

# Maths in Games

**Schiffe versenken - Strategiespiel oder doch nur Glück?**

**Grundlagen des Spiels**

**Vorwissen aus der Stochastik**

**Mensch-Computer-Vergleich**

**Strategien**

**Dokumentation**

**Online-Seminararbeit:**



[https://drive.google.com/drive/folders/11ErKsciWEpD7DKMk0umpLK6J-1\\_1H2UU?usp=sharing](https://drive.google.com/drive/folders/11ErKsciWEpD7DKMk0umpLK6J-1_1H2UU?usp=sharing)

Veröffentlichung: 07. Dez. 2022

Simon Völkl

voelklsimon02@gmail.com

Lehrkraft: Herr Christian Maag

Fächer: Mathematik & Englisch

Schuljahr: 2022/23

# Inhaltsverzeichnis

I. Abstract .....	4
II. Einleitung in die Seminararbeit .....	5
III. Hintergründe des Schiffeversenkens .....	6
IV. Spielregeln.....	7
1. Grundprinzip .....	7
2. Spielablauf .....	7
a) Verteilung der Flotte.....	7
b) Angriffe auf die gegnerische Flotte.....	9
3. Mögliche Zusatzregeln.....	10
V. Vorwissen für die Stochastik.....	11
1. Ergebnisse und Baumdiagramm .....	11
2. Ereignisse und Wahrscheinlichkeiten .....	12
3. Urnenmodelle und Parität .....	13
VI. Strategien beim Schiffeversenken.....	14
1. Der Mensch-Computer-Vergleich.....	14
2. Die Verteilung der Flotte .....	14
a) Zufällige Schiffsplatzierungen .....	15
b) Rand- und Eckpositionierungen .....	16
3. Die Angriffsphase .....	18
a) Triviale Zufallsangriffe.....	18
b) Rand- und Eckschüsse.....	19
c) Einzeichnen der Fehlschüsse und Schiffsanzahlanalyse .....	20
d) Jagd- und Zielmodus .....	21
e) Diagonale Angriffe und Parität.....	23
VII. Abschließende Worte.....	25
VIII. Dokumentation des Mensch-Computer-Vergleichs zum Schiffeversenken.....	26
1. Erstellung des ersten Flottengenerators .....	26
a) Zielsetzung für das Programm.....	26
b) Darstellung des Flottengenerators als Blackbox-System .....	27
c) Hypothese .....	27
d) Coding des Programms .....	28
e) Die Konsolenausgabe beim Programmablauf .....	29

f) Auswertung des Programms .....	30
g) Aussage über die Reliabilität des Flottengenerators.....	31
2. Studie zum Schiffeversenken.....	33
a) Zielsetzung der Umfrage .....	33
b) Hypothese.....	34
c) Befragung der Versuchsteilnehmer .....	34
d) Ergebnisse der Umfrage .....	35
3. Interview zur Behebung der Problematik des ersten Flottengenerators.....	38
4. Erstellung des zweiten Flottengenerators.....	39
a) Zielsetzung .....	39
b) Das Coding und die Konsolenausgabe.....	39
c) Auswertung des Flottengenerators 2.1 .....	39
d) Auswertung des Flottengenerators 2.2 .....	40
e) Auswertung des Flottengenerators 2.3 .....	41
f) Darstellungen der Heatmaps des zweiten Flottengenerators.....	42
g) Die Vorhersehbarkeit des Generators.....	45
IX. Anhang.....	46
1. Das Interview mit Herrn Carrasco in Textform .....	46
2. Die Umfragen in Papierform.....	47
3. Vertiefung der Formel des Jagd- und Zielmodus.....	48
a) Herleitung der Formeln .....	48
b) Analyse der Graphen .....	49
X. Abbildungsverzeichnis .....	50
XI. Literaturverzeichnis.....	52
XII. Erklärung .....	566

# I. Abstract

*„Schiffe versenken ist ein Glücksspiel [...] [und] prinzipiell gibt es keine Taktik für Glücksspiele“,*

behauptet Micaela Dörr von der Ratgeber-Redaktion Helpster.<sup>1</sup> Das Ziel der Seminararbeit ist, diese Aussage zu bekräftigen oder zu widerlegen. Es wird dabei diskutiert, ob das Schiffeversenken ein Glücksspiel ist und es Strategien zur deutlichen Erhöhung der Gewinnchancen gibt.

Die Arbeit stützt sich vor allem auf der Spielliteratur von Nils Hesse „Spielend gewinnen“ und Werken von Siegbert A. Warwitz, da zu dem Schlachtspiel sonst nur spärlich Fachschrifttum vorhanden ist. Außerdem wurden einige wissenschaftliche Arbeiten, wie Nick Berrys „Battleship“-Analyse zu Strategien beim Schiffeversenken in die Untersuchung mit aufgenommen. Mathematisches Wissen zur Stochastik wurde vor allem den Büchern „Lehrbuch der Stochastik“ von Nawrotzki und Tantons „Encyclopedia of Mathematics“ entnommen.

Zu Beginn werden die Hintergründe des Spiels, der grundsätzliche Ablauf des Schiffeversenkens und zusätzliche Regeln erläutert. Anschließend vertieft sich die Seminararbeit in die Stochastik, weshalb ein Einblick in die mathematischen Grundkenntnisse dazu gegeben wird. Darauffolgend wird ein Einblick in den Mensch-Computer-Vergleich in der Form einer Dokumentation gegeben, die zwei grundlegende Versionen einer Anwendung zum Flottengenerieren, eine Expertenmeinung zur bestmöglichen Programmierung und eine Umfrage von 50 Versuchsteilnehmern beinhaltet. Die Auswertung erfolgt mithilfe von Heatmaps, welche die Daten grafisch vereinfachend darstellen.

Auf den Ergebnissen basierend werden Strategien zur bestmöglichen Verteilung der Flotten ausgewertet. Weitere Methoden analysiert man beim Angreifen auf die gegnerische Flotte mit Hilfe von spielinternen und stochastischen Elementen.

Die Untersuchung zeigt, dass das Schiffeversenken zwar ein Glücksspiel ist, die Gewinnwahrscheinlichkeit aber signifikant von den gewählten Strategien und Spielfähigkeiten abhängt, weshalb Micaela Dörr von Helpster widersprochen werden muss.

***Diese Arbeit ist im Rahmen des Seminars „Maths in Games“ von Herrn Christian Maag an der FOSBOS-Weiden entstanden. Sie erhielt den 2. Platz im Fach Mathematik der Dr. Hans Riegel-Fachpreise 2023 an der Universität Regensburg.***

---

<sup>1</sup> Dörr, *Schiffe versenken - diese Taktik verhilft Ihnen zum Sieg*, [https://www.helpster.de/schiffe-versenken-diese-taktik-verhilft-ihnen-zum-sieg\\_113630](https://www.helpster.de/schiffe-versenken-diese-taktik-verhilft-ihnen-zum-sieg_113630), Letzter Zugriff am 07.09.2022

## II. Einleitung in die Seminararbeit

Im Rahmen der Seminararbeit soll die folgende Aussage von Micaela Dörr, einer Redakteurin von Helpster, auf ihren Wahrheitsgehalt untersucht werden:

*„Schiffe versenken ist ein Glücksspiel [...] [und] prinzipiell gibt es keine Taktik für Glücksspiele“<sup>2</sup>*

Der Kerngehalt dieser Behauptung fasst sich aus den beiden Fragen zusammen, ob das Schiffeversenken ein **Glücksspiel** ist und es **Strategien zur Steigerung der Gewinnwahrscheinlichkeit** gibt. Zur Beantwortung dieser Überlegungen werden somit die theoretischen Vorkenntnisse zum Schiffeversenken und der Stochastik verständlich dargestellt, um infolgedessen mögliche Strategien zu analysieren und daraus einen Entschluss fassen zu können.

---

<sup>2</sup> Dörr, *Schiffe versenken - diese Taktik verhilft Ihnen zum Sieg*, [https://www.helpster.de/schiffe-versenken-diese-taktik-verhilft-ihnen-zum-sieg\\_113630](https://www.helpster.de/schiffe-versenken-diese-taktik-verhilft-ihnen-zum-sieg_113630), Letzter Zugriff am 07.09.2022

### III. Hintergründe des Schiffeversenkens

Das Spiel „**Schiffe versenken**“, auch unter den Namen „Flottenmanöver“ oder „Seeschlacht“ bekannt, ist ein harmloses Kriegsspiel, das zumeist mit Stift und Papier gespielt wird<sup>3</sup>.

Über die geschichtlichen Hintergründe des Schiffeversenkens ist nicht viel bekannt; Literatur dazu ist nur spärlich vorhanden. Der Germanist Warwitz konnte allerdings anhand von Zeitzeugenberichten nachweisen, dass das Spiel schon kurz vor dem Beginn des 20. Jahrhunderts bereits seinen Namen hatte, die genaue Herkunft ist jedoch unerforscht<sup>4</sup>. „**Basilinda**“, ein Spiel aus dem Jahr 1890 von E. I. Horsman, soll laut Whitehill ein Vorläufer des späteren Spiels Schiffe versenken sein<sup>5, 6, 7</sup>. Dort sitzen sich zwei Spieler gegenüber und bewegen Figuren auf einem Spielfeld umher, wobei sich eine Trennwand zur Blockierung der Sicht auf die gegnerische Seite in der Mitte befindet. Da weder die Ursprungszeit noch die Herkunft des Spiels genau bestimmt werden könnten, müsste noch **historische Recherche** zum Schiffeversenken betrieben werden<sup>8</sup>.

In der heutigen Zeit wird Schiffe versenken „meist nur mit **Papier und Bleistift** [...] in Schulen oder Universitäten gespielt“<sup>9, 10</sup>. Dabei gibt es verschiedene Spielvarianten, die sich mit unterschiedlichen Regeln und sogar kommerziellen Varianten in der Form von Spielbrettern spielen lassen<sup>11</sup>. Zuletzt kamen auch digitale Versionen des Spiels in der Form von Apps<sup>12</sup> oder Websites<sup>13</sup> auf.

Bild wegen Urheberrecht  
ausgeblendet.

Abbildung 1: Eine kommerzielle  
Variante des Schiffeversenkens als  
Spielbrett

<sup>3</sup> Vgl. Warwitz, *Vom Sinn des Spielens. Reflexionen und Spielideen*, S. 139-140

<sup>4</sup> Vgl. Warwitz, *Spiele anderer Zeiten und Völker - mit Kindern entdecken und erleben*, S. 25

<sup>5</sup> Vgl. Whitehill, *Games: American Boxed Games and Their Makers, 1822-1992: With Values*

<sup>6</sup> Vgl. Unbekannter Autor, *1890 ANTIQUE BOARD GAME OF BASILINDA BY E.I. HORSMAN HALMA CO RARE WAR GAME*, <https://www.worthpoint.com/worthopedia/1890-antique-board-game-basilinda-1748619764>, Letzter Zugriff am 31.07.2022

<sup>7</sup> Vgl. Polczynski, *Battleship Games*, <https://thebiggamehunter.com/games-one-by-one/battleship-games/>, Letzter Zugriff am 02.08.2022

<sup>8</sup> Vgl. Ebd.

<sup>9</sup> Hesse, *Spielend gewinnen*, S. 61

<sup>10</sup> Vgl. Warwitz, *Vom Sinn des Spielens. Reflexionen und Spielideen*, S. 139-140

<sup>11</sup> Vgl. Hesse, *Spielend gewinnen*, S. 61

<sup>12</sup> Vgl. smuttlewerk interactive, *Schiffe Versenken Fleet Battle*, <https://play.google.com/store/apps/details?id=de.smuttlewerk.fleetbattle>, Letzter Zugriff am 31.07.2022

<sup>13</sup> Vgl. Battleship-game, *Schiffe Versenken*, <http://de.battleship-game.org/>, Letzter Zugriff am 31.07.2022

## IV. Spielregeln

Für die praktische Analyse des Spiels Schiffe Versenken ist es von Nöten, die genauen Regeln zu verstehen.

### 1. Grundprinzip

Schiffe versenken wird bei den meisten Varianten zu zweit, einander gegenüber an einem Tisch sitzend, gespielt. Jeder der sogenannten Admirale hat jeweils zwei Blätter; eins für die eigenen Boote, das andere für die gegnerischen Schiffe. Damit man bei dem Spiel nicht betrügen kann, wird eine Trennwand zwischen den beiden Parteien aufgestellt, die die Sicht auf die Blätter des Gegners versperrt.

Viele Websites bieten statt der analogen Spielart die Möglichkeit an, digital gegen einen **Computer** zu spielen, der einem bestimmten Algorithmus folgt<sup>14</sup>. Da das Spielverhalten von Menschen im Vergleich zu Computern unterschiedlich ist, sollte das eigene Vorgehen im Spiel dem Gegner entsprechend angepasst werden.

### 2. Spielablauf

Die Spieler müssen zuerst in der Flottenverteilungsphase eine bestimmte Anzahl von Schiffen auf einer Karte platzieren, um danach in der Angriffsphase Schüsse auf die gegnerischen Boote zu feuern. Wenn ein Spieler alle Schiffe des Gegners zerstört, gewinnt er das Spiel.

#### a) Verteilung der Flotte

Das Spiel beginnt mit der Verteilung der Flotten: Jeder Spieler zeichnet jeweils eine **Seekarte** mit 10x10 Kästchen für die eigenen Schiffe und für die Angriffe auf den Gegner. Die Karten werden mit Koordinaten versehen, sodass am oberen Rand waagrecht die Buchstaben A-J und am linken Rand senkrecht die Zahlen 1-10 stehen<sup>15, 16</sup>. Darauf folgend **positionieren die Admirale ihre Schiffe** auf den eigenen Seekarten: Es sollen ein Schlachtschiff mit der Länge



	A	B	C	D	E	F	G	H	I	J
1										
2										
3										
4										
5										
6										
7										
8										
9										
10										

Abbildung 2: Seekarte mit den dazugehörigen Koordinaten

<sup>14</sup> Vgl. BattleshipOnline, *Battleship Game*, <https://www.battleshiponline.org/>, Letzter Zugriff am 31.07.2022

<sup>15</sup> Vgl. Warwitz, *Vom Sinn des Spielens. Reflexionen und Spielideen*, S. 139

<sup>16</sup> Vgl. Barshap, *Crypto-Battleships or How to play Battleships game over the Blockchain?*, <https://arxiv.org/pdf/1807.08142.pdf>, S. 1-2, Letzter Zugriff am 02.08.2022

von 5 Kästchen, zwei Kreuzer mit 4 Kästchen, drei Zerstörer mit der Länge 3 und vier U-Boote mit 2 Kästchen gesetzt werden<sup>17, 18</sup>.

Zwar gibt es zwischen den verschiedenen Spielvarianten große Unterschiede bei der Anzahl und Bezeichnung der Boote, dennoch sind es aber meist mehr kurze als lange Boote.<sup>19</sup> In einigen Abwandlungen des Schiffeversenkens können auch Boote mit der **Größe eines Kästchens** platziert werden. Da aber die geringe Länge zu einer Unvorhersehbarkeit der Position auf der Seekarte mit der zwangsläufigen Folge von Zufallsschüssen führt, die keiner Strategie entsprechen, werden diese oftmals nicht im Spiel genutzt.

Bei der geheimen Platzierung der Boote sollten die **Regeln** eingehalten werden:

Die Schiffe müssen innerhalb des Spielfelds waagrecht oder senkrecht gesetzt werden. Eine diagonale Aufstellung eines Bootes ist ungültig, wie auch eine Platzierung eines Schiffs in ungeradlinigen Formen. Zwischen den Schiffen ist ein Abstand von mindestens einem Kästchen in alle Richtungen einzuhalten, da sie sich keinesfalls berühren dürfen<sup>20</sup>.

	A	B	C	D	E	F	G	H	I	J
1			x	x	x	x	x			x
2										x
3		x	x	x						
4						x	x			
5										
6		x			x				x	x
7		x			x		x			
8		x					x		x	
9							x		x	
10	x	x	x	x			x		x	

Abbildung 3: Beispielhafte Aufstellung der Schiffe

Henning Thielemann berechnete am 8. Januar 2012 mit einem Haskell-Programm die Anzahl der Möglichkeiten, Boote beim Schiffeversenken anzuordnen. Die kombinatorischen Berechnung ergab, dass es „[a]uf einem 10 x 10-Spielfeld [...] etwa 26,5 Billionen [...] Möglichkeiten“<sup>21</sup> gäbe, die Schiffe mit dieser Spielvariante zu platzieren, wobei „Drehungen und Spiegelungen nicht herausgerechnet“<sup>22</sup> wurden.

<sup>17</sup> Vgl. Take ist Serious, *Schiffe versenken*, <https://take-it-serious.de/schiffe-versenken/>, Letzter Zugriff am 02.08.2022

<sup>18</sup> Vgl. Thielemann, *Kombinatorik bei "Schiffe versenken"*, [https://www.researchgate.net/publication/333579639\\_Kombinatorik\\_bei\\_Schiffe\\_versenken](https://www.researchgate.net/publication/333579639_Kombinatorik_bei_Schiffe_versenken), Letzter Zugriff am 02.08.2022

<sup>19</sup> Vgl. Milton Bradley Company, *Battleship*, <https://www.hasbro.com/common/instruct/battleship.pdf>, S.1, Letzter Zugriff am 02.08.2022

<sup>20</sup> Take It Serious, *Schiffe versenken*, <https://take-it-serious.de/schiffe-versenken/>, Letzter Zugriff am 02.08.2022

<sup>21</sup> Thielemann, *Kombinatorik bei "Schiffe versenken"*, [https://www.researchgate.net/publication/333579639\\_Kombinatorik\\_bei\\_Schiffe\\_versenken](https://www.researchgate.net/publication/333579639_Kombinatorik_bei_Schiffe_versenken), Letzter Zugriff am 02.08.2022

<sup>22</sup> Ebd.

## b) Angriffe auf die gegnerische Flotte

Wenn die Admirale mit der Verteilung ihrer Flotten fertig sind, beginnt die **Angriffsphase**. Hier schießen die Spieler in abwechselnden Zügen aufeinander, um die gegnerischen Schiffe zu versenken und das Spiel zu gewinnen.<sup>23</sup>

Befolgt man die normalen Spielregeln, kann jeder Spieler pro Runde abwechselnd einen Schuss auf den Konkurrenten tätigen. Dazu sagt der angreifende Admiral eine Koordinate, wie beispielsweise „A-5“, woraufhin der verteidigende Spieler auf seine Seekarte sieht. Wenn sich auf der Position kein Schiff befindet, antwortet der Verteidigende „**Wasser**“, da kein Boot getroffen wurde. Falls auf der Koordinate ein Teil eines Schiffs ist, so muss der Spieler mit „**Treffer**“ reagieren, weil ein Boot getroffen wurde. In dem Fall, dass alle Teile eines Schiffs angegriffen wurden, geht das Boot unter; der verteidigende Admiral entgegnet daraufhin mit „**Versenkt!**“. Aufgrund der Abstandsregel zwischen den Schiffen sollte um untergegangene Boote herum „Wasser“ notiert werden.

Diese Informationen sollten von beiden Seiten mitgeschrieben werden:

Der Angreifer auf dem Blatt für den Gegner, der Verteidiger auf dem Papier der eigenen Flotte. Meist wird dazu für „Wasser“ ein Punkt „.“ in das Kästchen eingezeichnet, für einen „Treffer“ ein „x“-Zeichen und im Falle einer Schiffsversenkung werden die entsprechenden Felder als „■“ ausgemalt. Mit dieser Mitschrift beider Parteien werden Möglichkeiten zum Betrügen verhindert und die Spieler erhalten einen guten Überblick über die Spielsituation. Zudem können doppelte Schüsse auf ein Kästchen des Verteidigers verhindert werden.<sup>24, 25</sup>

	A	B	C	D	E	F	G	H	I	J
1	x							.		.
2	x		.	.	.					
3			.	■	.		.		x	
4			.	■	.				x	
5			.	.	.		.		x	
6	.									
7										.
8	.	.	.	.	.			.		
9	.	■	■	■	.					
10	.	.	.	.	.					.

Abbildung 4: Beispielhafte Seekarte eines Angreifers

<sup>23</sup> Vgl. Milton Bradley Company, *Battleship*, <https://www.hasbro.com/common/instruct/battleship.pdf>, S. 2, Letzter Zugriff am 02.08.2022

<sup>24</sup> Vgl. Milton Bradley Company, *Battleship*, <https://www.hasbro.com/common/instruct/battleship.pdf>, S. 2, Letzter Zugriff am 02.08.2022

<sup>25</sup> Vgl. Barshap, *Crypto-Battleships or How to play Battleships game over the Blockchain?*, <https://arxiv.org/pdf/1807.08142.pdf>, S. 1-2, Letzter Zugriff am 02.08.2022

### 3. Mögliche Zusatzregeln

In vielen **Spielvarianten** findet man zusätzliche Regeln, die das Schiffeversenken abwechslungsreicher gestalten sollen. Im Folgenden sind deshalb die bekanntesten und am häufigsten genutzten Varianten aufgelistet.

Die **Seemine** (🚫) ist in der Flottenverteilungsphase auf ein unbesetztes Feld in die Verteidigungs-Seekarte zu setzen. Wenn der gegnerische Admiral die nur einmal vorhandene Mine trifft, muss er eine Runde aussetzen, weshalb sie auf eine Koordinate gesetzt werden sollte, an der man einen intuitiven Angriff erwartet, wie zum Beispiel in den Ecken des Spielfeldes oder neben eines eigenen Bootes.

Die **Küstenbatterie (K)** ist einmalig in einer Runde in der Angriffsphase einsetzbar. Sie fungiert wie ein normaler Angriffsschuss, wobei ein Treffer das Boot des Gegenspielers sofort versenkt, als wären alle Teile des Schiffs getroffen worden. Da man für Schiffe der Längen vier oder fünf sehr viele Schüsse benötigen würde, um sie zu versenken, sollte die Küstenbatterie verwendet werden, wenn man an einer Position auf der Seekarte ein großes Schiff vermutet und der Gegner bestenfalls schon viele kleinere Schiffe verloren hat.

Einmalig einsetzbar ist auch die in einem Angriff anwendbare **Rakete (R)**. Sie lässt sich auf eine Koordinate auf der Seekarte feuern, um das angezielte Feld, sowie die direkt umliegenden Kästchen zu treffen. Wie in der Abbildung 5 erkennbar ist, wird die Trefferreichweite der Rakete in vielen Varianten unterschiedlich aufgefasst. Nützlich ist sie an Positionen auf der Karte, wo Boote vermutet werden; sie lassen sich aber auch gut in der Kombination mit anderen Strategien einsetzen.

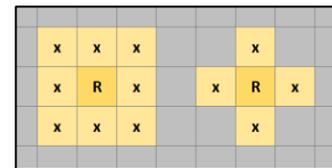


Abbildung 5: Die Spielweise der Rakete ist in vielen Varianten unterschiedlich

Der **Salvenmodus** ist ein dauerhafter Spielmodus in der Angriffsphase, bei der ein Angreifer so viele Schüsse auf den gegnerischen Admiral feuern kann, wie das größte Schiff des Schützen Kästchen lang ist. Dabei sollten die beiden Antwortvarianten bezüglich des Feedbacks an den Angreifer beachtet werden: Der Verteidiger kann entweder wie im normalen Spiel zu jedem angegriffenen Feld eine genaue Auskunft geben oder er teilt dem angreifenden Admiral nur die Anzahl der „Wasser“- oder „Treffer“-Schüsse und die Summe der „versunkenen Boote“ ohne der Nennung der jeweiligen Koordinaten mit<sup>26, 27</sup>.

In der weiteren Betrachtung des Spiels werden die **Zusatzregeln nicht verwendet**, da deren Einfluss auf das Schiffeversenken und die dazugehörigen Berechnungen kaum vorhersehbar wären.

<sup>26</sup> Take It Serious, *Schiffe versenken*, <https://take-it-serious.de/schiffe-versenken/>, Letzter Zugriff am 02.08.2022

<sup>27</sup> Vgl. Polczynski, *Battleship Games*, <https://thebiggamehunter.com/games-one-by-one/battleship-games/>, Letzter Zugriff am 02.08.2022

## V. Vorwissen für die Stochastik

Zum besseren Verständnis der folgenden Seiten zur Analyse der Spielstrategien beim Schiffeversenken ist eine Einführung in die Grundkenntnisse der Stochastik nötig.

### 1. Ergebnisse und Baumdiagramm

Ein **Zufallsexperiment** ist eine Untersuchung, deren Ausgang nicht vorhersehbar ist. Der Ausgang eines solchen Experiments wird als **Ergebnis**  $\omega$  bezeichnet. Der **Ergebnisraum**  $\Omega$  gibt die Menge aller möglichen Ergebnisse eines Zufallsexperiments an. Beispielsweise wäre der Ergebnisraum eines einfachen Münzwurfs mit den möglichen Ausgängen „Kopf“ und „Zahl“  $\Omega_{\text{Münzwurf}} = \{K; Z\}$ . Die Anzahl der Elemente eines Ergebnisraums heißt **Mächtigkeit des Ergebnisraums**  $|\Omega|$  und wäre beim Münzwurf  $|\Omega_{\text{Münzwurf}}| = 2$ .<sup>28</sup>

Vor allem mehrstufige Zufallsexperimente lassen sich besonders übersichtlich und einfach mithilfe von **Baumdiagrammen** veranschaulichen. Diese sind horizontal oder vertikal darstellbar, wie es in der Abbildung 6 zum einfachen Münzwurf gezeigt wird.<sup>29</sup>

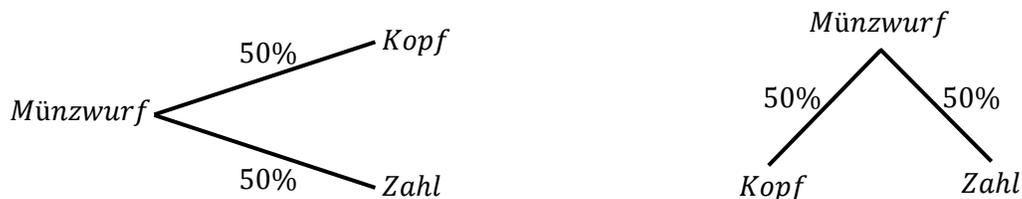


Abbildung 6: Horizontales und vertikales Baumdiagramm zum einfachen Münzwurf

<sup>28</sup> Vgl. Henze, *Stochastik für Einsteiger*, S. 3-6

<sup>29</sup> Vgl. Rumsey, *Wahrscheinlichkeitsrechnung für Dummies*, S.67-74

## 2. Ereignisse und Wahrscheinlichkeiten

Eine Teilmenge  $E$  des Ergebnisraums  $\Omega$  eines Zufallsexperiments ist das **Ereignis  $E$**  von  $\Omega$ , welches beim Ende des Zufallsexperiments mit dem dazugehörigen Ergebnis als Ausgang auftritt.

Bei  $n$  Durchführungen kann die **absolute Häufigkeit  $H_n(E)$**  des Eintretens eines Ereignisses bestimmt werden. Aus dieser Anzahl der Ausgänge des Zufallsexperiments mit dem Ereignis  $E$  und den Wiederholungen  $n$  berechnet man die **relative Häufigkeit  $h_n(E)$**  mit der Formel  $h_n(E) = \frac{H_n(E)}{n}$ . Diese stellt die ungenaue Wahrscheinlichkeit dar, wie oft ein Ereignis  $E$  in Abhängigkeit zu den Durchführungen aufgetreten ist.

Das **Gegeneignis  $\bar{E}$**  eines Ereignisses ist mit  $h_n(\bar{E}) = 1 - h_n(E)$  zu berechnen, wodurch man die Wahrscheinlichkeit, mit der das Ereignis nicht eintritt, erhält. Zudem können einelementige Ereignisse auch als **Elementarereignisse** bezeichnet werden.<sup>30</sup>

Mit dem **empirischen Gesetz der großen Zahlen** kann bei sehr vielen Durchführungen  $n$  eines Zufallsexperiments die relative Häufigkeit eines Ereignisses einen festen Wert  $p$  zwischen 0 und 1 annehmen. Jedem Ereignis  $E$  des Ergebnisraums  $\Omega$  wird eine eigene **Wahrscheinlichkeit  $P$**  zugeordnet, wenn die sogenannte Wahrscheinlichkeitsverteilung<sup>31</sup> mit den Kolmogorow-Axiomen der Nichtnegativität, Normierung und Additivität erfüllt ist.<sup>32</sup> **Laplace-Experimente** sind Versuche, bei denen alle Ergebnisse und damit auch Elementarereignisse gleich wahrscheinlich sind. Die Wahrscheinlichkeit eines beliebigen Ereignisses ist  $P(E) = \frac{\text{Anzahl } E_{\text{günstig}}}{\text{Anzahl } E_{\text{möglich}}} = \frac{|E|}{|\Omega|}$ .<sup>33</sup>

Mehrstufige Zufallsexperimente werden mit Hilfe der zwei **Pfadregeln**<sup>34</sup> auf die resultierenden Wahrscheinlichkeiten ausgewertet: Zuerst können zur Berechnung einer Pfadwahrscheinlichkeit die Wahrscheinlichkeiten entlang eines Pfades multipliziert werden, woraufhin man die zusammengehörigen Pfadwahrscheinlichkeiten miteinander addiert.<sup>35</sup>

---

<sup>30</sup> Vgl. Cornelsen, *Mathematik Berufliche Oberschule Bayern*, S. 158-160

<sup>31</sup> Aufgrund der Ausführlichkeit und Komplexität der Definition werden die drei Eigenschaften der Wahrscheinlichkeitsverteilung nicht weiter erklärt.

<sup>32</sup> Vgl. Nawrotzki, *Lehrbuch der Stochastik*, S. 103-112

<sup>33</sup> Vgl. Maas, *Stochastik kompakt für Dummies*, S. 110-113

<sup>34</sup> In der *Abbildung 11* zu dem *Baumdiagramm von 30 perfekten Zufallsangriffen* sind diese Regeln grafisch dargestellt.

<sup>35</sup> Vgl. Cornelsen, *Mathematik Berufliche Oberschule Bayern*, S. 172-173

### 3. Urnenmodelle und Parität

Viele Probleme und Zufallsexperimente in der stochastischen Mathematik lassen sich auf die **Urnenmodelle** zurückführen, wie es beispielhaft in der Grafik 7 abgebildet ist. Dabei unterscheidet man zwischen den Modellen die Beachtung der Reihenfolge und ob die Kugeln in der Urne nach dem Ziehen zurückgelegt werden.

Bild wegen Urheberrecht ausgeblendet.

Die **hypergeometrische Verteilung** entspricht einem speziellen Urnenmodell und lässt sich mit der Formel  $P(E_{Farbkugel}) = \frac{\binom{n}{k} \binom{N-n}{M-k}}{\binom{N}{M}}$

Abbildung 7: Beispielhafte Darstellung eines Urnenmodells

berechnen, wobei  $N$  die gesamten Kugeln in der Urne darstellt und  $M$  die Anzahl der gewünschten Farbkugeln ist. Die zu ziehenden Kugeln werden durch  $n$  repräsentiert, während  $k$  der Anzahl der farbigen Kugeln entspricht. Der Formelabschnitt  $\binom{n}{k}$  bedeutet beispielsweise  $\frac{n!}{(n-k)!k!}$ , was ein Ziehen der Kugeln aus der Urne ohne Zurücklegen und ohne Betrachtung der Reihenfolge bedeutet.<sup>36</sup>

Zwei Zahlen, die beide entweder gerade oder ungerade sind, besitzen die gleiche **Parität**.<sup>37</sup> Ist eine Zahl also gerade während die andere ungerade ist, besitzen sie in der Gesamtbetrachtung somit eine unterschiedliche Parität. Manchmal wird diese Parität auch offener gehandhabt: James Tanton schreibt, dass bei einer Parität auch einer von zwei möglichen Zuständen gemeint sein kann.<sup>38, 39</sup>

<sup>36</sup> Vgl. Georgii, *Stochastik: Einführung in die Wahrscheinlichkeitstheorie und Statistik*, S. 36

<sup>37</sup> Vgl. Die Parität ist vor allem in der Strategie der *Diagonalen Angriffe und Parität* von Bedeutung.

<sup>38</sup> Vgl. Tanton, *Encyclopedia of Mathematics*, S. 378-379

<sup>39</sup> Vgl. Musiol, *Taschenbuch der Mathematik*, S. 606

## VI. Strategien beim Schiffeversenken

Nach der Klärung des nötigen Hintergrundwissens zum Schiffe Versenken und des stochastischen Überblicks soll nun die Ausgangsfrage geklärt werden, ob der Gewinn im Spiel nur vom Glück abhängt oder auch strategische Elemente eine maßgebliche Rolle spielen.<sup>40</sup> Dazu wurde ein Mensch-Computer-Vergleich durchgeführt, der wichtige Ergebnisse für die Verteilung der Flotte aber vor allem für die Angriffsphase liefert.

### 1. Der Mensch-Computer-Vergleich

Um die Analyse zu möglichen Strategien beim Schiffeversenken widerlegen oder belegen zu können, wurde zuvor ein sogenannter Mensch-Computer-Vergleich vollzogen.

In einer anonymen **Umfrage** von 50 Schülern und Lehrern der FOSBOS-Weiden wurde gefragt, wie sie ihre Schiffe auf einer Seekarte der Größe 10x10 verteilen würden. Danach sollten sie einen blinden Angriff auf einen Gegner tätigen, woraufhin sie schlussendlich noch verwendete Taktiken angeben konnten. Parallel dazu erstellte ich mit den Programmiersprachen Java und C# **verschiedene Anwendungen** zur Berechnung möglicher Flottenaufstellungen. Die Ergebnisse der Umfrage und der Softwares trug ich jeweils in Excel-Tabellen ein, um Heatmaps zu erhalten, welche die Resultate grafisch einfach darstellten.<sup>41</sup>

An diesem Punkt der Arbeit ist die **Ansicht der „VII. Dokumentation des Mensch-Computer-Vergleichs beim Schiffeversenken“** zum besseren Verständnis der nächsten Kapitel sehr empfehlenswert. Die Abbildungen, auf die sich im Laufe der Strategieanalyse bezogen wird, befinden sich auch auf diesen Seiten.

### 2. Die Verteilung der Flotte

Um im Spiel seltener getroffen zu werden, empfiehlt es sich, einige Taktiken für die Verteilung der Schiffe anzuwenden.

---

<sup>40</sup> Vgl. Aczel, *Der ganz normal verteilte Zufall*, S. 107-108

<sup>41</sup> Alle verwendeten Dateien sind auf Google Drive frei verfügbar und unter der Online-Weiterleitung im Teil XI. *Literaturverzeichnis* erhältlich.

### a) Zufällige Schiffsplatzierungen

Eine Möglichkeit, um weniger oft getroffen zu werden, ist, die Unvorhersehbarkeit der eigenen Schiffspositionen zu erhöhen. Dies gelingt am besten, indem die Boote **zufällig platziert** und **Muster vermieden** werden.

Zunächst sollte versucht werden, keine **parallelen Schiffe** zu setzen. Im nebenstehendem Bild ist ein Muster zu erkennen, das vom Gegenspieler einfach erratbar wäre. Die sich repetierenden, nach unten gerichteten Boote lassen den Gegner bereits nach einigen Treffern vermuten, wo sich noch weitere, ähnlich verteilte Boote befinden. Der Umfrage zum Verteilen der Flotten nach, setzen 10% der Befragten ihre Schiffe übermäßig oft waagrecht und 6% vor allem senkrecht auf. Aufstellungen wie von diesen Teilnehmern wären demnach sehr leicht aufzudecken, da sich die Boote meist parallel zueinander befänden.

	A	B	C	D	E	F	G	H	I	J
1		x		x		x		x		
2		x		x		x		x		
3		x						x		
4								x		
5										

Abbildung 8: Beispiel eines leicht erratbaren Musters

Ähnlich sollte sich auch von **symmetrischen Aufstellungen** ferngehalten werden. Wie Seth Brown von TheSpruceCrafts schreibt, sucht der Gegner stets nach Auffälligkeiten und Möglichkeiten zum erfolgreichen Angriff und analysiert daher das Setzverhalten des Spielers. Eine asymmetrische Verteilung der Boote erschwert es dem Gegner, Muster in der Verteidigung zu finden.<sup>42, 43</sup>

Ein **Beispiel** für diese Muster in der Schiffsverteilungsphase ist der Flottengenerator 2.1. Wie bereits in der Dokumentation beschrieben wurde, versucht das Programm, die Boote unvorhersehbar zu platzieren. Zwar funktioniert dies in der Spielfeldmitte mit einer ungefähren Abweichung von nur  $\Delta P_{Ausgabe\ mittig} = Ausgabe_{max,mittig} - Ausgabe_{min,mittig} \approx 29,56\% - 26,37\% = 3,19\%$ , **am Rand** im Vergleich zur gesamten Karte jedoch durchschnittlich **5,30% öfter**. Der Abweichung zufolge entsteht das kaum übersehbare Muster in den Heatmaps und die Positionen der Schiffe werden vorhersehbarer.

Dennoch hat die Verwendung des Flottengenerators den Vorteil, völlig zufällig Schiffspositionen bestimmen zu können. Während ein **Mensch** aufgrund der Voreingenommenheit, äußeren Faktoren und Stimmungen keine Zufallszahlen oder Zufallspositionen generieren kann, greifen **Computer** mithilfe einer Crypto-Zahlenkette, einer Random()-Funktion oder anderen Werkzeugen in der Programmierung auf zufällige Zahlen zu und können damit zum Großteil unvorhersehbare Aufstellungen generieren.<sup>44</sup>

<sup>42</sup> Vgl. Brown, *How to Win at Battleship*, <https://www.thesprucecrafts.com/how-to-win-at-battleship-411068>, Letzter Zugriff am 07.09.2022

<sup>43</sup> Vgl. UltraBoardGames, *Tips to win Battleship*, <https://www.ultraboardgames.com/battleship/tips.php>, Letzter Zugriff am 03.09.2022

<sup>44</sup> Vgl. L'Ecuyer, *History of uniform random number generation*, <https://www.informs-sim.org/wsc17papers/includes/files/016.pdf>, Letzter Zugriff am 02.09.2022

## b) Rand- und Eckpositionierungen

Überwiegend häufige **Rand-** oder einzelne **Eckpositionierungen** traten in der Studie bei fast jedem dritten Teilnehmer auf, weshalb diese Spielweise genauer analysiert werden sollte.

„[...] Prinzipiell [ist es] vorteilhaft, keine eigenen [...] Boote am Spielfeldrand zu platzieren“<sup>45</sup>, behauptet die Internetseite 50PLUS in einem Artikel zum Schiffeversenken. Grund dafür sei eine erhöhte Chance auf Folgetreffer und die damit einhergehende gesamte Zerstörung eines Schiffs nach dem ersten erfolgreichen Treffer auf das Boot.<sup>46</sup> Damit ist gemeint, dass ein Schiff nach einem ersten Treffer am Rand oder vor allem in den Ecken leichter zerstört werden könne, da es **weniger mögliche Ausbreitungsrichtungen** gäbe. In der Grafik 9 ist ein beispielhafter Treffer auf drei Schiffe dargestellt, in der das Eckschiff nur zwei, das Schiff am Rand drei, und das Mittige sogar 4 potenzielle Richtungen besitzt, in denen es platziert sein könnte. Feststellbar ist die deutlich wahrscheinlichere Aufklärung des Eckschiffs nach einem Initialtreffer, da es hier weniger Ausbreitungsoptionen gibt.<sup>47</sup>

	A	B	C	D	E	F	G
1	x	? <sub>2</sub>		? <sub>1</sub>	x	? <sub>3</sub>	
2	? <sub>1</sub>				? <sub>2</sub>		
3			? <sub>1</sub>				
4		? <sub>4</sub>	x	? <sub>2</sub>			
5			? <sub>3</sub>				
6							

Abbildung 9: Erhöhte Vorhersehbarkeit nach einem Treffer am Rand oder in der Ecke

Wird nun versucht, die **Ränder zu meiden**, erfährt man bei der Platzierung der Schiffe einerseits ein **erhebliches Problem**, alle Boote in der Mitte unterzubringen, andererseits setzt man die Schiffe dadurch automatisch konzentrierter und in einem gewissen Muster in die Seekarte, wie man in der Abbildung 10 sieht. Sollte es doch ausführbar sein, alle Boote auf die Seekarte zu platzieren und die Randfelder zu ignorieren, spielt man damit statt auf 100 Kästchen, nun auf einer 8x8-Karte mit 64 Kästchen. Der Gegner sollte es damit sehr leicht haben, durch die unweigerlich entstehenden Muster die Strategie zu verstehen und einen erheblichen Vorteil im Spiel zu erlangen.

	A	B	C	D	E	F	G	H	I	J
1										
2		x	x	x	x	x			x	
3									x	
4		x	x	x	x		x		x	
5							x		x	
6	x		x	x	x					
7	x						x	x	x	
8			x							
9			x		x	x		x	x	
10			x							

Abbildung 10: Ein gescheiterter Versuch, alle Schiffe in die Mitte zu platzieren

Dennoch hat es einige Vorteile, die Mitte auszunutzen: Der Physik Doktorand Alexander A. Alemi von der Internetseite „TheVirtuosi“ hat ähnlich wie Thielemann betrachtet, wie Schiffe auf dem Spielfeld gesetzt werden können.<sup>48</sup> Er fand heraus, dass man **in der Mitte die meisten Wege** fände, ein Boot zu platzieren,

<sup>45</sup> Unbekannter Autor, *Schiffe versenken – erfolgreiche Spielstrategien*, <https://www.50plus.de/article/schiffe-versenken-die-erfolgreichsten-spielstrategien.html>, Letzter Zugriff am 01.09.2022

<sup>46</sup> Vgl. Ebd.

<sup>47</sup> Vgl. Hesse, *Spielend gewinnen*, S. 61

<sup>48</sup> Vgl. Alemi, *Alexander A. Alemi*, <https://www.alexalemi.com/>, Letzter Zugriff am 02.09.2022

wobei die Anzahl an Möglichkeiten mit der Schiffslänge anstiege. Dadurch könne der Gegner nach einem Initialtreffer nur schwierig die Ausbreitungsrichtungen erkennen.<sup>49, 50</sup>

Vergleicht man dieses Ergebnis mit den Heatmaps der relativen Häufigkeit der Flottengeneratoren 2.1 oder 2.3, wird hier eine **Ausprägung am Rand des Spielfelds** bemerkt. Auch in der Umfrage findet man vage Anzeichen in der breitflächigen Heatmap dafür, dass die Schiffe eher am Rand gesetzt wurden. Somit kann darauf geschlossen werden, dass ein einzelnes Boot zwar in der Mitte perfekt positioniert ist, eine **Flotte aber breit verteilt sein muss**. Eine überwiegende Platzierung in einem Teil der Karte stellt keine zufriedenstellende und effektive Lösung dar.

Deshalb empfiehlt Nils Hesse, **nur kleine Schiffe am Rand** oder in die Ecke zu platzieren, sodass die **großen Boote in der Mitte** mit einem gewissen Abstand zu den Rändern stationiert sind.<sup>51</sup> Mit dieser Strategie können 2er- oder 3er-Schiffe am Rand selbst nach einem Initialtreffer nur mit Schwierigkeiten weiter aufgedeckt werden, da die geringe Länge beim Setzen der Schiffe dazu beiträgt, mehr Abstand zu anderen Booten oder zum Seekartenrand einhalten zu können. Die großen Schiffe in der Mitte dagegen haben laut Alexander A. Alemi mehr Ausbreitungsmöglichkeiten, wodurch ein Aufdecken deutlich erschwert wird.<sup>52</sup>

---

<sup>49</sup> Vgl. Swanson, *The mathematically proven winning strategy for 14 of the most popular games*, <https://www.washingtonpost.com/news/wonk/wp/2015/05/08/how-to-win-any-popular-game-according-to-data-scientists/#Battleship>, Letzter Zugriff am 01.09.2022

<sup>50</sup> Alemi, *The Linear Theory of Battleship*, <http://thevirtuosi.blogspot.com/2011/10/linear-theory-of-battleship.html>, Letzter Zugriff am 07.09.2022

<sup>51</sup> Vgl. Hesse, *Spielend gewinnen*, S. 61

<sup>52</sup> Vgl. Alemi, *The Linear Theory of Battleship*, <http://thevirtuosi.blogspot.com/2011/10/linear-theory-of-battleship.html>, Letzter Zugriff am 07.09.2022

### 3. Die Angriffsphase

„Um ein Spiel des Schiffeversenkens zu gewinnen, muss man zwei Sachen versuchen: Die Wahrscheinlichkeit maximieren, jede Runde einen Treffer zu erhalten und hoffen, dass der Gegner nicht das gleiche macht.“<sup>53</sup> Um dies zu bewerkstelligen werden nun verschiedene Strategien auf ihre Nützlichkeit im Angriff analysiert.

#### a) Triviale Zufallsangriffe

Bei einigen Umfrageteilnehmern konnten beim Angriff zwar feste Strategien erfasst werden, die Großzahl aber versuchte den Gegner mit **Zufallsschüssen** aufzudecken.

Geht man vereinfacht davon aus, dass die Wahrscheinlichkeit, ein Schiffsfeld zu treffen, auf jedem Kästchen gleich ist, kann ein Angriff basierend auf Zufallstreffern als Baumdiagramm dargestellt werden. Aus diesem Diagramm werden mithilfe der Pfadregeln die Wahrscheinlichkeiten der Zwischenereignisse entlang eines Pfades multipliziert und die gemeinsamen Endereignisse summiert. Beispielsweise ist die Trefferchance bei 100 Kästchen und davon insgesamt 30 von Schiffen besetzte Felder gleich  $P(\text{Treffer}) = \frac{30}{100} = 30\%$ .

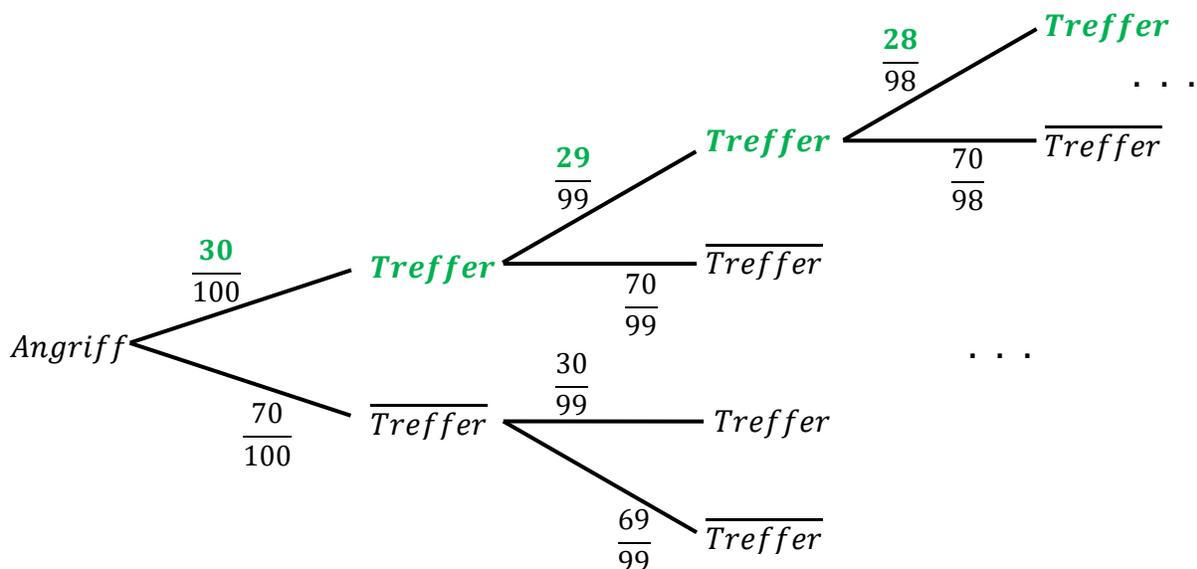


Abbildung 11: Baumdiagramm von 30 perfekten Zufallsangriffen

Nick Berry, Technologieberater und der Vorsitzender bei DataGenetics, schilderte in einem Blog, wie ineffizient Zufallsangriffe beim Schiffeversenken wären. In seiner **abgeänderten Spielversion** mit weniger Schiffen berechnete er die Wahrscheinlichkeit zum Treffen aller Schiffe. Er stellte dabei fest, dass er mit

<sup>53</sup> Swanson, *The mathematically proven winning strategy for 14 of the most popular games*, <https://www.washingtonpost.com/news/wonk/wp/2015/05/08/how-to-win-any-popular-game-according-to-data-scientists/#Battleship>, Letzter Zugriff am 01.09.2022, übersetzt von Simon Völkl

Zufallsangriffen bei einem von 6.650.134.872.937.201.800 Spielen die minimale Anzahl an Schüssen bräuchte.<sup>54</sup>

Die **Wahrscheinlichkeit**, bei diesen Spielregeln **alle Schiffe** mit einer minimalen **Schussanzahl von 30** zu treffen, läge bei  $P(\text{sofortiger Gewinn}) = \frac{\binom{\text{Schiffe gesamt}}{\text{anvisierte Schiffe}} \cdot \binom{\text{Felder gesamt} - \text{Schiffe gesamt}}{\text{Schüsse gesamt} - \text{anvisierte Schiffe}}}{\binom{\text{Felder gesamt}}{\text{Schüsse gesamt}}} = \frac{\binom{30}{30} \cdot \binom{70}{0}}{\binom{100}{30}} \approx 3,4 \cdot 10^{-24}\%$ . In manchen Spielen kann es sogar zu 100 benötigten Angriffen kommen, bis alle Boote des Gegners versenkt sind. Diese sehr geringe Wahrscheinlichkeit zeigt, dass Zufallsangriffe in der Angriffsphase **ineffizient** sind und gemieden werden sollten.<sup>55</sup>

Vergleichbar zu den intuitiven Zufallsangriffen wäre eine Implementierung von **1er-Schiffen** in das Spiel. Anders als größere Boote passen sie in jede Lücke zwischen den längeren Schiffen. Da aber vor allem im Endspiel das Auffinden der 1er-Schiffe aufgrund ihres geringen Platzverbrauchs rein vom Zufall abhängt, werden diese Schiffe meist von den Spielern gemieden.

## b) Rand- und Eckschüsse

Anhand der Umfrage und den Flottengeneratoren 2.1 und 2.3 wird untersucht, ob **Rand- und Eckschüsse** gegen den jeweiligen Gegner effektiv genutzt werden können.

Laut der **Umfrage** wurden die **Ecken von Spielern** nur zu durchschnittlich 26% mit Booten besetzt. Da die mittlere Chance, ein Schiff auf der gesamten Seekarte zu treffen,  $P(\text{Treffer}) = \frac{30}{100} = 30\%$  ist, sind Schüsse auf die Ecken gegen menschliche Spieler weniger effektiv als auf anderen Felder. Auch festzustellen ist eine **leicht erhöhte Setzquote am Rand** mit 30,08%, welche im Vergleich zu den 30% des Durchschnittstreffers kaum eine Auswirkung auf die Trefferwahrscheinlichkeit hat.

Bei der Studie wurde auch ein **Verhaltensmuster** unter den Teilnehmern bemerkt: Obwohl nur wenige ihre Schiffe in die Ecken setzten, griffen trotzdem 44% der Admirale mindestens einmal die Ecken an, was eine deutliche Abweichung von den 26% der Eckplatzierungen darstellt. Dabei stellt sich die Frage, ob die Menschen über die Angriffsmuster des Gegners nachdenken und deshalb versuchen, den Angriffen mit einer vermiedenen Eckplatzierung auszuweichen. Mit den bestehenden Studieninformationen kann dazu noch kein Entschluss gefasst werden. Außerdem sollte die geringe Anzahl der Umfrageteilnehmer beachtet werden, weshalb die Ergebnisse nur teilweise interpretierbar sind und nur ein ungenaues Maß darstellen. Anzuzweifeln ist darum, ob die Studie für alle Spieler repräsentativ ist.

---

<sup>54</sup> Vgl. Berry, *Battleship*, <https://www.datagenetics.com/blog/december32011/>, Letzter Zugriff am 07.09.2022

<sup>55</sup> Vgl. Ebd.

Anders verhält sich das **Programm 2.1**, das die Flotte absteigend setzt. Hier wurden die **Ecken** zu durchschnittlich 29,63%, also minimal geringer als die Durchschnittstrefferquote, besetzt. Eher auffällig dagegen sind die Kästchen, die jeweils **zwei Felder vom Eck entfernt am Rand** liegen, da hier die Wahrscheinlichkeit für einen Treffer dem mittleren Wert von 34,62% entspricht. Erklärt werden kann dieses Verhalten durch die bereits in der Dokumentation erklärte Vorhersehbarkeit der Generatoren. Diesen „Berechnungsfehler“ zu annullieren, wäre mit dem Flottengenerator kaum umsetzbar.

hn(E)	A	B	C	D	E
1	30	33	35	34	33
2	34	27	28	27	27
3	35	28	29	29	28
4	35	27	29	29	28
5	34	26	28	28	28
6	33	26	28	28	28

Abbildung 12: An diesen Feldern werden besonders häufig Schiffe positioniert

Offensichtlicher ist die Generierung der **Anwendung 2.3**, in der die Boote aufsteigend platziert werden, Fehlerdurchläufe aber aussortiert sind. Signifikant ist die erhöhte Wahrscheinlichkeit auf einen Ecktreffer mit 36,94%, wobei auch Schüsse auf die Kästchen am Rand, die zwei Felder vom Eck distanziert liegen, mit einer mittleren Wahrscheinlichkeit von 43,95% häufiger erfolgreich sind.

Diese Eck- und Randanalyse lässt sich, auf den Gegner typ angepasst, **als Strategie effektiv anwenden**. Nicht nur erhöht man hiermit vor allem gegen Computer die Wahrscheinlichkeit auf gelungene Schüsse, es lässt sich auch die eigene Flotte bei der Verteilung mit diesen Erkenntnissen anpassen.

### c) Einzeichnen der Fehlschüsse und Schiffsanzahlanalyse

Weitere Strategien sind das Eintragen der Fehlschüsse auf der Seekarte und die Vorhersage der Schiffslänge in Abhängigkeit von den bereits gesichteten Schiffen. Dazu wird das **Beispiel** in der nebenstehenden Abbildung untersucht:

Die schwarzen Kästchen sind bereits versunkene Schiffe, während die dunkelblauen Felder **automatisch mit „Wasser“ ausgefüllt** wurden. Aufgrund der Regel, ein Kästchen zwischen Booten freizuhalten, kann man die Flächen als **Nicht-Schiffs-Felder** markieren. Hellblaue Kästchen wurden bereits zuvor vom Spieler als Wasser gekennzeichnet.

	A	B	C	D	E	F	G	H	I	J
1	.	■	.	■	.	.	■	■	.	.
2	.	■	.	■	■	.	.	.	.	.
3	.	■	.	?	?	?	x	.	■	.
4	.	■	.	.	■	.	?	.	■	.
5	.	.	.	.	.	.	?	.	■	.
6	■	.	.	■	■	.	?	.	■	.
7	■	.	.	.	.	.	?	.	.	.

Abbildung 13: Beispielsituation. Hier werden zwei Strategien angewendet

In dieser Situation wurde auf dem Feld **G-3** ein Schiff „x“ getroffen, wobei die Ausbreitungsrichtung sowie die Länge des Schiffs noch unklar sind.

Durch die bereits zuvor zerstörten Boote lassen sich die Richtungen nach oben und nach rechts ausschließen, da dort automatisch Wasser gesetzt wurde. Nach **links** sind noch drei weitere Felder unbesetzt, nach **unten** noch vier, welche mit „?“ hervorgehoben sind. Daher ist es möglich, die zuvor vier potenziellen Richtungen auf zwei zu beschränken, wodurch die Wahrscheinlichkeit, beim nächsten Treffer bei einem Angriff auf ein Feld neben dem „x“ richtig zu liegen, **von 25% auf 50% ansteigt**.<sup>56</sup>

Sieht man sich die bereits gefundenen Boote an, wird festgestellt, dass zwei 4er- und zwei 2er-Kreuzer gesetzt wurden. Somit begrenzt sich die Anzahl der bis hier unentdeckten Schiffe auf ein 5er-, drei 3er- und zwei 2er-Boote auf dem Spielfeld. Das hier getroffene Schiff kann kein 4er-Schiff sein, weshalb das Feld D-3 für dieses Boot bereits ausgeschlossen werden kann.

Werden die **in die Lücken einsetzbaren Schiffe zusammenaddiert**, kann man in die linke Richtung somit 5 Schiffe passen, nach unten jedoch sogar 6 Schiffe.<sup>[Zum Verständnis: 57]</sup>

Daraus ergibt sich die Wahrscheinlichkeit, bei dem nächsten Angriff auf ein Kästchen neben dem „x“ richtig zu liegen nach unten  $P(\text{Treffer unten}) = \frac{6}{5+6} = \frac{6}{11} \approx 54,54\%$  und nach links  $P(\text{Treffer links}) = \frac{5}{11} \approx 45,45\%$ .

Dank der Analyse dieser Beispielssituation konnte die ursprüngliche Wahrscheinlichkeit **von 25%** bei vier möglichen Richtungen **auf die ca. 54,54% nach unten** mit dem Einzeichnen des Wassers und der Schiffsanzahlanalyse erhöht werden.

#### d) Jagd- und Zielmodus

Der „**Hunt- and Target-Mode**“, wie ihn Nick Berry in seinem Blog beschreibt, ist eine von Admiralen oft intuitiv gewählte Spielstrategie.

Im **Jagdmodus** versucht der Spieler einen Kreuzer auf der Seekarte durch einen Schuss aufzudecken. Trifft er ein Boot, so geht er in den **Zielmodus** über, in dem er die direkt anliegenden Kästchen in alle vier Richtungen beschießt, um das versteckte Schiff zu zerstören. Beim Versenken des Schiffs geht der Angreifende danach wieder in den Jagdmodus über.<sup>58</sup>

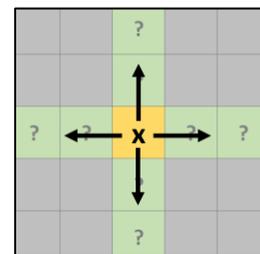


Abbildung 14: Die Angriffsrichtungen im Zielmodus

In dem Programm des Bloggers funktioniert diese Strategie, da er nur 17 Schiffskästchen statt 30 Kästchen wie im normalen Spiel benutzt, weshalb seine

<sup>56</sup> Vgl. Hesse, *Spielend gewinnen*, S. 61

<sup>57</sup> Nach links: 3x3er, 2x2er → summiert 5. Nach unten: 1x5er, 3x3er, 2x2er → summiert 6.

<sup>58</sup> Vgl. Berry, *Battleship*, <https://www.datagenetics.com/blog/december32011/>, Letzter Zugriff am 07.09.2022

durchschnittliche Aufdeckrate  $P(\text{Aufdeckrate}) = \frac{17}{100} = 17\%$  beträgt. Geht man davon aus, das Feld am Anfang oder Ende eines Schiffs getroffen zu haben, ist die Wahrscheinlichkeit 25%, die korrekte Ausbreitungsrichtung zu wählen, was zwar für Berrys Programm deutlich mehr als die Aufdeckrate ist, **der 30%-Trefferrate der üblichen Regeln aber signifikant unterliegt.**

Allerdings muss man bei einem erfolgreichen Schuss davon ausgehen, eventuell die **Mitte eines Bootes** getroffen zu haben. Gelingt ein Schuss, prüft man im Zielmodus die umliegenden Seiten mit Schüssen, welche bei einem Initialtreffer in der Mitte eines Schiffs **sogar zweimal erfolgreich** sein können.

Wie in der Abbildung 15 zu erkennen ist, gibt es bei einem 2er-Kreuzer keinen mittleren Schiffsabschnitt, bei einem Schiff der Länge 3 oder größer schon. Diesen mittleren Abschnitt zu treffen, ist wahrscheinlicher, je länger das Schiff ist und lässt sich mit der Formel

$$P(E_{\text{Mitteltreffer}}) = \frac{\text{Schiffslänge} - 2}{\text{Schiffslänge}}$$

berechnen, wobei  $E_{\text{Mitteltreffer}}$  das Ereignis darstellt, die Mitte zu treffen. Daraus erhält man folgende Ergebnisse zu  $P(E_{\text{Mitteltreffer}})$ :

Länge	$P(E_{\text{Mitteltreffer}})$	$P(E_{\text{Richtige Richtung}})$
2er	0%	25%
3er	33,3%	33,3%
4er	50,0%	37,5%
5er	60,0%	40%
...	...	...

Abbildung 16: Wahrscheinlichkeiten eines Treffers in der Mitte eines Schiffs

Grafisch lässt sich die Berechnung der **Wahrscheinlichkeit**, die **korrekte Ausbreitungsrichtung** nach dem Initialschuss zu entdecken, mit einem Baumdiagramm einfach darstellen:

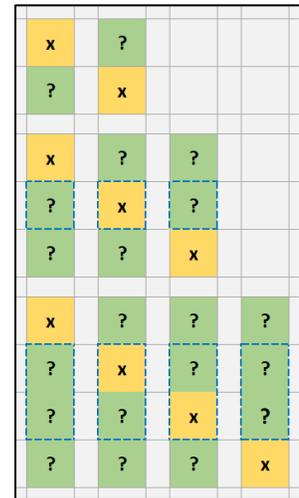


Abbildung 15: Mögliche vertikale Aufstellungen von 2er-, 3er- und 4er-Schiffen

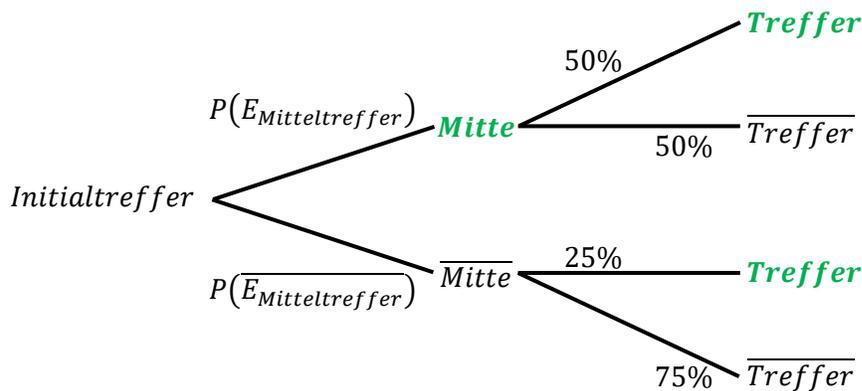


Abbildung 17: Baumdiagramm zur Berechnung von  $P(E_{\text{richtige Richtung}})$

Die Wahrscheinlichkeit des Ereignisses, die **korrekte Richtung** zu entdecken, wird somit mit der Formel

$$P(E_{\text{richtige Richtung}}) = P(E_{\text{Mitteltreffer}}) * 0,5 + P(\overline{E_{\text{Mitteltreffer}}}) * 0,25 \text{ errechnet,}$$

wobei  $P(\overline{E_{\text{Mitteltreffer}}}) = 1 - P(E_{\text{Mitteltreffer}})$  ist.

Nach weiterem Umformen der Gleichung folgt  $P(E_{\text{richtige Richtung}}) = \frac{\text{Schiffslänge}-1}{2 * \text{Schiffslänge}}$ , die eine **gebrochen rationale Funktion** ist.

Eine **weitere Erklärung** zu diesem Zusammenhang und die **Vertiefung** in dessen Bedeutung befindet sich im Anhang 3. *Vertiefung der Formel des Jagd- und Zielmodus*. Die **Ergebnisse** der Rechnungen sind in der Abbildung 16 aufgelistet.

Um diese Erkenntnisse nutzen zu können, muss man **zuvor** die bereits besprochene **Schiffsanzahlanalyse** anwenden, damit die Anzahl und die möglichen Längen der Boote annähernd bestimmbar werden. Aus der Tabelle erkennt man dennoch generell, dass die Verwendung des Zielmodus hilfreich ist, wenn weniger Zerschiffe als andere Bootslängen verborgen sind, damit eine größtmögliche Wahrscheinlichkeit zur Aufklärung der exakten Ausbreitungsrichtung zu erhalten ist.

### e) Diagonale Angriffe und Parität

Rund ein Viertel der Studienteilnehmer griffen in einem **Kreuzmuster** an, mit dem sie das **Spielfeld in vier Teile aufteilen**.<sup>59</sup> Diese Strategie hat einige Vorteile:

<sup>59</sup> Vgl. Brown, *How to Win at Battleship*, <https://www.thesprucecrafts.com/how-to-win-at-battleship-411068>, Letzter Zugriff am 07.09.2022

Zum einen beginnt der Angreifende, die mittleren und damit die **oft besetzten Felder**<sup>60</sup> **zu kontrollieren**. Zum anderen teilt er das Spielfeld durch die diagonalen Angriffe in vier Teilbereiche ein, wodurch sich der Spieler besser auf der Seekarte orientieren kann.

Wird diese Strategie selbst mit Abständen zwischen den einzelnen Schüssen bis in die Ecken fortgesetzt, sollte man bereits die **meisten Kreuzer aufgedeckt** haben.<sup>61</sup> Ein möglicher diagonalen Angriff ist in der Abbildung 18 dargestellt, wobei die eingetragenen Zahlen der Wichtigkeit der Schüsse entsprechen. In der Mitte können aufgrund der Nähe zu anderen Angriffen zwei „5er“-Schüsse ausgelassen werden.

	A	B	C	D	E	F	G	H	I	J
1	3									3
2		1							1	
3			2					2		
4				4			4			
5					5					
6						5				
7				4			4			
8			2					2		
9		1							1	
10	3									3

Abbildung 18: Die Wichtigkeit der Schüsse beim diagonalen Angriff

Möchte man das Muster weiter ausbreiten, empfiehlt sich die Benutzung eines „**Schachbrettmusters**“, das aus vielen Feldern mit **unterschiedlichen Paritäten**<sup>62</sup> besteht. Mit dieser Taktik werden zu verbrauchende Schüsse gespart, da Schiffe **nur horizontal oder vertikal** mit jeweils minimal zwei Feldern Länge versteckt sein können. Zudem kann es einfach an das diagonale Muster angepasst und weiterverwendet werden.<sup>63</sup>

Das Muster lässt sich im Spielverlauf **auf die zu suchende Bootsgröße anpassen**, wie es in der nebenstehenden Grafik zu sehen ist. Beispielsweise wird im oberen Bereich nach 2er-Schiffen gesucht. Im späteren Spiel ist es oftmals von Nöten, das Muster auf größere **Abstände auszuweiten**, weshalb in der Abbildung mittig auf 3er- und unten auf 4er-Boote geschossen wird.<sup>64</sup>

x	1	x	x	x	x	x	x	x	x	x
1	x	1	x	x	x	x	x	x	x	x
x	1	x	x	x	x	x	x	x	x	x
x	1	2	x		x		x			
	x	1	2	x		x		x		
		x	1	2	x		x		x	
x	1	2	3	x			x			
	x	1	2	3	x			x		
		x	1	2	3	x			x	

Abbildung 19: Verschiedene Anwendungen des Schachbrettmusters

Mit dieser Strategie lassen sich allerdings **keine 1er-Kreuzer** aufdecken, da man hierbei die triviale Methode eines Angriffs entlang der Reihen verwenden müsste, die dem Konzept der Parität keinesfalls entspräche.

<sup>60</sup> Vgl. Alemi, *The Linear Theory of Battleship*, <http://thevirtuosi.blogspot.com/2011/10/linear-theory-of-battleship.html>, Letzter Zugriff am 07.09.2022

<sup>61</sup> Vgl. Hesse, *Spielend gewinnen*, S. 61

<sup>62</sup> Vgl. De-Academic, *Parität (Mathematik)*, <https://de-academic.com/dic.nsf/dewiki/1077848>, Letzter Zugriff am 07.09.2022

<sup>63</sup> Vgl. Berry, *Battleship*, <https://www.datagenetics.com/blog/december32011/>, Letzter Zugriff am 07.09.2022

<sup>64</sup> Vgl. wikiHow, *Bei Schiffe versenken gewinnen*, <https://de.wikihow.com/Bei-Schiffe-versenken-gewinnen>, Letzter Zugriff am 07.09.2022

## VII. Abschließende Worte

Abschließend werden die gesammelten Erkenntnisse und Schlussfolgerungen der Seminararbeit erläutert.

Die **Unterschiede in den Verhaltensmustern** zwischen den Teilnehmern der Umfrage und den Flottengeneratoranwendungen kristallisierten sich hauptsächlich in der Dokumentation des Mensch-Computer-Vergleichs und der Angriffsstrategie der Rand- und Eckschüsse heraus. Auch in anderen Strategien war die umfassende Analyse sehr hilfreich und oft anwendbar.

Stark beeinflussbar ist die Qualität der **eigenen Flottenverteilung** durch zufällige Schiffsplatzierungen oder der richtigen Verwendung der Rand- und Eckfelder. In der **Angriffsphase** dagegen führen erheblich mehr Taktiken zu einer erhöhten Gewinnwahrscheinlichkeit:

Während Zufallsangriffe ineffektiv sind, bringen einfache Tricks, wie das Einzeichnen des Wassers nach dem Versenken eines Bootes oder diagonale Angriffe am Anfang des Spiels einen prägenden Vorteil. Die komplexeren Strategien der Schiffsanzahlanalyse oder der Parität, die vor allem in der Kombination mit dem Kreuzmuster Wirkung zeigt, haben ebenfalls ein deutliches Gewicht auf den Spielverlauf. Selbst der Jagd- und Zielmodus konnte sich nach der genauen mathematischen Untersuchung als verwendbar und überaus empfehlenswert beweisen.

Bei der denkbaren Implementierung von **1er-Booten** in das Schiffeversenken ist auffällig, wie sehr das Spiel mehr von Glück abhängig ist und die besprochenen Strategien ihre Erfolgchancen verlieren. Schiffe versenken ist ein **Glücksspiel**, da die gegnerische Seekarte nie aufgedeckt ist und Schüsse meist mit Glück ohne eine Vorahnung auf die Zielkoordinaten gefeuert werden müssen. Dennoch bringen die genannten Strategien, ob in der Verteidigung oder im Angriff, einen erheblichen, kaum zu vernachlässigbaren Vorteil für die **Gewinnwahrscheinlichkeit** in den Partien mit sich.

Anhand der Erkenntnisse dieser Seminararbeit **widerspreche ich somit der Aussage**, „Schiffe versenken [wäre] [...] ein Glücksspiel [und es gäbe] [...] keine Taktik für Glücksspiele“<sup>65</sup> von Micaela Dörr auf der Ratgeber-Redaktion Helpster. Nicht nur ist diese **Behauptung erwiesenermaßen falsch**, die Qualität dieses Artikels ist dadurch auch anzuzweifeln.

---

<sup>65</sup> Dörr, *Schiffe versenken - diese Taktik verhilft Ihnen zum Sieg*, [https://www.helpster.de/schiffe-versenken-diese-taktik-verhilft-ihnen-zum-sieg\\_113630](https://www.helpster.de/schiffe-versenken-diese-taktik-verhilft-ihnen-zum-sieg_113630), Letzter Zugriff am 07.09.2022

## VIII. Dokumentation des Mensch-Computer-Vergleichs zum Schiffeversenken

Im Folgenden werden das menschliche Verhaltensmuster beim Schiffeversenken mit den Ausgaben eigens erstellter Computerprogramme analysiert und verglichen.

### 1. Erstellung des ersten Flottengenerators

Zuerst werden die Funktionsweise sowie die Zuverlässigkeit des „Flottengenerators“ als Software für den Vergleich überprüft.

#### a) Zielsetzung für das Programm

Das Ziel der Anwendung ist es, dem Benutzer eine **Karte mit zufällig generierten Booten** zu geben, wobei die Regeln des Schiffeversenkens dabei eingehalten werden müssen.

Dem Nutzer wird dabei die Möglichkeit gegeben, verschiedene **Variablen festzulegen**, um die Konsolenausgabe zu verändern. Beispielsweise sollten die einzelnen Schiffsanzahlen, aber auch die Größe der Spielkarte einstellbar sein. Aus den Eingaben folgend generiert das Programm die Flotte und es wird eine visuelle Hilfe in der Form einer **Karte** angezeigt.

Für die weiterführende Evaluation und Bewertung der Zuverlässigkeit der Software ist es von Nöten, dem Nutzer auf Nachfrage einen Output zu erzeugen, welcher bei jeder Iteration  $n$  der Anwendung summiert wird. Aus dieser Ausgabe sollen in Microsoft Excel mit sogenannten Heatmaps die **absoluten Häufigkeiten  $H_n(E)$**  und die **relativen Häufigkeiten  $h_n(E)$**  durch verschiedene Grautöne grafisch dargestellt werden.<sup>66</sup>



Abbildung 20: Beispielhafte Darstellung einer Heatmap

Wenn es möglich ist, mehrere Tausende Durchführungen des Programms zu realisieren und diese durch die Heatmaps auszuwerten, lässt sich dadurch eine sichere Aussage über die Funktionsfähigkeit des Programms treffen. Die höchste Zuverlässigkeit der Software wäre erreicht, wenn alle Kästchen zu einer annähernd gleichen relativen Häufigkeit  $h_n(E)$  bei mindestens 1000 Iterationen mit Booten belegt werden würden.

<sup>66</sup> Vgl. Nelles, *Excel 2013 im Controlling*, S. 844-858.

## b) Darstellung des Flottengenerators als Blackbox-System

Die Eingaben werden durch das Programm in einen Output in der Form einer Karte mit den eingetragenen Schiffen ausgegeben. Dieser Prozess lässt sich stark vereinfacht als „**Blackbox-System**“ abbilden, wie es in der nachfolgenden Abbildung zu sehen ist. Mittig befindet sich der Prozess Flottengenerator, der die Variablen von links als **Eingangsgrößen** in eine Konsolenanwendung nach rechts als **Ausgangsgrößen** umwandelt. Bestimmte **Störgrößen** von oben haben einen Einfluss auf das System; die **Restriktoren** von unten beschränken die Anwendung. Die erzeugte Karte mit den eingetragenen Schiffen wird bei jeder Durchführung des Programms summiert, in Excel in eine Grafik umgewandelt und als Heatmap dargestellt.<sup>67</sup>

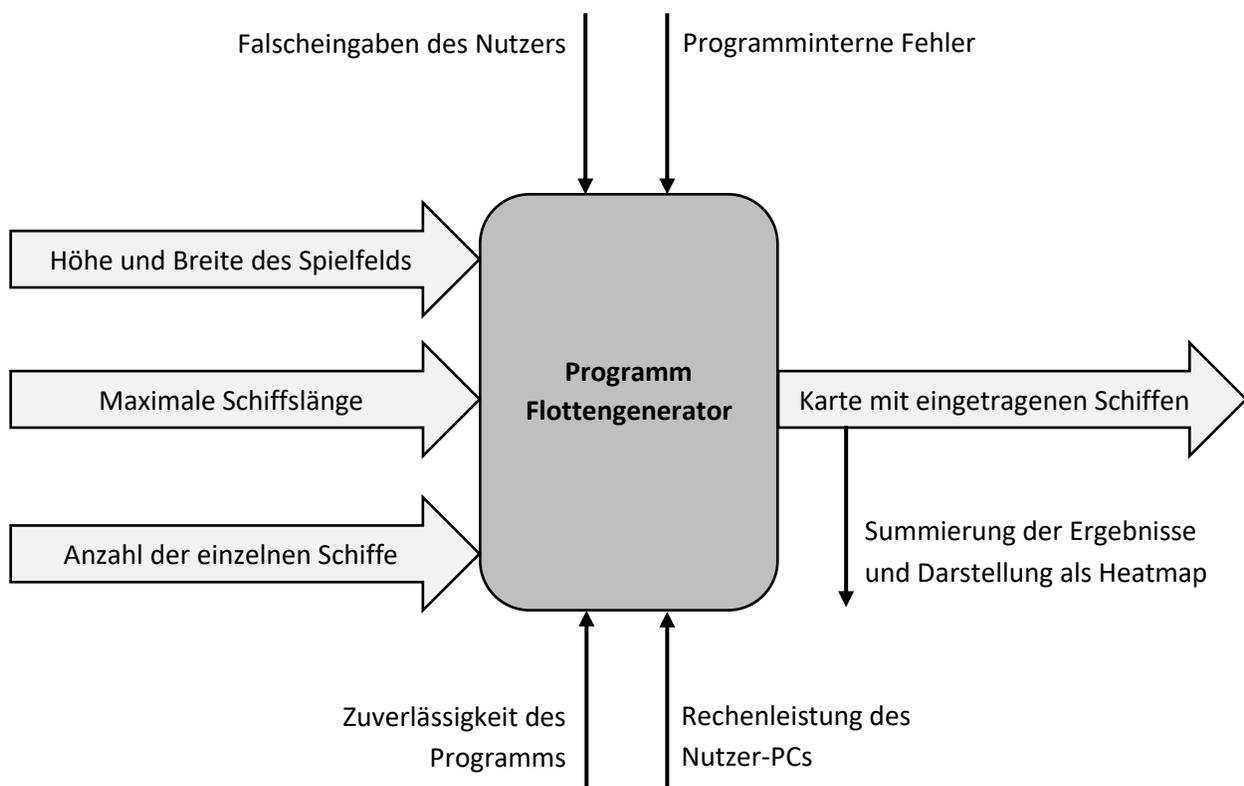


Abbildung 21: Blackbox-System des Flottengenerators

## c) Hypothese

In der Erarbeitung der Schiffe-versenken-Software soll die Hypothese geprüft werden, ob es möglich ist, mithilfe eines Computers einen fehlerfreien, regelkonformen und effizienten Generator für die Bootsplatzierung zu erstellen. Dabei wird die **Reliabilität**, also die Zuverlässigkeit, anhand **mehrerer Kriterien** geprüft:

<sup>67</sup> Vgl. Dworatschek, *Einführung in die Datenverarbeitung*, S. 48-53.

Die Fehleranfälligkeit wird mit der Häufigkeit der Anwendungsabstürze bestimmt. Nach mehrfachem Testen des Generators kann auch eine zuverlässige Aussage über die Einhaltung der Regeln getroffen werden, während die Effizienz der Software durch die Dauer der Berechnungszeit der einzelnen Arbeitsschritte und die Ausnutzung des CPUs im Task-Manager des PCs beobachtbar ist.

Zu erwarten sind Fehler im Flottengenerator, die im Verlauf der Erstellung der Anwendung behoben werden müssen. Außerdem könnte es dazu kommen, stärkere und schwächere Zonen auf der Heatmap zu erhalten, die durch eine ungenaue Programmierung entstehen. Einer nicht auffindbaren Lösung verschuldet, sind auch Fehlermeldungen, Abstürze des Programms oder nicht endende Softwareabläufe möglich.

Wenn es realisierbar ist, diese Probleme zu lösen, sollte man einen Flottengenerator erhalten, der völlig unvorhersehbar, aber regelkonform Schiffe auf der Karte platziert.

#### **d) Coding des Programms**

Das Coding erfolgte mit Java über das open-source Programmierwerkzeug Eclipse.<sup>68</sup> Die Anwendung zum Flottengenerieren sollte möglichst klar strukturiert sein, wodurch es folgendermaßen gegliedert wurde:

1. Initialisierung der Variablen
2. Abfrage der Höhe und Breite des Spielfelds
3. Ausgabe der Karte ohne die eingetragenen Boote
4. Eingabe der zu platzierenden Schiffe
  - 4.1 Abfrage der maximalen Schiffslänge
  - 4.2 Festlegung der Anzahl der jeweiligen Schiffslängen (Schleife)
5. Generierung der Schiffe auf der Karte (Schleife)
  - 5.1 Zufallsgenerator zur Festlegung der Startpositionen
  - 5.2 Zufallsgenerator für die Ausbreitungsrichtung der Schiffe von den Startpositionen
  - 5.3 Markierung der erzeugten Boote auf der Karte
6. Output der gesamten Karte
7. Zusammenfassung der einzelnen Ergebnisse zu einer Heatmap mit einem weiteren Java-Programm und in Excel.

Der gesamte Code zur Software des Flottengenerators ist auf GitHub über die Weiterleitung mit dem folgenden Link verfügbar: [https://drive.google.com/drive/folders/11ErKscIWEpD7DKMk0umpLK6J-1\\_1H2UU?usp=drive\\_link](https://drive.google.com/drive/folders/11ErKscIWEpD7DKMk0umpLK6J-1_1H2UU?usp=drive_link) (siehe XI. Literaturverzeichnis)

---

<sup>68</sup> Vgl. Eclipse Foundation, *Home: Eclipse*, <https://www.eclipse.org/>, Letzter Zugriff am 29.07.2022

## e) Die Konsolenausgabe beim Programmablauf

Folgende Ausgaben werden bei dem Start der Anwendung erzeugt:

```
Flotte generieren
Bitte geben Sie Anzahl der Felder in die Höhe an:
10
Höhe: 10

Bitte geben Sie Anzahl der Felder in die Breite an:
10
Breite: 10
```

Zuerst wird die **maximale Höhe und Breite** des Spielfelds vom Nutzer abgefragt und als Konsolennachricht als Feedback zurückgegeben.

```
0 | A B C D E F G H I J
-----
1 | . . . . . . . . . .
2 | . . . . . . . . . .
3 | . . . . . . . . . .
4 | . . . . . . . . . .
5 | . . . . . . . . . .
6 | . . . . . . . . . .
7 | . . . . . . . . . .
8 | . . . . . . . . . .
9 | . . . . . . . . . .
10| . . . . . . . . . .
```

Daraus wird die an die Eingaben angepasste **Karte generiert**, sodass dem Benutzer nach dem Input ein **Überblick** verschafft wird. Die Karte ist hier mit der linken Spalte von 1 bis 10 für die Höhe beschriftet. Die Breite wird mit Buchstaben von A bis J ausgegeben. Da die Schiffe noch nicht in die Karte eingetragen sind, befinden sich dort nur Punkte.

```
Bitte geben Sie die von Ihnen gewünschte maximale Schiffslänge eines Schiffes an:
5
Maximale Schiffslänge: 5
```

```
Bitte geben Sie die Anzahl der einzelnen Schiffslängen an:
Anzahl der 1er Schiffe:
0
0x 1er Schiffe
Anzahl der 2er Schiffe:
4
4x 2er Schiffe
Anzahl der 3er Schiffe:
3
3x 3er Schiffe
Anzahl der 4er Schiffe:
2
2x 4er Schiffe
Anzahl der 5er Schiffe:
1
1x 5er Schiffe

Überblick über die Schiffe:
2er Schiffe: 4
3er Schiffe: 3
4er Schiffe: 2
5er Schiffe: 1
```

Für die **Eingabe der Schiffsanzahlen**, muss festgelegt werden, wie viele Kästchen das größte Boot belegen soll.

Der Nutzer kann nun die Anzahl der einzelnen Schiffslängen festlegen und erhält einen **Überblick** über die getätigten Eingaben.

Abbildung 22: Konsolenausgabe des Flottengenerators

```

Die Schiffe werden nun auf zufälligen Positionen auf der Karte platziert.
6-6 10-8 3-5 4-8 8-6 9-10 5-10 2-7 9-3 1-4
0 | A B C D E F G H I J K L M N O P Q R
-----
1 | . . . x x . . . . .
2 | . . . . . x x . . .
3 | . x x x x . . . . .
4 | . . . . . . . x . x
5 | . . . . . . . x . x
6 | . x x x x x . x . .
7 | . . . . . . . . . x
8 | . . . . . x x x . x
9 | . x x . . . . . . x
10 | . . . . x x x x . .

```

Abbildung 23: Konsolenausgabe des Flottengenerators

Zum Schluss gibt das Programm die **Startpositionen der Schiffsgenerierung** aus, welche für die Überprüfung der Zuverlässigkeit des Programms benötigt wird.

Die erzeugte Karte enthält zudem die Schiffe, die als „x“ eingezeichnet wurden. Mit dieser Übersicht kann außerdem leicht eine Aussage über die Regelkonformität getroffen werden.

### f) Auswertung des Programms

Die Auswertung des Flottengenerators erfolgt mithilfe einer **Excel-Tabelle**. Die Kästchen, in denen sich Boote befinden, werden als „1“ anstatt einem „x“ markiert. In die Tabelle eingesetzt summiert Excel die Teilergebnisse zusammen, sodass die absoluten und relativen Häufigkeiten der Programmiterationen angezeigt werden.

Mit der bedingten Formatierungsregel **abgestufte Farbskala** ist es möglich, die Ausgaben in Abhängigkeit von den minimalen und maximalen Werten in Graustufen darstellen zu lassen. Die daraus entstehende Grafik wird als Heatmap bezeichnet und hilft, eine Übersicht über besonders häufig oder selten besetzte Kästchen zu erhalten.

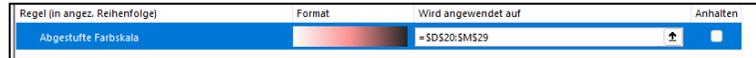


Abbildung 24: Die Formatierungsregel „abgestufte Farbskala“ in Excel

### g) Aussage über die Reliabilität des Flottengenerators

Aus den eingegebenen Werten ergeben sich die folgenden Tabellen zur absoluten Häufigkeit  $H_{200}(E)$  und der relativen Häufigkeit  $h_{200}(E)$ .

Das  $H_{200}(E)$ -Diagramm zeigt, wie oft ein Kästchen bei 200 Durchführungen durch ein Boot besetzt war. Daraus ergibt sich die Heatmap. Felder mit dunklen Flächen bedeuten eine hohe Anzahl von Schiffen; helle Flächen gelten für eine niedrige Menge von Schiffen, die sich bei  $n=200$  Wiederholungen jeweils auf den Kästchen befanden.

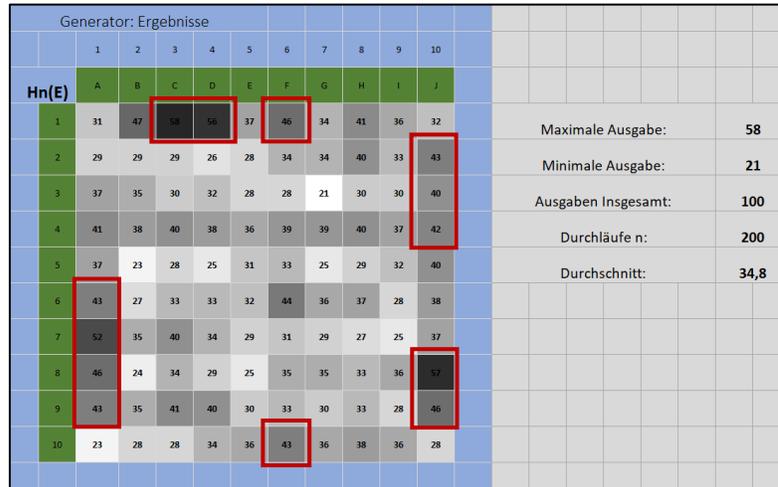


Abbildung 25: Heatmap zur absoluten Häufigkeit

Aus dem Muster der Karte ist erkennbar, dass **vor allem am äußersten Rand**, jedoch nicht in den Ecken selbst viele Boote platziert wurden, wie es mit den roten Rechtecken um diese Zonen markiert wurde. In der Mitte des Spielfelds ist die Verteilung der Schiffe im Vergleich zum Rand sehr ausgeglichen; nur wenige Stellen stechen dabei mit besonders vielen oder wenigen Schiffen heraus, die allerdings keinem erkennbaren Muster folgen.

Aus dem vorherigen Diagramm wird in Excel die  $h_{200}(E)$ -Grafik erstellt, die die relative Häufigkeit in Prozent angibt. Das Muster in der Grafik entspricht dem der Darstellung von  $H_{200}(E)$ , wobei die **Betrachtung der prozentualen Belegung** der Kästchen für einen Vergleich der verschiedenen Erhebungen mit unterschiedlichen Durchführungen  $n$  wichtig ist. So würde nur die Untersuchung der absoluten Häufigkeiten der Kästchenbelegung einen falschen Schluss zur Folge haben. Stattdessen müssen die relativen Häufigkeiten, die hier in Prozent angegeben sind, analysiert werden.

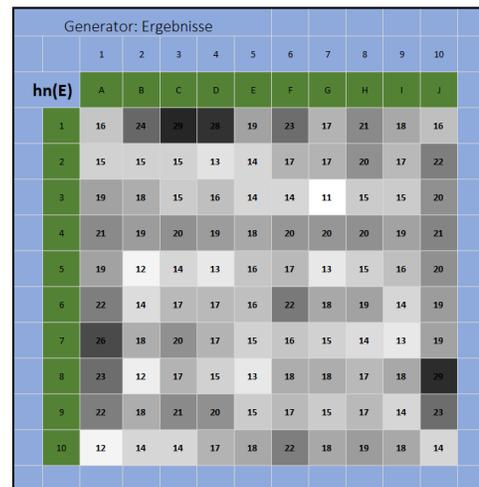


Abbildung 26: Heatmap zur relativen Häufigkeit

Da diese Anwendung nur 200-mal durchgeführt wurde, können aus den einzelnen Werten des **h<sub>200</sub>(E)**-Diagramms **nur die ungefähren Wahrscheinlichkeiten**, ob die Software in einem Durchlauf ein Schiff auf ein bestimmtes Kästchen setzt, abgeschätzt werden. Das Gesetz der großen Zahlen trifft somit hierauf nicht zu.<sup>69</sup>

Der Flottengenerator erfuhr bei den Ausführungen eine **sehr hohe Fehleranfälligkeit**. Da das Programm oft einfrore und nicht mehr weiter arbeitete, war es nicht möglich, viele Werte für die Auswertung ausgeben zu lassen. Zwar hielt der Generator alle festgelegten Regeln ein, die Effizienz verfehlte die Erwartungen aber weit. Regelmäßig war der leistungsstarke CPU „Intel® Core™ i7-8700“ des PCs, auf dem die Java-Anwendung lief, einer Auslastung von 95-100% ausgesetzt.

Daraus lässt sich schließen, dass der Flottengenerator zwar regelkonform funktioniert, jedoch die Fehler in der Programmierung oftmals zu unkontrollierbaren Softwareabstürzen und damit zur **Unbrauchbarkeit des Generators** zur Erstellung der Heatmaps führen. Dennoch ist die Anwendung in der Einzelnutzung ohne Wiederholungen zur Verteilung der Schiffe nützlich, wobei man beachten sollte, dass vor allem die in der Abbildung 25 rot markierten Kästchen öfter von Booten belegt werden.

---

<sup>69</sup> Vgl. Nawrotzki, *Lehrbuch der Stochastik*, S. 103-112

## 2. Studie zum Schiffeversenken

Für den Vergleich des Computers mit dem Menschen ist eine **Auswertung einer Umfrage** mit möglichst vielen Studienteilnehmern nötig. Aus diesem Grund wurde eine Befragung zum Schiffeversenken durchgeführt, um die **Verhaltensmuster** der Beteiligten zu erfassen.

### a) Zielsetzung der Umfrage

Das Ziel der Studie ist es, verschiedene Muster beim Platzieren der Boote oder beim Angriff auf den Gegner aufzudecken und anonym evaluiert als Heatmap grafisch darzustellen. Die Studie ist im Anhang der Seminararbeit einsehbar.

Dabei sollen die Teilnehmer der Studie drei verschiedene Aufgaben bearbeiten: Zuerst **verteilen** die Befragten eine bestimmte Anzahl an Schiffen auf einer Karte mit 10x10 Kästchen. Die genaue Anzahl der einzelnen Schiffe ist in der Abbildung 27 zu erkennen, die auch die Teilnehmer erhielten. Um die Umfrage mit dem Flottengenerator vergleichen zu können, müssen die gleichen Regeln eingehalten werden, wie in der Software.

Anzahl der Schiffe	Schiffsgröße
4x	2 □□
3x	3 □□□
2x	4 □□□□
1x	5 □□□□□

Abbildung 27: Die Anzahl der zu verteilenden Schiffe

Darauffolgend **greifen** die Teilnehmer **einen Gegner an**, indem sie zehn Schüsse auf eine 10x10 Karte verteilen. Da es bei der Studie keinen realen Gegner gibt, sondern die Spieler nur ihre zehn Angriffe ohne Rückmeldung tätigen, entspricht der Flottenangriff nicht dem eines echten Spiels. So erhalten sie kein Feedback, ob sie ein Boot oder das Wasser getroffen oder ein Schiff versenkt haben, was bei der Auswertung beachtet werden sollte. Zuletzt werden die Teilnehmer befragt, ob sie eine bestimmte **Strategie** beim Verteilen oder Angreifen der Flotten verfolgen.

Die Verteilungen der Schiffe sind nach den Befragungen **in eine Tabelle** in Excel **einzeln einzutragen**. Bestimmte Muster, die sich aus dem Setzen der Boote oder dem Angriff erkennen lassen, können dabei direkt notiert werden. Aus den festgehaltenen Werten werden in Excel Heatmaps zu den absoluten und relativen Häufigkeiten erstellt. Bei 50 oder mehr Befragungen sollte es möglich sein, eine sichere Aussage über die Verhaltensmuster der Teilnehmer für den Vergleich des Computerprogramms mit der Studie treffen zu können.

## b) Hypothese

Durch die Umfrage soll die Hypothese geprüft werden, ob Menschen beim Schiffeversenken bestimmte Verhaltensmuster aufweisen. Die Muster werden anhand mehrerer Kennzeichen untersucht:

Die grafische Darstellung der Heatmap zur absoluten Häufigkeit  $H_n(\mathbf{E})$  und zur relativen Häufigkeit  $h_n(\mathbf{E})$  wird aus den eingetragenen Werten der Umfrage generiert. Die **Fehlerquote** wird aus falschen oder unmöglichen Angaben der Teilnehmer bestimmt. **Strategien oder Auffälligkeiten** beim Flottenverteilen oder dem Angriff, wie beispielsweise die überwiegende Platzierung der Schiffe oder Schüsse an den Ecken oder am Rand des Spielfelds, werden auch analysiert.

Erwarten kann man eine **hohe Fehlerquote** unter den Befragten, da das Spiel Schiffe versenken im Vergleich zu anderen Gelegenheitsspielen oder Brettspielen weniger bekannt ist<sup>70, 71, 72</sup> und deshalb neue Spieler die Regeln erst verstehen müssen, wodurch Unrichtigkeiten öfter auftreten könnten. Zudem können **Muster** in den Heatmaps beim Setzen der Boote entstehen, die einen Aufschluss darüber geben, auf welchen Kästchen am meisten Schiffe platziert werden. Betrachtet man die Umfragen zum Angriff einzeln, sollte man Strategien, wie Eckschüsse, diagonale Angriffe oder Randtreffer erkennen können. Wenn es möglich sein sollte, 50 oder mehr Studienteilnehmer zu befragen, kann man mit den Ergebnissen eine annähernd sichere Aussage über die Verhaltensmuster der Menschen beim Schiffeversenken tätigen.

## c) Befragung der Versuchsteilnehmer

Die **anonyme Studie** erfolgte vom 11. bis 31. Juli 2022 in der FOSBOS Weiden. Dabei wurden Schüler, vorzugsweise aus unterschiedlichen Ausbildungsrichtungen und Lehrer der Schule, mithilfe von Blättern, auf welchen die Umfrage abgedruckt war, befragt. Bei Fragen zum Spiel oder zu den Regeln wurde den Teilnehmern geholfen; Vorschläge zum bestmöglichen Lösen der Aufgaben wurden allerdings nicht gegeben. Insgesamt beträgt die **Zahl der Befragten 50**, wodurch die Studienergebnisse als relativ sicher für das allgemeine Verhalten von Menschen beim Schiffeversenken verstanden werden können.<sup>73</sup> Dennoch wäre es erstrebenswert, weitere Forschung zu dem Setzverhalten der Spieler zu betreiben. Der Großteil der Umfragen befindet sich **in Papierform im Anhang**.

---

<sup>70</sup> Vgl. Baumgartner, *Die 35 BESTEN Brettspiele Klassiker*, <https://hiptoy.de/brettspiel-klassiker/#FAQ>, Letzter Zugriff am 25.07.2022

<sup>71</sup> Vgl. Gulz, *Top 10: Die beliebtesten Gesellschaftsspiele des Jahres 2021*, <https://www.ideal.de/magazin/freizeit/die-beliebtesten-gesellschaftsspiele-des-jahres?cmpReload=true>, Letzter Zugriff am 25.07.2022

<sup>72</sup> Vgl. Suhr, *Die beliebtesten Gesellschaftsspiele der Deutschen*, <https://de.statista.com/infografik/19742/umfrage-welche-gesellschaftsspiele-die-deutschen-zuhause-haben/>, Letzter Zugriff am 25.07.2022

<sup>73</sup> Stand vom 29.07.2022. Danach kamen keine weiteren Umfragen mehr zur Auswertung hinzu.

### d) Ergebnisse der Umfrage

Die Studie wurde mithilfe einer Excel-Datei ausgewertet, wobei die in die Tabellen eingetragenen Werte Heatmaps grafisch darstellten. Zudem wurden bei jeder einzelnen Umfrage Auffälligkeiten oder von den Teilnehmern niedergeschriebene Strategien notiert, woraus eine Auflistung der Fehler und Taktiken folgte.

Da für die Auswertung der Ergebnisse und den Vergleich mit dem Computer nur das  $h_{50}(E)$ -Diagramm nötig ist, wird die absolute Häufigkeit hier nicht weiter behandelt.

Betrachtet man die Heatmap der relativen Häufigkeit mit den Angaben in Prozent, so fallen v.a. die rot markierten Flächen auf, bei denen die Teilnehmer besonders oft, also ab 36% oder häufiger, Schiffe platzierten.

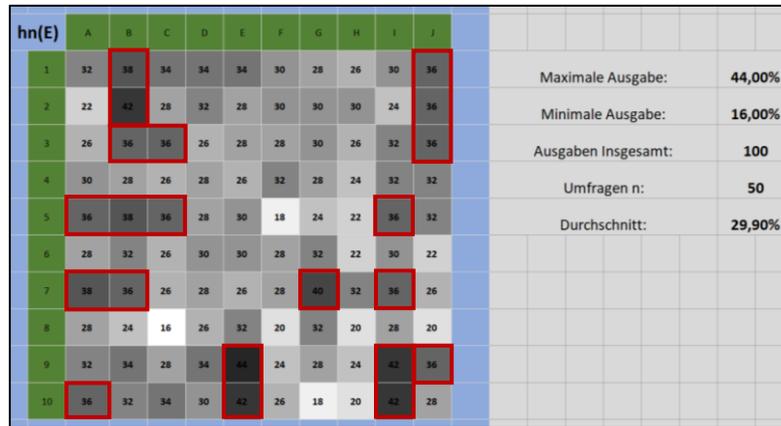


Abbildung 28: Heatmap zur relativen Häufigkeit

Legt man das **Diagramm breitflächiger** an, indem der durchschnittliche Prozentwert von jeweils vier Kästchen zu einer Kachel zusammengerechnet wird, ergibt sich eine noch übersichtlichere Darstellung der  $h_{50}(E)$ -Heatmap, wie es in der Abbildung 29 zu sehen ist.

Hier werden vor allem die dunklen Kacheln erkennbar, in denen mehr Boote platziert wurden. Anhand des Musters wird deutlich, dass **überwiegend am Rand und in den Ecken** Schiffe von den Befragten gesetzt wurden; die Mitte wurde gemieden.

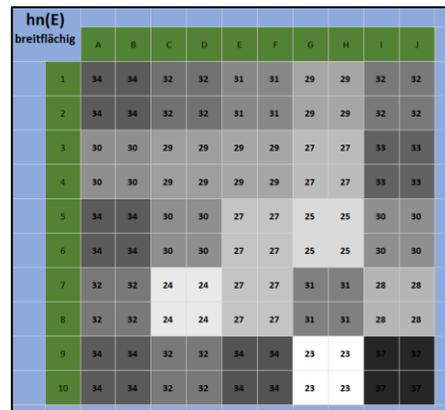


Abbildung 29: Heatmap zur breitflächigen relativen Häufigkeit

Untersucht man die abgegebenen Umfragen auf die **Fehlerquote**, ist eine **hohe Zahl an Unrichtigkeiten** festzustellen. Nach der Auswertung mit der Excel-Tabelle zur Studie machten 15 Personen insgesamt 49 Fehler beim Verteilen der Flotte.

Fehler	Ungültige Eingaben	0
	Insgesamte Fehler beim Flotte verteilen	49
	Insgesamte Fehler beim Angriff	0
	Anzahl der Personen mit mind. einem Fehler	15

Abbildung 30: Die Fehlerquote der Umfrage

Durchschnittlich ergibt das unter den Personen mit Unrichtigkeiten im Mittel  $\emptyset_{\text{Fehler pro Teilnehmer mit Unrichtigkeiten}} = \frac{\text{insgesamte Fehler}}{\text{Anzahl aller Teilnehmer mit Unrichtigkeiten}} = \frac{49}{15} \approx 3,3$  Fehler pro Teilnehmer mit Unrichtigkeiten.

Unter allen Teilnehmern der Umfrage waren es somit durchschnittlich

$\emptyset_{\text{Fehler pro Teilnehmer (alle)}} = \frac{\text{insgesamte Fehler}}{\text{Anzahl aller Teilnehmer}} = \frac{49}{50} \approx 0,98$  Fehler pro Teilnehmer. Wie bereits in der Hypothese vermutet wurde, ist die Fehlerquote mit fast einem Fehler pro Befragten sehr hoch.

Beim **Flottenverteilen** konnten einige **Auffälligkeiten** anhand der Abbildung 31 beobachtet werden: 32 der 50 Teilnehmer, also  $h_{50}(\text{Eckplatzierung}) = \frac{32}{50} = 64\%$ , positionierten mindestens ein Schiff in die Eckkästchen der Karte.

Strategien		
Flotte verteilen	Schiffe in die Ecken platziert	32
	Schiffe nicht in die Ecken platziert	18
	Schiffe v.a. waagrecht	5
	Schiffe v.a. senkrecht	3
	Schiffe v.a. am Rand platziert	34
	Schiffe v.a. mittig platziert	16

Abbildung 31: Strategien beim Verteilen der Flotte

Etwa  $h_{50}(v.a. \text{waagrecht}) = \frac{5}{50} = 10\%$  setzten ihre

Boote besonders oft **waagrecht**, während  $h_{50}(v.a. \text{senkrecht}) = \frac{3}{50} = 6\%$  die Schiffe übermäßig **senkrecht** verteilten. Bereits aus der breitflächigen Heatmap zur relativen Häufigkeit deutbar, setzten die meisten Befragten mit  $h_{50}(v.a. \text{Rand}) = \frac{34}{50} = 68\%$  die Boote vor allem direkt am **Rand** verlaufend oder ihn berührend. Dagegen stellten  $h_{50}(v.a. \text{mittig}) = \frac{16}{50} = 32\%$  die überwiegend in die **Mitte** des Spielfeldes.

Beim **Angriff** auf die Flotten konnten auch einige Auffälligkeiten festgestellt werden. Da die Befragten bei den einzelnen Angriffen kein Feedback erhielten, müssen die Schüsse auf den „imaginären Gegner“ als Blindschüsse verstanden werden.

Flotte angreifen	Schüsse in die Ecken	22
	Schüsse nicht in die Ecken	28
	Schüsse v.a. am Rand	20
	Schüsse v.a. mittig	19
	Schüsse v.a. diagonal	13

Abbildung 32: Strategien beim Angriff auf Flotten

Obwohl 64% der Spieler ein Boot in mindestens eine Ecke platzierten, schossen nur  $h_{50}(\text{Eckschüsse}) = \frac{22}{50} = 44\%$  in ein oder mehr Eckkästchen. Auffällig waren bei  $h_{50}(\text{v. a. Randschüsse}) = \frac{20}{50} = 40\%$  der Teilnehmer die häufig vorkommenden Angriffe an den Rand des Spielfelds, wobei jedoch 68% ihre Boote dorthin platzierten.

$h_{50}(\text{Diagonale Angriffe}) = \frac{13}{50} = 26\%$  der Befragten griffen in einem Muster an, das einer **Diagonalen** ähnelte. So verliefen die Schüsse dabei wie ein großes „X“ auf der Karte, das das Spielfeld in vier Zonen aufteilt. Nur wenige gaben ein Muster an, das genau dem des Beispiels aus der Abbildung 33 ähnelte, weshalb auch Näherungen daran als diagonaler Angriff verstanden wurden.

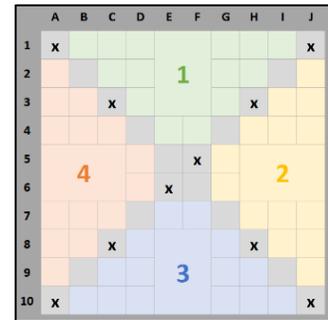


Abbildung 33: Beispiel eines diagonalen Angriffs mit einer Einteilung in vier Zonen

### 3. Interview zur Behebung der Problematik des ersten Flottengenerators

Da der erste Flottengenerator zur Erstellung der Heatmaps unbrauchbar war, sollte eine neue Software programmiert werden. Mit meinen Kenntnissen in der Java-Programmierung würde die Software jedoch ähnliche Fehler und Ungenauigkeiten im Coding enthalten, da die Anwendung erneut sehr unübersichtlich wäre.

Aus diesem Grund ersuchte ich eine **Expertenmeinung eines Fachinformatikers für Anwendungsentwicklung** auf, um diese Problematik zu beheben, sodass ich auf den mir bereits durch ein Praktikum bekannten Fachinformatiker, Herrn Carrasco Luis stieß. Ich sendete ihm bereits im Voraus die Dateien des Flottengenerators zu, damit er einen Überblick erhält. Anschließend führten wir einen Anruf über Teams, um die Probleme zu klären.

**Im Anhang** befindet sich die **Mitschrift** des Anrufs zwischen Herrn Carrasco und mir vom 28. Juli 2022. Das Video befindet sich in der Onlineversion der Seminararbeit in dem Ordner **3 Interview OOP**.

Herrn Carrasco zufolge sollte die Software mit **OOP**, also einer objektorientierten Programmierung, erfolgen. Die Vorteile davon seien die deutlich erhöhte Übersichtlichkeit, die Einteilung der Software in mehrere Arbeitsschritte und damit die Möglichkeit der Wiederverwendung der Anwendung.<sup>74, 75</sup>

---

<sup>74</sup> Vgl. Carrasco, *Interview zur Behebung der Problematik des ersten Flottengenerators*, am 28.07.2022

<sup>75</sup> Da ich nur wenig Wissen über diesen Teilbereich in der Programmierung in Java verfüge, war eine längere Einlesezeit für dieses Thema nötig. In der OTH-Bibliothek Weiden fand ich zahlreiche Werke zum Coding mit OOP in Java. Dieses Wissen konnte ich auch auf das Programmieren mit C# für den Flottengenerator übertragen.

## 4. Erstellung des zweiten Flottengenerators

Aufgrund der Unbrauchbarkeit des ersten Flottengenerators zur Erstellung der Heatmaps und der hohen Fehleranfälligkeit sollte ein **neues Programm mithilfe der OOP** erstellt werden. Nach einigen Anpassungen der Anwendung hat der zweite Flottengenerator **drei verschiedene Versionen**.

### a) Zielsetzung

Beim Coding der zweiten Software sollten, wie auch zuvor, die Regeln eingehalten werden. Hierbei ist es wichtig, dass der Generator eine **sehr niedrige bis keine Fehleranfälligkeit** besitzt. Gleichzeitig wird aber auch eine möglichst hohe Effizienz verlangt, weshalb das Programm eine niedrige Ausnutzung des CPUs des Nutzers und eine geringe Berechnungszeit haben sollte.

Dank der Programmierung mit OOP ist eine **höhere Übersichtlichkeit** im Programm zu erwarten, wodurch Bugs weniger oft vorkommen und unnötige Code-Wiederholungen im Code zu einer erhöhten Performance führen sollten.

### b) Das Coding und die Konsolenausgabe

Nach der Einholung einer Expertenmeinung lernte ich die objektorientierte Programmierung genauer kennen. Da ich davor bereits einige Lehrgänge zu C# besuchte, erstellte ich die zweite Anwendung in dieser Sprache mit Visual Studio 2022.

Die Gliederung der neuen Software entspricht grob der des vorherigen Generators, wobei nun **vermehrt Methoden aufgerufen** wurden. Der gesamte Code der drei Versionen des zweiten Flottengenerators ist unter dem weiterleitenden Forms-Link [https://drive.google.com/drive/folders/11ErKscIWEpD7DKMk0umpLK6J-1\\_1H2UU?usp=drive\\_link](https://drive.google.com/drive/folders/11ErKscIWEpD7DKMk0umpLK6J-1_1H2UU?usp=drive_link) einsehbar.

### c) Auswertung des Flottengenerators 2.1

Der zweite Flottengenerator zeigt eine **stärkere Leistung** als die erste Software auf:

In der Anwendung wird im Normalfall versucht, ein Schiff bei 500 Versuchen auf ein zufälliges Feld in eine von vier Richtungen zu setzen. Sollte es nicht positioniert werden können, so wird die aktuelle Durchführung trotz der fehlenden Schiffe in die Heatmap aufgenommen. Die zuvor sehr hohe Fehlerquote ist im neuen Programm bei **100.000 Wiederholungen** auf etwa ein Zehntel mit **2,07%** gefallen. Durch diese vernachlässigbar kleine Zahl können auch deutlich einfacher Heatmaps generiert werden, da mehr Iterationen hintereinander möglich sind.

Mithilfe einer kleinen Fehlerbehebung ist es mit dem zweiten Flottengenerator auch möglich, fehlerhafte Durchläufe zu überspringen, womit sich die Anwendung problemlos 10.000-mal oder mehr wiederholen lässt. Die Berechnungszeit ist mit ca. **70 Sekunden für 100 Wiederholungen** höher als bei der ersten Software, die Performance aber mit ungefähr 15% CPU-Auslastung pro geöffneter Anwendung deutlich geringer als zuvor. Aufgrund der hohen Effizienz des Programms lassen sich auch vier, ohne Hintergrundprogramme sogar fünf Konsolenanwendungen gleichzeitig ausführen, sodass die erhöhte Berechnungszeit damit wieder ausgeglichen ist.

Feststellen konnte man, dass das Programm manchmal keine Lösung finden konnte, was aber weniger an der Software lag, sondern an dem Spiel „Schiffe versenken“ selbst. Erkennbar aus der Abbildung 34, kann es selten zu Aufstellungen durch den Generator kommen, die ein weiteres **Setzen von noch fehlenden Schiffen nicht ermöglicht**. In der Abbildung fehlt beispielsweise ein 2er-Schiff, das aber aufgrund der Regeln mit einem Kästchen Abstand zu anderen Schiffen platziert werden müsste. Ob die unmöglichen Aufstellungen absichtlich in das Spiel eingebaut wurden, lässt sich mit der Heatmap zwar nicht klären, dennoch wirkt sich dieses Vorkommen aber verheerend auf die Berechnung mithilfe eines Programms aus.



Abbildung 34: Eine unmögliche Aufstellung der Schiffe durch den Generator

#### d) Auswertung des Flottengenerators 2.2

Der Flottengenerator 2.2 ähnelt dem ursprünglichen zweiten Generator, wobei nun die Schiffe in der **Reihenfolge von kurz nach lang** auf der Spielkarte platziert werden. In der Version 2.1 des Programms geschieht dies mit einer Anordnung, in der das Schiff mit der Länge 5 zuerst positioniert wird und die nächstkleineren darauf folgen.

Erstaunlicherweise ist die **Fehlerquote auf etwa 72,12% gestiegen**. Zu erklären ist das hohe Auftreten der unmöglichen Aufstellungen dadurch, dass die kleinen Schiffe, die zuerst platziert werden, den großen Schiffen später die Möglichkeit nehmen, platziert zu werden. Wie aus der Abbildung 35 ablesbar, werden hier beispielsweise bei **6500 Wiederholungen** des Programms vor allem 5er-, aber auch viele 4er-Schiffe nicht platziert.

Durchläufe:	<b>6500</b>
Fehleranzahl insgesamt:	<b>4688</b>
fehlende Schiffe: 1x5er	<b>4142</b>
fehlende Schiffe: 1x5er, 1x4er	<b>512</b>
fehlende Schiffe: 1x5er, 2x4er	<b>13</b>
Fehlerquote:	<b>72,12%</b>

Abbildung 35: Die hohe Fehlerquote und die dabei nicht platzierten Schiffe

Die Berechnungszeit ist wegen der oft auftretenden Fehler auf **ca. 8 Minuten bei 100 Iterationen** gestiegen, weshalb die Versuche, ein Schiff zu positionieren, aus Gründen des erhöhten Zeitanspruchs von 500 auf 300 angepasst werden mussten, wobei diese Änderung nur einen minimalen Einfluss auf das Ergebnis hat.

### e) Auswertung des Flottengenerators 2.3

Der Flottengenerator 2.3 unterscheidet sich nur in dem Umgang mit Fehlern mit dem Generator 2.2.

Die oft vorkommenden und nicht vernachlässigbar fehlerhaften Durchführungen werden nun nicht in die Analyse aufgenommen, sodass für den Nutzer eine **Heatmap ohne Fehler** ausgegeben wird. Falls das Programm nicht alle Schiffe auf die Spielkarte platzieren kann, wird die aktuelle Iteration zurückgesetzt und komplett wiederholt. Da bei einer Fehlerquote von ca. 72,12% sehr viele der Versuche, eine richtige Karte zu erstellen, scheitern, steigt die Berechnungszeit auf **ungefähr 20 Minuten pro 100 Wiederholungen**, wobei diese Version keine Fehler in der Schiffssetzung vorzufinden sind.

## f) Darstellungen der Heatmaps des zweiten Flottengenerators

Im Folgenden werden die Ergebnisse der drei Generatoren grafisch dargestellt. Zur besseren Lesbarkeit und Übersichtlichkeit werden die Grafiken auf drei Seiten aufgeteilt.

### Flottengenerator 2.1: Schiffsgröße absteigend und mit Fehlern

Bei 100.000 Iterationen des Programms liegt die Fehlerquote auf 2,07%. Da die Anwendung so oft wiederholt wurde und das Gesetz der großen Zahlen daher zutrifft<sup>76</sup>, können die relativen Häufigkeit jedes Kästchens als Wahrscheinlichkeit verstanden werden.

Flottengenerator 2.1										
Hn(E)	A	B	C	D	E	F	G	H	I	J
1	29747	33064	34570	34039	33453	33607	34350	34707	33626	29713
2	33722	26576	28189	27265	26527	26539	27309	28093	26644	33287
3	34854	27696	29468	29105	28210	28248	28986	29335	27693	34662
4	34543	27302	29217	29201	28404	28263	29401	29170	27553	34433
5	33581	26413	28167	28254	27502	27584	28216	27839	26466	33303
6	33294	26469	28043	28206	27687	27904	28469	28290	26607	33334
7	34263	27331	29110	28950	28545	28457	29029	29012	27237	34262
8	34474	27902	29505	28785	28300	28229	29085	29555	27945	34564
9	33396	27067	28424	27269	26583	26367	27128	28012	26905	33627
10	29299	33252	34000	34137	33578	33395	34298	34499	33330	29761

Flottengenerator 2.1										
hn(E)	A	B	C	D	E	F	G	H	I	J
1	30	33	35	34	33	34	34	35	34	30
2	34	27	28	27	27	27	27	28	27	33
3	35	28	29	29	28	28	29	29	28	35
4	35	27	29	29	28	28	29	29	28	34
5	34	26	28	28	28	28	28	28	26	33
6	33	26	28	28	28	28	28	28	27	33
7	34	27	29	29	29	28	29	29	27	34
8	34	28	30	29	28	28	29	30	28	35
9	33	27	28	27	27	26	27	28	27	34
10	29	33	35	34	34	33	34	34	33	30

Flottengenerator 2.1										
hn(E)	A	B	C	D	E	F	G	H	I	J
1	31	31	31	31	30	30	31	31	31	31
2	31	31	31	31	30	30	31	31	31	31
3	31	31	29	29	28	28	29	29	31	31
4	31	31	29	29	28	28	29	29	31	31
5	30	30	28	28	28	28	28	28	30	30
6	30	30	28	28	28	28	28	28	30	30
7	31	31	29	29	28	28	29	29	31	31
8	31	31	29	29	28	28	29	29	31	31
9	31	31	31	31	30	30	31	31	31	31
10	31	31	31	31	30	30	31	31	31	31

Abbildung 36: Heatmaps des Flottengenerators 2.1

<sup>76</sup> Vgl. Nawrotzki, *Lehrbuch der Stochastik*, S. 103-112

## Flottengenerator 2.2: Schiffsgröße aufsteigend und mit Fehlern

Bei 6500 Iterationen des Programms liegt die Fehlerquote bei 72.12%. Die Software wurde zwar ausreichend oft durchgeführt, die vielen Fehler verfälschen allerdings das Ergebnis signifikant, wie es der Vergleich zum Generator 2.3 auf der nächsten Seite zeigt. Aus diesem Grund sollte diese Version des Generators nicht weiter zur Analyse verwendet werden.

Flottengenerator 2.2										
Hn(E)	A	B	C	D	E	F	G	H	I	J
1	1629	1867	2134	2271	2066	2065	2207	2162	1809	1638
2	1787	1496	1497	1467	1389	1440	1466	1552	1447	1796
3	2149	1553	1584	1574	1537	1595	1562	1583	1504	2123
4	2242	1482	1605	1531	1513	1552	1495	1560	1464	2202
5	2024	1401	1561	1498	1459	1544	1465	1557	1451	2017
6	2074	1392	1576	1478	1464	1542	1459	1528	1426	2040
7	2215	1448	1568	1472	1504	1466	1511	1660	1492	2216
8	2170	1534	1649	1544	1574	1567	1586	1637	1515	2144
9	1837	1415	1586	1468	1406	1346	1427	1527	1435	1811
10	1702	1796	2162	2254	2055	2110	2285	2216	1880	1688

Flottengenerator 2.2										
hn(E)	A	B	C	D	E	F	G	H	I	J
1	25	29	33	35	32	32	34	33	28	25
2	27	23	23	23	21	22	23	24	22	28
3	33	24	24	24	24	25	24	24	23	33
4	34	23	25	24	23	24	23	24	23	34
5	31	22	24	23	22	24	23	24	22	31
6	32	21	24	23	23	24	22	24	22	31
7	34	22	24	23	23	23	23	26	23	34
8	33	24	25	24	24	24	24	25	23	33
9	28	22	24	23	22	21	22	23	22	28
10	26	28	33	35	32	32	35	34	29	26

Flottengenerator 2.2										
hn(E)	A	B	C	D	E	F	G	H	I	J
1	26	26	28	28	27	27	28	28	26	26
2	26	26	28	28	27	27	28	28	26	26
3	29	29	24	24	24	24	24	24	28	28
4	29	29	24	24	24	24	24	24	28	28
5	27	27	24	24	23	23	23	23	27	27
6	27	27	24	24	23	23	23	23	27	27
7	28	28	24	24	24	24	25	25	28	28
8	28	28	24	24	24	24	25	25	28	28
9	26	26	29	29	27	27	29	29	26	26
10	26	26	29	29	27	27	29	29	26	26

Durchläufe:	<b>6500</b>
Fehleranzahl insgesamt:	<b>4688</b>
fehlende Schiffe: 1x5er	<b>4162</b>
fehlende Schiffe: 1x5er, 1x4er	<b>512</b>
fehlende Schiffe: 1x5er, 2x4er	<b>14</b>
Fehlerquote:	<b>72,12%</b>

Abbildung 37: Heatmaps des Flottengenerators 2.2

### Flottengenerator 2.3: Schiffsgröße aufsteigend und ohne Fehler

Bei 5000 Iterationen des Programms liegt die Fehlerquote bei exakt 0%. Vergleicht man diese Heatmaps mit den Grafiken des Generators 2.2 stellt man bei den  $h_n(E)$ -Diagrammen fest, dass die Prozentwerte des fehlerhaften Flottengenerators niedriger sind als die der Version 2.3. Außerdem sind die farblichen Ausprägungen dieses Programms stärker, wodurch Muster einfacher zu erkennen sind.

Da 5000 Durchführungen für das Gesetz der großen Zahlen ausreichend sein sollten, können die relativen Häufigkeiten der Felder dieser Anwendung als Wahrscheinlichkeiten gesehen werden.<sup>77</sup>

Flottengenerator 2.3										
Hn(E)	A	B	C	D	E	F	G	H	I	J
1	1842	1782	2180	2147	2018	2070	2174	2193	1750	1863
2	1706	927	1138	943	894	899	914	1210	921	1755
3	2198	1265	1730	1450	1463	1490	1442	1650	1214	2169
4	2106	966	1461	1133	1207	1258	1169	1484	980	2141
5	1967	869	1470	1201	1270	1317	1188	1462	883	1989
6	2034	883	1480	1251	1349	1325	1271	1526	895	1980
7	2191	934	1429	1144	1195	1179	1115	1459	967	2150
8	2223	1208	1652	1518	1449	1482	1476	1696	1221	2197
9	1689	902	1201	959	868	890	962	1203	899	1758
10	1828	1787	2215	2166	2002	1993	2131	2206	1689	1855

Flottengenerator 2.3										
hn(E)	A	B	C	D	E	F	G	H	I	J
1	37	36	44	43	40	41	43	44	35	37
2	34	19	23	19	18	18	18	24	18	35
3	44	25	35	29	29	30	29	33	24	43
4	42	19	29	23	24	25	23	30	20	43
5	39	17	29	24	25	26	24	29	18	40
6	41	18	30	25	27	27	25	31	18	40
7	44	19	29	23	24	24	22	29	19	43
8	44	24	33	30	29	30	30	34	24	44
9	34	18	24	19	17	18	19	24	18	35
10	37	36	44	43	40	40	43	44	34	37

Flottengenerator 2.3										
hn(E)	A	B	C	D	E	F	G	H	I	J
1	31	31	32	32	29	29	32	32	31	31
2	31	31	32	32	29	29	32	32	31	31
3	33	33	29	29	27	27	29	29	33	33
4	33	33	29	29	27	27	29	29	33	33
5	29	29	27	27	26	26	27	27	29	29
6	29	29	27	27	26	26	27	27	29	29
7	33	33	29	29	27	27	29	29	33	33
8	33	33	29	29	27	27	29	29	33	33
9	31	31	33	33	29	29	33	33	31	31
10	31	31	33	33	29	29	33	33	31	31

Durchläufe:	5000
Fehler:	0

Abbildung 38: Heatmaps des Flottengenerators 2.3

<sup>77</sup> Vgl. Nawrotzki, *Lehrbuch der Stochastik*, S. 103-112

### g) Die Vorhersehbarkeit des Generators

Zwar werden bei den neuen Anwendungen sehr verlässlich und unvorhersehbar Boote auf die Seekarte platziert, bei der Betrachtung der Heatmap der relativen Häufigkeiten **am Beispiel der Version 2.1** stellt man jedoch eine **deutliche Ausprägung am Rand** fest.

Erklärt werden kann diese Abweichung, die einen maximalen Unterschied von  $\Delta P_{Ausgabe} = AusgabeMax - AusgabeMin = 34,85\% - 26,37\% = 8,49\%$  darstellt, durch die **vereinfachte Platzierung** eines Schiffs **am Rand des Spielfelds** aufgrund der nicht vorhandenen, bereits gesetzten Boote.

Möchte man ein 2er-Boot wie in der Abbildung 39 positionieren, so muss laut den Regeln je ein Kästchen in alle Richtungen unbesetzt sein. Da außerhalb der Seekarte, wo sich normalerweise die Nummerierung befindet, kein Schiff aufgestellt sein kann, werden diese mit „/“ markierten Kästchen nicht weiter beachtet. Daraus folgt, dass **nur die übrigen „?-Kästchen** für die Platzierung

**relevant** sind. Bei jedem dieser Felder wird geprüft, ob es durch ein anderes Schiff bereits in Anspruch genommen wurde, wodurch eine niedrigere Anzahl der „?-Kästchen zu einer höheren Wahrscheinlichkeit einer möglichen Positionierung des Bootes führt. Deshalb wird **am Rand eine höhere Setzwahrscheinlichkeit** gemessen als in anderen Bereichen des Spielfelds.

	A	B	C	D	E
	/	/	/	/	
1	?	x	x	?	
2	?	?	?	?	
3					
4		?	?	?	
5		?	x	?	
6		?	x	?	
7		?	?	?	
8					
9	/	?	?	?	
10	/	x	x	?	
	/	/	/	/	

Abbildung 39: Betrachtung des möglichen Umfelds bei der Schiffsplatzierung

## IX. Anhang

### 1. Das Interview mit Herrn Carrasco in Textform

Im Folgenden ist die Mitschrift des Anrufs zwischen Herrn Carrasco und mir am 28. Juli 2022 dargestellt. Änderungen an der Originalversion des Niedergeschriebenen, die wegen grammatikalischer Fehlern gemacht wurden oder zur besseren Lesbarkeit beitragen, wurden kenntlich gemacht.

**Völkl Simon:** *Hallo, sehr geehrter Herr Carrasco.*

**Herr Carrasco Luis, Fachinformatiker:** *Hallo Herr Völkl, es freut mich Sie kennenzulernen.*

**Völkl Simon:** *Nun, ich nehme an, Sie haben bereits die Dokumente zum Flottengenerator erhalten, [den] ich kreiert habe?*

**Herr Carrasco:** *Ja, natürlich. Darum soll es heute gehen?*

**Völkl Simon:** *Ja, wie Sie bereits [...] der E-Mail entnehmen konnten, stell[e] ich mir schon seit längerem die Frage, wie ich das Programm zum Generieren der Schiffe verbessern könnte. Das Problem ist nämlich, dass das Programm selbst nach Öfteren Überarbeiten immer noch Probleme oder [...] Fehler bei der Durchführung aufweist.*

**Herr Carrasco:** *Wie sehen denn diese Probleme in Ihrem Programm aus?*

**Völkl Simon:** *[...] Zuerst einmal sollte man verstehen, dass das Programm mindestens 1000, vielleicht sogar 10- oder 100-Tausend-mal durchgeführt werden sollte, sodass man [...] eine sichere Aussage über die Funktionsfähigkeit des Programms treffen [kann]. Das Problem ist nur, dass d[a]s Programm nicht so oft ausführbar ist, da es in dem Arbeitsschritt des Verteilens der Schiffe steckenbleibt. Sie haben bereits den Code gesehen: Hätten Sie [...] eine Idee, wie man dieses Problem lösen könnte?*

**Herr Carrasco:** *[...] Eine Möglichkeit dieses Problem zu lösen wäre die objektorientierte Programmierung zu verwenden. Wenn Sie diese Programmierung, also [...] [auch] OOP [genannt], [...] verwenden [...], könnten Sie das Programm viel übersichtlicher gestalten und dadurch deutlich schneller Ihren Fehler in diesem Programm finden.*

**Völkl Simon:** *Ja.*

**Herr Carrasco:** *Darüber hinaus kann aber auch, wenn Sie [...] dieses Programm in einzelne Arbeitsschritte einteilen, [...] dabei die Wiederverwendbarkeit deutlich gesteigert werden. Dadurch, dass Sie vielleicht diese [...] Schritte [...] auch in anderen Programmen nutzen können, bzw. Sie diese auch aufrufen können.*

**Völkl Simon:** *Ja, genau. D[a]s bräuchte ich dann [...] auch, ja. Okay, also mit OOP würde der Flottengenerator somit viel besser funktionieren, sagen Sie?*

**Herr Carrasco:** Richtig.

**Völkl Simon:** *Alles klar. Gut, dann [...] werde ich demnächst in der Bibliothek der Fachhochschule nach der Fachliteratur dazu sehen und versuchen, die Software neu aufsetzen.*

**Herr Carrasco:** Dann wünsche ich Ihnen dabei viel Erfolg!

**Völkl Simon:** *Ich danke Ihnen.*

**Herr Carrasco:** Wenn Sie irgendwelche Probleme haben [sollten], sagen Sie gerne Bescheid. Ich helfe Ihnen sehr gerne.

**Völkl Simon:** *Natürlich. Ich danke Ihnen sehr für dieses Interview und für die Hilfe, und für das Angebot natürlich auch. Und auf Wiedersehen!*

**Herr Carrasco:** Tschüss.

## **2. Die Umfragen in Papierform**

Anbei befinden sich die Umfragen der Studienteilnehmer in Papierform. Da manche Umfragen digital abgegeben wurden, ergibt die Gesamtzahl der Blätter weniger als die der eigentlich 50 Befragten.

### 3. Vertiefung der Formel des Jagd- und Zielmodus

Durch diese Vertiefung soll ein erweiterter Einblick in die wohl am mathematisch anspruchsvollste Strategie gegeben werden.

#### a) Herleitung der Formeln

In die ursprüngliche Formel  $P(E_{\text{richtige Richtung}}) = P(E_{\text{Mitteltreffer}}) * 0,5 + P(\overline{E_{\text{Mitteltreffer}}}) * 0,25$

(I) kann der dazugehörige Zusammenhang  $P(\overline{E_{\text{Mitteltreffer}}}) = 1 - P(E_{\text{Mitteltreffer}})$  (II) eingesetzt werden. Mit II in I folgt:

$$P(E_{\text{richtige Richtung}}) = P(E_{\text{Mitteltreffer}}) * 0,5 + [1 - P(E_{\text{Mitteltreffer}})] * 0,25$$

$$P(E_{\text{richtige Richtung}}) = P(E_{\text{Mitteltreffer}}) * 0,5 + 0,25 - 0,25 * P(E_{\text{Mitteltreffer}})$$

$$P(E_{\text{richtige Richtung}}) = 0,25 * P(E_{\text{Mitteltreffer}}) + 0,25 \text{ (III)}$$

$$\text{mit } P(E_{\text{Mitteltreffer}}) = \frac{\text{Schiffslänge} - 2}{\text{Schiffslänge}} \text{ (IV) in III:}$$

$$P(E_{\text{richtige Richtung}}) = 0,25 * \frac{\text{Schiffslänge} - 2}{\text{Schiffslänge}} + 0,25$$

$$P(E_{\text{richtige Richtung}}) = \frac{\text{Schiffslänge} - 2}{4 * \text{Schiffslänge}} + 0,25$$

$$P(E_{\text{richtige Richtung}}) = \frac{\text{Schiffslänge} - 2 + \text{Schiffslänge}}{4 * \text{Schiffslänge}}$$

$$P(E_{\text{richtige Richtung}}) = \frac{\text{Schiffslänge} - 1}{2 * \text{Schiffslänge}}$$

$$\text{oder verkürzt mit Schiffslänge } \hat{=} s: \mathbf{p(s)} = \frac{s - 1}{2 * s}$$

$P(E_{\text{richtige Richtung}})$  beschreibt damit die Wahrscheinlichkeit, mit einem Schuss in eine beliebige der vier möglichen Richtungen<sup>78</sup> des davor getätigten Initialtreffers erneut dasselbe Schiff erfolgreich zu beschießen. Dagegen ist  $P(E_{\text{Mitteltreffer}})$  die Chance, auf den Mittelteil eines Bootes mit dem ersten Treffer gestoßen zu sein. Die resultierende Gleichung, die nun nur noch von der Schiffslänge abhängt, entspricht einer **gebrochen rationalen Funktion**.

---

<sup>78</sup> Mögliche Richtungen sind hier *oben, unten, links und rechts*

## b) Analyse der Graphen

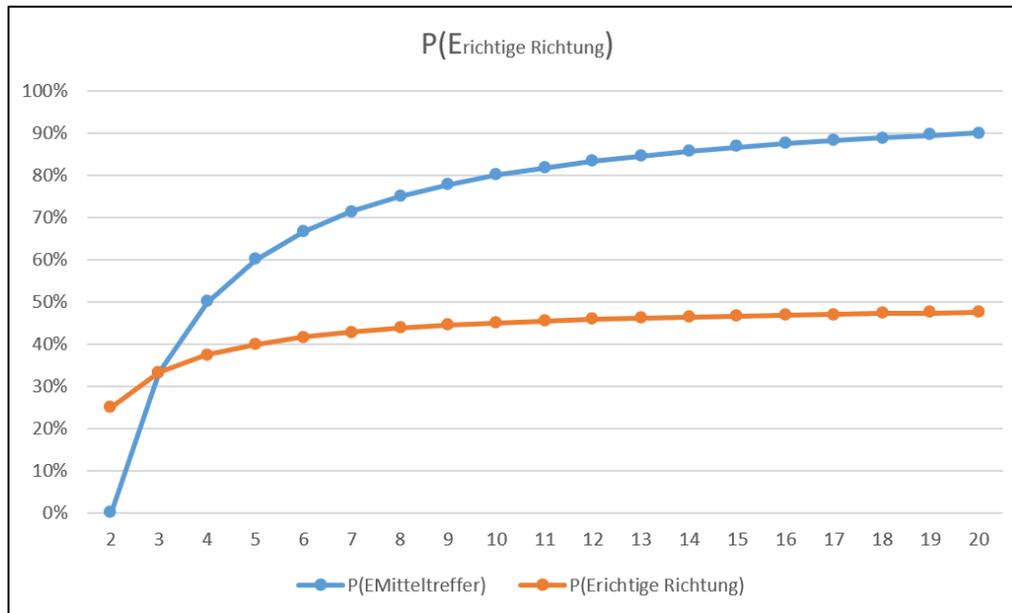


Abbildung 40: Diagramm zu Graphen des Jagd- und Zielmodus

Aus dem dargestellten Diagramm können einige nützliche Informationen herausgelesen werden:

**Unter einer Schiffslänge von 2** ist die **Formel  $p(s)$  nicht einsetzbar**, da beispielsweise  $p(0)$  mit dem Teilen durch null einen mathematischen Fehler liefert. Auch  $p(1) = 0$  ergibt wenig Sinn, weil man bei einem 1er-Schiff keine weiteren Kästchen des Bootes nach dem Initialtreffer abschießen kann.

Daraus und mit den Informationen aus den Graphen der Abbildung lässt sich schließen, dass die Verwendung des Jagd- und Zielmodus nur für Schiffe der Größe 2 oder länger von Nutzen ist. Aus diesem Grund wird der Graph  $G_p$  durch die **Definitionsmenge**, wie auch schon im Diagramm, auf  $D_p = [2; \infty[$  eingegrenzt. Es sollten auch nur ganze Zahlen, hier speziell **natürliche Zahlen** in die Formel  $p(s)$  eingesetzt werden, da Schiffe ausschließlich in einer integren Form vorkommen. Aus dem Graphen und weiteren Berechnungen kann auch das **Grenzverhalten** abgelesen werden.

Die Gleichung  $m(s)$  soll hier  $P(E_{Mitteltreffer})$  entsprechen, wobei  $s$  die aktuelle Schiffslänge darstellt.

$\lim_{s \rightarrow \infty} p(s) = 0,50^- \cong 50\%^-$ . Die Wahrscheinlichkeit, die richtige Richtung zu wählen, erreicht selbst bei einer annähernd unendlichen Schiffslänge **niemals 50%**, da eine waagrechte Asymptote bei  $y = 0,50$  vorliegt.

$\lim_{s \rightarrow \infty} m(s) = 1,00^- \cong 100\%^-$ . Die Chance auf einen Mitteltreffer nähert sich in der Unendlichkeit nur **knapp unter 100%** an, was auch durch eine waagrechte Asymptote bei  $y = 1,00$  zu erklären ist.

## X. Abbildungsverzeichnis

Abbildung 1: Eine kommerzielle Variante des Schiffeversenkens als Spielbrett .....	6
Bild aus: Amazon-Media, <a href="https://m.media-amazon.com/images/I/71xjwFsKXvL._AC_SL1500_.jpg">https://m.media-amazon.com/images/I/71xjwFsKXvL._AC_SL1500_.jpg</a> , Letzter Zugriff am 31.07.2022.	
Abbildung 2: Seekarte mit den dazugehörigen Koordinaten.....	7
Abbildung 3: Beispielhafte Aufstellung der Schiffe .....	8
Abbildung 4: Beispielhafte Seekarte eines Angreifers.....	9
Abbildung 5: Die Spielweise der Rakete ist in vielen Varianten unterschiedlich.....	10
Abbildung 6: Horizontales und vertikales Baumdiagramm zum einfachen Münzwurf.....	11
Abbildung 7: Beispielhafte Darstellung eines Urnenmodells .....	13
Bild aus: Wikipedia, <a href="https://de.wikipedia.org/wiki/Datei:Urn_problem_qtl2.svg">https://de.wikipedia.org/wiki/Datei:Urn_problem_qtl2.svg</a> , Letzter Zugriff am 12.09.2022	
Abbildung 8: Beispiel eines leicht erratbaren Musters.....	13
Abbildung 9: Erhöhte Vorhersehbarkeit nach einem Treffer am Rand oder in der Ecke .....	13
Abbildung 10: Ein gescheiterter Versuch, alle Schiffe in die Mitte zu platzieren .....	15
Abbildung 11: Baumdiagramm von 30 perfekten Zufallsangriffen .....	18
Abbildung 12: An diesen Feldern werden besonders häufig Schiffe positioniert .....	20
Abbildung 13: Beispielssituation. Hier werden zwei Strategien angewendet.....	20
Abbildung 14: Die Angriffsrichtungen im Zielmodus .....	21
Abbildung 15: Mögliche vertikale Aufstellungen von 2er-, 3er- und 4er-Schiffen .....	21
Abbildung 16: Wahrscheinlichkeiten eines Treffers in der Mitte eines Schiffs.....	22
Abbildung 17: Baumdiagramm zur Berechnung von $P(E_{\text{richtige Richtung}})$ .....	23
Abbildung 18: Die Importanz der Schüsse beim diagonalen Angriff .....	24
Abbildung 19: Verschiedene Anwendungen des Schachbrettmusters .....	24
Abbildung 20: Beispielhafte Darstellung einer Heatmap.....	26
Abbildung 21: Blackbox-System des Flottengenerators .....	27
Abbildung 22: Konsolenausgabe des Flottengenerators .....	30
Abbildung 23: Konsolenausgabe des Flottengenerators .....	30
Abbildung 24: Die Formatierungsregel „abgestufte Farbskala“ in Excel .....	30
Abbildung 25: Heatmap zur absoluten Häufigkeit.....	31
Abbildung 26: Heatmap zur relativen Häufigkeit .....	31
Abbildung 27: Die Anzahl der zu verteilenden Schiffe .....	33
Abbildung 28: Heatmap zur relativen Häufigkeit .....	35
Abbildung 29: Heatmap zur breitflächigen relativen Häufigkeit .....	35

Abbildung 30: Die Fehlerquote der Umfrage.....	36
Abbildung 31: Strategien beim Verteilen der Flotte .....	36
Abbildung 32: Strategien beim Angriff auf Flotten .....	36
Abbildung 33: Beispiel eines diagonalen Angriffs mit einer Einteilung in vier Zonen .....	37
Abbildung 34: Eine unmögliche Aufstellung der Schiffe durch den Generator .....	40
Abbildung 35: Die hohe Fehlerquote und die dabei nicht platzierten Schiffe .....	40
Abbildung 36: Heatmaps des Flottengenerators 2.1.....	42
Abbildung 37: Heatmaps des Flottengenerators 2.2.....	43
Abbildung 38: Heatmaps des Flottengenerators 2.3.....	44
Abbildung 39: Betrachtung des möglichen Umfelds bei der Schiffsplatzierung .....	45
Abbildung 40: Diagramm zu Graphen des Jagd- und Zielmodus .....	49
Abbildung 41: QR-Code zur Weiterleitung an die Onlineversion der Arbeit.....	Deckblatt, 52
Abbildung 42: FOS-BOS-Weiden Logo .....	Deckblatt

## XI. Literaturverzeichnis

Die gesamte Seminararbeit mit allen verwendeten Dateien ist auf Google Drive verfügbar. Der Weiterleitungs-Link zu dem Ordner ist [https://drive.google.com/drive/folders/11ErKscIWEpD7DKMk0umpLK6J-1\\_1H2UU?usp=drive\\_link](https://drive.google.com/drive/folders/11ErKscIWEpD7DKMk0umpLK6J-1_1H2UU?usp=drive_link), wobei alternativ auch der hier abgebildete QR-Code eingescannt werden kann.



Abbildung 41: QR-Code zur Weiterleitung an die Onlineversion der Arbeit

- Abts, Dietmar. *Grundkurs JAVA: Eine Einführung in das objektorientierte Programmieren mit Beispielen und Übungsaufgaben*. Braunschweig: Vieweg, 2000.
- . *Grundkurs JAVA: Von den Grundlagen bis zu Datenbank- und Netzanwendungen*. Wiesbaden: Springer Fachmedien Wiesbaden, 2020.
- Aczel, Amir. *Der ganz normal verteilte Zufall*. Heidelberg: Spektrum Akademischer Verlag, 2011.
- Alemi, Alexander A. „Alexander A. Alemi.“ *AlexAlemi-Website*. Februar 2022. <https://www.alexalemi.com/> (Zugriff am 02. September 2022).
- Balzert, Helmut. *Java: Der Einstieg in die Programmierung*. Herdecke: W3L-Verl., 2008.
- . *Java: Objektorientiert programmieren*. Herdecke: W3L-Verl., 2010.
- Barshap, Guy. „Cryptography and Security: Computer Science: Cornell University.“ *Cornell University-Website*. 21. Juli 2018. <https://arxiv.org/abs/1807.08142> (Zugriff am 31. Juli 2022).
- Battleship-game. „Schiffe Versenken.“ *Schiffe Versenken: Battleship-game-Website*. 2020. <http://de.battleship-game.org/> (Zugriff am 2022. Juli 31).
- BattleshipOnline. „Battleship Game: Battleship Online.“ *Battleship-Online-Website*. 31. Juli 2022. <https://www.battleshiponline.org/> (Zugriff am 31. Juli 2022).
- Bauer, Heinz. *Wahrscheinlichkeitstheorie und Grundzüge der Maßtheorie*. Berlin : de Gruyter, 1978.
- Baumgartner, Jana. „Brettspiel-Klassiker: Hiptoys.“ *Hiptoys-Website*. 23. Juli 2022. <https://hiptoys.de/brettspiel-klassiker/#FAQ> (Zugriff am 24. Juli 2022).
- Behnen, Konrad. *Grundkurs Stochastik*. Stuttgart: Teubner, 1995.
- Berry, Nick. „Battleship: DataGenetics.“ *DataGenetics-Website*. 2013. <https://www.datagenetics.com/blog/december32011/> (Zugriff am 06. September 2022).
- Bewersdorff, Jörg. *Glück, Logik und Bluff*. Wiesbaden: Springer Spektrum, 2018.

- Bol, Georg. *Wahrscheinlichkeitstheorie: Einführung*. München: Oldenbourg, 2004.
- Bosch, Karl. *Elementare Einführung in die Wahrscheinlichkeitsrechnung*. Braunschweig : Vieweg, 1999.
- Brown, Seth. „How to Win at Battleship: The Spruce Crafts.“ *The Spruce Crafts-Website*. 08. November 2020. <https://www.thesprucecrafts.com/how-to-win-at-battleship-411068> (Zugriff am 07. September 2022).
- Carrasco, Luis, Interview geführt von Simon Völkl. *Interview zur Behebung der Problematik des ersten Flottengenerators* (28. Juli 2022).
- Christof, Maas. *Stochastik kompakt für Dummies*. Weinheim: Wiley-VCH, 2019.
- Cornelsen. *Mathematik Berufliche Oberschule Bayern*. Berlin: Cornelsen Verlag GmbH, 2019.
- de-academic. „Parität (Mathematik): de-academic.“ *de-academic-Website*. 2022. <https://de-academic.com/dic.nsf/dewiki/1077848> (Zugriff am 07. September 2022).
- Doberkat, Ernst-Erich. *Einführung in die objektorientierte Programmierung mit Java*. München: Oldenbourg, 2002.
- Dörr, Micaela. „Schiffe versenken - diese Taktik verhilft Ihnen zum Sieg: Helpster.“ *Helpster-Website*. unbekannt. [https://www.helpster.de/schiffe-versenken-diese-taktik-verhilft-ihnen-zum-sieg\\_113630](https://www.helpster.de/schiffe-versenken-diese-taktik-verhilft-ihnen-zum-sieg_113630) (Zugriff am 07. September 2022).
- Dworatschek, Sebastian. *Einführung in die Datenverarbeitung*. Berlin: De Gruyter, 2020.
- Eclipse Foundation. „Home: Eclipse.“ *Eclipse-Website*. 29. Juli 2022. <https://www.eclipse.org/> (Zugriff am 29. Juli 2022).
- Georgii, Hans-Otto. *Stochastik: Einführung in die Wahrscheinlichkeitstheorie und Statistik*. Berlin: De Gruyter, 2008.
- Gulz, Carolin. „Magazin: Freizeit: idealo.“ *idealo-Website*. 23. November 2020. <https://www.ideal.de/magazin/freizeit/die-beliebtesten-gesellschaftsspiele-des-jahres?cmpReload=true> (Zugriff am 24. Juli 2022).
- Hafner, Robert. *Wahrscheinlichkeitsrechnung und Statistik*. Wien: Springer, 1989.
- Harris, Aisha. „How to Win at Battleship:Slate.“ *Slate-Website*. 16. Mai 2012. <https://slate.com/culture/2012/05/how-to-win-at-battleship.html> (Zugriff am 03. September 2022).
- Henze, Norbert. *Stochastik für Einsteiger*. Wiesbaden: Springer Spektrum, 2017.
- Hesse, Nils. *Spielend gewinnen*. Wiesbaden: Springer Fachmedien, 2015.
- Inden, Michael. *Der Weg zum Java-Profi*. Heidelberg: dpunkt, 2015.
- Krengel, Ulrich. *Einführung in die Wahrscheinlichkeitstheorie und Statistik*. Wiesbaden: Vieweg, 2005.
- Krüger, Karl-Heinz. *Grundwissen Stochastik*. Stuttgart: UTB, 2020.

- L'Ecuyer, Pierre. „HISTORY OF UNIFORM RANDOM NUMBER GENERATION: Informs-Sim.“ *Informs-sim-Website*. 2017. <https://www.informs-sim.org/wsc17papers/includes/files/016.pdf> (Zugriff am 02. September 2022).
- Laplace, Pierre-Simon. *Théorie analytique des probabilités*. Paris: Courcier, 1820.
- Milton Bradley Company. „Battleship: Instruct: Common.“ *Battleship*. 1990. <https://www.hasbro.com/common/instruct/battleship.pdf> (Zugriff am 31. Juli 2022).
- Musiol, Gerhard, und Heiner Mühlig. *Taschenbuch der Mathematik*. Frankfurt am Main: Verlag Harri Deutsch, 2005.
- Nawrotzki, Kurt. *Lehrbuch der Stochastik*. Frankfurt am Main: Harri Deutsch, 1994.
- Nelles, Stephan. *Excel 2013 im Controlling*. Bonn: Vierfarben, 2014.
- Polczynski, Jim. „Battleship Games: Games one by one: The Big Game Hunter.“ *The Big Game Hunter-Website*. 2003. <https://thebiggamehunter.com/games-one-by-one/battleship-games/> (Zugriff am 2022. Juli 03).
- Rosenthal, Jeffrey S. *Vom Blitz getroffen. Die seltsame Welt des Zufalls*. Frankfurt am Main: Eichborn, 2007.
- Rumsey, Deborah. *Wahrscheinlichkeitsrechnung für Dummies*. Weinheim: WILEY-VCH, 2007.
- smuttlewerk interactive. „Schiffe Versenken Fleet Battle: Apps: Store: Google Play.“ *Google Play-Website*. 31. Juli 2022. <https://play.google.com/store/apps/details?id=de.smuttlewerk.fleetbattle> (Zugriff am 31. Juli 2022).
- Suhr, Frauke. „Themen: Gesellschaftsspiele: Statista.“ *Statista-Website*. 23. Oktober 2019. <https://de.statista.com/infografik/19742/umfrage-welche-gesellschaftsspiele-die-deutschen-zuhause-haben/> (Zugriff am 24. Juli 2022).
- Swanson, Ana. „Wonk: News: The Washington Post.“ *The Washington Post-Website*. 08. Mai 2015. <https://www.washingtonpost.com/news/wonk/wp/2015/05/08/how-to-win-any-popular-game-according-to-data-scientists/#Battleship> (Zugriff am 01. September 2022).
- Take It Serious. „Schiffe versenken: Take It Serious.“ *Take It Serious-Website*. 02. August 2022. <https://take-it-serious.de/schiffe-versenken/> (Zugriff am 02. August 2022).
- Tanton, James. *Encyclopedia of mathematics*. New York: Facts On File, 2005.
- Thielemann, Henning. „Publikationen: ResearchGate.“ *ResearchGate-Website*. Mai 2019. [https://www.researchgate.net/publication/333579639\\_Kombinatorik\\_bei\\_Schiffe\\_versenken](https://www.researchgate.net/publication/333579639_Kombinatorik_bei_Schiffe_versenken) (Zugriff am 27. Juli 2022).
- UltraBoardGames. „Tips to win Battleship: UltraBoardGames.“ *UltraBoardGames-Website*. 2020. <https://www.ultraboardgames.com/battleship/tips.php> (Zugriff am 03. September 2022).
- unbekannt. „Schiffe versenken – erfolgreiche Spielstrategien: 50Plus.de.“ *50Plus.de-Website*. kein Datum. <https://www.50plus.de/article/schiffe-versenken-die-erfolgreichsten-spielstrategien.html> (Zugriff am 01. September 2022).

—, „Toys, Dolls Games & Puzzles : Worthopedia: Home: Worthpoint.“ *Worthpoint-Website*. Datum der Veröffentlichung unbekannt. <https://www.worthpoint.com/worthopedia/1890-antique-board-game-basilinda-1748619764> (Zugriff am 2022. Juli 31).

Virtuosi), Alemi (The. „The Linear Theory of Battleship: The Virtuosi.“ *The Virtuosi-Website*. 03. Oktober 2011. <http://thevirtuosi.blogspot.com/2011/10/linear-theory-of-battleship.html> (Zugriff am 01. September 2022).

Warwitz, Siegbert A. *Spiele anderer Zeiten und Völker - mit Kindern entdecken und erleben*. Baltmannsweiler: Schneider Verlag Hohengehren, 1998.

—, *Vom Sinn des Spielens. Reflexionen und Spielideen*. Baltmannsweiler: Schneider Verlag Hohengehren, 2021.

Whitehill, Bruce. *Games: American Boxed Games and Their Makers, 1822-1992: With Values*. Wisconsin: KP Books, 1992.

wikiHow. „Bei Schiffe versenken gewinnen: Spiele: wikiHow.“ *wikiHow-Website*. unbekannt. <https://de.wikihow.com/Bei-Schiffe-versenken-gewinnen> (Zugriff am 07. September 2022).

## **XII. Erklärung**

Hiermit erkläre ich, dass ich die Seminararbeit selbständig erarbeitet und keine anderen als die angegebenen Hilfsmittel und Quellen benutzt habe. Die Stellen der Arbeit, die anderen Quellen im Wortlaut oder dem Sinn nach entnommen wurden, sind durch Angaben der Herkunft kenntlich gemacht. Dies gilt auch für Zitate, bildliche Darstellungen sowie für Quellen aus dem Internet.

07. Dezember 2022

Simon Völkl