

---

# **Erstellung eines Demonstrationsversuchs für Unterwasser-Akustik**

---

**Zulassungsarbeit**

**Marco Utz**

**Erlangen Centre for Astroparticle Physics**

**Physikalisches Institut 1**

**Friedrich-Alexander-Universität**



**Erlangen-Nürnberg**

**März 2011**

## Inhaltsverzeichnis

1	Einleitung.....	3
2	Theoretische Grundlagen.....	6
2.1	Schallausbreitung.....	6
2.2	Schallwellen an Grenzflächen.....	8
2.3	Folgerungen und Annahmen im Model.....	9
3	Hauptkomponenten des Modells.....	10
3.1	Messaufbau.....	12
3.1.1	Befestigungsort der Sensoren.....	12
3.1.2	Befestigungsart der Sensoren.....	21
3.1.3	Verstärkung des Signals.....	21
4	Rekonstruktionsstrategien.....	22
4.1	Mathematische Überlegungen zur Bestimmung der Koordinaten des Klopfpunktes aus den Messdaten.....	22
4.2	Bestimmung der Koordinaten des Klopfpunktes aus den Messdaten.....	24
4.3	Simulation zur Überprüfung der Anwendbarkeit des Suchalgorithmus.....	24
4.4	Ermittlung des Anfangszeitpunktes der einzelnen Sensoren.....	27
4.4.1	Ermittlung durch einen einfachen Trigger.....	28
4.4.2	Ermittlung mittels der ersten Knickstellen.....	29
4.4.3	Ermittlung mittels der ersten Extremstellen des Signals.....	30
4.4.3.1	Optimierung mittels Fallunterscheidung.....	30
4.4.4	Ermittlung mittels Kombination erster Knickstellen mit ersten Extremstellen.....	32
4.4.5	Fehlereinflüsse bei der Messung und was zu beachten ist.....	33
4.4.6	Abschätzung der Größe systematischer und statistischer Fehler.....	33
4.4.7	Verbesserung der Ergebnisse - Mittelwert aus den Kombinationen mehrerer Sensoren.....	35
5	Darstellung der Ergebnisse.....	36
5.1	Wiederholbarkeit der Messung eines Klopfpunktes.....	36
5.2	Darstellung der wesentlichen Algorithmen an ausgewählten Beispielen.....	39
6	Zusammenfassung und Ausblick.....	41
<b>Anhang</b>		
7	Aufbau des Programms und Implementierung der Algorithmen.....	46
7.1	Hauptprogramm „Berechnungsprogramm.vi“ (Teil I).....	47
7.1.1	Unterprogramm „letzte 3 Runden in Array.vi“.....	49
7.1.2	Unterprogramm „Mittelwert.vi“.....	49
7.1.3	Unterprogramm „Knickfinder.vi“.....	49
7.1.4	Unterprogramm „Extremafinder.vi“.....	50
7.2	Hauptprogramm „Berechnungsprogramm.vi“ (Teil II).....	51
7.2.1	Fall A – schwaches Klopfen.....	52
7.2.1.1	Unterprogramm „10 Kombinationen der 5 Sensoren bei Knicks.vi“ (Teil I).....	52
7.2.1.2	Unterprogramm „Werte in Array schreiben.vi“.....	52
7.2.1.2.1	Unterprogramm „Werte aus Array auslesen.vi“.....	53
7.2.1.3	Unterprogramm „10 Kombinationen der 5 Sensoren bei Knicks.vi“ (Teil II).....	53
7.2.2	Fall B – starkes Klopfen.....	54
7.2.2.1	Unterprogramm „10 Kombinationen der 5 Sensoren bei Extrema.vi“ (Teil I).....	54
7.2.2.2	Unterprogramm „Fallunterscheidung.vi“.....	55
7.2.2.2.1	Unterprogramm „nächstes Extremum Finder.vi“.....	56
7.2.3	Unterprogramm „Punkt suchen.vi“.....	57
7.3	Hauptprogramm „Berechnungsprogramm.vi“ (Teil III).....	58
7.3.1	Unterprogramm „Nullpaarestreicher.vi“.....	58
7.3.2	Unterprogramm „schlechte Werte Vernichter.vi“.....	59
7.3.3	Unterprogramm „Graphik.vi“.....	60
8	Kurze Erläuterung zur Handhabung des Programms.....	61

# 1 Einleitung

Am Neutrino-Teleskop ANTARES wird mittels optischer Sensoren das Cherenkovlicht detektiert, das von geladenen Teilchen (Myonen) im Wasser erzeugt wird, welche bei der Wechselwirkung von Neutrinos mit Sauerstoffkernen oder auch Wasserstoffkernen entstehen. Diese Teilchen sind hochrelativistisch und bewegen sich im Wasser mit einer Geschwindigkeit, die über der dortigen Lichtgeschwindigkeit liegt. Das Licht wird auf einem Kegelmantel in Flugrichtung emittiert, dessen Öffnungswinkel von der Geschwindigkeit des Teilchens und vom Brechungsindex des Mediums abhängt. Das von den Photosensoren nachgewiesene Lichtsignal wird elektronisch aufbereitet, digitalisiert an Land geschickt und abgespeichert. Mit Hilfe dieser gespeicherten Daten wird die zeitliche Struktur aller dieser Signale analysiert und die Spuren der Myonen durch den Detektor rekonstruiert. Das Neutrino-Teleskop besteht aus zwölf "strings", die jeweils am Boden verankert sind und von einer Boje am 480 m entfernten Ende straff nach oben gehalten werden. An den "strings" befinden sich auf 25 "storeys" je drei Photosensoren, die das Cherenkovlicht nachweisen [6]. Die Anlage nutzt ein Gesamtvolumen von etwa  $0,03 \text{ km}^3$ . Für die Zukunft ist geplant, sensitivere Neutrino-Detektoren (KM3NeT) von etwa  $1 \text{ km}^3$  Größe zu installieren. Allerdings gibt es, da Licht eine Absorptionslänge von nur etwa 60 m hat, eine obere Grenze für das Detektionsvolumen solcher optischen Cherenkov-Detektoren.

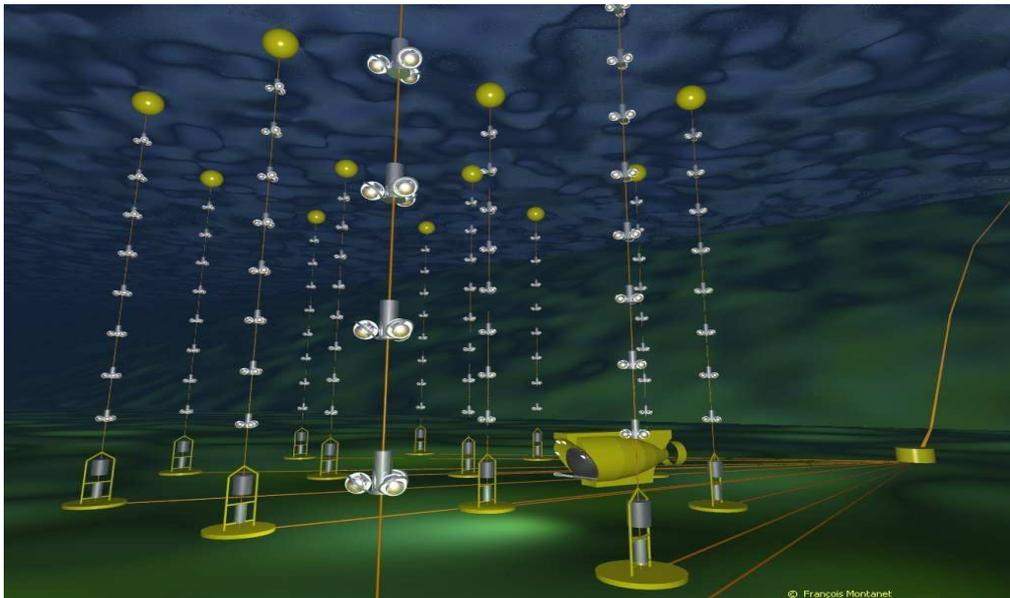


Abbildung 1-1: Künstlerische Darstellung des ANTARES-Neutrino-Teleskops [5]

Eine Möglichkeit zur weiteren Vergrößerung dieses nutzbaren Volumens bietet das sog. thermoakustische Modell, das von Askariyan im Jahre 1979 entwickelt wurde. Neutrinos erzeugen durch Wechselwirkung mit Kernen/Nukleonen im Wasser eine hadronische Kaskade mit elektromagnetischen Anteilen. Das thermoakustische Modell beschreibt die Erzeugung akustischer Signale durch diese Kaskade, die ihre Energie entlang ihrer Bahn an das umgebende Wasser abgibt und es damit aufheizt. Das erhitzte Wasser expandiert und erzeugt einen hydrodynamischen Puls, also ein Schallsignal, welches charakteristische Eigenschaften besitzt. Der Vorteil gegenüber der Detektion von

Cherenkov–Strahlung liegt in der weit größeren Absorptionslänge von Schall in Wasser (diese liegt bei dem hier relevanten Frequenzbereich bei etwa 1000 m) [1]. Zur Untersuchung dieser Nachweismethode sind neben den oben erwähnten optischen Sensoren auch eine Reihe akustischer Sensoren (Hydrophone) an bestimmten Stellen der Strings angebracht.

### Ziel dieser Arbeit

Ziel dieser Arbeit ist es ein „stand alone“ Modell zu schaffen, das das Prinzip einer akustischen Quellenrekonstruktion aufgezeigt und nachstellt. Dieses Modell soll in erster Linie dazu dienen, dem Laien die Arbeit des Lehrstuhls in ANTARES näher zu bringen. Dazu wird ein Aquarium herangezogen, welches mit Wasser gefüllt ist. Mittels Piezosensoren werden Klopfsignale analysiert, die an der Frontfläche des Aquariums eingeleitet werden. Diese Analyse beinhaltet folgende Schritte:

- Mittels eines Suchalgorithmus werden zunächst Anfangszeitpunkte für die Detektion des Klopfsignals durch die einzelnen Sensoren bestimmt
- Aus diesen Anfangszeitpunkten werden anschließend Laufzeitdifferenzen zwischen dem Klopfpunkt und dem jeweiligen Sensor errechnet
- Ein weiterer Suchalgorithmus sucht systematisch nach einem möglichen Emissionspunkt, für den diese gemessenen Laufzeitdifferenzen zwischen Emissionspunkt (Klopfpunkt) und den einzelnen Sensoren am wahrscheinlichsten ist
- Dieses systematische Suchen wird mit den Messwerten aus insgesamt 10 Sensorkonfigurationen (mit je drei Sensoren) durchgeführt.
- Anschließend wird aus den so ermittelten Koordinaten der Mittelwert bestimmt und das Ergebnis graphisch dargestellt (siehe Abbildung 1-2).
- Das gesamte Berechnungsprogramm wurde mit dem graphischen Programmiersystem Labview erstellt.

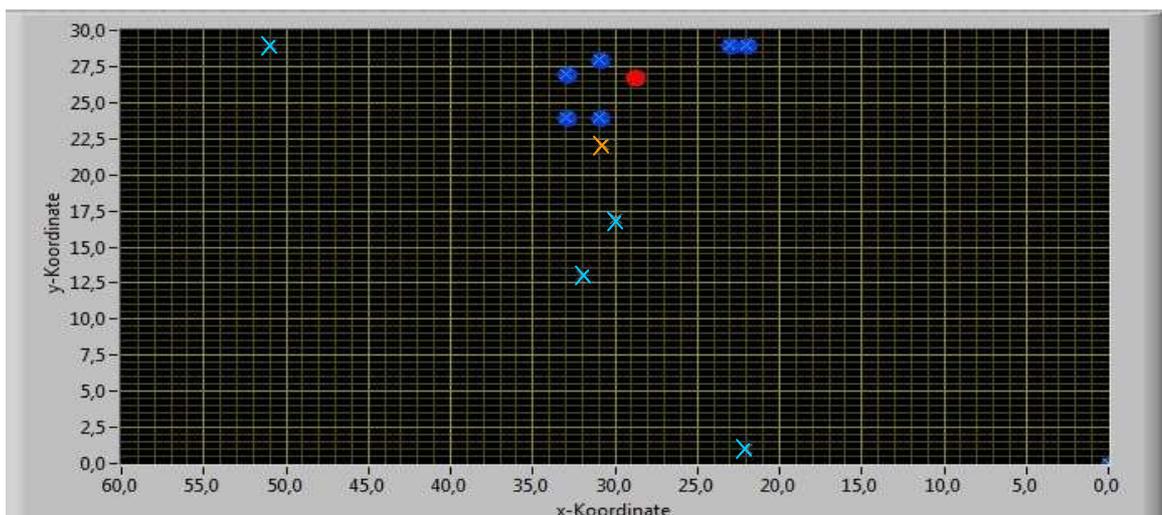


Abbildung 1-2: Graphische Darstellung der vom Programm errechneten Klopfpunkte („rot“: letztlich bestimmter Mittelwert der aus mehreren Konfigurationen berechneten Werten „blau“. Das Programm schließt iterativ die schlechtesten Messwerte aus (die 4 hellblauen Kreuze) Oranges Kreuz: Mittelwert vor dem Ausschließen)

---

## Behandelte Fragen

Die Erstellung eines solchen Modells wirft einige Problemstellungen auf, die im Zuge dieser Arbeit gelöst wurden:

- Können die Sensoren beliebig angebracht werden, oder müssen Randbedingungen bedacht werden?
- Wie kann man das Schallsignal, das direkt vom Emissionspunkt zum Sensor gelangt, von der Vielzahl an Signalen, die unter anderem aufgrund von Reflexionen und durch Schallanteile durch verschiedene Medien entstehen, zeitlich trennen (Signalidentifikation)?
- Sind die Sensoren sensibel genug (braucht man evtl. einen Verstärker)?
- Wie legt man den Anfangszeitpunkt der Signalerkennung fest?
- Wie lassen sich aus den von den Sensoren aufgenommenen Signalen die Koordinaten des Emissionspunktes berechnen (Rekonstruktionsalgorithmen)?

## Aufbau dieser Arbeit

Im Folgenden werden zunächst einige theoretische Grundlagen zur Schallausbreitung insbesondere an Grenzflächen dargestellt. Anschließend werden die im Modell verwendeten Komponenten genauer beschrieben. Im Anschluss folgt ein Kapitel über Untersuchungen, mit denen die Frage nach einer möglichst guten Sensorpositionierung beantwortet werden kann. Das nächste Kapitel beschreibt die mathematischen Überlegungen, die dem Modell zugrunde liegen und die letztlich gewählte Rekonstruktionsstrategie für die Ortung des Emissionspunktes. Am Ende des Kapitels werden die wesentlichen Fehlereinflüsse auf die Messungen dargestellt und Überlegungen zur Abschätzung des Fehlers für die Ortung aufgezeigt. Danach folgt ein Kapitel in dem einige Ergebnisse an Beispielen dargestellt werden. Im Kapitel 7 werden die einzelnen Komponenten (vor allem die verwendeten Algorithmen) des Programms genauer erklärt. Zuletzt folgen eine kurze Anleitung zur Handhabung des Programms und der Ausblick.

## 2 Theoretische Grundlagen

In diesem Kapitel werden die für das Modell wichtigen Grundlagen der Schallausbreitung dargestellt. Für den interessierten Leser wird an dieser Stelle auf [2] und [3] hingewiesen; hieraus wurden die folgenden Ausführungen größtenteils entnommen.

### 2.1 Schallausbreitung

#### Schallgeschwindigkeit

Schall ist die Ausbreitung lokaler Druckschwankungen in Medien. Dabei ist die Schallgeschwindigkeit in verschiedenen Medien unterschiedlich groß. Für Schallgeschwindigkeit in Flüssigkeiten und Gasen gilt die Gleichung

$$c = \sqrt{\frac{\kappa}{\rho}} \quad \text{Gleichung 1}$$

wobei  $\kappa$  das Kompressionsmodul und  $\rho$  die Dichte des Mediums ist.

Die Schallgeschwindigkeit im Wasser (bei 20°C) ist mit etwa  $c_w = 1480 \text{ m/s}$  größer als die in (trockener) Luft (bei 20 °C). Hier beträgt die Schallgeschwindigkeit  $c_L = 344 \text{ m/s}$ . Diese und weitere Werte, die im Modell relevant sind, finden sich in Tabelle 1.

#### Schalldruck

Der Schalldruck  $p$  ist die Druckänderung in einem homogenen, kompressiblen Medium durch Kompression der Moleküle. Auf die Begrenzungsflächenbereiche des Wellenträgers übt der Schalldruck eine Normalkraft  $F_N$  aus. Es gilt:

$$p = \frac{F_N}{A} \quad \text{Gleichung 2}$$

Hierbei ist  $A$  das Flächenelement der Begrenzungsfläche.

Der Effektivwert des Schalldrucks harmonischer Schallwellen bei sinusförmiger Erregung beträgt:

$$p_{\text{eff}} = \frac{\hat{p}}{\sqrt{2}} \quad \text{Gleichung 3}$$

Wobei  $\hat{p}$  die Schalldruckamplitude ist.

Schallkennimpedanz (Schallwiderstand)

Die Schallkennimpedanz  $Z_0$  ist ein Maß für die Geschwindigkeit, mit der die Moleküle eines Mediums auf eine Druckstörung reagieren. Für ebene Wellen gilt mit der Schallgeschwindigkeit  $c$  im Medium:

$$Z_0 = \rho \cdot c \quad \text{Gleichung 4}$$

Die Schallkennimpedanz  $Z_0$  hängt über die Dichte  $\rho$  und die Schallgeschwindigkeit  $c$  vom statischen Druck  $p_s$  und der Temperatur  $T$  des Mediums ab. In der Tabelle 1 sind die Schallkennimpedanzen, für die Medien, die im vorliegenden Modell relevant sind mit aufgelistet.

	Schallgeschwindigkeit in $\frac{m}{s}$	Schallkennimpedanz $Z_0$ in $\frac{kg}{m^2 \cdot s}$
Glas	5300	$13,0 \cdot 10^6$
Wasser bei 20°C	1480	$1,48 \cdot 10^6$
Luft bei 20°C	344	416

Tabelle 1: Schallgeschwindigkeit und Schallkennimpedanz der im Modell vorkommenden Medien

Schallintensität

Die Schallintensität einer harmonischen Schallwelle beträgt:

$$I = \frac{p_{eff}^2}{Z} \quad \text{Gleichung 5}$$

Schalleistung

Die Schalleistung  $P$  einer Schallquelle ergibt sich zu

$$P = \int_A I dA \quad \text{Gleichung 6}$$

wenn auf ein Flächenelement  $dA$  die Schallintensität  $I$  einfällt.

Schallintensität einer kugelförmigen (punktförmigen) Schallquelle

Für eine punktförmige Quelle mit der Schalleistung  $P$  gilt für die Schallintensität im Abstand  $r$ :

$$I = \frac{P}{4 \cdot \pi \cdot r^2} \quad \text{Gleichung 7}$$

## 2.2 Schallwellen an Grenzflächen

Beim Übergang einer Schallwelle ( $p_e, I_e$ ) von einem Medium 1 in ein anderes Medium 2, wird diese zum Teil reflektiert ( $p_r, I_r$ ) und zum Teil dringt sie als transmittierte Welle in das zweite Medium ein ( $p_t, I_t$ ). Dieser transmittierte Teil wird auf eine bestimmte Strecke (siehe weiter unten „Punkt 5“ unter Folgerungen und Annahmen) absorbiert und in Wärme umgewandelt.  $p$  ist dabei jeweils der Schalldruck und  $I$  die Intensität des jeweiligen Schallanteils. Für die Beschreibung der Intensitäten der verschiedenen Anteile definiert man drei Größen:

- Schallreflexionsgrad
- Schalltransmissionsgrad
- Schallabsorptionsgrad

### Schallreflexionsgrad

Bezieht man die Intensität  $I_r$  des an einer Grenzfläche reflektierten Schallanteils auf die Intensität  $I_e$  des auf diese Grenzfläche auftreffenden Schallanteils, so erhält man eine dimensionslose Größe mit einem Wertebereich von 0% bis 100%. Diese Größe wird als Schallreflexionsgrad  $\rho_s$  bezeichnet. Für den Schallreflexionsgrad einer Grenzfläche zwischen einem Medium 1 mit der Schallkennimpedanz  $Z_1$  und einem Medium 2 mit der Schallkennimpedanz  $Z_2$  gilt bei senkrechtem Einfall:

$$\rho_s = \left( \frac{Z_2 - Z_1}{Z_2 + Z_1} \right)^2 \quad \text{Gleichung 8}$$

### Schalltransmissionsgrad

Bezieht man die Intensität  $I_t$  des transmittierten Schallanteils auf die Intensität  $I_e$  des auf diese Grenzfläche auftreffenden Schallanteils, erhält man ebenfalls eine solche dimensionslose Größe – den Schalltransmissionsgrad  $\tau_s$ .

Für diesen gilt:

$$\tau_s = \left( \frac{4Z_2Z_1}{Z_2 + Z_1} \right)^2 = 1 - \rho_s \quad \text{Gleichung 9}$$

### Schallabsorptionsgrad

Der in das Medium 2 transmittierte Anteil  $I_t$  des Schalls wird zu einem Teil  $I_a$  absorbiert und in Wärme umgewandelt. Diesen als Absorptionsgrad bezeichneten Wert erhält man, wenn man den absorbierten Teil  $I_a$  der transmittierten Intensität  $I_t$  auf die Intensität  $I_e$  des auf diese Grenzfläche auftreffenden Schallanteils bezieht.

## Brechung

Ist die Schallgeschwindigkeit im Medium 2 größer als im Medium 1, so wird ein schräg einfallendes Schallsignal vom Einfallslot weg, andernfalls zum Einfallslot hin gebrochen. Analog zum Snellius'schen Brechungsgesetz aus der Optik lautet das Brechungsgesetz der Akustik:

$$\frac{\sin \vartheta_1}{c_1} = \frac{\sin \vartheta_2}{c_2}$$

Gleichung 10

### 2.3 Folgerungen und Annahmen im Model

- Die Schallgeschwindigkeit kann im Modell als konstant angenommen werden
- Aufgrund der sehr verschiedenen Schallkennimpedanzen von Luft verglichen mit Glas oder Wasser dringt nach Gleichung 8 so gut wie kein Schall vom Wasser/Glas nach außen an die Luft. Für den Schallreflexionsgrad an der Grenzfläche zwischen Glas und Luft ergibt sich nach Gleichung 8 unter Verwendung der Werte aus Tabelle 1:

$$\rho_s = \left( \frac{13,0 \cdot 10^6 - 416}{13,0 \cdot 10^6 + 416} \right)^2 = 0,99987$$

d.h. der Schall wird in guter Näherung zu 100% reflektiert.

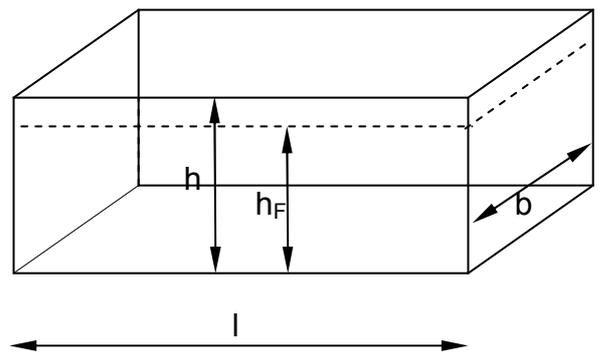
- Im vorliegenden Modell wird davon ausgegangen, dass es sich bei der Schallquelle (dem Klopfpunkt) um eine punktförmige Quelle handelt, von der aus sich der Schall in alle Richtungen gleichmäßig ausbreitet. Die Intensität ist unter dieser Annahme auf gedachten Kugelschalen um das Emissionszentrum konstant und nimmt mit der Entfernung vom Zentrum ab (siehe Gleichung 7).
- Da die Entfernungen im vorliegenden Modell sehr klein sind, kann davon ausgegangen, dass die Intensität des Schalls auf dem Wege zu den Sensoren aufgrund von Absorption nicht merklich abnimmt. Aber:
- Aufgrund der Schallabsorption in den verschiedenen Medien und Energieverlusten bei den Reflexionen an den Wänden flacht das Signal mit der Zeit ab. Dieser Zeitraum beträgt etwa 0,1 Sekunde (aus dem Messaufbau bestimmt). Das heißt, dass das Signal nach etwa 150 m abgeklungen ist. Mit den Maßen des Aquariums ergibt sich so, dass das Schallsignal nach etwa 500 Reflexionen „verschwunden“ ist. Wenn man dies mit dem oben bestimmten Wert für  $\rho_s = 0,99987$  nachrechnet ergibt sich, dass die Schallintensität nach den besagten 500 Reflexionen noch etwa 93,7% des anfänglichen Wertes betragen müsste. Der Wert 0,99987 gibt lediglich an welcher Anteil des Schalls beim Grenzübergang von Glas zu Luft reflektiert wird. Der Schall muss aber bei jeder Reflexion an den Wänden zweimal den Übergang Glas/Wasser überwinden, wobei jeweils Absorptionseffekte und weitere Reflexionseffekte das Signal abschwächen. Dies erklärt auch die weit geringere Absorptionslänge des Schalls im Aquarium.

### 3 Hauptkomponenten des Modells

#### Aquarium

Zur Modellierung des akustischen Detektionsvolumens wird ein Aquarium benutzt, das die folgenden Abmaße hat:

Länge	$l$	60 cm
Breite	$b$	30 cm
Höhe	$h$	35 cm
Füllhöhe	$h_F$	30 cm
Füllmenge	$V_F$	54 l



#### Messaufnahme

Die gesamte Messaufnahme lässt sich in drei Komponenten aufgliedern:

- Der Schall, der sich nach allen Richtungen gleichermaßen im Aquarium ausbreitet wird von Sensoren detektiert.
- Die Sensoren sind an eine Datenaquisitionskarte (DAQ-Karte) angeschlossen.
- Diese wird mit einem Computer verbunden, auf dem sich das Berechnungsprogramm befindet. Die folgende Abbildung zeigt den schematischen Aufbau:

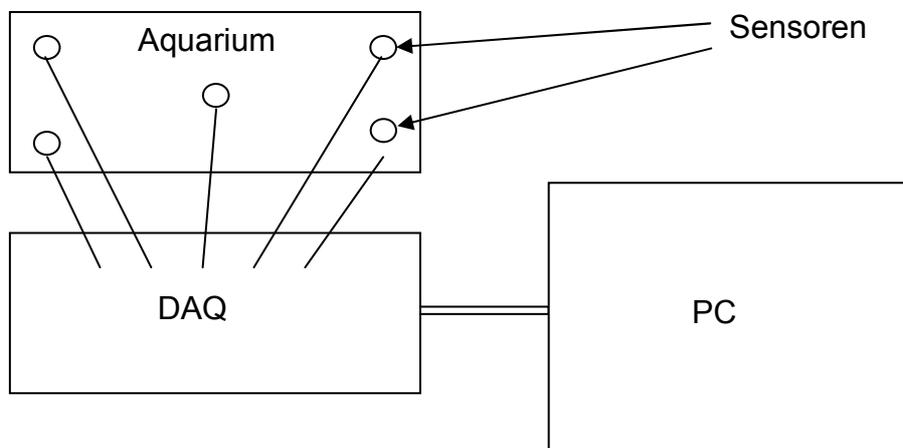


Abbildung 3-1: Schema der Messaufnahme

## Sensoren

Zur Registrierung des Schallsignals werden Piezosensoren, die nach dem Prinzip des piezoelektrischen Effekts arbeiten, verwendet.

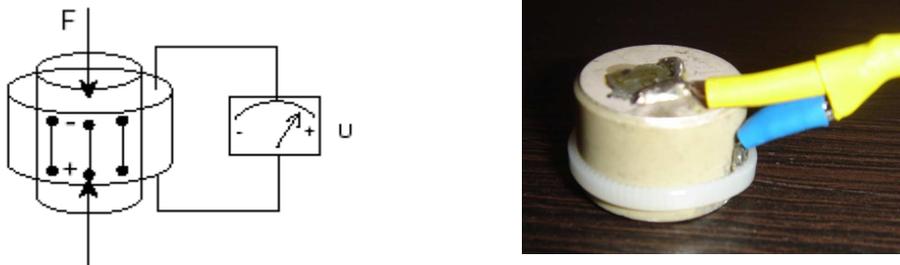


Abbildung 3-2: links: Prinzip eines Piezosensors [4] rechts: Bild eines verwendeten Sensors

Wirkt auf einen piezoelektrischen Körper eine Kraft, so entstehen durch dielektrische Verschiebung Oberflächenladungen und ein elektrisches Feld baut sich auf. Dieser Effekt wird als direkter piezoelektrischer Effekt bezeichnet [1][4]. Die Ursache für diesen Effekt ist eine Anisotropie des Kristallgitters. Die Zentren der positiven bzw. negativen Ladungen lassen sich z.B. durch mechanische Einwirkung gegeneinander verschieben. Bringt man nun an den Stirnflächen eines solchen Kristalls Elektroden an, so kann das entstandene Feld als elektrische Spannung abgegriffen werden. Werden die Elektroden kurzgeschlossen, gleichen sich die Oberflächenladungen in Form eines Stromes aus [4].

Ein im vorliegenden Modell verwendeter Sensor ist in der Abbildung 3-2 rechts dargestellt. Auf der oberen Stirnfläche ist der Anschluss verlötet, der zweite wurde mittels eines Kabelbinders gegen die Elektrode gepresst. Beide Anschlüsse wurden isoliert, um einem Kurzschluss vorzubeugen.

## DAQ-Karte

Die Piezosensoren sind über eine Datenakquisitionskarte (DAQ – Karte) mit einem PC verbunden. Die Karte hat insgesamt 26 Eingänge, von denen im Modell nur 6 benötigt werden. 5 davon für die 5 verwendeten Sensoren, der 6te wurde kurzgeschlossen und stellt das Nullpotential der Spannung dar. Das von den Sensoren aufgenommene Signal wird diskret abgetastet, wobei die Abtastrate im verwendeten Programm über den PC eingestellt werden kann. Diese beträgt im vorliegenden Modell für jeden Kanal:

$$f = 64kHz .$$

Die Abtastrate kann im vorliegenden Modell nicht höher gewählt werden, da der hier entwickelte Algorithmus sonst versagen würde. Aus dieser gewählten Abtastrate ergibt sich eine Auflösung für die Abstandsmessungen zwischen Sensoren und Emissionspunkt von:

$$l = \frac{c_s}{f} = 2,32cm \quad (\text{siehe 4.4})$$

Das Eigenrauschen der Messaufnahme wurde für die 5 Sensoren im Betrieb (d.h. dass die Sensoren im Aquarium unter Wasser angebracht waren) über eine Messschleife (also aus 5 x 4000 Werten) ermittelt, wobei aus diesen Werten jeweils der Mittelwert gebildet wurde um anschließend die Standardabweichung zu berechnen. Diese ergab für die 20000 verwendeten Werte:

$$\bar{U}_{Rausch} = 0,002V$$

### Digitale Verarbeitung

Zur Programmierung des Berechnungsprogramms wird die graphische Programmierungs- und Steuerungsumgebung Labview von National Instruments verwendet [7]. Als Gerüst für dieses Berechnungsprogramm wurde ein mit der DAQ-Karte mitgeliefertes Programm (siehe Kapitel 7.1.4) verwendet. Dieses wurde dann auf das vorliegende Problem abgestimmt.

## **3.1 Messaufbau**

In diesem Kapitel wird dargestellt, wo und wie die Sensoren auf Grund verschiedener Kriterien am günstigsten anzubringen sind, und ob eine Verstärkung des Signals nötig ist oder nicht.

### **3.1.1 Befestigungsort der Sensoren**

Der optimale Ort der Befestigung der Sensoren ist von vielen Einflüssen abhängig. Die wichtigsten werden hier aufgelistet:

#### I) Anteile des sich ausbreitenden Schalls

Bei der Ausbreitung des Schalls in einem solchen Modell sind prinzipiell drei verschiedene Schallanteile zu berücksichtigen:

- a) ein Anteil, der sich direkt von der Quelle zum Empfänger (Sensor) durch das Wasser ausbreitet
- b) Anteile, die aus Reflexionen an den Wänden herrühren
- c) Anteile, die sich durch den Glaskörper ausbreiten (zum einen direkt von der Quelle, zum anderen aber auch vom Wasser ausgehend)

Bei den anfänglichen Versuchen wurden die Sensoren von außen an das Aquarium geklebt. Hierbei muss der zuerst genannte Anteil durch geeignete Wahl der Sensoranbringung zeitlich separiert werden, da sonst eine Unterscheidung der Anteile nicht möglich ist.

- Abschirmung der Reflexionen

Um die Reflexionen zeitlich abzuschirmen, müssen die Sensoren in einem gewissen Abstand  $x$  zu den Kanten des Aquariums angebracht werden (siehe Abbildung 3-3), so dass die Laufwege der reflektierten Schallwellen deutlich größer werden, als die Wege der direkt durchs Wasser laufenden Wellen. Wie bereits erwähnt, beträgt die Auflösung der Schallwege im Modell ca. 2,313 cm, d.h. dass zwei Schallsignale nur dann zeitlich trennbar sind, wenn ihre Laufwege sich um mindestens 2,313 cm unterscheiden. Die Länge  $l$  in der Abbildung 3-3 muss mindestens  $15\text{cm} + 1,16\text{cm}$  betragen. Damit ergibt sich für die Distanz  $x$  des Klopfpunktes (und eines Sensors) von der Wand:

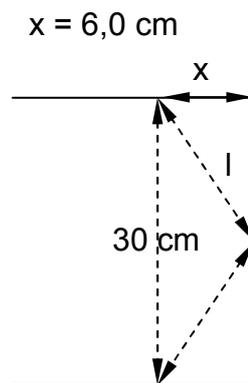


Abbildung 3-3: Skizze des rechten Teils des Aquariums von oben zur Veranschaulichung des (direkten/reflektierten) Schallsignals

- Abschirmung der Anteile durchs Glas

Die zeitliche Trennung der Schallwellen durch das Glas gestaltet sich schwieriger. Die Schallgeschwindigkeit in Glas ( $c_G = 5300\text{ m/s}$ ) ist etwa viermal so groß, wie die Schallgeschwindigkeit im (reinen) Wasser ( $c_W = 1480\text{ m/s}$ ). Man muss zum Trennen dieser beiden Schallsignale Punkte auf dem Aquarium wählen, zu denen der direkte Weg durch das Wasser mindestens viermal kleiner ist, als der kürzeste Weg entlang der Wand. Bei dem hier verwendeten Aquarium findet man solche Punkte nur in kleinen Arealen. Die Klopfpunkt- und die Sensorregion müssten sehr stark eingeschränkt werden, so dass eine vernünftige Rekonstruktion nicht mehr gewährleistet wäre. Um diese Areale auf die gesamte Fläche des Aquariums zu vergrößern, bräuchte man ein Aquarium mit möglichst geringer Breite, damit die Schallwege im Wasser klein genug sind. Letztlich wurde entschieden, die Sensoren im Wasser anzubringen. Hierzu wurden diese in eine Kupferhülse gebracht, in welche sie wasserdicht in Epoxydharz eingegossen wurden. Die folgende Abbildung zeigt einen solchen eingegossenen Sensor:



Abbildung 3-4: Bild eines eingegossenen Sensors

Bei dieser Art der Anbringung spielen die Schallwellen, die durchs Glas gelangen, nur noch eine vernachlässigbare Rolle. Zwar gelangt ein gewisser Anteil des Schalls nach wie vor durch das Glas schneller zu den Sensoren, jedoch ist dessen Intensität so gering, dass er nicht mehr wahrnehmbar ist. Im Hinblick auf die oben erwähnten Schallanteile ist daher nur darauf zu achten, dass die Sensoren den genannten Abstand zu den Wänden des Aquariums haben müssen, um die Reflexionen klar von den direkten Signalen separieren zu können.

## II) Minimierungs- bzw. Lokalisierungsalgorithmus

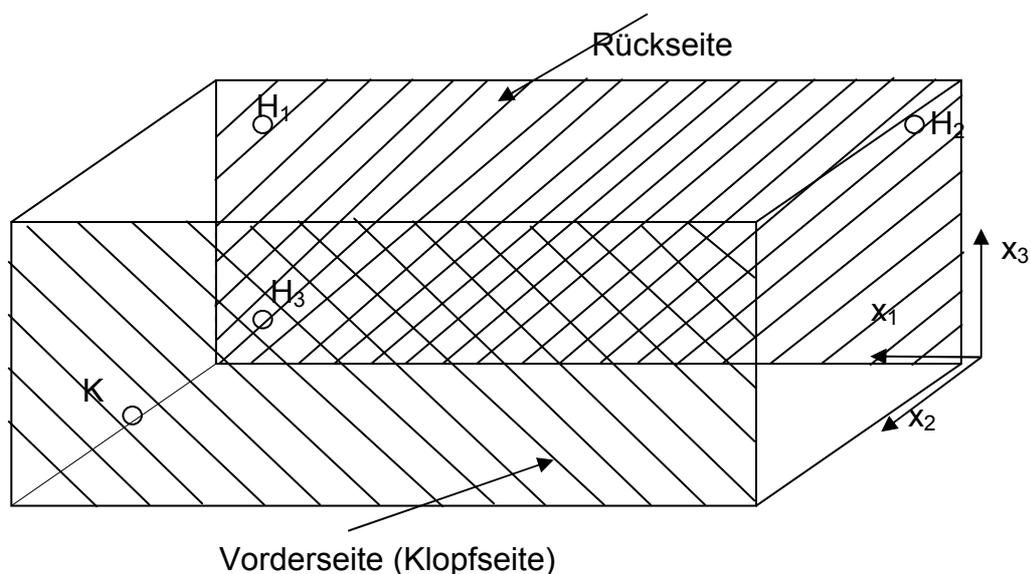
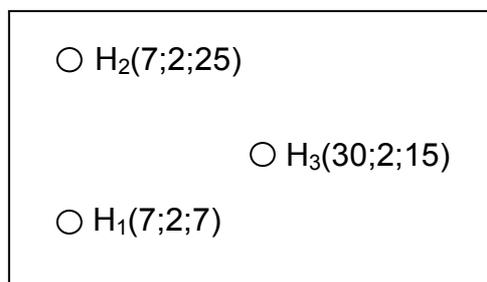


Abbildung 3-5: Veranschaulichung des Modells mit den Sensoren  $H_1$ ,  $H_2$  und  $H_3$  (es handelt sich hier um eine beliebige Anordnung) auf der Rückseite des Aquariums, des Klopfpunktes  $K$  auf der Vorderseite und des verwendeten Koordinatensystems

Mit den entwickelten Algorithmen (siehe Kapitel 4) wurde für verschiedene Sensorkonfigurationen untersucht, wie gut die Auflösung eines Punktes auf der Klopffläche ist. D.h. wie eindeutig die Zuordnung gemessener Laufwegdifferenzen zu Klopfpunktkoordinaten ist. Dabei wurden zwei verschiedene Sensorkonfigurationen verwendet, bei denen jeweils die Laufwegdifferenzen zu jedem möglichen Klopfpunkt berechnet und in einer Tabelle abgespeichert wurden. Diese Untersuchungen werden im Folgenden kurz dargestellt.

### Untersuchte Sensorkonfigurationen

Konstellation I)



Konstellation II)

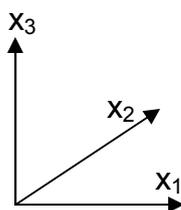
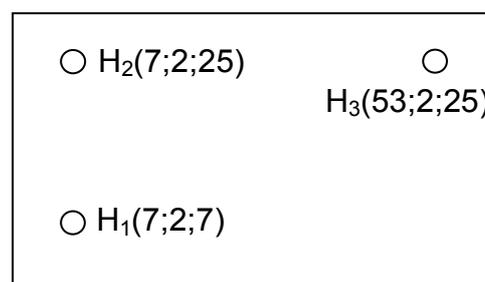


Abbildung 3-6: Schemata der zwei verwendeten Sensorkonfigurationen (Blick auf die Rückseite des Aquariums)

Konstellation I wurde so gewählt, dass die Sensoren relativ nahe beieinander liegen, während sie bei der Konstellation II möglichst große Abstände haben. Zu den beiden Konfigurationen wurden nun alle Klopfpunkte auf der Frontseite des Aquariums durchlaufen. Hierbei wurde zuerst berechnet, welcher der Sensoren dem Klopfpunkt jeweils am nächsten ist. Anschließend wurden jeweils die Laufwegdifferenzen der beiden anderen Sensoren bezüglich diesem einen berechnet. Diesen Laufwegdifferenzen kann theoretisch in eindeutiger Weise ein Emissionspunkt zugeordnet werden. Da im Modell jedoch (wegen der fest eingestellten Abtastrate) nur diskrete Weglängen bestimmt werden können, ist dies hier nicht der Fall. Die verwendeten Ausschnitte sind in der folgenden schematischen Darstellung der Klopffläche (Frontfläche) gekennzeichnet.

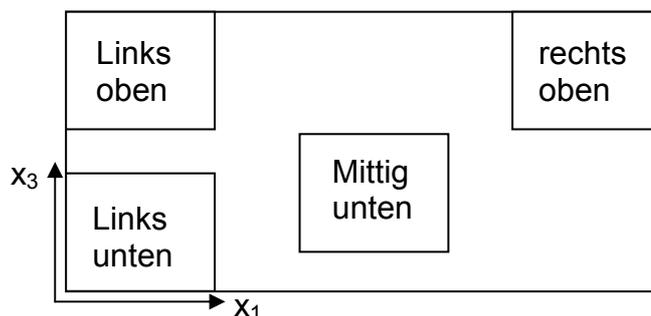


Abbildung 3-7: qualitative Darstellung der gewählten Ausschnitte bei der Analyse

In den folgenden Graphiken sind aus der Verteilung der berechneten Laufwegdifferenzen jeweils die Ausschnitte zu sehen, die in Abbildung 3-7 aufgezeigt sind. Die Differenz des dritten Sensors beträgt hierbei stets „0“ ( $s_0$  – siehe Kapitel 4.2). An den Rändern der Graphik sind die Koordinaten des jeweiligen Klopfpunktes angegeben. In den einzelnen Graphiken sind Bereiche in denen Punkte liegen zu denen dieselben beiden berechneten Laufwegdifferenzen gehören in einer Farbe markiert.

Ausschnitte zur Konstellation I)

Links oben

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15																							
30	3	6	3	6	3	6	3	6	3	5	3	5	3	5	3	4	3	4	3	4	3	4	3	3	3	3	3	3	3	3								
29	3	6	3	6	3	6	3	6	3	5	3	5	3	5	3	5	3	4	3	4	3	4	3	4	3	3	3	3	3	3	3	3						
28	3	6	3	6	3	6	3	5	3	5	3	5	3	5	3	5	3	4	3	4	3	4	3	4	3	3	3	3	3	3	3	2						
27	3	6	3	6	3	6	3	5	3	5	3	5	3	5	3	5	3	4	3	4	3	4	3	4	3	3	3	3	3	3	3	2						
26	2	6	2	6	2	5	2	5	2	5	2	5	2	5	2	5	3	5	3	5	3	5	3	4	3	4	3	4	2	3	2	3	2	3	2	3	2	2
25	2	6	2	6	2	5	2	5	2	5	2	5	2	4	2	4	2	4	2	4	2	4	2	4	2	3	2	3	2	3	2	2	2	2	2	2	2	
24	2	6	2	5	2	5	2	5	2	5	2	5	2	4	2	4	2	4	2	4	2	4	2	3	2	3	2	3	2	2	2	2	2	2	2	2	2	
23	2	6	2	5	2	5	2	5	2	5	2	4	2	4	2	4	2	4	2	4	2	4	2	3	2	3	2	2	2	2	2	2	2	2	2	2	2	
22	2	5	2	5	2	5	2	5	2	5	2	4	2	4	2	4	2	4	2	4	2	3	2	3	2	3	2	2	2	2	2	2	2	2	2	2	2	
21	1	5	1	5	1	5	1	5	1	4	1	4	1	4	1	4	1	4	1	3	1	3	1	3	1	2	1	2	1	2	1	2	1	2	1	2	1	2
20	1	5	1	5	1	5	1	5	1	4	1	4	1	4	1	4	1	4	1	3	1	3	1	3	1	2	1	2	1	2	1	2	1	2	1	1	1	
19	1	5	1	5	1	5	1	4	1	4	1	4	1	4	1	4	1	3	1	3	1	3	1	3	1	2	1	2	1	2	1	2	1	1	1	1	1	
18	1	5	1	5	1	4	1	4	1	4	1	4	1	4	1	4	1	3	1	3	1	3	1	3	1	2	1	2	1	2	1	1	1	1	1	1	1	
17	0	5	0	5	0	4	0	4	0	4	0	4	0	3	0	3	0	3	0	3	0	3	0	3	0	2	0	2	0	2	0	2	0	1	0	1	0	1

Abbildung 3-8: Laufwegdifferenzen zu den Sensoren  $H_1$  und  $H_3$  zum Ausschnitt „links oben“ der Konstellation I). Auf der x - und y - Achse sind die Klopfpunktkoordinaten in cm aufgetragen. Die Zahlen im Raster stellen die Laufwegdifferenzen zu den beiden Sensoren dar, die vom gewählten Klopfpunkt am weitesten entfernt sind. Paare gleicher Laufwegdifferenzen sind in dergleichen Farbe dargestellt.

Links unten

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15																								
16	0	5	0	5	0	4	0	4	0	4	0	4	0	3	0	3	0	3	0	3	0	3	0	2	0	2	0	2	0	1	0	1	0	1	0	1			
15	1	5	1	5	1	4	1	4	1	4	1	4	1	4	1	3	1	3	1	3	1	3	1	2	1	2	1	2	1	1	1	1	1	1	1	1	1		
14	1	5	1	5	1	5	1	4	1	4	1	4	1	4	1	4	1	3	1	3	1	3	1	2	1	2	1	2	1	2	1	1	1	1	1	1	1		
13	1	5	1	5	1	5	1	4	1	4	1	4	1	4	1	4	1	3	1	3	1	3	1	3	1	2	1	2	1	2	1	1	1	1	1	1	1		
12	1	5	1	5	1	5	1	5	1	4	1	4	1	4	1	4	1	4	1	3	1	3	1	3	1	2	1	2	1	2	1	1	1	1	1	1	1		
10	2	5	2	5	2	5	2	5	2	4	2	4	2	4	2	4	2	4	2	4	2	4	2	3	2	3	2	3	2	2	2	2	2	2	2	2	2		
9	2	5	2	5	2	5	2	5	2	5	2	4	2	4	2	4	2	4	2	4	2	4	2	3	2	3	2	3	2	2	2	2	2	2	2	2	2		
8	2	5	2	5	2	5	2	5	2	5	2	4	2	4	2	4	2	4	2	4	2	4	2	3	2	3	2	3	2	2	2	2	2	2	2	2	2		
7	2	6	2	5	2	5	2	5	2	5	2	4	2	4	2	4	2	4	2	4	2	4	2	3	2	3	2	3	2	2	2	2	2	2	2	2	2		
6	2	6	2	5	2	5	2	5	2	5	3	5	3	4	3	4	3	4	3	4	3	4	3	3	2	3	2	3	2	3	2	2	2	2	2	2	2		
5	3	6	3	6	3	5	3	5	3	5	3	5	3	5	3	4	3	4	3	4	3	4	3	4	3	3	3	3	3	3	3	3	3	2	3	2	3	2	
4	3	6	3	6	3	5	3	5	3	5	3	5	3	5	3	4	3	4	3	4	3	4	3	4	3	3	3	3	3	3	3	3	3	2	3	2	3	2	
3	3	6	3	6	3	5	3	5	3	5	3	5	3	5	3	4	3	4	3	4	3	4	3	4	3	3	3	3	3	3	3	3	3	3	3	3	2	3	2
2	3	6	3	6	3	6	3	5	3	5	3	5	3	5	3	4	3	4	3	4	3	4	3	4	3	3	3	3	3	3	3	3	3	3	3	3	2	3	2
1	3	6	3	6	4	6	4	5	4	5	4	5	4	5	4	4	4	4	4	4	4	4	4	4	4	4	3	4	3	3	3	3	3	3	3	2	3	2	

Abbildung 3-9: Laufwegdifferenzen zu den Sensoren  $H_2$  und  $H_3$  zum Ausschnitt „links unten“ der Konstellation I) siehe auch Legende zu Abbildung 3-8

*Mittig unten*

	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
15	2 1	3 1	3 1	3 2	3 2	4 2	4 3	4 3	5 3	5 3	5 4	5 4	5 4	6 4	6 5
14	2 1	3 1	3 1	3 2	3 2	4 2	4 3	4 3	4 3	5 4	5 4	5 4	5 4	6 4	6 5
13	2 1	2 1	3 2	3 2	3 2	4 3	4 3	4 3	4 3	5 4	5 4	5 4	5 4	6 5	6 5
12	2 1	2 1	3 2	3 2	3 2	4 3	4 3	4 3	4 4	5 4	5 4	5 4	5 5	6 5	6 5
11	2 1	2 2	3 2	3 2	3 2	3 3	4 3	4 3	4 4	5 4	5 4	5 4	5 5	5 5	6 5
10	2 1	2 2	2 2	3 2	3 3	3 3	4 3	4 3	4 4	4 4	5 4	5 5	5 5	5 5	6 5
9	2 2	2 2	2 2	3 2	3 3	3 3	4 3	4 4	4 4	4 4	5 4	5 5	5 5	5 5	6 5
8	2 2	2 2	2 2	3 3	3 3	3 3	3 3	4 4	4 4	4 4	4 4	5 5	5 5	5 5	5 5
7	2 2	2 2	2 2	2 3	3 3	3 3	3 4	4 4	4 4	4 4	4 5	5 5	5 5	5 5	5 6
6	1 2	2 2	2 3	2 3	3 3	3 3	3 4	4 4	4 4	4 5	4 5	5 5	5 5	5 5	5 6
5	1 2	2 2	2 3	2 3	3 3	3 4	3 4	3 4	4 4	4 5	4 5	5 5	5 5	5 6	5 6
4	1 2	2 2	2 3	2 3	2 3	3 4	3 4	3 4	4 4	4 5	4 5	4 5	5 5	5 6	5 6
3	1 2	1 3	2 3	2 3	2 3	3 4	3 4	3 4	4 5	4 5	4 5	4 5	5 6	5 6	5 6
2	1 2	1 3	2 3	2 3	2 4	3 4	3 4	3 4	3 5	4 5	4 5	4 5	4 6	5 6	5 6
1	1 3	1 3	2 3	2 3	2 4	2 4	3 4	3 4	3 5	4 5	4 5	4 5	4 6	5 6	5 6

Abbildung 3-10: Laufwegdifferenzen zu den Sensoren H<sub>2</sub> und H<sub>3</sub> zum Ausschnitt „mittig unten“ der Konstellation I) siehe auch Legende zu zu Abbildung 3-8

*Rechts oben*

	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
30	8 5	8 5	8 6	8 6	8 6	8 6	8 6	8 6	8 6	8 7	9 7	9 7	9 7	9 7	9 7
29	8 5	8 6	8 6	8 6	8 6	8 6	8 6	8 6	8 7	8 7	9 7	9 7	9 7	9 7	9 7
28	8 6	8 6	8 6	8 6	8 6	8 6	8 6	8 7	8 7	8 7	9 7	9 7	9 7	9 7	9 7
27	8 6	8 6	8 6	8 6	8 6	8 6	8 7	8 7	8 7	8 7	9 7	9 7	9 7	9 7	9 7
26	8 6	8 6	8 6	8 6	8 6	8 7	8 7	8 7	8 7	8 7	9 7	9 7	9 7	9 7	9 7
25	8 6	8 6	8 6	8 6	8 7	8 7	8 7	8 7	8 7	8 7	9 7	9 7	9 7	9 7	9 8
24	8 6	8 6	8 6	8 7	8 7	8 7	8 7	8 7	8 7	8 7	9 7	9 7	9 7	9 8	9 8
23	7 6	8 6	8 7	8 7	8 7	8 7	8 7	8 7	8 7	8 7	9 8	9 8	9 8	9 8	9 8
22	7 6	8 7	8 7	8 7	8 7	8 7	8 7	8 7	8 7	8 8	8 8	9 8	9 8	9 8	9 8
21	7 6	8 7	8 7	8 7	8 7	8 7	8 7	8 7	8 8	8 8	8 8	9 8	9 8	9 8	9 8
20	7 7	7 7	8 7	8 7	8 7	8 7	8 7	8 7	8 8	8 8	8 8	8 8	8 8	9 8	9 8
19	7 7	7 7	8 7	8 7	8 7	8 7	8 7	8 8	8 8	8 8	8 8	8 8	8 8	9 8	9 8
18	7 7	7 7	8 7	8 7	8 7	8 7	8 8	8 8	8 8	8 8	8 8	8 8	8 8	8 8	9 8
17	7 7	7 7	7 7	8 7	8 7	8 8	8 8	8 8	8 8	8 8	8 8	8 8	8 8	8 8	8 8
16	7 7	7 7	7 7	8 7	8 8	8 8	8 8	8 8	8 8	8 8	8 8	8 8	8 8	8 8	8 8

Abbildung 3-11: Laufwegdifferenzen zu den Sensoren H<sub>1</sub> und H<sub>2</sub> zum Ausschnitt „rechts oben“ der Konstellation I) siehe auch Legende zu Abbildung 3-8

Ausschnitte zur Konstellation II)

*Links oben*

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15															
30	3	13	3	13	3	13	3	12	3	12	3	12	3	11	3	11	3	11	3	10	3	10	3	9	3	9	3	9	3	8
29	3	13	3	13	3	13	3	12	3	12	3	12	3	11	3	11	3	11	3	10	3	10	3	10	3	9	3	9	3	8
28	3	13	3	13	3	13	3	12	3	12	3	12	3	11	3	11	3	11	3	10	3	10	3	10	3	9	3	9	3	8
27	3	13	3	13	3	13	3	12	3	12	3	12	3	11	3	11	3	11	3	10	3	10	3	10	3	9	3	9	3	8
26	2	13	2	13	2	13	2	13	2	12	3	12	3	12	3	11	3	11	3	10	2	10	2	10	2	9	2	9	2	8
25	2	13	2	13	2	13	2	13	2	12	2	12	2	12	2	11	2	11	2	10	2	10	2	10	2	9	2	9	2	8
24	2	13	2	13	2	13	2	13	2	12	2	12	2	12	2	11	2	11	2	10	2	10	2	10	2	9	2	9	2	8
23	2	13	2	13	2	13	2	12	2	12	2	12	2	11	2	11	2	11	2	10	2	10	2	10	2	9	2	9	2	8
22	2	13	2	13	2	13	2	12	2	12	2	12	2	11	2	11	2	11	2	10	2	10	2	10	2	9	2	9	2	8
21	1	13	1	13	1	13	1	12	1	12	1	12	1	11	1	11	1	11	1	10	1	10	1	10	1	9	1	9	1	8
20	1	13	1	13	1	13	1	12	1	12	1	12	1	11	1	11	1	11	1	10	1	10	1	10	1	9	1	9	1	8
19	1	13	1	13	1	13	1	12	1	12	1	12	1	11	1	11	1	11	1	10	1	10	1	10	1	9	1	9	1	8
18	1	13	1	13	1	13	1	12	1	12	1	12	1	11	1	11	1	11	1	10	1	10	1	10	1	9	1	9	1	8
17	0	13	0	13	0	13	0	12	0	12	0	12	0	11	0	11	0	11	0	10	0	10	0	10	0	9	0	9	0	8
16	0	13	0	13	0	13	0	12	0	12	0	12	0	11	0	11	0	11	0	10	0	10	0	10	0	9	0	9	0	8

Abbildung 3-12: Laufwegdifferenzen zu den Sensoren H<sub>1</sub> und H<sub>3</sub> zum Ausschnitt „links oben“ der Konstellation I) siehe auch Legende zu Abbildung 3-8

*Links unten*

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15															
15	0	13	0	13	0	13	0	12	0	12	0	12	0	11	0	11	0	11	0	10	0	10	0	10	0	9	0	9	0	8
14	1	13	1	13	1	13	1	12	1	12	1	12	1	11	1	11	1	11	1	10	1	10	1	10	1	9	1	9	1	8
13	1	14	1	13	1	13	1	13	1	12	1	12	1	12	1	11	1	11	1	11	1	10	1	10	1	9	1	9	1	9
12	1	14	1	14	1	13	1	13	1	13	1	13	1	12	1	12	1	12	1	11	1	11	1	11	1	10	1	10	1	9
11	1	14	1	14	1	13	1	13	1	13	1	13	1	12	1	12	1	12	1	11	1	11	1	11	1	11	1	10	1	9
10	2	14	2	14	2	14	2	13	2	13	2	13	2	12	2	12	2	12	2	11	2	11	2	11	2	10	2	10	2	9
9	2	14	2	14	2	14	2	13	2	13	2	13	2	12	2	12	2	12	2	11	2	11	2	11	2	10	2	10	2	9
8	2	14	2	14	2	14	2	14	2	13	2	13	2	13	2	12	2	12	2	12	2	11	2	11	2	10	2	10	2	10
7	2	15	2	14	2	14	2	14	2	13	2	13	2	13	2	12	2	12	2	12	2	11	2	11	2	11	2	10	2	10
6	2	15	2	14	2	14	2	14	2	14	3	13	3	13	3	13	3	13	3	12	3	12	2	11	2	11	2	11	2	10
5	3	15	3	15	3	14	3	14	3	14	3	13	3	13	3	13	3	13	3	12	3	12	3	12	3	11	3	11	3	10
4	3	15	3	15	3	14	3	14	3	14	3	13	3	13	3	13	3	13	3	12	3	12	3	12	3	11	3	11	3	10
3	3	15	3	15	3	14	3	14	3	14	3	13	3	13	3	13	3	13	3	12	3	12	3	12	3	11	3	11	3	10
2	3	15	3	15	3	15	3	14	3	14	3	14	3	13	3	13	3	13	3	13	3	12	3	12	3	11	3	11	3	10
1	3	15	3	15	4	15	4	14	4	14	4	14	4	13	4	13	4	13	4	12	4	12	4	12	4	11	3	11	3	10

Abbildung 3-13: Laufwegdifferenzen zu den Sensoren H<sub>2</sub> und H<sub>3</sub> zum Ausschnitt „links unten“ der Konstellation II) siehe auch Legende zu Abbildung 3-8

*Mittig unten*

	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
15	0 8	0 7	0 7	0 6	0 6	0 5	0 5	0 4	0 4	0 3	0 3	0 2	0 2	0 1	0 1
14	1 8	1 8	0 7	0 7	0 6	0 6	0 5	0 5	0 4	0 4	0 3	0 3	0 2	0 1	0 1
13	1 8	1 8	1 7	1 7	1 6	1 6	1 5	1 5	1 4	1 4	1 3	1 3	1 2	1 2	1 1
12	1 8	1 8	1 7	1 7	1 6	1 6	1 6	1 5	1 4	1 4	1 3	1 3	1 2	1 2	1 1
11	1 9	1 8	1 8	1 7	1 7	1 6	1 6	1 5	1 5	1 4	1 4	1 3	1 3	1 2	1 2
10	2 9	1 8	1 8	1 7	1 7	1 6	1 6	1 5	1 5	1 4	1 4	1 3	1 3	1 2	1 2
9	2 9	2 8	2 8	2 8	2 7	2 7	2 6	2 6	2 5	2 5	2 4	2 4	1 3	1 2	1 2
8	2 9	2 9	2 8	2 8	2 7	2 7	2 6	2 6	2 5	2 5	2 4	2 4	2 3	2 3	2 2
7	2 9	2 9	2 8	2 8	2 7	2 7	2 6	2 6	2 5	2 5	2 4	2 4	2 3	2 3	2 2
6	2 9	2 9	2 8	2 8	2 8	2 7	2 7	2 6	2 6	2 5	2 5	2 4	2 4	2 3	2 3
5	3 10	3 9	3 9	3 8	3 8	3 7	2 7	2 6	2 6	2 5	2 5	2 4	2 4	2 3	2 3
4	3 10	3 9	3 9	3 8	3 8	3 7	3 7	3 6	3 6	3 5	3 5	3 4	2 4	2 3	2 3
3	3 10	3 9	3 9	3 8	3 8	3 8	3 7	3 7	3 6	3 6	3 5	3 5	3 4	3 4	3 3
2	3 10	3 9	3 9	3 9	3 8	3 8	3 7	3 7	3 6	3 6	3 5	3 5	3 4	3 4	3 3
1	3 10	3 10	3 9	3 9	3 8	3 8	3 7	3 7	3 6	3 6	3 5	3 5	3 4	3 4	3 3

Abbildung 3-14: Laufwegdifferenzen zu den Sensoren H<sub>2</sub> und H<sub>3</sub> zum Ausschnitt „mittig unten“ der Konstellation II) siehe auch Legende zu Abbildung 3-8

*Rechts oben*

	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61
30	10 8	11 9	11 9	12 9	12 10	12 10	13 11	13 11	13 11	14 12	14 12	14 12	15 13	15 13	15 13
29	10 8	11 9	11 9	11 10	12 10	12 10	13 11	13 11	13 11	14 12	14 12	14 12	14 13	15 13	15 13
28	10 8	10 9	11 9	11 10	12 10	12 10	12 11	13 11	13 11	13 12	14 12	14 12	14 13	15 13	15 13
27	10 8	10 9	11 9	11 10	12 10	12 10	12 11	13 11	13 11	13 12	14 12	14 12	14 13	15 13	15 13
26	10 8	10 9	11 9	11 10	11 10	12 10	12 11	13 11	13 12	13 12	14 12	14 12	14 13	14 13	15 13
25	10 8	10 9	11 9	11 10	11 10	12 10	12 11	12 11	13 12	13 12	13 12	14 13	14 13	14 13	15 13
24	10 8	10 9	10 9	11 10	11 10	12 10	12 11	12 11	13 12	13 12	13 12	14 13	14 13	14 13	14 13
23	9 8	10 9	10 9	11 10	11 10	11 10	12 11	12 11	12 11	13 12	13 12	13 12	14 13	14 13	14 13
22	9 8	10 9	10 9	10 10	11 10	11 10	12 11	12 11	12 11	13 12	13 12	13 12	14 13	14 13	14 13
21	9 8	9 9	9 10	10 10	11 10	11 10	11 11	12 11	12 11	12 12	13 12	13 12	13 13	14 13	14 13
20	9 8	9 9	9 10	10 9	10 10	11 10	11 11	12 11	12 11	12 12	13 12	13 12	13 13	14 13	14 13
19	9 8	9 9	9 9	10 9	10 10	11 10	11 11	11 11	12 11	12 12	12 12	13 12	13 13	13 13	14 13
18	8 8	8 9	9 9	9 10	10 10	11 10	11 11	11 11	12 11	12 12	12 12	13 12	13 13	13 13	13 13
17	8 8	8 9	9 9	9 10	10 10	10 10	11 11	11 11	11 11	12 12	12 12	12 12	13 13	13 13	13 13
16	8 8	8 8	8 9	9 9	9 9	10 10	10 10	11 11	11 11	12 12	12 12	12 12	13 13	13 13	13 13

Abbildung 3-15: Laufwegdifferenzen zu den Sensoren H<sub>1</sub> und H<sub>2</sub> zum Ausschnitt „rechts oben“ der Konstellation II) siehe auch Legende zu Abbildung 3-8

---

Aus diesen Graphiken lässt sich schließen, dass es bei dem verwendeten Algorithmus günstiger ist, die Sensoren mit möglichst großen Abständen zueinander anzubringen. Vor allem der Vergleich der zwei Ausschnitte „rechts oben“ macht dies deutlich. Bei der Konstellation, bei der die Sensoren näher beieinander sind, gibt es zu gemessenen Laufwegdifferenzen sehr große Areale möglicher Punkte. Bei der zweiten Konstellation hingegen sind die Arealgrößen in diesem Ausschnitt nicht merklich anders als in der Mitte oder bei Klopfpunkten in der Nähe der Sensoren (Ausschnitte links/mittig). Die Ausschnitte, bei denen die Emissionspunkt Nahe den Sensoren liegen, zeigen keine Unterschiede zwischen den beiden Konstellationen.

### III) Hardware (DAQ-Karte)

Aufgrund von Übersprechern beeinflussen sich die an die DAQ-Karte angeschlossenen Sensoren gegenseitig (siehe hierzu Abbildung 4-8 in Kapitel 4.4.3.1). D.h. wenn ein Sensor ein Signal erhält, reagieren die anderen Sensoren hierauf auch. Die Anschlüsse der Karte sind nummeriert. Der Einfluss der Sensoren ist relativ stark ausgeprägt von Sensoren kleinerer Anschlussnummer auf Sensoren, die eine größere Anschlussnummer haben. Der Einfluss in umgekehrte Richtung ist dagegen geringer. Aufgrund dieser Beobachtungen sollte der letzte Sensor (Sensor H<sub>5</sub>) möglichst nicht am weitesten von einem Klopfpunkt entfernt sein. Dies ist am besten gegeben, wenn dieser mittig angeordnet wird. Die anderen vier Sensoren sollten so angeordnet werden, dass möglichst immer ein unbeeinflusster mit einem stärker beeinflussten Sensor beieinander (auf einer vertikalen Koordinate  $x_3$  bzw. bei einer horizontalen Koordinate  $x_1$ ) sind.

Die Abbildung 3-16 zeigt eine sich somit ergebende mögliche Konfiguration von 5 Sensoren. Prinzipiell wurde diese auch im späteren Modell benutzt.

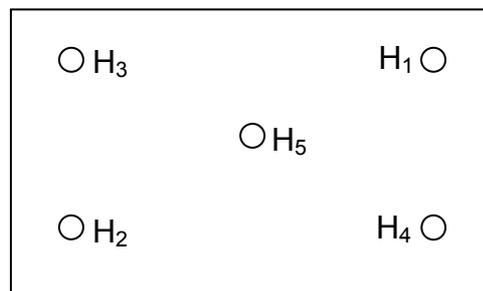


Abbildung 3-16: Skizze der optimalen Verteilung der Sensoren aufgrund des Übersprechverhaltens

### **3.1.2 Befestigungsart der Sensoren**

Für den Prototyp wurden die Sensoren an Schnüren frei hängend ins Wasser getaucht. Für das spätere Vorführen des Modells ist eine feste Fixierung geplant. Hierzu sollen an den Sensoren Haken angebracht werden, um diese an einem in das Aquarium gehängtes Gitter zu befestigen.

### **3.1.3 Verstärkung des Signals**

Anfangs war nicht klar, ob das Signal gegenüber dem Rauschen der Messapparatur ausgeprägt genug sein würde bzw. ob die Sensoren die möglicherweise sehr kleinen Druckschwankungen merklich registrieren würden oder ob es nötig würde die einzelnen Sensoren mit Verstärkern zu versehen. Die ersten Messungen ließen sofort erkennen, dass bereits ein relativ schwaches Signal sehr gut detektiert und an die Messaufnahme weitergegeben wird. Als Klopfgegenstände wurden verschiedene Materialien ausprobiert und es stellte sich heraus, dass ein möglichst hochfrequenter Ton durch z.B. einen metallischen Gegenstand eingeleitet werden sollte, da dumpfe (niederfrequente) Töne durch z.B. Holz oder die Finger kaum registriert werden. Unter diesen Umständen kann von einer Verstärkung des Signals abgesehen werden.

## 4 Rekonstruktionsstrategien

Den Kern des Modells bildet das Programm (siehe Kapitel 7) zur Berechnung des Emissionspunktes (Kloppunkt). Es geht um die Berechnung zweier markanter Größen (also um die zugrunde liegende Rekonstruktionsstrategie):

- Zum einen müssen aus den gemessenen Signalverläufen der einzelnen Sensoren Anfangszeitpunkte ermittelt werden, aus denen Laufwegdifferenzen zwischen Emitter und Sensor bestimmt werden
- Zum anderen müssen aus diesen Laufwegdifferenzen Koordinaten berechnet werden

In diesem Kapitel wird dargestellt, welche mathematischen Überlegungen zur Berechnung dieser Werte angestellt wurden, und die Ansätze aufgezeigt, die später (siehe Kapitel 7) in Algorithmen implementiert wurden.

### 4.1 Mathematische Überlegungen zur Bestimmung der Koordinaten des Kloppunktes aus den Messdaten

In diesem Kapitel werden Lösungswege zur Quellenrekonstruktion aufgezeigt. Zur besseren Verständlichkeit werden folgende Bezeichnungen eingeführt:

die insgesamt 5 Sensoren werden stets mit  $H_1$ ,  $H_2$  usw. bezeichnet und deren Abstand zur Emissionsstelle entsprechend mit  $h_1$ ,  $h_2$  usw.

Bei den Berechnungen wird zunächst der Sensor ermittelt werden, der dem Emissionspunkt am nächsten liegt. Dieser Sensor erhält dann die neue Bezeichnung  $S_0$  und sein Abstand zum Emissionspunkt entsprechend die Bezeichnung  $s_0$ . Die beiden anderen Sensoren werden in diesem Zusammenhang ebenfalls umbenannt: Der Sensor  $H_{a_1}$  wird zu  $S_1$  und der Sensor  $H_{a_2}$  zu  $S_2$  für  $a_1 < a_2$ . Ihre Abstände zum Emissionspunkt werden entsprechend mit  $s_1$  und  $s_2$  bezeichnet.

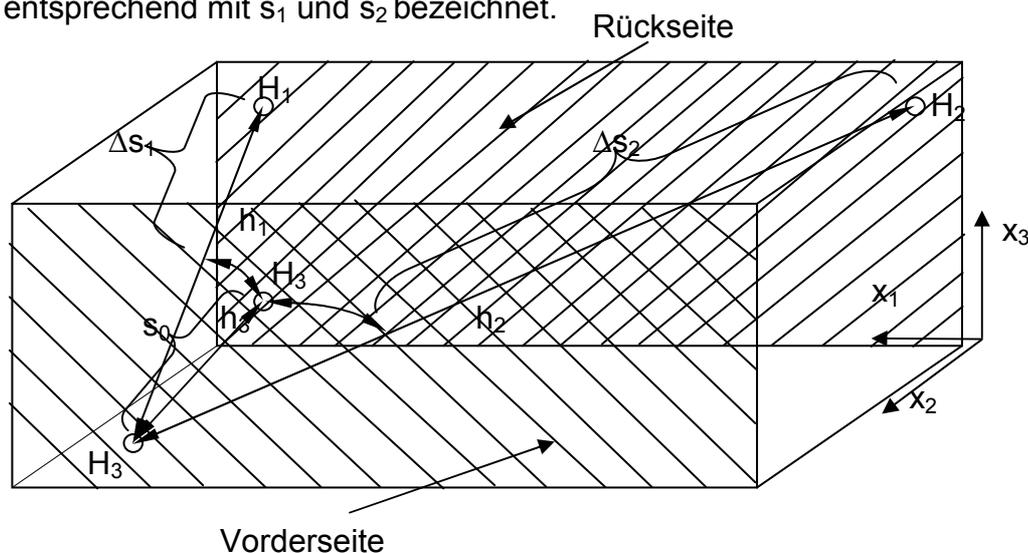


Abbildung 4-1: Veranschaulichung des Modells mit Kloppunkt  $K$  auf der Vorderseite und drei Sensoren  $H_1$ ,  $H_2$  und  $H_3$  auf der Rückseite des Aquariums

Die Abbildung 4-1 stellt das vorliegende Problem dar. Auf der Vorderseite des Aquariums wird im Punkt  $K$  geklopft. Daraufhin breitet sich der Schall unter anderem im Wasser nach allen Richtungen gleichermaßen aus. Der Schall trifft nacheinander bei den drei Sensoren  $H_{i,(i=1..3)}$  ein und legt jeweils die Strecken  $h_{i,(i=1..3)}$  zurück. Die Strecke bis zu dem Sensor  $S_0(x_{1S_0}; x_{2S_0}; x_{3S_0})$  der dem Klopfpunkt  $K(x_{1K}; x_{2K}; x_{3K})$  am nächsten liegt wird wie oben beschrieben mit  $s_0$  bezeichnet und berechnet sich zu:

$$s_0 = \sqrt{\sum_{i=1}^3 (x_{iK} - x_{iS_0})^2} \quad \text{Gleichung 11}$$

Diesen Weg  $s_0$  hat der Schall in die Richtung eines jeden Sensors zurückgelegt, ehe ein erstes Signal  $t_0$  von der Messeinrichtung wahrgenommen wird. Dieses Signal stellt den Startzeitpunkt der Messung dar. D.h. zum Zeitpunkt  $t_0$  hat der Schall den Weg  $s_0$  zurückgelegt und muss nun noch die Wege  $\Delta s_1$  und  $\Delta s_2$  zu den beiden anderen Sensoren zurücklegen. Für diese Wege gilt:

$$\Delta s_{i=0..2} = s_i - s_0 = \sqrt{\sum_{i=1}^3 (x_{iK} - x_{iS_i})^2} - \sqrt{\sum_{i=1}^3 (x_{iK} - x_{iS_0})^2} \quad \text{Gleichung 12}$$

Bei den vorliegenden Messungen werden Zeitpunkte (und damit Zeitunterschiede) gemessen. Mit der Schallgeschwindigkeit in reinem Wasser bei 20°C ( $c_W = 1480$  m/s) lassen sich daraus Wegdifferenz berechnen.

Gleichung 12 liefert somit drei Gleichungen in den drei Unbekannte  $x_{1K}$ ,  $x_{3K}$ , und  $s_0$ :

$$\begin{aligned} \text{I)} \quad \Delta s_0 &= \sqrt{(x_{1K} - x_{1S_0})^2 + (x_{2K} - x_{2S_0})^2 + (x_{3K} - x_{3S_0})^2} - s_0 && \text{mit } \Delta s_0 = 0 \\ \text{II)} \quad \Delta s_1 &= \sqrt{(x_{1K} - x_{1S_1})^2 + (x_{2K} - x_{2S_1})^2 + (x_{3K} - x_{3S_1})^2} - s_0 \\ \text{III)} \quad \Delta s_2 &= \sqrt{(x_{1K} - x_{1S_2})^2 + (x_{2K} - x_{2S_2})^2 + (x_{3K} - x_{3S_2})^2} - s_0 \end{aligned}$$

Da nur an der Vorderseite geklopft wird, bleibt die  $x_2$ -Koordinate des Punktes  $K$  immer gleich (hier 30cm). Damit kann man dieses System einfacher schreiben:

$$\begin{aligned} \text{I)} \quad \Delta s_0 &= 0 \\ \text{II)} \quad \Delta s_1 &= \sqrt{(x_{1K} - x_{1S_1})^2 + (30\text{cm})^2 + (x_{3K} - x_{3S_1})^2} - s_0 \\ \text{III)} \quad \Delta s_2 &= \sqrt{(x_{1K} - x_{1S_2})^2 + (30\text{cm})^2 + (x_{3K} - x_{3S_2})^2} - s_0 \end{aligned}$$

Gleichungssystem I: zur Bestimmung der Klopfpunktkoordinaten

## 4.2 Bestimmung der Koordinaten des Klopfpunktes aus den Messdaten

Dieses Gleichungssystem ist zwar analytisch lösbar, es wurde hier aber vorgezogen, einen eigenen Algorithmus im verwendeten Programm „Labview“ zu erstellen. Im Folgenden wird dargestellt, welcher Ansatz zur Erstellung eines passenden Algorithmus gewählt wurde. Der Ansatz basiert auf der Idee, die „beste“ Klopffläche systematisch zu suchen. Hierbei wird die Klopffläche schrittweise mit Schrittweite  $\Delta z$  abgetastet. Die Schrittweite wurde im Modell fest gewählt mit  $\Delta z = 1\text{cm}$ . Nun werden in jedem Schritt  $n$  zunächst die Abstände der drei Sensoren zum jeweiligen Klopfpunkt berechnet um anschließend (durch Größenvergleich) den Sensor  $S_0$  mit dem geringsten Abstand zu ermitteln. Dann wird der Weg  $(s_0)_n$  mit Gleichung 11 und daraus die Differenzen  $(\Delta s_1)_n$  und  $(\Delta s_2)_n$  mit Gleichung 12 berechnet.

Es werden also die Punkte

$$K_n(\Delta z \cdot m_l; 30; \Delta z \cdot m_h); \Delta z \in Q; m_l \in \left[0; \frac{60\text{cm}}{\Delta z[\text{cm}]} \right]; m_h \in \left[0; \frac{30\text{cm}}{\Delta z[\text{cm}]} \right]; n = 1 \dots m_l \cdot (m_h + 1)$$

beginnend im Punkt  $K_0(0;30;0)$  bis zum Punkt  $K_n(60;30;31)$  durchlaufen und in jedem dieser Punkte die drei Unbekannten  $(s_0)_n$ ,  $(\Delta s_1)_n$  und  $(\Delta s_2)_n$  des Gleichungssystems I berechnet. Anschließend wird die Summe dieser Differenzen (Istwert) gebildet und mit dem gemessenen Wert (Sollwert) verglichen, indem die Differenz zwischen Istwertsumme und Sollwertsumme gebildet wird.

$$D_n = \left| \left[ (\Delta s_{1,ist})_n - (\Delta s_{1,soll})_n \right] + \left[ (\Delta s_{2,ist})_n - (\Delta s_{2,soll})_n \right] \right| \quad \text{Gleichung 13}$$

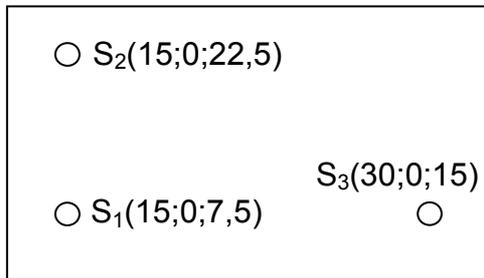
Nun wird die Stelle mit dem minimalen Wert von  $D_n$  gesucht. Diese Stelle gibt den Klopfpunkt wieder.

## 4.3 Simulation zur Überprüfung der Anwendbarkeit des Suchalgorithmus

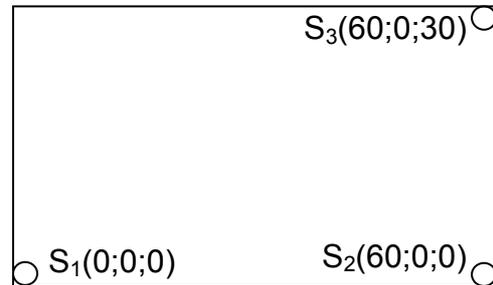
Es war nicht klar, ob die Minimierung mittels Gleichung 13 eine ideale Methode zum auffinden des Klopfpunktes ist. Daher wurde die Anwendbarkeit dieses Algorithmus mittels einer Simulation bestätigt. Dabei wurden für drei verschiedene Sensorkonfigurationen mit jeweils 1000 (Zufalls-) Emissionspunkten (wahllos auf der Vorderseite siehe Abbildung 4-1) zuerst die zugehörigen Laufwegdifferenzen berechnet, um anschließend über den vorliegenden Algorithmus den jeweiligen Emissionspunkt zu bestimmen. Anschließend wurden die Differenzen zwischen den ermittelten Werten und den Sollwerten untersucht. Im Folgenden werden die Ergebnisse dieser Simulation kurz dargestellt.

## Verwendete Sensorkonstellationen

Konstellation I)



Konstellation II)



Konstellation III)

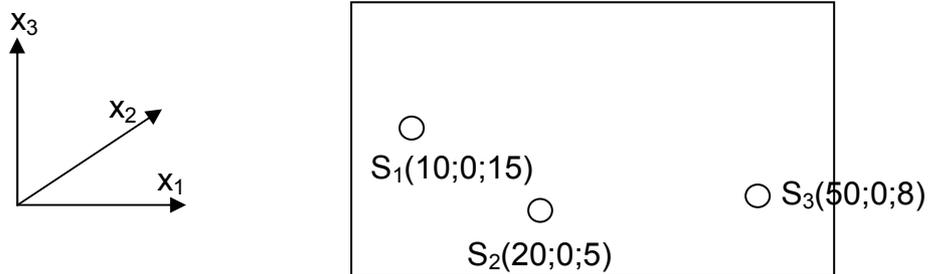


Abbildung 4-2: Schemata der drei verwendeten Sensorkonstellationen

Die Verteilung der Abweichungen zwischen den Koordinaten der ermittelten Werte und den Koordinaten der zugehörigen Sollwerte zeigen die folgenden Graphiken:

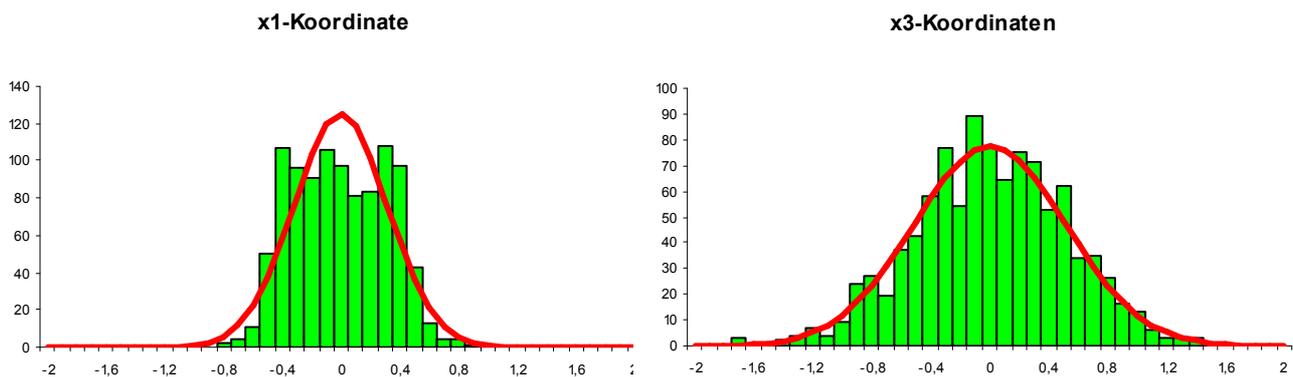


Abbildung 4-3: Verteilung der Abweichungen von den Sollwerten der  $x_1$ -Koordinate bzw. der  $x_3$ -Koordinate bei Konstellation I) in cm bei 1000 Simulationen. grün: gemessene Verteilung; rot: aus ermittelten Kennwerten bestimmte Näherung für die Normalverteilung

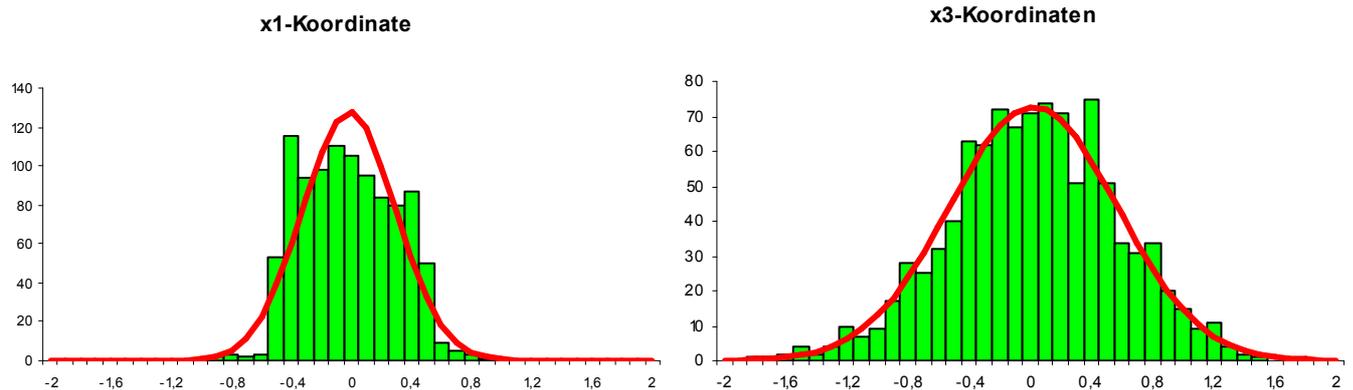


Abbildung 4-4: Verteilung der Abweichungen von den Sollwerten der  $x_1$ -Koordinate bzw. der  $x_3$ -Koordinate bei Konstellation II) in cm bei 1000 Simulationen. grün: gemessene Verteilung; rot: aus ermittelten Kennwerten bestimmte Näherung für die Normalverteilung

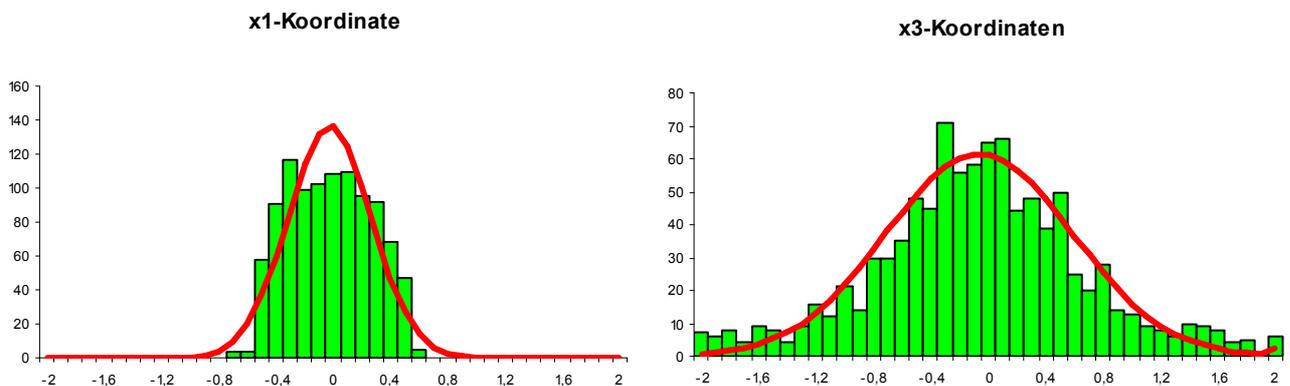


Abbildung 4-5: Verteilung der Abweichungen von den Sollwerten der  $x_1$ -Koordinate bzw. der  $x_3$ -Koordinate bei Konstellation III) in cm bei 1000 Simulationen. grün: gemessene Verteilung; rot: aus ermittelten Kennwerten bestimmte Näherung für die Normalverteilung

Zu jeder der drei Konstellationen wurde die mittlere absolute Abweichung vom Sollwert und die zugehörige Standardabweichung für die beiden berechneten Koordinaten  $x_1$  und  $x_3$  bestimmt. Die folgende Tabelle zeigt die zugehörigen Werte:

	Konstellation I)	Konstellation II)	Konstellation III)
Mittlere Abweichung $x_1$	-0,043	-0,0154	-0,0244
Standardabweichung $x_1$	0,32	0,313	0,292
Mittlere Abweichung $x_3$	0,021	0,0186	-0,0644
Standardabweichung $x_3$	0,516	0,548	0,848

Die berechneten Werte streuen nur sehr gering um die Sollwerte. Außerdem kann man in den Graphiken gut erkennen, dass die berechneten Werte näherungsweise gaußverteilt sind, so dass der verwendete Algorithmus verwendet werden kann.

#### 4.4 Ermittlung des Anfangszeitpunktes der einzelnen Sensoren

Das Kernstück des Programms bildet die Ermittlung der in Gleichung 13 erwähnten Sollwerte, also der gemessenen Werte für  $\Delta s_1$  und  $\Delta s_2$ . Diese beiden Werte werden indirekt als Zeiten  $\Delta t$  [in Skalenteilen Skt] gemessen und anschließend in Weglängen  $\Delta s$  [in cm] umgerechnet.

Das Programm nimmt eine bestimmte Anzahl an Messpunkten, mit einer voreingestellten Abtastrate  $f$  [in Skt/s], pro Sekunde auf. Der Schall breitet sich mit der Geschwindigkeit  $c_w$  [in cm/s] durch das Wasser aus. Mit diesen Werten lässt sich eine Umrechnung von „ $\Delta t$  in Skalenteilen“ in „ $\Delta s$  in cm in Flüssigkeit“ durchführen, denn es gilt:

$$\Delta s = \Delta t \cdot c_w / f \quad \text{Gleichung 14}$$

Mit dieser Gleichung lassen sich die gemessenen Skalenteile (Zeitdifferenzen  $\Delta t$  in Skt) in Streckenstücke ( $\Delta s$  in cm) umrechnen. Mit der im Programm verwendeten Abtastrate von  $f = 64\text{kHz}$  ergibt sich mit der Schallgeschwindigkeit  $c_w = 1480\text{m/s}$ :

$$1\text{Skt} = 2,313 \text{ cm}$$

Es geht also in erster Linie darum Zeitdifferenzen zwischen den Sensoren zu ermitteln. Hierzu musste festgelegt werden, welche Stellen des Messsignals der drei Sensoren (siehe Abbildung 4-6) die jeweiligen Startzeitpunkte der Schallsignale darstellen. Hierzu wurde eine Reihe von Ansätzen untersucht. Diese werden in den folgenden Kapiteln 4.4.1 bis 4.4.4 dargestellt.

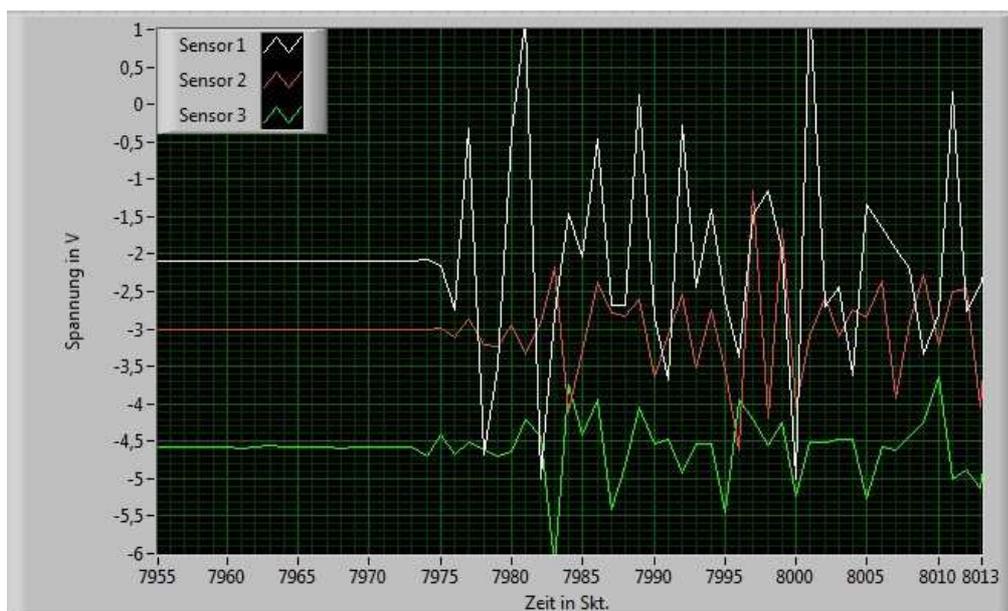


Abbildung 4-6: Signalverlauf der 3 Sensoren

#### 4.4.1 Ermittlung durch einen einfachen Trigger

In einem ersten Ansatz wurde versucht die Anfangszeitpunkte mit Hilfe eines in Labview vorprogrammierten tools Namens „trigger and gate“ zu ermitteln. Diesem führt man das Datenpaket zu. Man muss einen der Kanäle des Paketes für den Trigger auswählen und den Schwellwert einstellen. Zusätzlich ist es möglich auszuwählen, ob der Trigger bei steigendem oder fallendem Erreichen des Schwellwertes reagieren soll und nach wie vielen Messpunkten der Trigger erstmals wieder reagieren können soll. Dieser Trigger wurde verknüpft mit einem weiteren tool „Statistics“, mit dem es möglich ist sich einen Startzeitpunkt  $T_S$  (hier des Triggers) angeben zu lassen. Die Implementierung zu diesem Ansatz findet sich im Kapitel 7.1. Die ersten Berechnungen wurden mit diesen ermittelten Startzeitpunkten durchgeführt. Die hierbei berechneten Werte (Istwerte) wiesen allerdings erhebliche Abweichungen vom Sollwert (wirkliche Klopfstelle) auf. Dies hatte (wie sich letztlich herausstellte) den einfachen Grund, dass man nicht einen bestimmten Schwellwert vorgeben kann, sondern mittels untersuchender Algorithmen die Schwelle berechnen muss. Die Einstellung eines festen Schwellwertes stellt ein sehr grobes Vorgehen dar. Aufgrund der fest vorgegebenen scan-Rate wird das Schallsignal nur zu diskreten Zeitpunkten erfasst. So kann es leicht passieren, dass zwei aufeinander folgende erfasste Zeitpunkte Stellen darstellen, die unmittelbar vor bzw. hinter einem Extremum (Peak) liegen und bei hoher Frequenz einen viel kleineren Wert liefern, so dass der Trigger erst zu einem späteren Zeitpunkt reagiert. Eine korrekte Berechnung mit dieser Anordnung ist somit nur zufällig möglich. Daher war es für eine stabile (wiederholbare) Ermittlung der Anfangszeitpunkte unumgänglich, das von den einzelnen Sensoren aufgenommene Signal genauer zu analysieren. Aufgrund dieser Analyse wäre es möglich gewesen eine variable Schwelle zu programmieren. Diese Idee kam aber erst nach der Verwirklichung der folgenden Ideen, so dass dieser Ansatz nicht weiter verfolgt wurde. Um diese Analyse durchführen zu können, wurde ein Algorithmus erstellt, der das Programm anhält sobald geklopft wird und eine Graphik wie in Abbildung 4-7 dargestellt liefert.

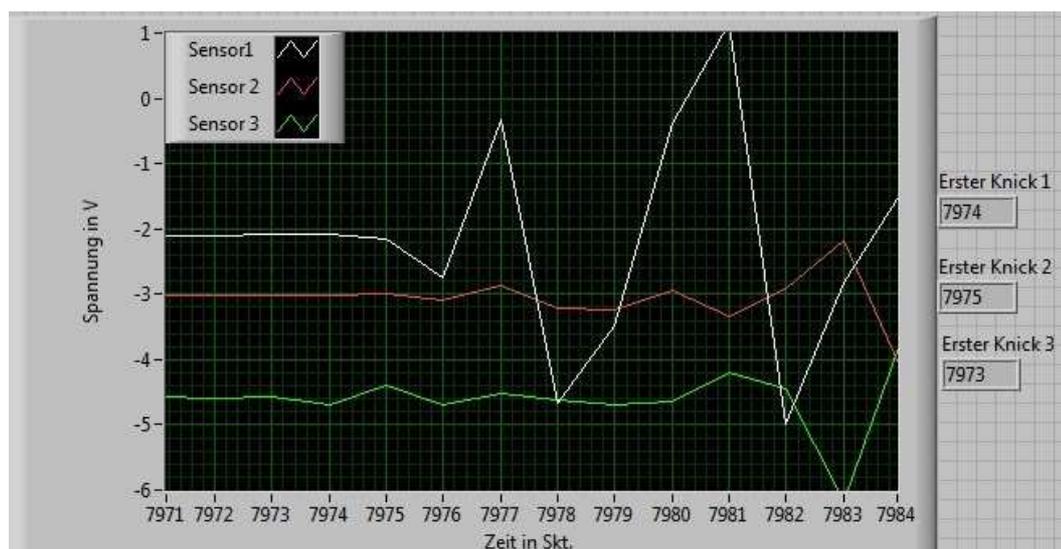


Abbildung 4-7: Graphische Darstellung eines typischen Signals dreier Sensoren von oben  $H_1$ ,  $H_2$ ,  $H_3$

#### 4.4.2 Ermittlung mittels der ersten Knickstellen

Die Idee, die Anfangszeitpunkte mittels Trigger zu bestimmen wurde bald verworfen. Wenn man sich die Sensordaten in Abbildung 4-7 ansieht, erkennt man, dass Rauschen und Klopfsignal mit dem Auge klar unterscheidbar sind. Das Signal-zu-Rausch-Verhältnis ist also bei starkem Klopfen gut. Die Abtastrate  $f$  wurde so gewählt, dass der Signalanfang als Knick im abgetasteten Signal zu erkennen ist. Es musste nun überlegt werden, wie man diese ersten Knicke von den im Rauschsignal vorkommenden Knicken unterscheiden kann. Eine genauere Untersuchung der Knicke ergab, dass es am sinnvollsten war, über Steigungsdifferenzen zu unterscheiden. Hierzu wurde aus den jeweils drei aufeinander folgenden Punkten  $P_{i-1}$ ,  $P_i$  und  $P_{i+1}$  zwei Steigungswerte und deren Differenz berechnet mit

$$m_1 = \frac{U_i - U_{i-1}}{t_i - t_{i-1}} = y_i - y_{i-1} \quad \text{Gleichung 15}$$

$$m_2 = \frac{U_{i+1} - U_i}{t_{i+1} - t_i} = y_{i+1} - y_i \quad \text{Gleichung 16}$$

und 
$$\Delta m = |m_1 - m_2| \quad \text{Gleichung 17}$$

Erst wenn diese Differenz (Gleichung 17) einen bestimmten Schwellwert  $S$  (dieser ist für jeden Sensor verschieden einstellbar) überschreitet, wird die Stelle als möglicher Anfangsknick des Klopfsignals genommen. Die  $t$ -Koordinaten sind Zeitpunkte (in Skt), die  $U$ -Koordinaten gemessene Spannungen (in V). für zwei aufeinander folgende Punkte gilt  $\Delta t = 1\text{Skt}$ . In der Abbildung 4-7 (rechts) sieht man, welche Werte dieser Algorithmus für  $S = 0,03$  berechnet. Die Implementierung (Unterprogramm „Knickfinder.vi“) zu diesem Ansatz findet sich im Kapitel 7.1.3.

### 4.4.3 Ermittlung mittels der ersten Extremstellen des Signals

Ein Problem der Methode der ersten Knicke ist, dass bei stärkerem Klopfen sich oft kein eindeutiger erster Knick ausmachen lässt. Wenn man die Schwelle zu klein wählt passiert es zum einen, dass ein Knick im Rauschsignal bereits die Bedingungen erfüllt (siehe die ermittelten Werte in Abbildung 4-7), zum anderen sind es Nebensignaleffekte der Messkarte, die zu Knicken führen, die so ermittelt werden. Stellt man die Schwelle zu groß ein, so werden die richtigen ersten Knicke oft einfach „übersehen“. Wenn man sich die Abbildung 4-7 genauer ansieht, so sieht man, dass es auch gar nicht so leicht auszumachen ist, wo denn dieser erste Knick ist. Was man aber gut unterscheiden kann, sind die ausgeprägten Extremwerte bei den Werten 7977 und 7983. Eine andere Idee für die Wahl des Anfangszeitpunktes war daher die Verwendung der ersten auftretenden Extrema. Der Algorithmus, mittels dem diese Extremstellen zu finden sind, steckt in der folgenden Bedingung 1. Die Implementierung hierfür findet sich im Kapitel 7.1.4 (Unterprogramm „Extremafinder.vi“).

Diese Extremstellen konnten mit der Bedingung 1 ermittelt werden:

$$|U(k-1)| < |U(k)| > |U(k+1)| \quad \wedge \quad |U(k) - U_m| = U_{abs} > C \in \mathbb{R} \quad \text{Bedingung 1}$$

$U(\dots)$  sind die Spannungswerte zu aufeinander folgenden Zeitpunkten

$C$  ist eine Schwelle, ab der ein Extremum als Anfangsextremum erkannt werden soll

$U_m$  der Spannungsmittelwert des jeweiligen Sensors, über einen Wertebereich einer Schleife im Ruhezustand. Dieser Mittelwert wird im Unterprogramm „Mittelwert.vi“ (siehe Kapitel 7.1.2) unter der Bedingung 6 berechnet.

#### 4.4.3.1 Optimierung mittels Fallunterscheidung

Für diese Extremwerte ergibt sich allerdings ein neues Problem: Es treten die bereits oben erwähnten Nebensignaleffekte (siehe Kapitel 3.1.1 – III) auf, die von der Messeinrichtung herrühren.

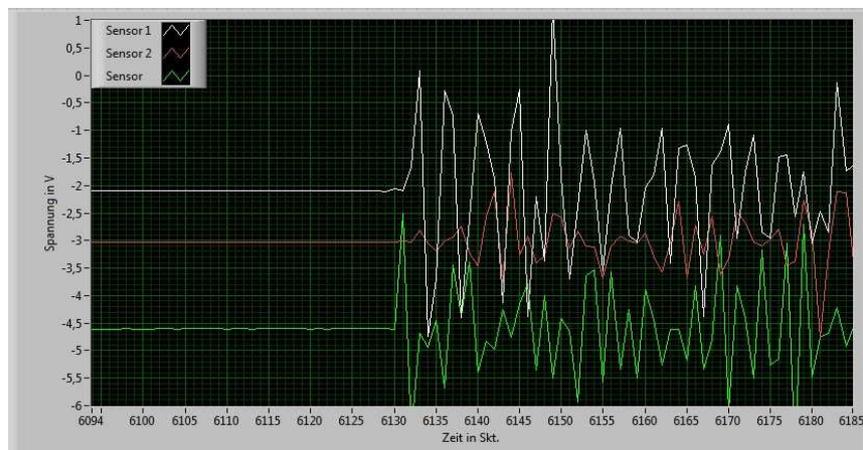


Abbildung 4-8: Zur Veranschaulichung der Nebensignaleffekte

Dieses Problem wird in Abbildung 4-8 verdeutlicht. Bei dieser Aufnahme wurde der Klopfpunkt so gewählt, dass er dem Sensor H<sub>1</sub> und Sensor H<sub>3</sub> etwa gleich nahe liegt, aber relativ weit entfernt vom Sensor H<sub>2</sub>. Man erkennt hier deutlich, dass der Sensor H<sub>2</sub> zum Zeitpunkt des ersten Peaks des Sensors H<sub>1</sub> auch einen kleinen Peak aufweist. Dieser Peak wird von dem Unterprogramm „Extremumfinder.vi“ als Anfangszeitpunkt des Sensors H<sub>2</sub> berechnet, obwohl der richtige Zeitpunkt (großer Peak bei 6142) erst später auftritt. Um diesen Fehler zu vermeiden, wurde ein Unterprogramm geschrieben, das mittels einer Fallunterscheidung solche falschen Peaks erkennt und zum nächsten richtigeren führt. Dabei untersucht das Unterprogramm „Fallunterscheidung.vi“, ob zwei bzw. drei Stellen erster Peaks (Anfangszeitpunkte T<sub>S</sub>) höchstens 2 Skt. voneinander entfernt sind.

$$|T_{S1} - T_{S2}| \leq 2 \quad |T_{S3} - T_{S2}| \leq 2 \quad |T_{S1} - T_{S3}| \leq 2 \quad \text{Bedingungen 2}$$

Wenn eine der Bedingungen 2 erfüllt ist, so wird ermittelt, welcher dieser Peaks der größte ist, um damit zu bestimmen, welcher Sensor der Klopfstelle am nächsten liegt. Anschließend wird ein relativer Größenvergleich zwischen den kleineren Peaks und diesem größten Peak gemacht:

$$U_{abs} > 0,3 \cdot U_{abs,max} \quad \text{Bedingung 3}$$

Es handelt sich bei den hier verwendeten 30% um einen empirisch gewonnenen Wert, der sich als geeignet herausstellte. Wenn nun diese Bedingung 3 für den kleineren Peak erfüllt ist, so entscheidet sich der Algorithmus für diesen – der Klopfpunkt liegt auf der Mittelsenkrechten (geometrischer Ort aller Punkte mit gleichem Abstand zu zwei vorgegebenen Punkten) zwischen den jeweiligen beiden Sensoren. Ist die Bedingung 3 dagegen nicht erfüllt, so werden die Werte des betroffenen Sensors erneut untersucht, diesmal aber eine weitere Bedingung berücksichtigt, nämlich dass das zu suchende Extremum mindestens um 20% größer sein muss, als das zuerst gefundene:

$$U_{abs,neu} > 1,2 \cdot U_{abs,alt} \quad \text{Bedingung 4}$$

Beim Aufsuchen des nächsten Extremums musste ein Kompromiss zwischen zwei Problemen gefunden werden:

- sucht man nach Extrema die viel größer sind, als das zuerst gefundene so wird meist eines genommen, das weit hinter dem „richtigen“ liegt
- nimmt man dagegen eines, das etwa genau so groß ist, wie das erste, so wird meist wieder eines gewählt das einem zu frühen Zeitpunkt entspricht.

Der gewählte Wert von 20% erwies sich als derjenige, mit dem dieser Kompromiss am ehesten gefunden wurde.

#### 4.4.4 Ermittlung mittels Kombination erster Knickstellen mit ersten Extremstellen

Tests haben gezeigt, dass mit letztem Algorithmus häufig gute Werte für die gesuchten Anfangszeiten gefunden werden. Allerdings muss unterschieden werden zwischen starkem und schwachem Klopfen. Bei starkem Klopfen ist dieser Algorithmus sehr zuverlässig, klopft man aber schwach, so versagt dieser Algorithmus häufig.

Abhilfe konnte hier durch eine Kombination der beiden Programme „Knickfinder.vi“ und „Extremafinder.vi“ in Zusammenhang mit „Fallunterscheidung.vi“ geschaffen werden. Das Programm untersucht zunächst, ob stark oder schwach geklopft wurde, in dem es die Extremwerte aus den 4000 Messerwerten der einzelnen Sensoren bestimmt und für diese untersucht ob:

$$\left| (U_{\max})_{abs}(H_1) \right| > 3V \vee \left| (U_{\max})_{abs}(H_2) \right| > 3V \vee \left| (U_{\max})_{abs}(H_3) \right| > 3V \quad \text{Bedingung 5}$$

Hierin sind die  $(U_{\max})_{abs}$  die absolut größten Spannungswerte innerhalb einer Messsignalreihe in der geklopft wurde. Dabei wurde festgelegt, dass sich ein starkes Klopfen daran erkennen lässt, dass wenigstens einer der von den Sensoren gemessenen größten Peaks größer als 3 Volt ist. Es wird also jeweils der größte Peak im Klopfsignalverlauf betrachtet und untersucht, ob dieser den Wert 3 Volt überschreitet oder nicht. Dann entscheidet sich das Programm bei schwachem Klopfen für die Werte aus dem Unterprogramm „Knickfinder.vi“ als Anfangszeitpunkte  $T_S$  bei starkem Klopfen dagegen werden hierfür die Werte aus dem Unterprogramm „Extremafinder.vi“ genommen.

#### **4.4.5 Fehlereinflüsse bei der Messung und was zu beachten ist**

Mit den in den Kapiteln 4.4.2 bis 4.4.4 beschriebenen Vorgehensweisen werden ausreichend gute Werte für die Anfangszeitpunkte der Sensorsignale ermitteln und daraus die Klopfkoordinaten berechnet. Allerdings sind die Messerwerte natürlich vielen Fehlereinflüssen unterworfen. In diesem Kapitel werden diese Fehlereinflüsse dargestellt und ein Schätzwert für die auftretenden absoluten Fehler bei der Messung ermittelt.

##### **Fehlereinflüsse**

###### systematische Fehler

- Bläschenbildung an Sensoroberfläche – das Wasser im Aquarium muss einige Tage stehen, da sonst Bläschenbildung auftritt; diese Bläschen benetzen die Sensoren und schirmen die Druckwellen ab – dadurch kann der Algorithmus versagen, wenn das Signal zu schwach ist.
- Auch dauert es einige Tage, bis die Oberflächen der Sensoren ganz mit Wasser benetzt sind. Durch anfängliche Lufteinschlüsse an den Sensoroberflächen kommt es zu Abschirmeffekten, so dass anfangs z.B. kein starkes Klopfen als solches wahrgenommen wird – auch in diesem Fall würde der Algorithmus evtl. versagen
- Abtastung (nur diskrete Messung)  
Ein Skalenteil entspricht einer bestimmten Wegdifferenz. Jeder Differenz zwischen Skalenteilen lassen sich somit nur Bereiche für die Wegdifferenzen zuordnen

###### statistische Fehler

- aufgrund der diskreten Abtastung kann es zu dem unter Kapitel 4.4.1 beschriebenen Abtastfehlern kommen. Durch das Verwenden erster Knicke bzw. Extrema wird der Fehlereinfluss zwar verringert, ist aber nicht beseitigt.
- Der Anfangszeitpunkt lässt sich mit den beschriebenen Ideen nur sehr grob finden. Um diesen immer richtig zu finden sind umfangreichere Fallunterscheidungen nötig
- Hierunter fällt auch, dass unterschiedlich starkes Klopfen genauer differenziert werden müsste – mit „stark“ und „schwach“ ist dies nur sehr grob differenziert und daher sehr ungenau

Es muss an dieser Stelle ausdrücklich betont werden, dass die hier verwendeten Vorgehensweisen (das Finden der Anfangszeitpunkte durch Knicke und Extremstellen) nur im vorliegenden Problem mit den hier benutzten Einstellungen so realisierbar sind. Eine höhere Abtastrate hätte beispielsweise zur Folge, dass es die hier so ausgeprägten Knicke (diese werden über Steigungen identifiziert) nicht mehr gäbe.

#### 4.4.6 Abschätzung der Größe systematischer und statistischer Fehler

Nach der Gleichung 14 lassen sich aus den gemessenen Zeitpunkten Wegdifferenzen berechnen. Da bei der Messung nur diskrete Zeitpunkte abgetastet werden, gibt es eine Grenze für die mögliche Genauigkeit der Koordinatenbestimmung nach Kapitel 4.2. Denn zu zwei Werten  $\Delta s_1$  und  $\Delta s_2$ , gibt es keinen eindeutigen Klopfpunkt, sondern einen Bereich, in dem dieser Punkt liegen könnte. Nähere Untersuchungen (Kapitel 3.1.1) hatten hierbei gezeigt, dass diese Ungenauigkeiten von der relativen Lage des Emissionspunktes zu den drei Sensoren abhängen. Zur Abschätzung des Fehlers wird hier nochmals auf einen der acht Ausschnitte aus Kapitel 3.1.1 zurückgegriffen (siehe Abbildung 3-9). Hier wurde für eine bestimmte Sensorkonfiguration (siehe Abbildung 3-6 – Konstellation I) der Emissionspunkt in einer Umgebung von 15cm x 15cm variiert. Bei dem Ausschnitt in Abbildung 3-9 wurde der Emissionspunkt auf der Vorderseite des Aquariums nach Abbildung 4-1 in der Nähe der Sensoren variiert. An den Rändern der Abbildung sind die Koordinaten des jeweiligen Emissionspunktes angegeben. Die Tabelle zeigt die errechneten Laufwegdifferenzen der beiden weiter vom Emissionspunkt entfernten Sensoren. In dem verwendeten Ausschnitt sind das die Sensoren  $H_2$  und  $H_3$  aus Abbildung 3-6. Es wurden jeweils die beiden Laufwegdifferenzen bzgl. dem dritten Sensor  $H_1$  (mit Laufwegdifferenz  $\Delta s_0=0$ ) berechnet. Die Farben zeigen die Bereiche mit genau gleichen Differenzen, also Bereiche mit Emissionspunkten, die zu den gleichen Laufwegdifferenzen führen. Im Mittel lässt sich aus der Darstellungen ein statistischer Fehler (für das Auffinden des Punktes nach dem in Kapitel 4.2 beschriebenen Algorithmen) für die berechnete Klopfpunktcoordinate angeben; dieser liegt bei etwa (aus den erwähnten Abbildungen ermittelt):

$$\Delta x_{M,i} = 2cm \quad i=1;2$$

D.h., dass bei der Anbringung der Sensoren darauf zu achten ist, dass diese mit einer Genauigkeit von etwa 2cm montiert werden müssen. Nimmt man an, dass die Bestimmung der Anfangszeitpunkte (nach den unter den Kapiteln 4.4.2 bis 4.4.4 beschriebenen Algorithmen) für jeden der drei zur Berechnung herangezogenen Sensoren mit einem Fehler von

$$\Delta t = 2Skt \quad \rightarrow \quad \Delta x_{A,i} = 4,63cm$$

behaftet ist, so ergibt sich eine Ungenauigkeit der Koordinaten des errechneten Punktes von:

$$\Delta x_{i,ges} = \sqrt{(2cm)^2 + (4,63cm)^2} = 5,04cm \quad i=1;2$$

---

#### **4.4.7 Verbesserung der Ergebnisse - Mittelwert aus den Kombinationen mehrerer Sensoren**

Um nun die im Kapitel 4.4.5 geschilderten Fehlereinflüsse möglichst zu kompensieren, ist es sinnvoll, die Berechnung der Koordinaten des Emissionspunktes als Mittelwert aus mehreren Messungen zu bilden. In Antares wird hierfür eine Vielzahl an akustischen Sensoren benutzt. Im vorliegenden Modell ist diese Zahl hauptsächlich durch den beschränkten Platz begrenzt. Außerdem ist es nicht sinnvoll, für ein einfaches Modell zu viele Sensoren heranzuziehen. Daher wurde beschlossen, das Modell mit insgesamt fünf Sensoren auszustatten. Bei dieser Anzahl von Sensoren ist es möglich, insgesamt 10 verschiedene Kombinationen (3 aus 5) zu bilden. Somit können für einen Klopfpunkt 10 verschiedene Positionen ermittelt werden. Dies hat den Vorteil, dass selbst dann ein Klopfpunkt berechnet werden kann, wenn der Klopfpunkt ungünstig bezüglich einer (oder sogar mehrerer) Sensorkonfiguration liegt (siehe Kapitel 3.1.1). Denn bei nur einem berechneten Punkt kommt es ab und an vor, dass die berechneten Koordinaten unbrauchbar sind. Hat man aber mehrere Messwerte, so ist es wahrscheinlicher dass mehrere davon brauchbar sind. Dabei ist allerdings auch zu bedenken, dass eine zu große Zahl unbrauchbarer Werte<sup>1</sup> bedeuten könnte, dass die Messung insgesamt schlecht war (z.B. wegen eines nicht berücksichtigten Sonderfalls). Deshalb wurde hier ein Algorithmus entworfen, der die Anzahl der unbrauchbaren Werte zählt. Übersteigt diese Zahl einen bestimmten Wert, so meldet das Programm, dass erneut geklopft werden muss, da die Messung nicht vertrauenswürdig ist.

Um das Messergebnis nun noch zusätzlich zu verbessern wurde ein weiterer Algorithmus erstellt, der durch systematisches Ausschließen der vom Mittelwert am weitesten entfernten Punkte, einen besseren Wert ermittelt. Hierzu werden die Abstände aller gemessenen Koordinatenpaare zu deren Mittelwert bestimmt und anschließend das Paar gestrichen, das den größten Abstand hat. Nun wird der Mittelwert erneut berechnet und das ganze wiederholt. Im vorliegenden Programm werden auf diese Weise die drei schlechtesten Messwerte ausgeschlossen.

---

<sup>1</sup> Mit einem schlechten Wert ist hier ein Nullpaar (0/0) gemeint, wenn also der Algorithmus aus Kapitel 4.2 keinen Wert findet, der besser ist als der schlechte Anfangswert (10)

## 5 Darstellung der Ergebnisse

In diesem Kapitel wird zum einen aufgezeigt, wie gut bestimmte Klopfpunkte errechnet werden und wie gut dabei die Wiederholbarkeit der Messergebnisse ist, zum anderen werden die wesentlichen Algorithmen bei der Laufwegdifferenzermittlung an Beispielen dargestellt. Bei der Durchführung der Testläufe wurde die folgende, Anbringung der Sensoren gewählt. Diese ist so auch in etwa für den späteren Einsatz des Modells vorgesehene. Bei den Maßen ist die Symmetrie zu beachten.

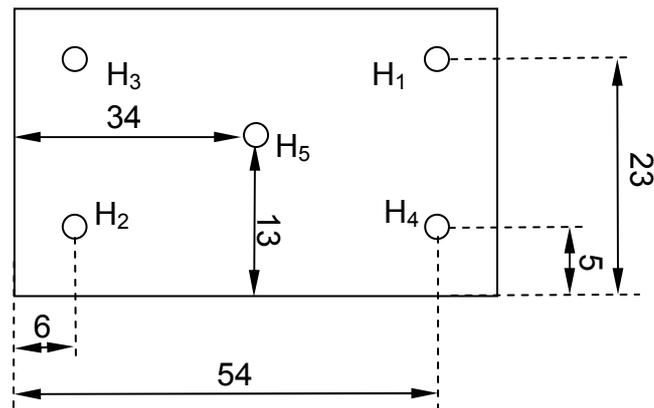


Abbildung 5-1: Abmessungen der Sensorverteilung

### 5.1 Wiederholbarkeit der Messung eines Klopfpunktes

Zunächst wurde an acht verschiedenen Stellen des Aquariums jeweils etwa 20-mal geklopft, um die Verteilung der jeweils berechneten Mittelwerte aufzuzeigen. Diese Verteilungen sind in den folgenden Graphiken dargestellt. Der besseren Übersicht wegen wurden hier zwei Graphiken erstellt:

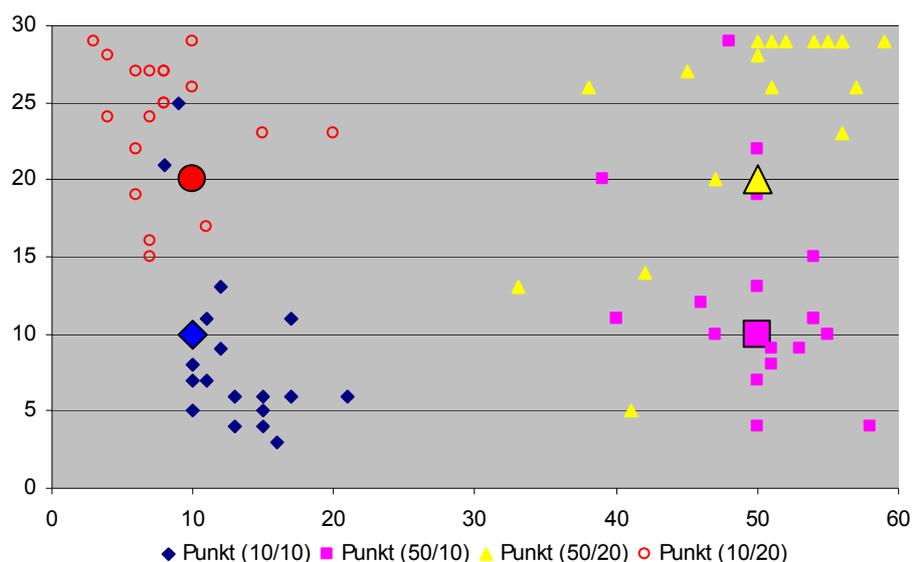


Abbildung 5-2 a: Darstellung der berechneten Mittelwerte aus den 10 Kombinationen der 5 Sensoren aus Abbildung 5-1 bei jeweils 20-maligem Klopfen für vier verschiedene Punkte. Große Symbole: tatsächliche Position des Klopfpunktes; kleine Symbole (in entsprechender Farbe): gemessene Positionen

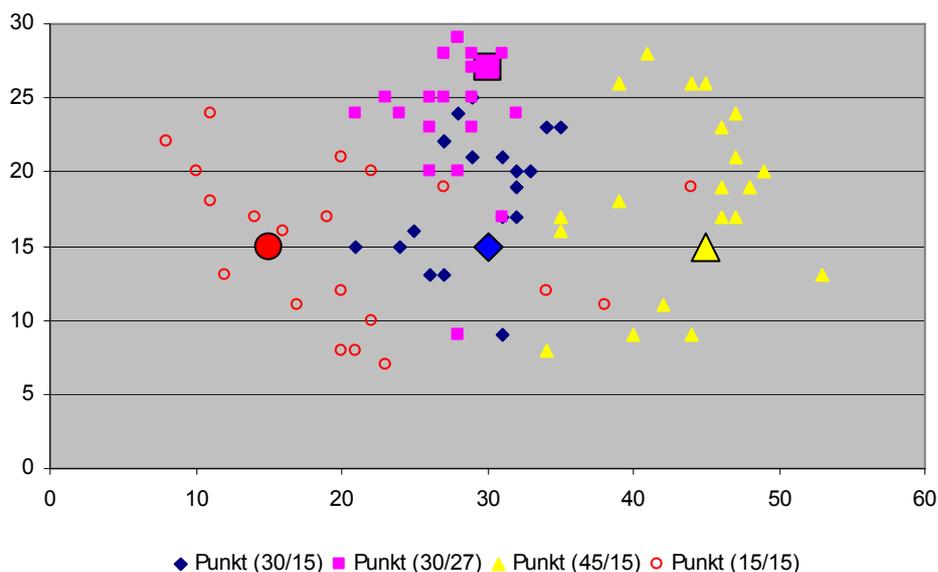


Abbildung 5-2 b: Darstellung der berechneten Mittelwerte aus den 10 Kombinationen der 5 Sensoren aus Abbildung 5-1 bei jeweils 20-maligem Klopfen für weitere vier verschiedene Punkte. Große Symbole: tatsächliche Position des Klopfpunktes; kleine Symbole (in entsprechender Farbe): gemessene Positionen

Dabei sind die fett eingezeichneten Markierungen jeweils die Sollwerte für den zu errechnenden Punkt. Die Abweichungen der errechneten Werte vom jeweiligen Sollwert wurden tabellarisch zusammengestellt und ausgewertet. Die folgenden Graphiken stellen die Häufigkeitsverteilung für die Abweichung der  $x_1$ -Koordinate und der  $x_3$ -Koordinaten von der jeweiligen Sollkoordinate dar:

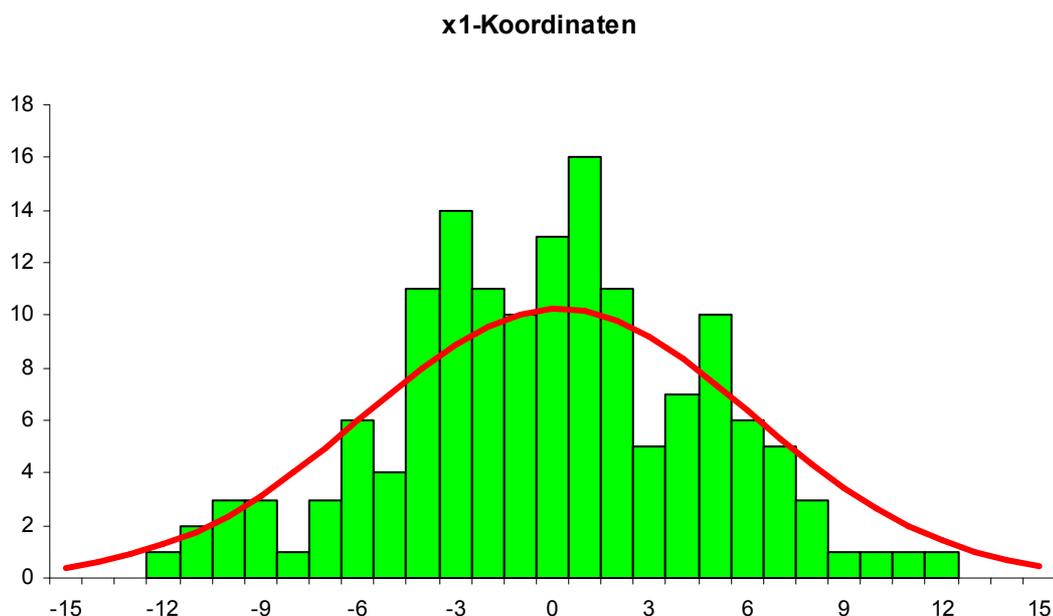


Abbildung 5-3: Verteilung der Differenzen zwischen Sollwert und Istwert der bei 20 Messungen berechneten Werte für die  $x_1$ -Koordinaten des Emissionspunktes (grün). rot: aus den ermittelten Kennwerten bestimmte Näherung für die zugehörige Gaußverteilung

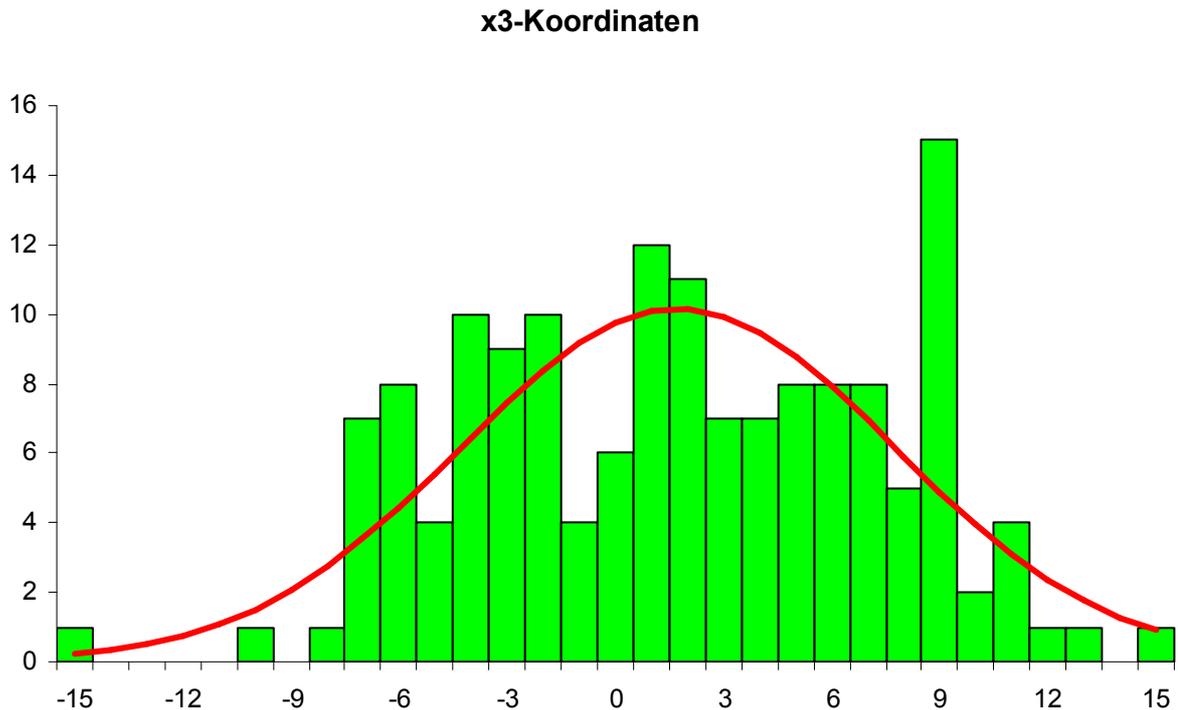


Abbildung 5-4: Verteilung der Differenzen zwischen Sollwert und Istwert der bei 20 Messungen berechneten Werte für die  $x_2$ -Koordinaten des Emissionspunktes (grün). rot: aus den ermittelten Kennwerten bestimmte Näherung für die zugehörige Gaußverteilung

Aus den beiden Verteilungen wurden die folgenden Mittelwerte mit zugehöriger Standardabweichung ermittelt:

	Mittlere Abweichung	Standardabweichung
$x_1$ -Koordinate	0,2	6,0
$x_3$ -Koordinate	1,7	6,0

Die Standardabweichungen in den Messungen stimmen relativ gut mit den in Kapitel 4.4.5 angenommenen Abweichungen überein, sind allerdings etwas größer. Dies liegt zum einen daran, dass die Algorithmen nach Kapitel 4.4 nicht alle Fälle berücksichtigen, die bei der Anfangszeitpunktsbestimmung auftreten. Solche Fälle führen zu den Ausreißern. Maßnahmen zur Verbesserung werden im Ausblick (siehe Kapitel 6) dargestellt. Zum anderen wurden bei der Abschätzung nur die größten Fehlereinflüsse berücksichtigt. Im Rahmen dieser Arbeit, bei der es darum geht ein einfaches Modell zu erstellen ist dies zur Abschätzung aber ausreichend.

## 5.2 Darstellung der wesentlichen Algorithmen an ausgewählten Beispielen

Im Folgenden wird ein Beispiel aufgezeigt, bei dem dargestellt wird, wie das Programm sich zunächst nach der unter Kapitel 4.4.4 beschriebenen Bedingung 5 für den Fall starken Klopfens entscheidet. Bei dem Beispiel wurde etwa in der Mitte zwischen dem Sensoren H<sub>2</sub> und H<sub>3</sub> geklopft. Es wurde so stark geklopft, dass die unter Kapitel 3.1.1 (Punkt III) dargestellten Nebensignaleffekte zu erkennen sind. Diese werden auch vom Programm erkannt so dass dieses mittels der programmierten Algorithmen zu den nächsten besseren Peaks wechselt.

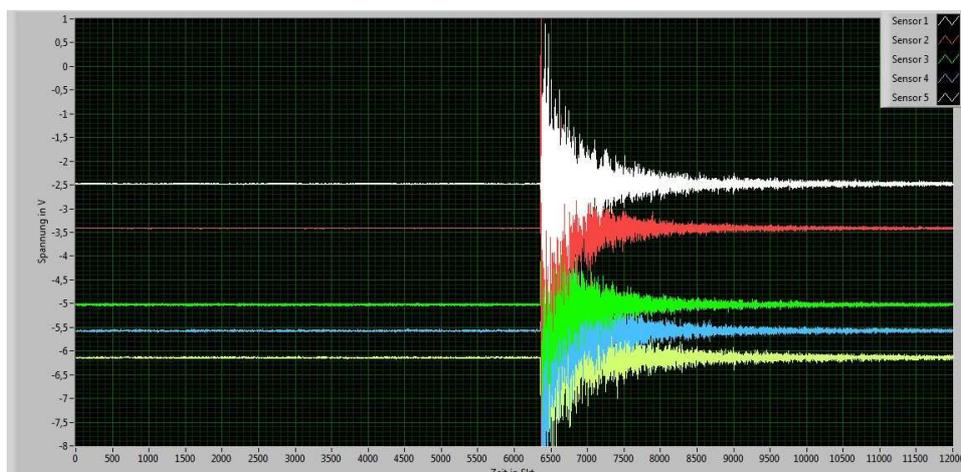


Abbildung 5-5: Darstellung des gesamten Plots der fünf Sensoren

In der Abbildung 5-5 ist gut zu erkennen, dass die Stärke des Klopfens so groß war, dass die größten auftretenden Peaks die 3 Volt überschreiten, so dass die Bedingung 5 erfüllt ist, und sich das Programm für die ermittelten ersten Extremstellen entscheidet. Der hier interessante Bereich des tatsächlichen Klopfmoments ist in der Abbildung 5-6 nochmals vergrößert dargestellt.

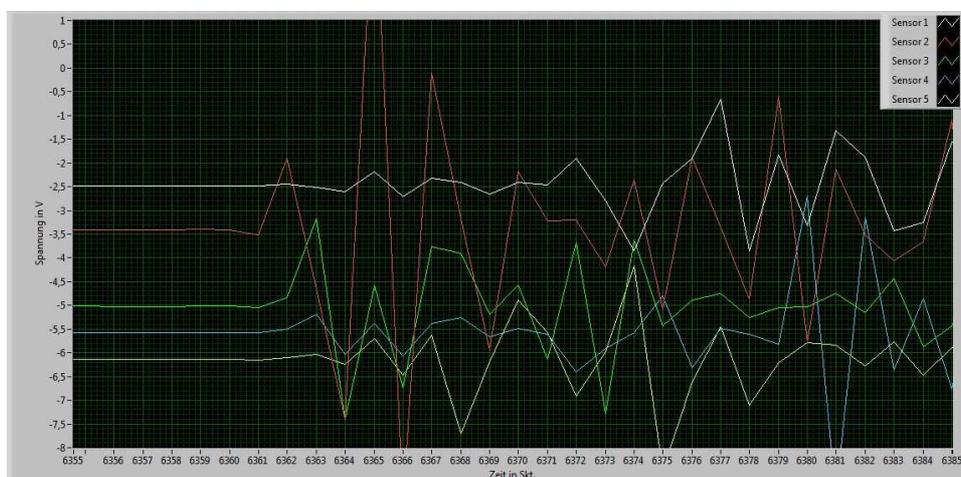


Abbildung 5-6: Vergrößerte Darstellung des gesamten Plots der fünf Sensoren auf den Bereich in dem das Klopfersignal zeitlich erfasst wird

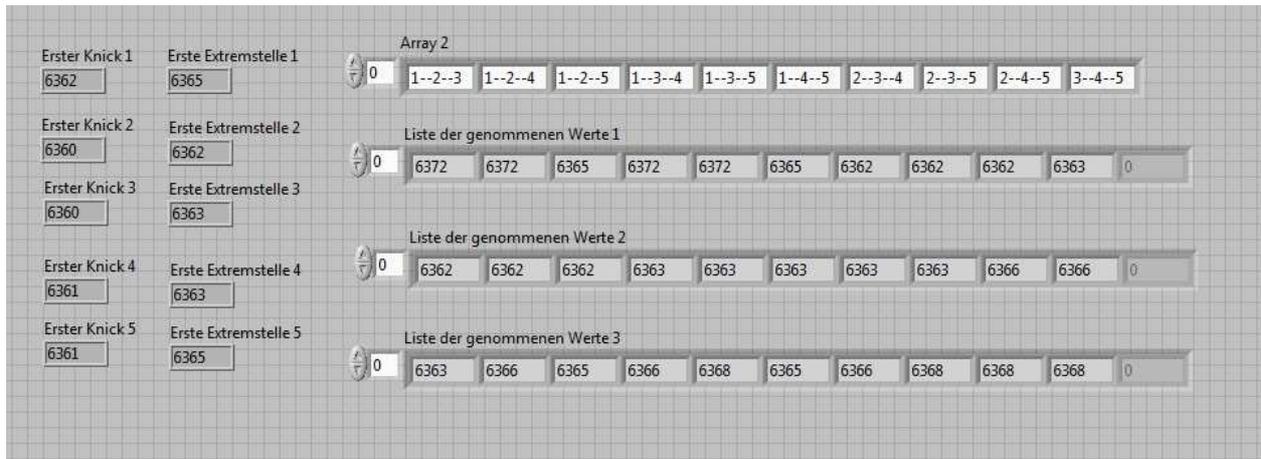


Abbildung 5-7: Werte für die Anfangszeitpunkte aus den oberen Plots – links die ermittelten Werte „erster Knicks“ und „erste Extremwerte“ – rechts die letztlich genommenen Werte für die einzelnen Sensorkonfigurationen nachdem die unter Kapitel 4.4.3.1 beschriebene (wegen des hier vorliegenden Falles starken Klopfens) Fallunterscheidung durchlaufen wurde

In der Abbildung 5-7 sieht man links, welche Werte das Programm zum einen als „erste Knickstellen“ zum anderen als „erste Extremwerte“ errechnet hat. Da die Bedingung 5 erfüllt ist (maximaler Peak über 3 Volt) nimmt das Programm hier die rechten Werte „erste Extremstellen“. Wie unter Kapitel 4.4.3.1 beschrieben durchläuft das Programm nun eine entsprechende Fallunterscheidung. Betrachtet man die Signale der ersten drei Sensoren (weiß/rot/grün) so erkennt man gut, dass die Sensoren  $H_2$  und  $H_3$  das Signal als erste wahrnehmen (da dies etwa in der Mitte zwischen den beiden eingeleitet wurde). Auch gut zu erkennen ist, dass der Sensor  $H_1$ , der das Signal eigentlich als letzter wahrnehmen sollte aufgrund der Nebensignaleffekte durch die anderen Signale einen Ausschlag zeigt. So sind die zunächst berechneten Werte für die „ersten Extremstellen“ dieser drei Sensoren relativ nahe beieinander ( $H_1$ : 6365 Skt;  $H_2$ : 6362 Skt;  $H_3$ : 6363 Skt). Da die Messerwerte von  $H_1$  und  $H_3$  (auch von  $H_2$  und  $H_3$  – es wird ja auch zwischen den beiden geklopft) nur zwei Skalenteile entfernt sind, vergleicht der Algorithmus die Peakhöhen dieser beiden Messerwerte und erkennt, dass dies nicht die richtige Anfangszeit des Sensors  $H_1$  ist. Letztlich entscheidet sich das Programm für den Peak bei 6372 Skt (siehe die Kombination 1-2-3 links in Abbildung 5-7. Sieht man sich die Werte in dieser Tabelle genauer an, so sieht man allerdings, dass der genommene Anfangswert für den Sensor  $H_1$  variiert. Der Algorithmus wurde an dieser Stelle so erweitert, dass der größte (auf Grund der Fallunterscheidungen) gefundene Skt – Wert für den jeweiligen Sensor zur letztlich Berechnung der Koordinaten herangezogen wird. Also in diesem Beispiel der Wert 6372 Skt für den Anfangszeitpunkt des Sensors  $H_1$ .

## 6 Zusammenfassung und Ausblick

Ziel dieser Arbeit war die modellhafte Nachbildung der Arbeit des Lehrstuhls auf ANTARES, wobei das Detektionsvolumen mit einem Aquarium nachgebildet wurde. Das Erstellen eines solchen Modells wirft einige Fragen auf, die im Zuge dieser Arbeit beantwortet wurden. So stellte sich heraus, dass, wenn die Klopfstärke und -frequenz groß genug sind, man von der Verwendung eines Verstärkers für die Piezosensoren absehen konnte. Die Verteilung der Sensoren im Aquarium hängt von mehreren Faktoren ab. Zum einen von der gewählten Abtastrate zum anderen von den störenden Einflüssen der Nebensignaleffekten der Sensoren und nicht zuletzt davon welche Abmaße das Aquarium besitzt. Mit den verwendeten Komponenten ergab sich, dass die Sensoren am günstigsten isoliert, unter Wasser, mit einem Abstand von etwa 6cm zu den Wänden, anzubringen sind. Außerdem sollte der Abstand zwischen den Sensoren möglichst groß sein (in den Ecken), wobei einer der Sensoren (der an der DAQ-Karte zuletzt angeschlossenen) in der Mitte der Verteilung liegen sollte. Der Kern der gesamten Modellierung war die Antwort auf die Frage, wie man nun aus den von den Sensoren aufgenommenen Signalen auf die gesuchten Koordinaten eines Emissionspunktes kommt. Realisiert wurde dies mittels eines Algorithmus, der durch systematisches Ablaufen der Klopffläche den Punkt findet, in dem die einzelnen Laufwegdifferenzen zu den einzelnen Sensoren am wenigsten vom gemessenen Wert abweichen. Die Ermittlung dieser Laufwegdifferenzen stellte sich als schwierigste Aufgabe heraus. Um diese zu bekommen, musste untersucht werden, wo man die Anfangszeitpunkte der für die Signalerkennung der einzelnen Sensoren am besten wählt. Zunächst wurde hier mit einem Trigger gearbeitet, der beim Überschreiten einer bestimmten Schwelle reagiert. Dies stellte sich aber als zu ungenau heraus. Günstigere Werte ergaben sich unter Verwendung von ersten Knicken im Signal im Falle schwachen Klopfens und erster Extremstellen im Falle starken Klopfens. Allerdings funktioniert diese Vorgehensweise so nur im vorliegenden Modell bei den hier verwendeten Einstellungen (v.a. der Abtastrate). Um die Ergebnisse zu verbessern wurden insgesamt 5 Sensoren herangezogen, um aus den damit 10 möglichen Berechnungen einen Mittelwert bilden zu können. Zusätzlich wurden über einen weiteren Algorithmus die schlechtesten drei Messwerte entfernt. Untersuchungen am endgültigen Algorithmus ergaben für die beiden Koordinaten Standardabweichungen von 6,0 cm für die  $x_1$ -Koordinate und ebenfalls 6,0 cm für die  $x_3$ -Koordinate. Diese Abweichungen sind hauptsächlich auf Fehler  $\Delta x_{M,i}$  bei der Messaufnahme (dieser liegt bei etwa 2cm) und Fehler  $\Delta x_{A,i}$  bei der Bestimmung des Anfangszeitpunktes der Signalerkennung der einzelnen Sensoren (etwa 4,63cm pro Sensor) zurückzuführen. Teilweise weichen die ermittelten Werte für die Koordinaten aber auch sehr von den Sollwerten ab. Dies hängt hauptsächlich damit zusammen, dass der entworfene Algorithmus nur eine begrenzte Anzahl von auftretenden Fällen bei der Anfangszeitpunktermittlung unterscheidet. Auswege zeigt der folgende Ausblick.

---

In erster Linie soll das hier erstellte Modell dem Laien die Arbeit des Lehrstuhls auf ANTARES aufzeigen. Dies wird z.B. an der „Langen Nacht der Wissenschaft“ so sein. Prinzipiell ist geplant das Modell als Ausstellungsobjekt zu benutzen, dass Schülern z.B. am Tag der offenen Tür und bei Besuchen durch Schulklassen vorgeführt werden soll.

Bei dem entwickelten Modell handelt es sich nur um eine sehr einfache Nachbildung des realen Problems. Von da her gäbe es deshalb noch eine Fülle an Möglichkeiten das Modell zu verbessern. Da es jedoch gar nicht sinnvoll wäre ein Modell zu entwickeln, das exakt dem realen Problem entspricht, sollen hier nur noch einige Aspekte aufgezeigt werden, die relativ leicht realisierbar wären und die Einfachheit eines solchen Modells trotzdem bewahren.

1) Im Kapitel 4.4.3.1 wurde aufgezeigt, wie mittels einer Fallunterscheidung untersucht wird, ob im Falle starken Klopfens wahrgenommene Extrema tatsächlich vom Klopfpunkt herrühren, oder ob diese von den anderen Sensoren stammen. Diese Fallunterscheidung sprengt allerdings bereits den Rahmen dieser Arbeit. Daher wurde es bei dieser belassen. Im Falle des schwachen Klopfens entscheidet sich das Programm für die „ersten Knicks“. Aber auch in diesem Fall ist es ab und an der Fall, dass diese Knicke von einem Fehlsignal herrühren. Es wäre daher sinnvoll, auch in diesem Fall eine solche Fallunterscheidung zu erstellen. Der Erfolg im anderen Fall gäbe hierzu Anlass.

3) Die Nebensignaleffekte treten manchmal mehrmals in Folge auf (vor allem wenn sehr stark geklopft wird) der entworfenen Algorithmus untersucht aber nur einmal die Sensordaten auf solche Nebensignaleffekte. Denkbar wäre hier ein mehrmaliges Vergleichen der Werte.

2) Eine weitere Möglichkeit das Modell abzuändern, wäre, dass man nicht nur an der Frontfläche des Aquariums klopfen kann, sondern auch an den Seiten bzw. am Boden. In diesem Fall müsste man allerdings auch einige Sensoren mehr benutzen. Ein Klopfpunkt wäre nun nicht mehr aus nur drei Sensoren berechenbar, sondern es müsste ein vierter Sensor herangezogen werden. Prinzipiell liefere die Berechnung aber genauso ab wie bisher.

3) Denkbar wäre auch, sich die Frequenzen des Klopfsignals genauer anzusehen und zu untersuchen, ob diese sich für verschiedene Klopfkörper unterscheiden. So wäre es möglich nicht nur zu bestimmen, woher das Signal kommt, sondern auch welcher Gegenstand das Signal erzeugt. Im realen Modell stellt dies nämlich tatsächlich einen Teil der Aufgabe dar. Bei den akustischen Messungen in ANTARES erhält man neben den zu messenden Signalen auch Störsignale von z.B. Delphinen, die als solche erkannt werden müssen.

4) Bisher errechnet das Programm immer einen Klopfpunkt, außer wenn die Anzahl der Nullpaare zu groß ist. Dann wird die Messung als unbrauchbar erkannt. Eine weitere Möglichkeit eine schlechte Messung zu erkennen, wäre das Berechnen der Standardabweichung des Mittelwertes. Ist diese zu groß, wird die Messung auch als unbrauchbar verworfen.

---

## Literaturverzeichnis

- [1] S. Kuch „Aufbau eines Hydrophon-Teststands und vorbereitende Studien zur akustischen Detektion ultrahochenergetischer Neutrinos“, Diplomarbeit, Erlangen 2003
- [2] Hering, Martin, Stohrer, „Taschenbuch der Mathematik und Physik“, Springer-Verlag 2009
- [3] Lerch, Sessler, Wolf, „Technische Akustik – Grundlagen und Anwendungen“, Springer-Verlag 2009
- [4] [http://www.uni-saarland.de/fak7/hartmann/cfn/Dokumente/Manuals/Piezoelektrischer\\_Effekt.pdf](http://www.uni-saarland.de/fak7/hartmann/cfn/Dokumente/Manuals/Piezoelektrischer_Effekt.pdf)
- [5] [http://antares.in2p3.fr/Gallery/3D/antares\\_pub.jpeg](http://antares.in2p3.fr/Gallery/3D/antares_pub.jpeg)
- [6] [http://www.innovations-report.de/html/berichte/physik\\_astronomie/bericht-20845.html](http://www.innovations-report.de/html/berichte/physik_astronomie/bericht-20845.html)
- [7] <http://www.ni.com/labview>

### ERKLÄRUNG NACH § 30 ABS. 6 LPO I

Ich versichere, dass ich die schriftliche Hausarbeit selbstständig verfasst und keine anderen Hilfsmittel als die angegebenen benutzt habe.

Die Stellen der Arbeit, die anderen Werken dem Wortlaut oder dem Sinn nach entnommen sind, wurden von mir unter Angabe der Quelle als Entlehnung kenntlich gemacht. Dies gilt ebenso für Zeichnungen, Kartenskizzen und bildliche Darstellungen.

---

# Anhang

## 7 Aufbau des Programms und Implementierung der Algorithmen

In diesem Kapitel sollen die einzelnen Unterprogramme für die Berechnung und die verwendeten Algorithmen etwas näher erläutert werden. Abbildung 7-1 stellt ein Schema des gesamten Programms und dessen wichtigste Bausteine dar. Das Berechnungsprogramm lässt sich grob in drei Teile aufgliedern wie die Abbildung zeigt. Eine genauere Aufspaltung der einzelnen Teile findet sich jeweils in den folgenden Kapiteln in denen diese Teile genauer beschrieben werden.

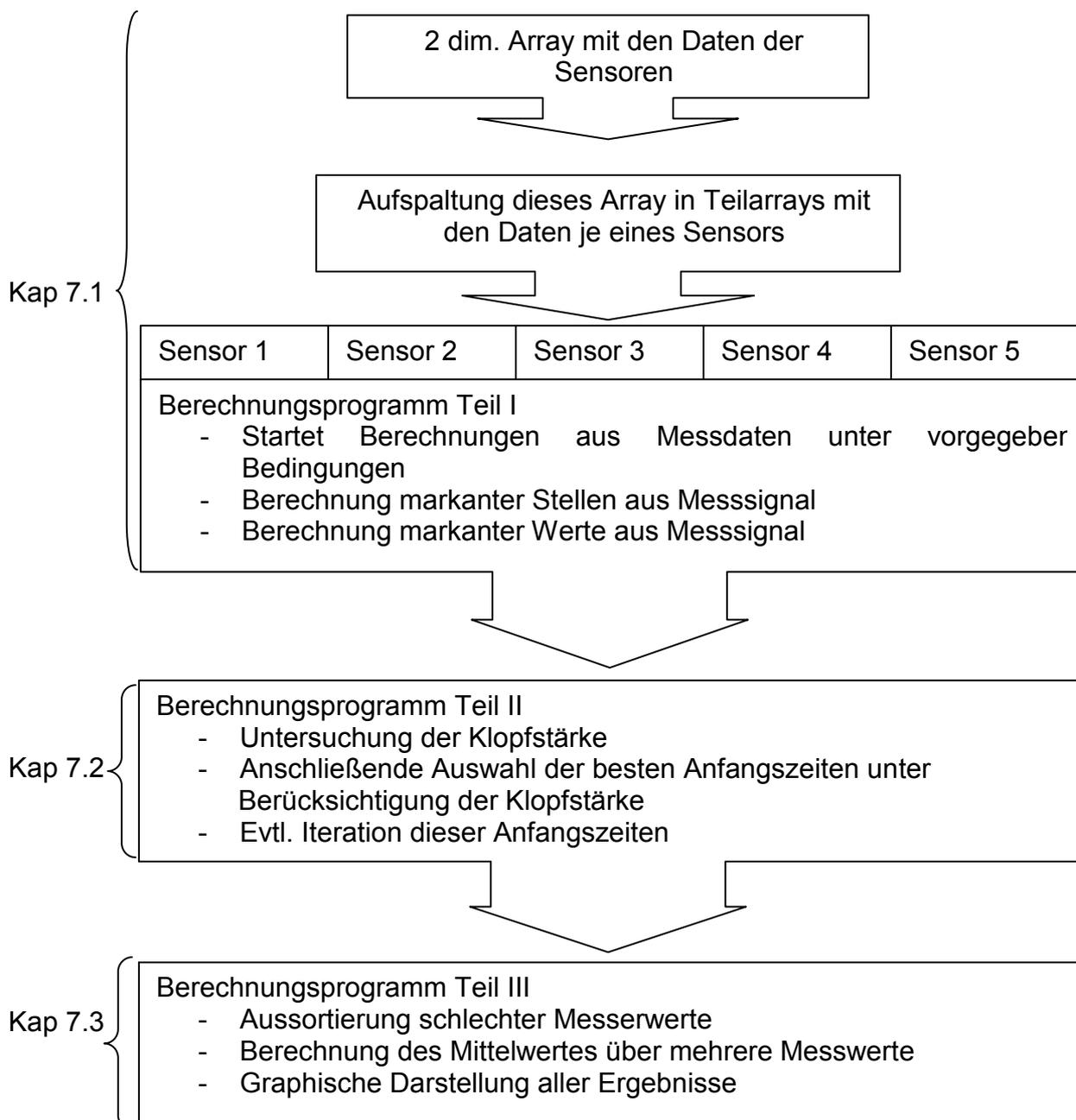


Abbildung 7-1: Schematische Darstellung der einzelnen Bausteine des Programms

## 7.1 Hauptprogramm „Berechnungsprogramm.vi“ (Teil I)

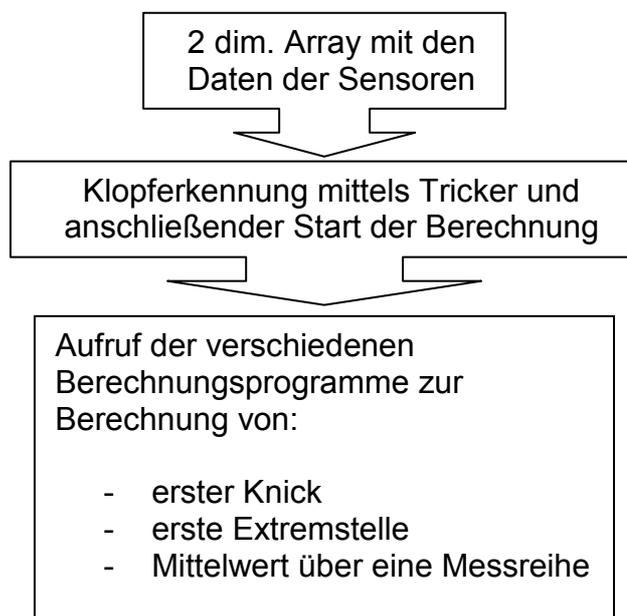


Abbildung 7-2: Schematische Darstellung des ersten Teils des Programms

Zur Messaufnahme sollte eines der Programme herangezogen werden, die in der Palette von Labview bereitstehen. Dieses war dann auf das vorliegende Problem abzustimmen. Am besten schien hierzu das Programm „Cont Acq&Graph(buffered, Channel Gain Queue) Channels.vi“ geeignet. Dieses Programm nimmt kontinuierlich eine vorgegebene Anzahl an Messwerten von der DAQ-Karte auf und verarbeitet diese in einer while-Schleife, welche solange läuft bis der Stop-button betätigt wird. Im Frontpanel des Programms kann eingestellt werden, welche Kanäle der Karte eingelesen werden sollen. Außerdem kann man einstellen, wie viele Werte pro Sekunde abgetastet werden (scan-rate) und wie viele von diesen Werten bei einem Schleifendurchgang dargestellt werden sollen. Die Messwerte werden in einem zweidimensionalen Array an die Schleife übergeben. Die einzelnen Spalten dieses Arrays sind die jeweiligen Kanäle. Dieses Programm wurde umbenannt in „Berechnungsprogramm.vi“ und stellt den Kern des gesamten Programms dar. In „Berechnungsprogramm.vi“ wurde ein Programmteil entworfen, der parallel für jeden Sensor getrennt durchlaufen wird. Am Eingang eines jeden solchen Programmteils steht die Trigger-Startwert Kombination, wie in Kapitel 4.4.1 beschrieben. Der Trigger reagiert auf das Überschreiten eines eingestellten Schwellwertes. Der Reaktionszeitpunkt wird von dem tool „statistics, dass hinter dem Trigger angebracht ist als Startzeitpunkt  $T_S$  zurückgegeben. Dieser Startzeitpunkt wird über einen Schieberegister solange gespeichert, bis ein neuer Startzeitpunkt folgt, der die folgenden Bedingungen erfüllt:

$$T_S \neq T_G \quad \wedge \quad T_S \neq 0 \quad \text{Bedingung 6}$$

Dabei ist  $T_G$  der zuletzt gespeicherte Wert für den Startzeitpunkt. Das Abspeichern dieses Wertes ist notwendig, da sonst am Ende des Schleifendurchlaufs der Wert auf

---

„0“ zurückginge und das Programm damit neue Koordinaten berechnen würde. Es sollen aber die berechneten Werte solange dargestellt werden, bis erneut geklopft wird. Die zweite Bedingung ist nötig, damit zum einen die Berechnungen, die ebenfalls unter der Bedingung 6 gestartet wird, erst nach dem ersten Klopfen beginnt. Zum anderen geht der Wert  $T_S$  nach dem Beenden des Triggers auf den Wert auf „0“ zurück, womit der erste Teil der Bedingung 4 erfüllt wäre und ungewollt eine Berechnung eingeleitet würde. Diese Trigger-Startwert-Kombination wird als Auslöser für den gesamten Berechnungsteil herangezogen. Da es einen enormen Rechenaufwand bedeuten würde, wenn das Programm bei jedem Schleifendurchlauf alle berechnenden Unterprogramme aufruft, wurde die Bedingung 4 als Auslöser für das Betreten einer Case-Struktur verwendet, welche die Unterprogramme aufruft, die diese Berechnungen durchführen.

Dabei handelt es sich um die folgenden Unterprogramme:

- Knickfinder.vi (Implementation von Kapitel 4.4.2)
- Extremafinder.vi (Implementation von Kapitel 4.4.3)
- Mittelwert.vi

Um nun für jeden Sensor eine eigene Rechnung durchführen zu können, werden dem eingangs erwähnten Array in der Schleife die einzelnen Spalten (Kanäle) entnommen. Hierzu wurde die Funktion „delete from array“ verwendet, mit der es möglich ist aus einem Array Elemente oder Teilarrays zu entnehmen. Da im vorliegenden Programm pro Schleifendurchlauf 4000 Messwerte übergeben werden, hat das Messarray eine Länge von 4000. Um diese nun für jeden Sensor genauer untersuchen zu können, müssen, wenn geklopft wird die Messdaten separiert werden. Dabei sollten der Schleifendurchlauf in dem geklopft wurde zusammen mit dem davor und dem dahinter liegenden erfasst werden um auch dann gut auswerten zu können, wenn das Klopfen am Anfang bzw. zum Ende der Messreihe eingeleitet wird. Hierfür wurde eigens das Unterprogramm „letzte 3 Runden in Array.vi“ programmiert. In diesem Unterprogramm wird ein Array, das eine Länge von 12000 ( $3 \cdot 4000$ ) hatte mit Daten befüllt. Es werden bei jedem Schleifendurchlauf die 4000 Messwerte beginnend beim Index 8000 hineingeschrieben und anschließend die Inhalte dieses Array um 4000 Plätze nach links verschoben (als Vorbereitung für das nächste Hineinschreiben der Werte), was mit dem tool „rotate Array“ möglich ist. Wird nun geklopft, so ist die Bedingung für die eingangs erwähnte Case-Struktur erfüllt. Das Array, das nun die letzten 4000, die aktuellen 4000 und die nächsten 4000 Werte beinhaltet wird dann zur genaueren Analyse an die berechnenden Unterprogramme übergeben.

### 7.1.1 Unterprogramm „letzte 3 Runden in Array.vi“

Diesem Unterprogramm werden die folgenden **Eingangsgrößen** übergeben:

- „Spaltennummer aus Messarray“  $G$  für das Array  $A_G$
- „Messarray“  $A_M$  (mit allen Kanälen)
- „Länge des Messarrays“  $I$  (buffer size = 4000)
- „initialisiertes Array“ (ein anfangs mit Nullen gefülltes Array  $A_I$  der Länge  $4I$  (also hier 16000))

Aus dem Array  $A_I$  wird über das tool „delete from array“ beginnend beim Index 0 ein Teilarray der Länge  $3I$  (also 12000) herausgeschnitten. Verwendet wird nun dieses herausgeschnittene (gelöschte) Array  $A_B$ . Dieses wird an ein tool übergeben, das „Rotate 1D array“ heißt. Mittels diesem wird das Array um die besagten -4000 Stellen rotiert und anschließend über das tool „Replace array subset“ beginnend beim Index 8000 die Spalte  $A_K$  aus dem Messarray, die den gewünschten Kanal darstellt geschrieben. Hierzu gibt es ein tool „Index array“, das aus einem Array Elemente oder Teilarrays (hier eine Spalte) zurückgibt. Neben diesem wird außerdem ein Array  $A_{MW}$  zur Mittelwertsberechnung aus dem Array  $A_I$  herausgenommen, welches beim Index 0 beginnt, und somit die 4000 Werte des letzten Schleifendurchlaufs beinhaltet.

Die **Ausgangsgrößen** sind nun:

- „Maximum“ (aus dem Messarray  $A_M$ )
- „Ergebnisarray“  $A_E$
- „Array zur Mittelwertsberechnung“ (Mittelwertsarray  $A_{MW}$ )

### 7.1.2 Unterprogramm „Mittelwert.vi“

Das Array  $A_{MW}$  ist die einzige Eingangsgröße des Unterprogramms „Mittelwert.vi“. Im Unterprogramm „Mittelwert.vi“ wird der Mittelwert aus den 4000 Messwerten des unmittelbar davor liegenden Schleifendurchlaufs berechnet. Hierzu dient eine while – Schleife, die von  $i = 0$  bis 4000 (array size) läuft, und dabei über „index array“ jeweils das Element an der Stelle  $i$  zu einem Shiftregister (mit Anfangswert 0) addiert. Am Ende wird diese Summe durch die „array size“ geteilt. Das Unterprogramm gibt den Mittelwert  $M$  über 4000 Werte zurück. Dieser Mittelwert wird an mehreren Stellen des Programms benötigt.

### 7.1.3 Unterprogramm „Knickfinder.vi“

Das Unterprogramm Knickfinder.vi sucht nach den ersten Knicks wie im Kapitel 4.4.2 beschrieben. Zu diesem Zweck wurde hier ein Array der Länge 100 (dieses soll im folgenden  $A_{100}$  genannt werden) definiert, in welches die Zeitpunkte (beginnend beim Zeitpunkt „ $T_G - 50$ “) geschrieben werden.

Dem Unterprogramm „Knickfinder.vi“ werden die folgenden **Eingangsgrößen** übergeben:

- „gespeicherter Klopzeitpunkt“ - Startzeitpunkt  $T_S$  (siehe Kapitel 4.4.1)
- „Messarraylänge“  $l$  (also 4000)
- „Datenarray“  $A_E$  (Ergebnisarray)
- „Mittelwert“  $M$
- „Steigungsschwelle“ (Schwellwert)  $S$  (siehe Kapitel 4.4.2) dieser wird im Hauptprogramm als Konstante eingegeben

Das Unterprogramm besteht im wesentlichen aus einer while-Schleife. Diese durchläuft unter Verwendung der Gleichungen 4, 5 und 6 alle Werte des Array  $A_{100}$  und endet, sobald  $\Delta m \geq S$  (siehe Gleichung 17) gilt, spätestens aber, wenn der Index  $i$  den Wert „Größe(Array  $A_{100}$ )-3“ (also 97) annimmt. So erhält man zu jedem Sensor einen Anfangszeitpunkt  $T_A$ . Um zu erreichen, dass möglichst selten ein Knick vor dem eigentlichen Klopfsignal genommen wird muss zusätzlich eine der folgenden Bedingungen zum Anhalten dieser Schleife erfüllt sein:

$$|A_{100}(i-1) - M| > 0,1 \quad |A_{100}(i) - M| > 0,1 \quad |A_{100}(i+1) - M| > 0,1 \quad \text{Bedingung 7}$$

Einziges **Ausgangsgröße** dieses Unterprogramms ist nun:

- „Erste Knick“  $Kn(i)$  des Sensors  $S_i$

#### 7.1.4 Unterprogramm „Extremfinder.vi“

Dieses Unterprogramm unterscheidet sich nur wenig vom Unterprogramm „Knickfinder.vi“. Auch hier sind die **Eingangsgrößen**:

- „gespeicherter Klopzeitpunkt“  $T_S$
- „Messarraylänge“  $l$
- „Datenarray“  $A_E$
- „Mittelwert“  $M$
- „Steigungsschwelle“  $S$

Das Unterprogramm sucht nach der ersten Extremstelle des Arrays  $A_{100}$ . Auch dies geschieht mittels einer while-Schleife. Diese endet auch, wenn der Laufindex  $i$  den Wert 97 (siehe Kapitel 7.1.3) annimmt oder wenn die Bedingung 1 aus Kapitel 4.4.3 erfüllt ist

$$|A(k-1)| < |A(k)| > |A(k+1)| \quad \wedge \quad |A(k) - A_m| = A_{abs} > C \in IR \quad \text{Bedingung 8}$$

Der Wert  $C$  ist mit dem Mittelwert  $M$  gekoppelt, und es gilt:

$$C = M + 0,2 \quad \text{Gleichung 18}$$

Die **Ausgangsgröße** dieses Unterprogramms ist:

- „erste Extremstelle“  $E(i)$  des Sensors  $S_i$

## 7.2 Hauptprogramm „Berechnungsprogramm.vi“ (Teil II)

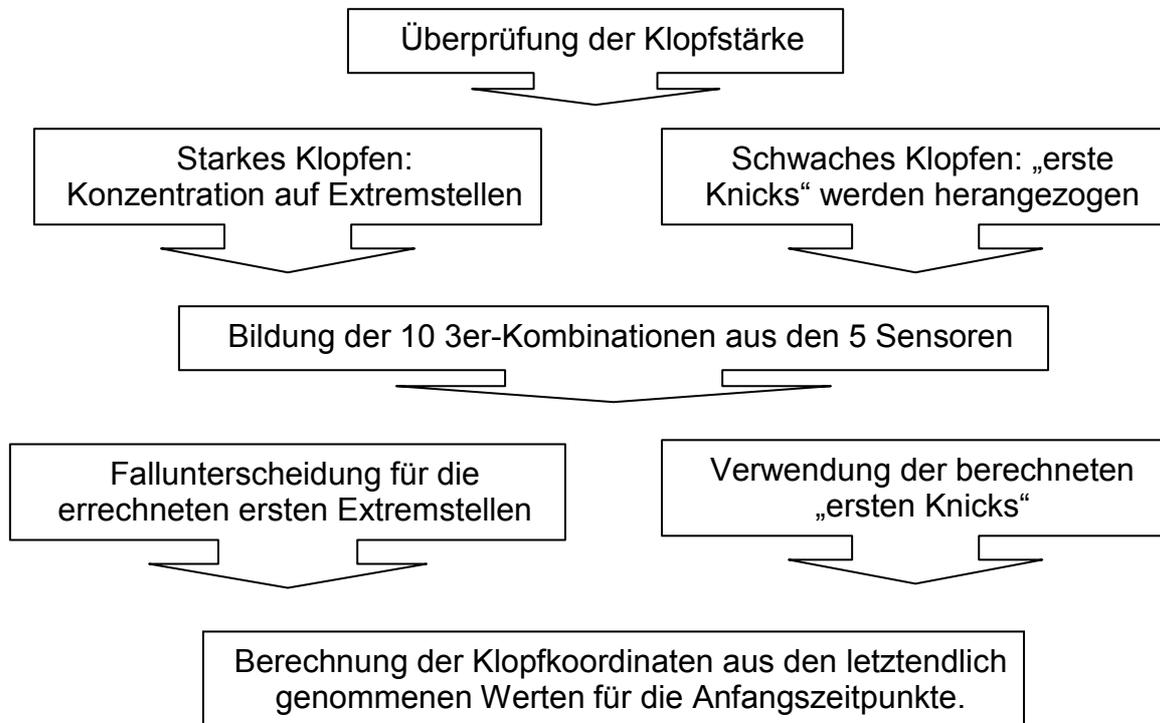


Abbildung 7-3: Schematische Darstellung des zweiten Teils des Programms

Als nächstes werden die Ergebnisarrays  $A_E$  aus Kapitel 7.1.1 der einzelnen Sensoren in einer Casestruktur auf deren jeweiliges absolutes Maximum untersucht. Hierzu dient das tool „Array Max & Min“. Die Case-Struktur wird nur durchlaufen, wenn Bedingung 6 für alle Sensoren erfüllt ist.

$$T_S \neq T_G \quad \wedge \quad T_S \neq 0 \quad \text{Bedingung 9}$$

Dann wird untersucht, ob eines der absoluten Maxima größer einem voreingestellten Wert  $C_1$  (im Programm beträgt dieser Wert 3) ist:

$$\exists i \text{ mit } \max_{abs}(A_{E,i}) > C_1 \quad (i = 1 \dots 5) \quad \text{Bedingung 10}$$

Wenn dies der Fall ist, so handelt es sich um starkes Klopfen (die Extremstellen sind die potentiellen Anfangszeitpunkte), wenn nicht, so war das Klopfen schwach, und es werden die „ersten Knicks“ als Anfangszeitpunkte genommen. Um die Klopfstärke zu untersuchen wurde die Bedingung 10 mit Hilfe einer weiteren Case-Struktur überprüft.

### 7.2.1 Fall A – schwaches Klopfen

Wenn nun Bedingung 10 nicht erfüllt ist, so werden die im Unterprogramm „Knickfinder.vi“ berechneten Stellen „erster Knick“ als „richtiger“ Startzeitpunkte  $T_S$  der jeweiligen Sensoren genommen. Nun muss das Programm die einzelnen 3-er Konfigurationen aus den 5 Sensoren bilden um für diese jeweils mit dem Unterprogramm „Punkte suchen.vi“ den Klopfpunkt zu errechnen. Zum Bilden dieser Kombinationen wurde das Unterprogramm „10 Kombinationen der 5 Sensoren bei Knicks.vi“ geschrieben.

#### 7.2.1.1 Unterprogramm „10 Kombinationen der 5 Sensoren bei Knicks.vi“ (Teil I)

Dieses Unterprogramm besteht im Wesentlichen aus einer for-Schleife, deren Index  $i$  von 0 bis 9 läuft und dabei mittels einer case-Struktur die 10 Werte – Kombinationen der 5 Sensoren an das Unterprogramm „Hauptprogramm.vi“ übergibt. Dieses Unterprogramm stellt lediglich ein Zwischenprogramm dar, welches zur jeweiligen Kombination die korrekten Sensorkoordinaten zusammen mit den zugehörigen Messwerten an das Unterprogramm „Punkt suchen.vi“ übergibt.

Die **Eingangsgrößen** dieses Unterprogramms sind:

- Mittelwert (für alle Sensoren)
- Koordinate  $x_1$  (für alle Sensoren)
- Koordinate  $x_2$  (für alle Sensoren)
- Koordinate  $x_3$  (für alle Sensoren)
- Knickstelle (für alle Sensoren)
- Ergebnisarray (für alle Sensoren)
- Schallgeschwindigkeit

Somit müssen an dieses Unterprogramm insgesamt 30 Werte übergeben werden. Leider können aber in Labview maximal 28 Terminals verbunden werden (mit Ausgängen). Somit ist diese Anzahl an Eingangswerten zu groß. Abhilfe konnte hier geschaffen werden, indem die meisten dieser Werte in ein Array gespeichert wurden, welches dann als eine Variable (pro Sensor) übergeben wurde. Damit reduzierten sich die **Eingangsgrößen** auf:

- „Wertearray“  $A_W$  (für alle Sensoren)
- „Ergebnisarray“  $A_E$  (für alle Sensoren)
- „Schallgeschwindigkeit“  $c_S$

Das Einschreiben der genannten Werte geschieht mittels eines Unterprogramms „Werte in Array schreiben.vi“ welches bereits in „Berechnungsprogramm.vi“ aufgerufen wird.

#### 7.2.1.2 Unterprogramm „Werte in Array schreiben.vi“

In diesem Unterprogramm werden in ein mit Nullen vorinitialisiertes Array der Länge 8 die folgenden Werte geschrieben:

- 
- erste Extremstelle (wird benötigt im Fall B)
  - erster Extremwert (wird benötigt im Fall B)
  - Gespeicherte Klopfstelle (wird benötigt im Fall B)
  - Mittelwert (wird sowohl im Fall A als auch B benötigt)
  - Koordinate x<sub>1</sub> (wird sowohl im Fall A als auch B benötigt)
  - Koordinate x<sub>2</sub> (wird sowohl im Fall A als auch B benötigt)
  - Koordinate x<sub>3</sub> (wird sowohl im Fall A als auch B benötigt)
  - Knickstelle (wird nur in diesem Fall A benötigt)

Diese Werte stellen also die **Eingangsgrößen** dieses Unterprogramms (jeweils für einen Sensor) dar. Die **Ausgangsgröße** ist:

- „Wertearray“  $A_W$

#### 7.2.1.2.1 Unterprogramm „Werte aus Array auslesen.vi“

Um nun diese Werte im Unterprogramm „10 Kombinationen der 5 Sensoren bei Knicks.vi“ benutzen zu können, wurde noch das Unterprogramm „Werte aus Array auslesen.vi“ benötigt, das diese Werte wieder benutzbar macht. Dieses Unterprogramm erhält als einzige **Eingangsgröße** den Wert:

- „Wertearray“  $A_W$

Die **Ausgangsgrößen** sind entsprechend:

- erste Extremstelle
- Extremwert
- Gespeicherte Klopfstelle
- Mittelwert
- Koordinate x<sub>1</sub>
- Koordinate x<sub>2</sub>
- Koordinate x<sub>3</sub>
- Knickstelle

Dabei wird das Wertearray  $A_W$  jeweils an das tool „Index Array“ übergeben. Dieses gibt als Ausgangsgröße jeweils den Wert an der gewünschten Stelle (1 ... 8) zurück.

#### 7.2.1.3 Unterprogramm „10 Kombinationen der 5 Sensoren bei Knicks.vi“ (Teil II)

Alle genannten Eingangsgrößen werden in diesem Unterprogramm an eine Case-Struktur übergeben. Diese besteht aus insgesamt 10 cases (den 10 Kombinationen der 5 Sensoren). In jedem einzelnen Fall werden nun die Werte von je drei verschiedenen Sensoren an das **Unterprogramm „Hauptprogramm.vi“** übergeben. Dieses gibt seinerseits die folgenden Werte an das Unterprogramm „Punkt suchen.vi“ weiter

- Koordinaten des ersten genommenen Sensors
- Erste Knickstelle dieses ersten genommenen Sensors
- Koordinaten des zweiten genommenen Sensors
- Erste Knickstelle dieses zweiten genommenen Sensors

- Koordinaten des dritten genommenen Sensors
- Erste Knickstelle dieses dritten genommenen Sensors
- Schallgeschwindigkeit

Und bekommt von diesem die folgenden Werte zurück:

- berechnete  $x_1$ -Koordinate der Klopfstelle
- berechnete  $x_2$ -Koordinate der Klopfstelle

Diese berechneten Koordinaten werden nun vom Unterprogramm „Hauptprogramm.vi“ direkt an das Unterprogramm „10 Kombinationen der 5 Sensoren bei Knicks.vi“ zurückgegeben. Dort werden diese getrennt in die Arrays „Liste der  $x_1$ -Koordinaten“ und „Liste der  $x_2$ -Koordinaten“ geschrieben. Hierzu werden mittels zweier shift-register zwei mit Nullen vorinitialisierte Arrays der Länge 10 definiert, welche an das tool „replace array subset“ übergeben werden. Dieses ersetzt den Wert an der Stelle Index, den man vorgeben kann. Der Eingang „Index“ dieses tools ist hierbei mit dem Index der for-Schleife (siehe Kapitel 7.2.1.1) gekoppelt, der die Werte 0 bis 9 durchläuft. Somit wird bei jedem Schleifendurchgang eine andere Sensorkonfiguration zur Berechnung herangezogen und die berechneten Werte in diese Arrays geschrieben.

Diese beiden Arrays stellen die **Ausgangsgrößen** des Unterprogramms „10 Kombinationen der 5 Sensoren bei Knicks.vi“ dar:

- Liste der  $x_1$  – Koordinaten
- Liste der  $x_2$  – Koordinaten

## **7.2.2 Fall B – starkes Klopfen**

Wenn die Bedingung 10 erfüllt ist, so werden die im Unterprogramm „Extremafinder.vi“ berechneten Stellen „erste Extremstelle“ als „richtiger“ Startzeitpunkte  $T_S$  der jeweiligen Sensoren genommen. Auch hier muss das Programm die einzelnen 3-er Konfigurationen aus den 5 Sensoren bilden um für diese jeweils mit dem Unterprogramm „Punkte suchen.vi“ den Klopfpunkt zu errechnen. Zum Bilden dieser Kombinationen wurde das Unterprogramm „10 Kombinationen der 5 Sensoren bei Extrema.vi“ geschrieben.

### **7.2.2.1 Unterprogramm „10 Kombinationen der 5 Sensoren bei Extrema.vi“ (Teil I)**

Der Aufbau dieses Unterprogramms ist fast derselbe wie im eben beschriebenen Unterprogramm „10 Kombinationen der 5 Sensoren bei Knicks.vi“. Der einzige Unterschied ist, dass die Werte von der case-Struktur zunächst an das Unterprogramm „Fallunterscheidung.vi“ übergeben werden und erst von dort zum Unterprogramm „Hauptprogramm.vi“ übergeben werden. Dieses Unterprogramm „Fallunterscheidung.vi“ wird im folgenden Kapitel näher erläutert.

### 7.2.2.2 Unterprogramm „Fallunterscheidung.vi“

Diesem Unterprogramm werden vom Unterprogramm „10 Kombinationen der 5 Sensoren bei Extrema.vi“ die folgenden Werte der jeweils genommenen Sensoren übergeben:

- „erste Extremstelle“
- „Erster Extremwert“
- „gespeicherte Klopfstelle“
- Messarray
- Mittelwert

Zurückgegeben wird von diesem Unterprogramm jeweils:

- „endgültig (genommene) Stelle“

Wie unter Kapitel 4.4.3.1 beschrieben beeinflussen sich die Sensoren zum Teil gegenseitig. D.h. dass sobald ein Sensor ein Signal bekommt, auch andere Sensoren reagieren. Allerdings sind die dabei auftretenden „falschen“ Peaks relativ klein verglichen mit den „richtigen“ ersten Peaks. Dies wird im Unterprogramm „Fallunterscheidung.vi“ ausgenutzt, um solche „Falschen“ Peaks auszufiltern. Dazu muss das Programm zunächst erkennen, dass es sich um einen potentiellen Kandidaten für einen „falschen“ Peak handelt. Denn dies sollte nur der Fall sein, wenn erste Peaks verschiedener Sensoren relativ nahe beieinander liegen. Als relativ nahe wurde hier ein Abstand von 2 Skt definiert. D.h. also, dass das Programm erst Überprüfungen startet, wenn der Abstand zwischen zwei ersten Peaks kleiner als zwei ist.

Dazu werden die folgenden Bedingungen 4 aus Kapitel 4.4.3.1 für die Extremstellen 1, 2 und 3 untersucht:

$$|T_{G1} - T_{G2}| \leq 2 \quad \text{Bedingung 11}$$

$$|T_{G3} - T_{G2}| \leq 2 \quad \text{Bedingung 12}$$

$$|T_{G1} - T_{G3}| \leq 2 \quad \text{Bedingung 13}$$

Je nach dem ob eine dieser Bedingungen erfüllt ist, wird dem Ereignis eine Zahl zugeordnet:

	Erfüllt	Nicht erfüllt
Bedingung 11	1	0
Bedingung 12	2	0
Bedingung 13	4	0

Diese Zahlen werden anschließend addiert, also:

$$\text{Summe} = \text{Zahl}(\text{Bed}10) + \text{Zahl}(\text{Bed}11) + \text{Zahl}(\text{Bed}12) \quad \text{Bedingung 14}$$

Die dabei möglichen Ergebnisse sind die Nummern der nun zu unterscheidenden Fälle:

- 1.Fall: Summe = 1
- 2.Fall: Summe = 2
- 3.Fall: Summe = 4
4. Fall: Summe = 3 ∨ 5 ∨ 6 ∨ 7
5. Fall: Summe = 0 ∨ sonst (default)

Zu Beginn berechnet das Unterprogramm die absoluten Peakhöhen, indem es den Betrag der Differenz aus dem jeweiligen Extremwert und dem Mittelwert des jeweiligen Sensors berechnet, also

$$D(i) = |E(i) - M(i)| \quad \text{Gleichung 19}$$

#### 1.Fall:

Hier überprüft das Programm nun zunächst welcher der beiden Extremwerte der größere ist, also ob  $D(1) > D(2)$  und geht je nach dem in einen der zwei cases. Anschließend wird der kleinere Wert durch den größeren Wert dividiert und so überprüft, ob der kleinere Peak wenigstens 30% des größeren Peaks ausmacht. Ist dies erfüllt, so wird davon ausgegangen, dass der Klopfpunkt tatsächlich etwa in der Mitte dieser beiden Sensoren liegt und die Extremstellen deshalb so nahe beieinander liegen. Ist jedoch der kleinere Peak kleiner als 30% vom größeren Peak, so sucht das Programm nach dem nächsten in Frage kommenden Extremum. Hierzu wurde das Unterprogramm „nächstes Extremum Finder.vi“ geschrieben.

#### **7.2.2.2.1 Unterprogramm „nächstes Extremum Finder.vi“**

Dieses Unterprogramm arbeitet im wesentlichen genauso wie im Kapitel 7.1.4 beschriebene Unterprogramm „Extremumfinder.vi“. Es hat dieselben Eingangsgrößen und es untersucht dieselben Bedingungen. Allerdings wird eine weitere Bedingung gefordert, nämlich dass das zu nehmende Extremum mind. um 30% größer sein muss als das zuletzt genommene, also:

$$A(k) \geq 1,3 \cdot E_{\text{vorher}} \quad \text{Bedingung 15}$$

Mit diesem Algorithmus findet dieses Unterprogramm einen späteren Wert, der an das Unterprogramm „10 Kombinationen der 5 Sensoren bei Extrema.vi“ zurückgegeben wird.

#### 2. Fall & 3. Fall

Diese beiden Fälle werden genauso behandelt, wie der erste.

#### 4. Fall:

Auch im 4. Fall untersucht das Programm zunächst, welcher der drei Extremwerte der größte ist. Hierzu werden die drei Werte an das Unterprogramm „größter von 3 Werten.vi“ übergeben, das den größten Wert zurückgibt. Dieses Unterprogramm arbeitet dabei ganz trivial mit zwei Case-Strukturen. Anschließend werden wieder die beiden kleineren Werte durch diesen größten dividiert und überprüft, ob diese wenigstens 30% des größten Wertes ausmachen. Wenn dies für einen oder sogar beide Werte nicht erfüllt ist, so sucht das Programm wieder über das Unterprogramm „Nächstes Extremum Finder.vi“ wie oben beschrieben nach dem nächsten in Frage kommenden Extremwert und gibt diesen zurück.

### **7.2.3 Unterprogramm „Punkt suchen.vi“**

Eines der wichtigsten Unterprogramme ist das bereits mehrfach erwähnte Unterprogramm „Punkt suchen.vi“. In diesem Unterprogramm wird nach dem im Kapitel 4.2 beschriebenen Vorgehen der „beste“ Klopfpunkt durch systematisches Suchen ermittelt.

Dem Unterprogramm werden die folgenden Eingangsgrößen übergeben:

- Koordinaten der drei betrachteten Sensoren
- Schallgeschwindigkeit
- Laufzeitdifferenz  $L_1$  des ersten Sensors (genommener Wert 1)
- Laufzeitdifferenz  $L_2$  des zweiten Sensors (genommener Wert 2)
- Laufzeitdifferenz  $L_3$  des dritten Sensors (genommener Wert 3)

Gleich zu Beginn wird aus der Schallgeschwindigkeit und der Laufzeitdifferenz eine Laufwegdifferenz berechnet.

Das Unterprogramm besteht im Wesentlichen aus zwei ineinander geschachtelten for-Schleifen. Die äußere ist für die Höhe der Klopffläche, die andere für die Breite zuständig. Die äußere Schleife beginnt bei der Höhe 0, in der die innere Schleife von 0 bis 60 durchlaufen wird (die Schrittweite im Programm beträgt 1), dann geht die äußere Schleife auf die Höhe 1, in der die innere Schleife von 0 bis 60 durchlaufen wird, usw. bis die Höhe 30 erreicht ist. Dann ist der Suchvorgang beendet.

In jedem so erreichten Punkt berechnet das Programm die Abstände zwischen dem Punkt und den drei Sensoren. Anschließend wird jeweils die Differenz aus diesen Abständen und den zugehörigen gemessenen Laufwegdifferenzen berechnet. Diese Differenzen werden addiert und die Summe in einem Shiftregister gespeichert, wenn der Wert kleiner ist als der zuletzt gespeicherte. Zusätzlich werden in diesem Fall ebenfalls über Shiftregister die zugehörigen Koordinaten gespeichert. Sind die beiden Schleifen nun fertig durchlaufen, so werden die zuletzt in den Shiftregistern gespeicherten Koordinaten zurückgegeben.

### 7.3 Hauptprogramm „Berechnungsprogramm.vi“ (Teil III)

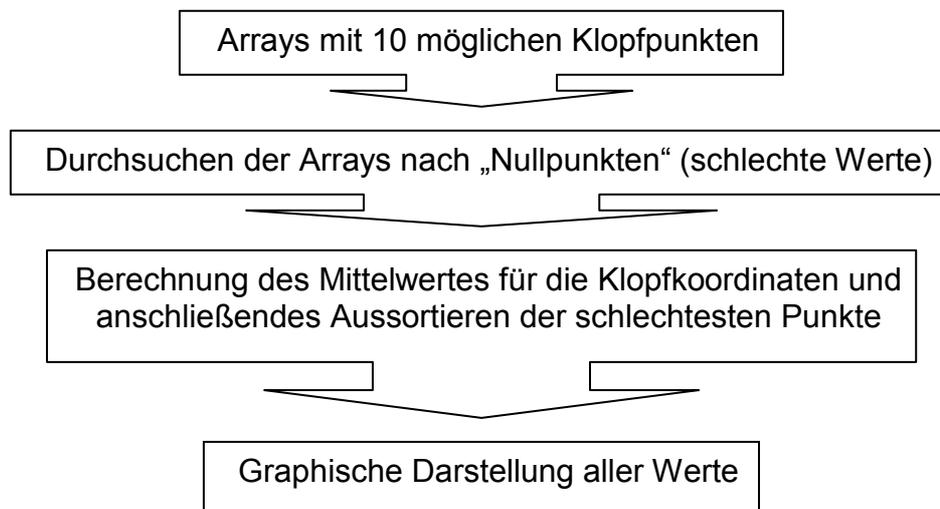


Abbildung 7-4: Schematische Darstellung des dritten Teils des Programms

Der letzte große Bereich des Berechnungsprogramms ist zum einen für die Berechnung und anschließende iterative Verbesserung des Mittelwertes für die Koordinaten des Klopfpunktes zuständig. Zum anderen werden alle gemessenen Koordinaten zusammen mit dem ursprünglichen Mittelwert und dem letztlich „besten“ Mittelwert graphisch dargestellt.

#### 7.3.1 Unterprogramm „Nullpaarestreicher.vi“

Nachdem in den unter Kapitel 7.2.1.3 bis 7.2.2.1 beschriebenen Unterprogrammen insgesamt 10 Wertepaare für Klopfpunkte aus den 10 Konfigurationen der 5 Sensoren berechnet worden sind, werden als **Eingangsgrößen** die beiden Arrays (siehe Kapitel 7.2.1.3)

- „Liste der  $x_1$  – Koordinaten“
- „Liste der  $x_2$  – Koordinaten“

an das Unterprogramm „Nullpaarestreicher.vi“. In diesem Unterprogramm werden, wie der Name schon sagt, die Nullpaare<sup>2</sup> gestrichen. Dazu werden die beiden Arrays zunächst an zwei Shiftregister einer for-Schleife übergeben. Diese for-Schleife läuft 10 mal (Länge der Koordinatenarrays). Darin werden die beiden Arrays zunächst addiert. So finden sich die Nullpaare relativ leicht, denn für diese ergibt die Summe jeweils den Wert Null. Dieses Summenarray wird dann an das tool „Search 1D Array“ übergeben. Bei diesem tool kann man nach einem Bestimmten Wert (hier 0) suchen und angeben wo im Array begonnen werden soll (hier beim Index 0). Als Ausgabe bekommt man den Index der Stelle, an der das gesuchte Element steht. Parallel dazu werden die beiden arrays jeweils zusammen mit diesem Index an das tool „delete from array“ übergeben, das die gefundene Stelle (mit der 0) aus den beiden arrays streicht. Die Schleife läuft insgesamt

<sup>2</sup> Nullpaare werden immer als Messfehler interpretiert

10 mal und löscht jeweils beim Auffinden eines Nullpaars die entsprechende Stelle aus den beiden Koordinatenarrays. Nach dem letzten Schleifendurchlauf befinden sich in den beiden Shiftregistern die beiden Arrays ohne die Nullpaare. Die Größe dieser Arrays wird mit dem tool „Array size“ bestimmt von der anfangs bestimmten Länge des ursprünglichen Arrays subtrahiert. Dies ergibt die Anzahl der Nullpaare.

Die **Ausgangsgrößen** dieses Unterprogramms sind somit:

- Anzahl der Nullpaare N
- „Liste der  $x_1$  – Koordinaten ohne Nullpaare“ (Array der Länge  $l = 10 - N$ )
- „Liste der  $x_2$  – Koordinaten ohne Nullpaare“ (Array der Länge  $l = 10 - N$ )

### 7.3.2 Unterprogramm „schlechte Werte Vernichter.vi“

Nun werden die Arrays (mit Länge  $l$ ) ohne Nullpaare (Nullfreie Arrays) an das Unterprogramm „schlechte Werte Vernichter.vi“ übergeben. Auch hier betreten die beiden Arrays über zwei Shiftregister eine For-Schleife. Dabei ist die Anzahl der Schleifendurchläufe frei wählbar (die Grundeinstellung ist 3, d.h. es werden die drei größten Ausreißer aus den Arrays gestrichen). In der Schleife werden die beiden Arrays zuerst an das Unterprogramm „Koordinaten Mittelwert.vi“ übergeben, das den Mittelwert aus den Arrayeinträgen bestimmt und diesen zurückgibt. Dann werden zwei Arrays deren Länge mit den Koordinatenarraylängen übereinstimmen über das tool „Initialize Array“ mit diesen Mittelwerten gefüllt. D.h., dass in diesen Arrays an jeder Stelle der Mittelwert steht. Diese Mittelwertsarrays werden nun von den Koordinatenarrays subtrahiert und diese Differenzen quadriert. Die beiden dabei entstandenen Arrays deren Einträge also solche Differenzenquadrate sind, werden nun addiert. Zuletzt wird aus dem Ergebnisarray die Wurzel gezogen. Dieses Array soll Abstandsarray heißen, da der eben beschriebene Algorithmus der Berechnung des Abstandes eines jeden Punktes vom anfangs berechneten Mittelwert entspricht:

$$d(P_i; P_m) = \sqrt{(x_{1i} - x_{1m})^2 + (x_{2i} - x_{2m})^2}; \quad \text{mit } i = 1 \dots k; k = 10 - N \quad \text{Bedingung 16}$$

Wobei  $n$  die Anzahl der Nullpaare ist.

Dieses so erzeugte Abstandsarray wird nun an das tool „Array Max & Min“ übergeben, das das Maximum eines Arrays ausgibt. In diesem Fall also die Stelle  $m$  im Abstandsarray, an der der größte Abstand zum Mittelwert steht. Nun werden aus den beiden nullpaarfreien Arrays mit Hilfe des tools „Delete from Array“ die Werte an der Stelle  $m$  gestrichen. Damit haben die beiden Arrays nur noch eine Länge von „ $l - 1$ “. Nun beginnt der zweite Durchlauf der Schleife. Zuletzt (nach 3 Durchläufen) hat das Koordinatenarray nur noch eine Länge von „ $l - 3$ “. Das Unterprogramm „schlechte Werte Vernichter.vi“ gibt an das Berechnungsprogramm die folgenden Variablen zurück:

- beste Werte  $x_1$
- beste Werte  $x_2$
- mittlere  $x_1$  – Koordinate
- mittlere  $x_2$  – Koordinate

### 7.3.3 Unterprogramm „Graphik.vi“

Das ureigentliche Ziel des gesamten Modells, nämlich die graphische Darstellung eines mittels Algorithmen berechneten Punktes, nach dem Eingang eines Klopfsignals wird in diesem letzten Unterprogramm verwirklicht. Dabei wurde entschieden, möglichst alle Wert darzustellen (natürlich ohne Nullpaare). Also die ursprünglichen Koordinaten mit deren ersten Mittelwerten, die nach der „Optimierung“ übrig gebliebenen Koordinaten und den schließlich daraus berechneten Mittelwerten. Dazu werden an das Unterprogramm „Graphik.vi“ die folgenden Größen übergeben:

- $x_1$  – Koordinaten ohne Nullpaare (alte Koordinaten)
- $x_2$  – Koordinaten ohne Nullpaare (alte Koordinaten)
- Mittelwert der  $x_1$  – Koordinaten ohne Nullpaare
- Mittelwert der  $x_2$  – Koordinaten ohne Nullpaare
- Beste  $x_1$  – Koordinaten (neue Koordinaten)
- Beste  $x_2$  – Koordinaten (neue Koordinaten)
- Mittelwert aus den besten  $x_1$  – Koordinaten
- Mittelwert aus den besten  $x_2$  – Koordinaten

Im Unterprogramm „Graphik.vi“ werden über das tool „Build XY Graph“ all diese Werte als Punkte in einem Koordinatensystem dargestellt. Dem tool werden dazu 2-dim. arrays übergeben, deren Zeilen Kanäle (Wertekategorie) darstellen in denen die Koordinaten für die verschiedenen Punkte stehen. Solche Arrays werden im Unterprogramm zuerst definiert. Hierzu dient das tool „Initialize Array“, das zunächst mit Nullen gefüllt wird und dessen erste Dimension mit 4 (Zeilen) und dessen zweite Dimension mit der Länge der Arrays mit den alten Koordinaten vordefiniert werden. Anschließend werden diese beiden Arrays mit Hilfe des tools „Replace Array subset“ mit den Eingangsgrößen gefüllt (hierfür mussten die beiden Mittelwerte auch in ein eindimensionales Array geschrieben werden, da es sonst Probleme mit den Datentypen gab). Das tool „Build XY Graph“ ordnet dann automatisch die Werte der jeweiligen Zeile des X- Arrays den korrekten entsprechenden Werten des Y-Arrays zu.

Um nun zwischen den verschiedenen Werten besser unterscheiden zu können, wurden die Form der Punkte und deren Farben variiert. Dazu kann im Register „Properties“ jeder einzelne Plot geeignet definiert werden. Die anfänglichen Werte (ohne Nullpaare“ wurden hierbei mit Kreuzchen in helleren Farben dargestellt. Die letztlich guten Werte durch dicke dunklere Kreise.

## 8 Kurze Erläuterung zur Handhabung des Programms

In diesem Kapitel soll aufgezeigt werden, wie das Programm zu bedienen ist. D.h. welche Konstanten wo eingegeben werden können und welche Programme beisammen sein müssen.

Liste der benötigten Programmteile

Die folgenden Programmteile müssen komplett in einem Ordner beisammen sein, um ein einwandfreies Arbeiten des Programms zu gewährleisten:

- Berechnungsprogramm.vi
- Punkt suchen.vi
- Extremafinder.vi
- Knickfinder.vi
- Nächstes Extremum Finder.vi
- Mittelwert.vi
- Hauptprogramm.vi
- 10 Kombinationen der 5 Sensoren bei Extrema.vi“
- 10 Kombinationen der 5 Sensoren bei Knicks.vi“
- Abstandsrechnung
- Fallunterscheidung.vi
- Graphik.vi
- Größte von 3 Zahlen.vi
- Kleinste von 3 Zahlen
- Koordinaten Mittelwert.vi
- Laufzeitdifferenzen von 3 Sensoren.vi
- Letzte drei Runden in Array.vi“
- Nullpaarzähler und aus Array streicher.vi
- Richtig runden.vi
- Schlechte Werte Vernichter.vi
- Werte aus array auslesen.vi
- Werte in Array schreiben.vi

Manche Werte (Konstanten) können bei Bedarf geändert werden. So zum Beispiel im Front Panel des „Berechnungsprogramm.vi“. Hier können die unter Kapitel 7.1 beschriebenen Werte „Buffer Size“ und „scan rate“ SR bei Bedarf geändert werden. Außerdem werden hier die Koordinaten der Sensoren (Koordinatensystem siehe Abbildung 3-5) und die Schallgeschwindigkeit in [cm/Skt.] eingetragen. Leider war es nicht möglich, die Schallgeschwindigkeit direkt aus den diese bestimmenden Werten zu errechnen – deshalb ist es nötig dies manuell zu tun. Der Zusammenhang lautet:

$$c_s [\text{Skt.}] = SR [\text{Hz}] / c_s [m / s] \quad \text{Gleichung 20}$$

Im Feld „Stop nach Klopfer Nr“ ist es möglich das Programm nach einem bestimmten Klopfer anzuhalten, um beispielsweise das Signal der Sensoren genauer analysieren zu können.