll-Wert zum Zeitpunkt t
$e(t)$	Regelabweichung zum Zeitpunkt t
A	Amplitude
R	Widerstand
R_r	Reihenwiderstand
R_p	Paralleler Widerstand
I	Strömungsstärke
ΔP	Treibende Druckdifferenz
r	Radius
l	Länge
η	Viskosität
MAP	Mittlerer Blutdruck
μm	Mikrometer
bpm	beats per minute (deut. Schläge pro Minute)
ml	Milliliter
s	Sekunden
mm	Millimeter

Abbildungsverzeichnis

1	Schematische Darstellung des Gefäßsystems und seiner zwei Kreisläufe	7
2	Schematische Darstellung des Herzens	8
3	Prinzip der indirekten Methode nach Riva-Rocci	10
4	Typische Blutdruckkurve mit wesentlichen Eigenschaften	11
5	Blutdruckkurven mit veränderten Amplituden	12
6	Allgemeine Darstellung eines Regelkreises	15
7	Prinzip des Stellglieds eines Mehrgrößensystems	16
8	Konzept des Regelkreises zur Blutdruckregulation	17
9	Zwei separate Sinuswellen für die Blutdruckkurve	18
10	Erste Blutdruckkurve auf 120/80 mmHg normalisiert über 10 Sekunden	18
11	Blutdruck in den Herzkammern	21
12	Druck in der Systole und Diastole	21
13	Querschnitt der verschiedenen Gefäße in mm	23
14	Lokalisierung von Sensoren und Ausbreitung von Druckwellen	27
15	Blutdruckkurven unter Verwendung der Parameter aus der Tab. 2/ Abb. 24	34
16	Ermittelte Drücke verschiedener Gefäße	35
17	Blutdruckkurven mit dynamischen Anpassungen	36
18	Ergebnis der Regelkreis-Simualtion	38
19	Input und Output des Regelkreises für die Stresssituation	40
20	Input und Output des Regelkreises für die Schocksituation	42
21	Input und Output des Regelkreises für die Radiusverengung	44
22	Input und Output des Regelkreises bei der Niereninsuffizienz	46
23	Blutdruckkurven mit markierten Maxima und Minima	58
24	Initiale Parameter für die Anwendungsbeispiele	61
25	Dynamische Parameter für die Anwendungsbeispiele	61

Tabellenverzeichnis

1	Gefäßarten und ihre Kenngrößen in μm	8
2	Initiale Parameter für die Anwendungsbeispiele	34
3	Zusätzliche dynamische Parameter	36
4	Soll-Parameter für den Regelkreis	37
5	Soll-Parameter für die Stresssituation	39
6	Soll-Parameter für die Schocksituation	41
7	Soll-Parameter für die Radiusverengung	43
8	Soll-Parameter bei der Niereninsuffizienz	45
9	Initiale Parameter für die Stresssituation	62
10	Dynamische Parameter für die Stresssituation	62
11	Initiale Parameter für die Schocksituation	63
12	Dynamische Parameter für die Schocksituation	63
13	Initiale Parameter für die Radiusverengung	64
14	Dynamische Parameter für die Radiusverengung	64
15	Initiale Parameter bei der Niereninsuffizienz	65
16	Dynamische Parameter bei der Niereninsuffizienz	65

Quelltextverzeichnis

1	Implementierte Funktion für die linke Herzkammer	20
2	Funktion zur Modellierung einer Aorta	22
3	Funktion zum Bestimmen des Widerstands R	24
4	Ausschnitt zur Bestimmung der Gesamtwiderstände aller Gefäßarten	25
5	Funktion zur Bestimmung des mittleren Blutdrucks	28
6	Implementierte Funktion für die rechte Herzkammer	57
7	Funktion zur Ermittlung des systolischen und diastolischen Blutdrucks	57
8	Funktion zur Darstellung der Druckwerte	58
9	Ein Abschnitt der Funktion, in dem der Regelkreis durchlaufen wird	59

Anlagen

Hier in den Anhängen sind unter anderem weitere wichtige Quelltexte aufgelistet, die jedoch zu umfangreich für den Haupttext sind, sowie kurze Quelltextausschnitte, auf die im Haupttext verwiesen wird. Für eine umfassende Übersicht der Implementierung ist der vollständige Quelltext auf GitHub unter folgendem Link verfügbar:

<https://github.com/lucas-diet/BPRegModel>.

```
1 def rightVentricle(self, shift=-0.5):
2     viskosityEffect = self.viscosity / 100
3     volumePressureConstant = 0.001
4     volumeEffect = volumePressureConstant * self.totalVolume
5     currentEDV = self.edv
6     currentESV = self.esv
7     currSV = self.strokeVolume
8
9     for i in range(0, len(self.time)):
10         t = self.time[i]
11
12         currentEDV = self.updateParameter(t, ctEDV, newEDV, currentEDV)
13         currentESV = self.updateParameter(t, ctESV, newESV, currentESV)
14         currSV = currentEDV - currentESV
15         elasticity = 1 + np.sin(2 * np.pi * self.heartRate * (t - shift) / 60)
16
17         if i == 0:
18             self.bloodVolume_RV[i] = currentEDV
19         else:
20             dVdt = currSV - elasticity * (self.bloodVolume_RV[i-1] - currentESV)
21             self.bloodVolume_RV[i] = self.bloodVolume_RV[i-1] + dVdt * self.dt
22
23         self.bloodPressure_RV[i] = elasticity * (self.bloodVolume_RV[i] - currentESV) *
24             0.15 + volumeEffect + viskosityEffect
```

Quelltext 6: Implementierte Funktion für die rechte Herzkammer

```
1 def findPeak(self, data):
2     systolicPeaks, _ = find_peaks(data)
3     diastolicPeaks, _ = find_peaks(-data)
4
5     filteredSystolicPeaks = systolicPeaks[0::2] # Jeden zweiten systolischen Peak
6     filteredDiastolicPeaks = diastolicPeaks[1::2] # Jeden zweiten diastolischen Peak
7
8     return filteredSystolicPeaks, filteredDiastolicPeaks
```

Quelltext 7: Funktion zur Ermittlung des systolischen und diastolischen Blutdrucks

```

1 def presPrinter(self, data):
2     types = ['Rechter Ventrikel', 'Linker Ventrikel', 'Aorta', 'Arterie', 'Arteriole',
5         'Kapillare', 'Venole', 'Vene', 'V. Cava']
6     type = iter(types)
7
8     for d in data:
9         print('\n', '##### ', next(type), ' #####', '\n')
10
11    sys, _ = self.findPeak(d)
12    _, dia = self.findPeak(d)
13
14    map = self.calculatePressure(d, sys, dia)
15
16
17    print('Systolischer Druck: ', np.mean(d[sys]), 'mmHg')
18    print('Diastolischer Druck: ', np.mean(d[dia]), 'mmHg')
19    print('Mittlerer Druck', map, 'mmHg')

```

Quelltext 8: Funktion zur Darstellung der Druckwerte

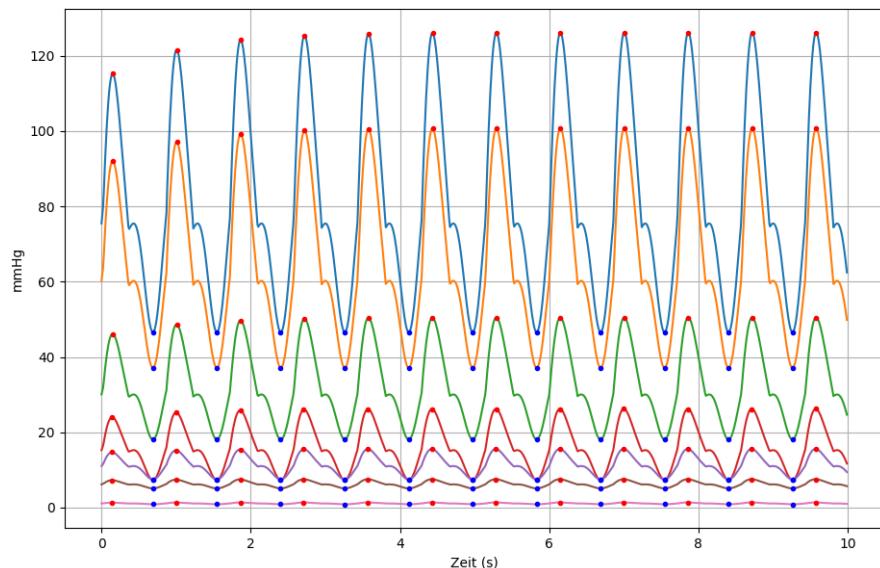


Abbildung 23: Blutdruckkurven mit markierten Maxima und Minima

```

1 def controlSystem(self, currVals, soHR, soLF, soVis, soTV, soEDV, soESV, runs, lens,
2     nums, ctSim):
3
4     h = Heart(self.radi, self.viscosity, self.heartRate, self.strokeVolume, self.edv,
5             self.esv, self.totalVolume, self.maxTime)
6
7     bs = BodySystem(self.radi, self.lumFactor, self.viscosity, self.heartRate,
8                     self.strokeVolume, self.edv, self.esv, self.totalVolume, self.maxTime)
9
10
11    h.heartSimulation(currVals[8], currVals[9], currVals[10], currVals[11])
12    bs.vesselSimulator(lens, nums, currVals[0], currVals[1], currVals[2], currVals[3],
13                      currVals[4], currVals[5], currVals[6], currVals[7])
14
15    isAorta, isArterie, isArteriol, isCapillare, isVenole, isVene, isVCava =
16        bs.getPressurs()
17
18    isRV = h.bloodPressure_RV
19    isLV = h.bloodPressure_LV
20
21    rwNorm = []
22    rwP = []
23
24
25    if len(currVals[1]) == 0:
26        pass
27    else:
28        self.heartRate = currVals[1][0]
29
30
31    tHR = [self.heartRate] + soHR
32    tVis = [self.viscosity] + soVis
33    tLF = [self.lumFactor] + soLF
34    tTV = [self.totalVolume] + soTV
35    tEDV = [self.edv] + soEDV
36    tESV = [self.esv] + soESV
37    tSim = [ctSim[0]] + ctSim
38
39
40    for i in range(0, runs):
41        nHR = tHR[i]
42        nVis = tVis[i]
43        nLF = tLF[i]
44        nTV = tTV[i]
45        nEDV = tEDV[i]
46        nESV = tESV[i]
47
48
49        soH = Heart(self.radi, nVis, nHR, self.strokeVolume, nEDV, nESV, nTV,
50                    self.maxTime)
51        soBS = BodySystem(self.radi, nLF, nVis, nHR, self.strokeVolume, nEDV, nESV, nTV,
52                         self.maxTime)
53        soS = Sensor(self.radi, nVis, nHR, self.strokeVolume, nEDV, nESV, self.maxTime)
54
55        soH.heartSimulation(ctSim, soEDV, ctSim, soESV)
56        soBS.vesselSimulator(lens, nums, ctSim, soHR, ctSim, soVis, currVals[4],
57                           currVals[5], currVals[6], currVals[7])

```

```

        currVals[5], ctSim, soTV)

42
43     ##### Regelstrecke #####
44     soAorta, soArterie, soArteriol, soCapillare, soVenole, soVene, soVCava =
45         soBS.getPressurs()
46     soRV = soH.bloodPressure_RV
47     soLV = soH.bloodPressure_LV
48     soPres = [soRV, soLV, soAorta, soArterie, soArteriol, soCapillare, soVenole,
49             soVene, soVCava]

50
51     ##### Regelabweichung #####
52     if i == 0:
53         rwRV = np.copy(soRV)
54         rwLV = np.copy(soLV)
55         rwAorta = np.copy(soAorta)
56         rwArterie = np.copy(soArterie)
57         ...
58
59         t = tSim[i]
60         rwRV[t:t+1] = isRV[t:t+1] - soRV[t:t+1]
61         rwLV[t:t+1] = isLV[t:t+1] - soLV[t:t+1]
62         rwAorta[t:t+1] = isAorta[t:t+1] - soAorta[t:t+1]
63         rwArterie[t:t+1] = isArterie[t:t+1] - soArterie[t:t+1]
64         ...
65
66         rwPres = [rwRV, rwLV, rwAorta, rwArterie, rwArteriol, rwCapillare, rwVenole,
67                 rwVene, rwVCava]
68         rwMins = [np.min(rwRV), np.min(rwLV), np.min(rwAorta), np.min(rwArterie),
69                  np.min(rwArteriol), np.min(rwCapillare), np.min(rwVenole), np.min(rwVene),
70                  np.min(rwVCava)]
71
72         for j in range(len(rwPres)):
73             normalized_array = []
74             array_length = len(rwPres[j])
75             for k, l in enumerate(rwPres[j]):
76                 diff = l - rwMins[j]
77                 if diff != 0:
78                     normalized_array.append(diff)
79                 else:
80                     # Den Mittelwert der benachbarten Elemente berechnen
81                     if k == 0: # Erstes Element
82                         new_value = (rwPres[j][k + 1] - rwMins[j]) / 2
83                     elif k == array_length - 1: # Letztes Element
84                         new_value = (rwPres[j][k - 1] - rwMins[j]) / 2
85                     else: # Mittelwert der beiden Nachbarn
86                         new_value = ((rwPres[j][k - 1] - rwMins[j]) + (rwPres[j][k + 1]
87                                     - rwMins[j])) / 2
88                     normalized_array.append(new_value)
89             rwNorm.append(normalized_array)

```

```

84
85     isRV = rwRV
86     isLV = rwLV
87     isAorta = rwAorta
88     isArterie = rwArterie
89     ...
90
91     for i in rwNorm[0:9]:
92         rwP.append(np.array(i))
93
94     soS.presPrinter(rwP)
95     print('\n', '#####')
96     self.controlSystemPlotter(rwP, bs.time)

```

Quelltext 9: Ein Abschnitt der Funktion, in dem der Regelkreis durchlaufen wird

```

radi = [20000, 4000, 20, 8, 20, 5000, 30000]    # in µm
viscosity = 50                                     # Wert zwischen 0 und 100
heartRate = 60
edv = 100                                         # Enddiastolische Volumen
esv = 40                                           # Endzystolisches Volumen
strokeVolume = edv - esv                          # Schlagvolumen
maxTime = 10                                        # in Sekunden
totalVolume = 5000                                 # in ml
nums = [1, 2, 4, 16, 4, 2, 1]                      # Anzale der Gefäße
lens = [200, 150, 100, 50, 100, 150, 300]          # in mm :: Längen der Gefäße
##### Extra Parameter für BodySystem #####
lims = [-17, 17]                                    # Achsenlänge; Radienplot
lumFactor = [1, 1, 1, 1, 1, 1, 1]                   # Faktor zum skalieren von Radien
#####

```

Abbildung 24: Initiale Parameter für die Anwendungsbeispiele

```

# Herzfrequenz
ctHR = [2, 4, 6, 9]; newHR = [40, 90, 80, 50]

# Viskosität
#ctVis = [5, 15, 20, 25]; newVis = [50, 60, 55, 50]

# Radius
#ctRadius = [2, 5, 7, 18]; newRadius = [0.9, 0.6, 0.4, 0.1]

# Gesamtvolumen
ctVol = [3, 5, 7, 8]; newVol = [5050, 6200, 7450, 5000]

#####
# Folgende Parameter müssen alle aktiv oder inaktiv sein
#####

# Enddiastolisches Volumen
#ctEDV = [5, 15, 20, 25]; newEDV = [50, 70, 70, 60]

# Endzystolisches Volumen
#ctESV = [5, 15, 20, 25]; newESV = [60, 30, 45, 60]

```

Abbildung 25: Dynamische Parameter für die Anwendungsbeispiele

Einflussgröße	Werte	Einheit
Radien	[20000, 4000, 20, 8, 20, 5000, 30000]	μm
Viskosität	50	%
Herzfrequenz	60	bpm
edv	70	ml
esv	20	ml
Zeit	30	s
Gesamtvolumen	5000	ml
Gefäßanzahl	[1, 2, 4, 16, 4, 2, 1]	#
Gefäßlänge	[200, 150, 100, 50, 100, 150, 300]	mm
Radiusfaktor	[1, 1, 1, 1, 1, 1, 1]	

Tabelle 9: Initiale Parameter für die Stresssituation

Einflussgröße	Wertbezeichnung	Werte	Einheit
Herzfrequenz	Zeitpunkte	[5, 15, 20, 25]	s
	Neue Werte	[50, 54, 50, 51]	bpm
Viskosität	Zeitpunkte	[5, 15, 20, 25]	s
	Neue Werte	[30, 35, 30, 30]	bpm
edv	Zeitpunkte	[5, 15, 20, 25]	s
	Neue Werte	[70, 70, 70, 70]	bpm
esv	Zeitpunkte	[5, 15, 20, 25]	s
	Neue Werte	[30, 25, 25, 30]	bpm

Tabelle 10: Dynamische Parameter für die Stresssituation

Einflussgröße	Werte	Einheit
Radien	[20000, 4000, 20, 8, 20, 5000, 30000]	μm
Viskosität	50	%
Herzfrequenz	70	bpm
edv	110	ml
esv	60	ml
Zeit	60	s
Gesamtvolumen	5000	ml
Gefäßanzahl	[1, 2, 4, 16, 4, 2, 1]	#
Gefäßlänge	[200, 150, 100, 50, 100, 150, 300]	mm
Radiusfaktor	[1, 1, 1, 1, 1, 1, 1]	

Tabelle 11: Initiale Parameter für die Schocksituation

Einflussgröße	Wertbezeichnung	Werte	Einheit
Herzfrequenz	Zeitpunkte	[2, 7, 20, 45]	s
	Neue Werte	[50, 57, 60, 51]	bpm
Viskosität	Zeitpunkte	[2, 32, 40, 50]	s
	Neue Werte	[50, 55, 50, 50]	bpm
edv	Zeitpunkte	[5, 15, 20, 25]	s
	Neue Werte	[110, 120, 125, 110]	bpm
esv	Zeitpunkte	[5, 15, 20, 25]	s
	Neue Werte	[60, 50, 55, 60]	bpm

Tabelle 12: Dynamische Parameter für die Schocksituation

Einflussgröße	Werte	Einheit
Radien	[20000, 4000, 20, 8, 20, 5000, 30000]	μm
Viskosität	50	%
Herzfrequenz	70	bpm
edv	110	ml
esv	60	ml
Zeit	300	s
Gesamtvolumen	5000	ml
Gefäßanzahl	[1, 2, 4, 16, 4, 2, 1]	#
Gefäßlänge	[200, 150, 100, 50, 100, 150, 300]	mm
Radiusfaktor	[1, 1, 1, 1, 1, 1, 1]	

Tabelle 13: Initiale Parameter für die Radiusverengung

Einflussgröße	Wertbezeichnung	Werte	Einheit
Herzfrequenz	Zeitpunkte	[1, 140, 210, 280]	s
	Neue Werte	[60, 67, 60, 60]	bpm
Viskosität	Zeitpunkte	[70, 140, 210, 280]	s
	Neue Werte	[50, 50, 50, 50]	bpm

Tabelle 14: Dynamische Parameter für die Radiusverengung

Einflussgröße	Werte	Einheit
Radien	[20000, 4000, 20, 8, 20, 5000, 30000]	μm
Viskosität	50	%
Herzfrequenz	70	bpm
edv	80	ml
esv	20	ml
Zeit	60	s
Gesamtvolumen	5000	ml
Gefäßanzahl	[1, 2, 4, 16, 4, 2, 1]	#
Gefäßlänge	[200, 150, 100, 50, 100, 150, 300]	mm
Radiusfaktor	[1, 1, 1, 1, 1, 1, 1]	

Tabelle 15: Initiale Parameter bei der Niereninsuffizienz

Einflussgröße	Wertbezeichnung	Werte	Einheit
Herzfrequenz	Zeitpunkte	[2, 15, 20, 40]	s
	Neue Werte	[60, 57, 66, 61]	bpm
Viskosität	Zeitpunkte	[2, 15, 20, 40]	s
	Neue Werte	[50, 60, 55, 50]	bpm
Gesamtvolumen	Zeitpunkte	[3, 5, 7, 8]	s
	Neue Werte	[5000, 5000, 5000, 5000]	bpm
edv	Zeitpunkte	[2, 5, 7, 10]	s
	Neue Werte	[70, 75, 70, 70]	bpm
esv	Zeitpunkte	[2, 15, 20, 40]	s
	Neue Werte	[20, 20, 25, 25]	bpm

Tabelle 16: Dynamische Parameter bei der Niereninsuffizienz



Eigenständigkeitserklärung

1. Hiermit versichere ich, dass ich die vorliegende Arbeit - bei einer Gruppenarbeit die von mir zu verantwortenden und entsprechend gekennzeichneten Teile - selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe.
Ich trage die Verantwortung für die Qualität des Textes sowie die Auswahl aller Inhalte und habe sichergestellt, dass Informationen und Argumente mit geeigneten wissenschaftlichen Quellen belegt bzw. gestützt werden. Die aus fremden oder auch eigenen, älteren Quellen wörtlich oder sinngemäß übernommenen Textstellen, Gedankengänge, Konzepte, Grafiken etc. in meinen Ausführungen habe ich als solche eindeutig gekennzeichnet und mit vollständigen Verweisen auf die jeweilige Quelle versehen. Alle weiteren Inhalte dieser Arbeit ohne entsprechende Verweise stammen im urheberrechtlichen Sinn von mir.
2. Ich weiß, dass meine Eigenständigkeitserklärung sich auch auf nicht zitierfähige, generierende KI-Anwendungen (nachfolgend „generierende KI“) bezieht.
Mir ist bewusst, dass die Verwendung von generierender KI unzulässig ist, sofern nicht deren Nutzung von der prüfenden Person ausdrücklich freigegeben wurde (Freigabeerklärung). Sofern eine Zulassung als Hilfsmittel erfolgt ist, versichere ich, dass ich mich generierender KI lediglich als Hilfsmittel bedient habe und in der vorliegenden Arbeit mein gestalterischer Einfluss deutlich überwiegt. Ich verantworte die Übernahme der von mir verwendeten maschinell generierten Passagen in meiner Arbeit vollumfänglich selbst.
Für den Fall der Freigabe der Verwendung von generierender KI für die Erstellung der vorliegenden Arbeit wird eine Verwendung in einem gesonderten Anhang meiner Arbeit kenntlich gemacht.
Dieser Anhang enthält eine Angabe oder eine detaillierte Dokumentation über die Verwendung generierender KI gemäß den Vorgaben in der Freigabeerklärung der prüfenden Person.
Die Details zum Gebrauch generierender KI bei der Erstellung der vorliegenden Arbeit inklusive Art, Ziel und Umfang der Verwendung sowie die Art der Nachweispflicht habe ich der Freigabeerklärung der prüfenden Person entnommen.
3. Ich versichere des Weiteren, dass die vorliegende Arbeit bisher weder im In- noch im Ausland in gleicher oder ähnlicher Form einer anderen Prüfungsbehörde vorgelegt wurde oder in deutscher oder einer anderen Sprache als Veröffentlichung erschienen ist.
4. Mir ist bekannt, dass ein Verstoß gegen die vorbenannten Punkte prüfungsrechtliche Konsequenzen haben und insbesondere dazu führen kann, dass meine Prüfungsleistung als Täuschung und damit als mit „nicht bestanden“ bewertet werden kann. Bei mehrfachem oder schwerwiegendem Täuschungsversuch kann ich befristet oder sogar dauerhaft von der Erbringung weiterer Prüfungsleistungen in meinem Studiengang ausgeschlossen werden.

Jena, 06.07.2024

Ort und Datum

L. Dietrich

Unterschrift



Freigabeerklärungen

Von der prüfenden Person konkret festzulegender Geltungsbereich
(z.B. Thema, Veranstaltung, Zeitraum):

Projektmodul Bioinformatik SoSe 2024 für Lucas Dietrich

Die folgenden Tools müssen nicht als Hilfsmittel deklariert werden und dürfen zur Erstellung von Seminar- und Abschlussarbeiten genutzt werden, auch wenn sie KI-gestützt sind:

- Textverarbeitungsprogramme, z. B. Word oder OpenOffice Writer
- Tabellenkalkulation, z. B. Excel oder LibreOffice Calc
- Rechtschreib- und Grammatikprüfung sowie -korrektur inkl. Werkzeugen in Textverarbeitungsprogrammen, z. B. DeepKomma
- Suchmaschinen
- Digitale Wörterbücher und Thesaurus
- Mindmap-Tools
- Recherchetools, z. B. wissenschaftliche Literatursuche via PubMed
- Recherchetools, die keine Ideen generieren, z. B. wissenschaftliche Literatursuche via Google Scholar
- eigene Erweiterungen...

Seminar- und Abschlussarbeiten sollen zuvorderst die Gedanken, Ideen und Erkenntnisse der Verfasserin bzw. des Verfassers beinhalten. Es muss klar erkennbar sein, ob und an welchen Stellen diese durch generierende KI ergänzt wurden. Im Folgenden finden Sie eine Auflistung der für die betroffene Prüfung erlaubten KI-Werkzeuge und wie deren Einsatz zu kennzeichnen ist. Die Dokumentation soll in einem separaten KI-Quellenverzeichnis erfolgen.

Erlaubte Werkzeuge:

- Textgenerierende** KI-Werkzeuge: Die wörtliche oder inhaltliche Übernahme aus KI-generierten Textquellen (einschließlich Quellcodes, mathematischen Ausdrücken etc.) ist erlaubt.
- Bildgenerierende** KI-Werkzeuge: Die direkte Übernahme aus KI-generierten Bildquellen ist erlaubt.
- Bildverarbeitende** KI-Werkzeuge: Die direkte Übernahme aus Bildquellen, die mittels KI-Werkzeug weiterverarbeitet werden, ist erlaubt.
- Übersetzung** durch KI-Werkzeuge: Die wörtliche Übernahme aus KI-generierten Übersetzungen ist erlaubt.
- eigene Erweiterungen...

Zur Dokumentation des KI-Einsatzes ist mindestens die Bezeichnung des verwendeten Werkzeugs und die Version erforderlich. Darüber hinaus sind die folgenden Angaben zu machen:

- Angabe der direkt übernommenen Generate des KI-Werkzeugs.
- Zeit und Datum der Nutzung des Werkzeugs.
- Die vollständige Eingabe in das Werkzeug, z. B. durch Prompts.
- Die vollständige Ausgabe des Werkzeugs.
- Falls vorhanden: Die Internetadresse, unter der das Werkzeug aufgerufen wurde.
- eigene Erweiterungen...

Bei der mehrstufigen Nutzung von KI-Werkzeugen in Form einer Sequenz von Überarbeitungen eines Inhalts sind alle Zwischenschritte einzeln zu dokumentieren. Dies, sowie Ein- und Ausgabe, können ggf. mit Links zu unveränderlichen Chatprotokollen sichergestellt werden.

15.05.2024

Datum

Thomas Krieg

Unterschrift der prüfenden Person

Dokumentation entsprechend der Freigabeerklärung

Gemäß der Freigabeerklärung auf Seite 68 finden Sie hier eine kurze Dokumentation zur Verwendung von KIs.

Zur Korrektur und Verbesserung des Ausdrucks wurde die Version 3.5 von ChatGPT verwendet.