

Meine Seminarausarbeitung

Viel Schreiber

Betreuer: Dipl.-Inf. Carl Coder

Zusammenfassung Ein schöner Abstract. Das ist einfach die Kurzzusammenfassung.

[Christopher]

1 Einleitung

1.1 Spielbeschreibung

Diese Ausarbeitung beschäftigt sich mit der Umsetzung von Schiffe-Versenken in beliebig viele Dimensionen. Die generelle Funktionsweise von dem normalen Schiffe-Versenken bleibt erhalten, muss jedoch um einige Dinge erweitert werden, um auch in höheren Dimensionen gut spielbar zu sein.

Wir betrachten das Spielgeschehen zur Einfachheit halber nur aus der Sicht eines Spielers. Zu Beginn platziert der Gegner eine feste und bekannte Anzahl an Schiffen auf seinem Spielfeld. Es wird davon ausgegangen, dass die Schiffspeditionen rein zufällig ausgewählt wurden. (Hier mehr Details mit Permutation) Natürlich sind die Schiffspeditionen dem Spieler nicht bekannt. Die Gesamtheit aller gewählten Schiffspeditionen wird auch die Schiffsverteilung genannt. Nun kann der Spieler anfangen, auf bestimmte Positionen auf dem Spielfeld, auch Zellen genannt, zu schießen. Nach jedem Schuss erfährt der Spieler von seinem Gegner, die Anzahl an getroffenen Schiffen. Falls keine Schiffe getroffen wurden ist die Anzahl 0. Da Schiffe überlappen können, kann die Anzahl auch größer als 1 sein. Sobald alle Schiffe versenkt wurden, ist das Spiel beendet. Das Ziel des Spielers ist es, mit möglichst wenig Schüssen alle Schiffe zu versenken.

1.2 Inhalt

Zu Beginn wird das beschriebene Spielprinzip formalisiert, sodass man damit mathematisch arbeiten kann.

Anschließend werden verschiedene Spielstrategien vorgestellt.

Danach werden verschiedene Möglichkeiten präsentiert, die Spielstrategien so gut wie möglich zu optimieren, damit sie auch mit sehr vielen Dimensionen immernoch effizient berechenbar sind.

Zum Schluss werden dann die Strategien mithilfe der vorherigen Kenntnisse implementiert und die Ergebnisse in Bezug auf Laufzeiteffizienz verglichen.

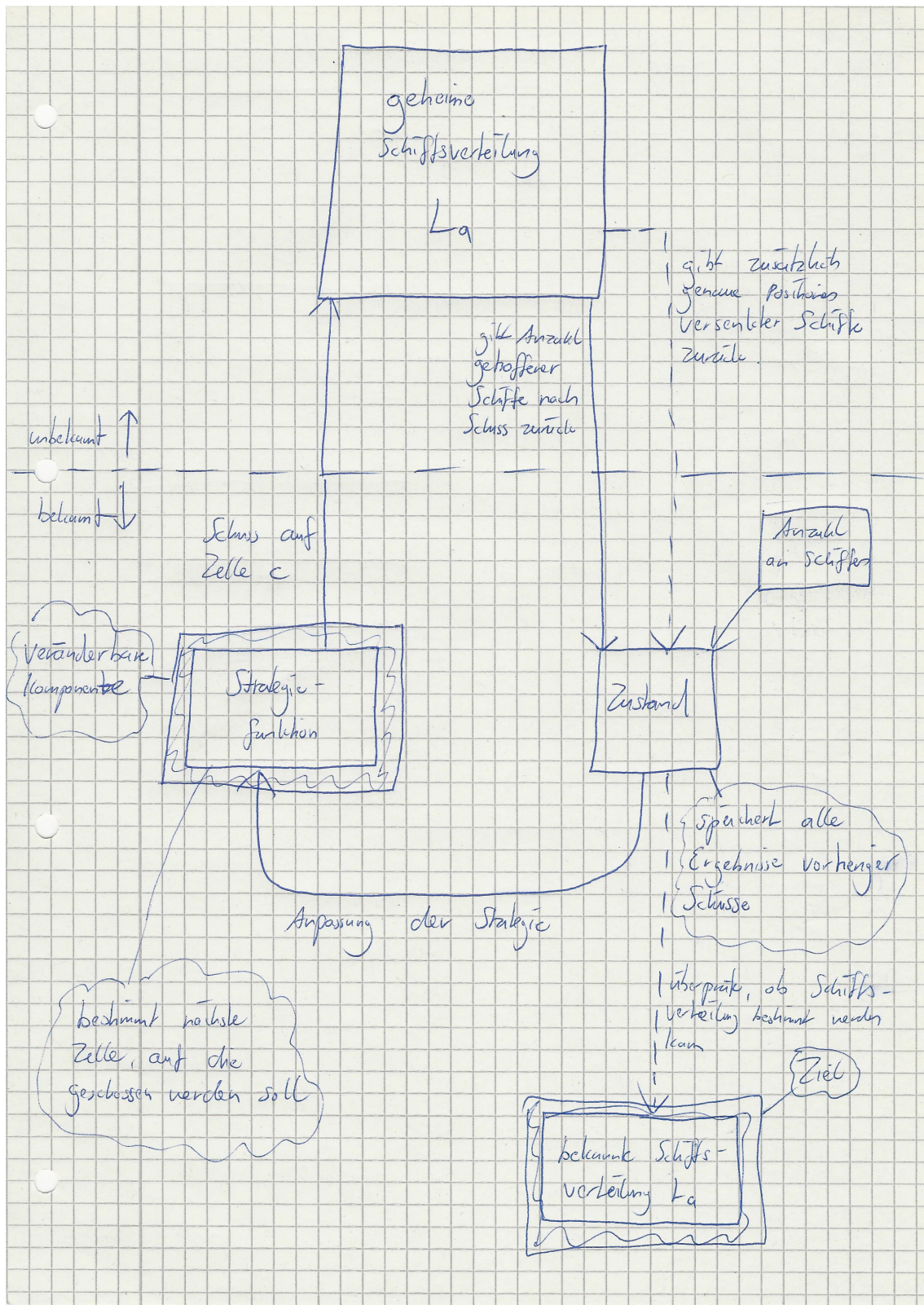


Abbildung 1. Bild

2 Formalisierung des Spielprinzips

Definition 1. Sei $C_{all} = \{1, \dots, N\}^d$ die Menge aller Zellen (cells) des Spielfeldes mit jeweils N Zellen in d Dimensionen.

Definition 2. Sei l eine mögliche Schiffsposition (location), welche mithilfe einer minimalen Ecke $c_{min}(l)$ und einer maximalen Ecke $c_{max}(l)$ bestimmt wird. Dann ist

$$cells(l) = \{c \in C \mid c_{min}(l) \leq c \leq c_{max}(l)\}$$

die Menge aller Zellen, welche sich innerhalb der möglichen Schiffsposition l befinden.

Definition 3. Sei

$$L_{all} = \{\{k \in C_{all} \mid i \leq k \leq j\} \mid i, j \in C_{all} \wedge i \leq j\}$$

die Menge aller möglichen Schiffspositionen.

Definition 4. Sei $ship_count$ die Anzahl der platzierten Schiffe.

Definition 5. Sei $all_distributions = \{L \subseteq L_{all} \mid |L| = ship_count\}$ die Menge von allen möglichen Schiffsverteilungen.

Definition 6. Sei $L_a \in all_distributions$ die eigentliche und geheime Schiffsverteilung.

Definition 7. Sei $L \in all_distributions$ und $c \in C_{all}$. Dann ist

$$hit(L, c) = |\{l \in L_{all} \mid c \in cells(l)\} \cap L|$$

die Treffer-Funktion, welche angibt, wie viele Schiffe bei einem Schuss auf Zelle c getroffen wurden, unter der Annahme, dass L die eigentliche Schiffsverteilung ist.

Definition 8. Sei $c \in C_{all}$. Dann ist

$$hit(c) = hit(L_a, c)$$

2.1 Zustände

Definition 9. Sei $F = \{(c_1, h_1), \dots, (c_t, h_t)\}$ der Zustand nach t Schüssen.

Definition 10. Sei F der momentane Zustand. Dann ist $C_{shot}(F) = \{c \in C_{all} \mid (c, h) \in F, h \in \mathbb{N}_0\}$ die Menge an Zellen, auf die bereits geschossen wurde.

Definition 11. Sei F der momentane Zustand. Dann ist $C_{left}(F) = C_{all} \setminus C_{shot}(F)$ die Menge an Zellen, auf die noch nicht geschossen wurde.

Definition 12. Sei F der momentane Zustand und $cell \in C_{left}(F)$ die beschossene Zelle. Dann ist

$$fire(L, F, cell) = F \cup \{(cell, hit(L, cell))\}$$

die Schuss-Funktion, welche die Zelle dem Zustand F hinzufügt.

Definition 13. Sei F ein Zustand und $c \in C_{left}(F)$ die beschossene Zelle. Dann ist

$$hypo_fire(F, c, h) = F \cup \{(c, h)\}$$

die hypothetische Schuss-Funktion, welche den Zustand so verändert, als hätte es h Treffer bei dem Schuss auf Zelle c gegeben. Es wird also kein echter Schuss abgegeben, es wird angenommen, dass ein Schuss mit h Treffern abgegeben wird.

Definition 14. Sei F ein Zustand. Dann ist

$$distributions(F)$$

die Menge aller Schiffsverteilungen, auf die die Treffer-Informationen vom Zustand F zutreffen. In anderen Worten, die Menge an zum Zustand F noch möglichen Schiffsverteilungen.

Definition 15. Sei F ein Zustand. Dann ist

$$distribution_count(F) = |distributions(F)|$$

Aussagen über Zustände

Definition 16. Sei F ein Zustand. Dann ist

$$finished(F) \Leftrightarrow \forall l \in L_a: cells(l) \subseteq C_{shot}(F)$$

wahr gdw. alle Schiffe zerstört wurden.

Definition 17. Sei F ein Zustand. Dann ist

$$determined(L, F) \Leftrightarrow distribution_count(F) = 1$$

wahr gdw. die richtige Schiffsverteilung bereits bestimmt ist.

Satz 1 Sei F ein Zustand. Sei außerdem $L \in distributions(F)$ eine mögliche Schiffsverteilung.

Dann ist

$$P(L = L_a \mid F) = \frac{1}{distribution_count(F)}$$

die Wahrscheinlichkeit, dass L die richtige Schiffsverteilung ist.

Beweis. TODO

Satz 2 Sei F ein Zustand. Sei außerdem $c \in C_{left}(F)$ die beschossene Zelle und $h \in \mathbb{N}_0$. Dann ist

$$P(hit(c) = h \mid F) = \frac{distribution_count(hypo_fire(F, c, h))}{distribution_count(F)}$$

die Wahrscheinlichkeit zu dem Zustand F , dass bei dem Schuss auf Zelle c genau h Schiffe getroffen werden.

Beweis. Vor dem Schuss gibt es noch im Zustand F genau $distribution_count(F)$ mögliche Schiffsverteilungen. Nach dem Schuss gibt es nur noch $distribution_count(hypo_fire(F, c, h))$ mögliche Schiffsverteilungen.

2.2 Schuss-Strategien

Definition 18. Eine Funktion der Form

$$strat: F \rightarrow C_{left}(F)$$

wird Strategiefunktion genannt. Diese weist jedem Zustand die Zelle zu, auf die als nächstes geschossen werden soll.

Definition 19. Sei

$$all_strategies(F) = \{strat: F \rightarrow C_{left}(F)\}$$

die Menge an allen möglichen Strategiefunktionen.

3 Greedy-Strategien

3.1 Greedy-Hit-Strategie

Die Greedy-Hit-Strategie versucht bei jedem Schuss die erwartete Anzahl an getroffenen Schiffen zu maximieren.

Definition 20. Sei F ein Zustand und $c \in C_{left}(F)$ die beschossene Zelle. Dann ist

$$\mathbb{E}(hit(c) \mid F) = \sum_{h=0}^{ship_count} P(hit(c) = h \mid F) * h$$

Definition 21. Sei F ein Zustand. Dann ist

$$strat_{greedy-hit}(F) = \max_{c \in C_{left}(F)} \mathbb{E}(hit(c) \mid F)$$

die Zelle, bei welcher bei Beschuss die Anzahl an erwarteten getroffenen Schiffen maximal ist.

3.2 Greedy-Distribution-Strategie

Die Greedy-Distribution-Strategie versucht bei jedem Schuss die erwartete Anzahl an ausgeschlossenen Schiffsverteilungen zu maximieren.

Definition 22. Sei F der momentane Zustand. Sei außerdem $c \in C_{left}(F)$ die beschossene Zelle und $h \in \mathbb{N}_0$. Dann ist

$$distributions_removed(L, F, c) = distribution_count(F) - distribution_count(fire(L, F, c))$$

die Anzahl an Schiffsverteilungen, die nach diesem Schuss ausgeschlossen werden könnten.

Satz 3 Sei F der momentane Zustand. Sei außerdem $c \in C_{left}(F)$ die beschossene Zelle. Dann ist

$$\mathbb{E}(distributions_removed(F, c)) = \frac{1}{distribution_count(F)} * \sum_{L \in distributions(F)} distributions_removed(L, F, c)$$

Beweis. Definition des Erwartungswertes.

Definition 23. Sei F ein Zustand. Dann ist

$$strat_{greedy-distribution}(F) = \max_{c \in C_{left}(F)} \mathbb{E}(distributions_removed(F, c))$$

die Zelle, bei welcher bei Beschuss die Anzahl an erwarteten ausgeschlossenen Schiffsverteilungen maximal ist.

3.3 Vergleich der Strategien

Auf den ersten Blick sieht es aus, dass die beiden vorgestellten Greedy-Strategien zwei verschiedene Definitionen für die gleiche Strategie sind. Im folgenden Abschnitt wird aber gezeigt, dass sich beide Strategien unterscheiden.

Definition 24. Sei F ein Zustand und $c \in C_{left}(F)$ die beschossene Zelle. Dann ist

$$asc(F, c) = \{[h_{min}, h_{max}] \mid \forall h, h' \in [h_{min}, h_{max}]: h > h' \Leftrightarrow P(hit(c) = h \mid F) > P(hit(c) = h' \mid F)\}$$

die Menge aller Intervalle, in denen die Funktion $P(hit(c) = h \mid F)$ streng monoton wachsend ist.

Außerdem ist

$$desc(F, c) = \{[h_{min}, h_{max}] \mid \forall h, h' \in [h_{min}, h_{max}]: h > h' \Leftrightarrow P(hit(c) = h \mid F) < P(hit(c) = h' \mid F)\}$$

die Menge aller Intervalle, in denen die Funktion $P(\text{hit}(c) = h \mid F)$ streng monoton fallend ist.

Außerdem ist

$$eq(F, c) = \{[h_{min}, h_{max}] \mid \forall h_{min} \leq h, h' \leq h_{max}: P(\text{hit}(c) = h \mid F) = P(\text{hit}(c) = h' \mid F)\}$$

die Menge aller Intervalle, in denen die Funktion $P(\text{hit}(c) = h \mid F)$ die Steigung 0 hat.

Natürlich sind alle Intervalle in den natürlichen Zahlen, da die Funktion $P(\text{hit}(c) = h \mid F)$ nur für natürliche Zahlen definiert ist.

Lemma 1. Sei F ein Zustand und $c \in C_{left}(F)$ die beschossene Zelle. Sei außerdem $h, h' \in \mathbb{N}$. Dann gilt:

Falls $\exists I \in asc(F, c): h, h' \in I$:

$$h > h' \Leftrightarrow P(\text{hit}(c) = h \mid F) > P(\text{hit}(c) = h' \mid F)$$

Falls $\exists I \in desc(F, c): h, h' \in I$:

$$h < h' \Leftrightarrow P(\text{hit}(c) = h \mid F) < P(\text{hit}(c) = h' \mid F)$$

Falls $\exists I \in eq(F, c): h, h' \in I$:

$$P(\text{hit}(c) = h \mid F) = P(\text{hit}(c) = h' \mid F)$$

Beweis. Definitionen der Intervalle.

Lemma 2. Sei F ein Zustand und $c \in C_{left}(F)$ die beschossene Zelle. Sei außerdem $h, h' \in \mathbb{N}$. Dann gilt:

$$\begin{aligned} & P(\text{hit}(c) = h \mid F) < P(\text{hit}(c) = h' \mid F) \\ \Leftrightarrow & \text{distribution_count}(\text{hypo_fire}(F, c, h)) < \text{distribution_count}(\text{hypo_fire}(F, c, h')) \end{aligned}$$

Beweis.

$$\begin{aligned} & P(\text{hit}(c) = h \mid F) < P(\text{hit}(c) = h' \mid F) \\ \Leftrightarrow & \frac{\text{distribution_count}(\text{hypo_fire}(F, c, h))}{\text{distribution_count}(F)} < \frac{\text{distribution_count}(\text{hypo_fire}(F, c, h'))}{\text{distribution_count}(F)} \\ \Leftrightarrow & \text{distribution_count}(\text{hypo_fire}(F, c, h)) < \text{distribution_count}(\text{hypo_fire}(F, c, h')) \end{aligned}$$

□

Korollar 1 Sei F ein Zustand und $c \in C_{left}(F)$ die beschossene Zelle. Sei außerdem $h, h' \in \mathbb{N}$. Dann gilt:

$$\begin{aligned} & P(\text{hit}(c) = h \mid F) > P(\text{hit}(c) = h' \mid F) \\ \Leftrightarrow & \text{distribution_count}(\text{hypo_fire}(F, c, h)) > \text{distribution_count}(\text{hypo_fire}(F, c, h')) \end{aligned}$$

Beweis. Negation des vorgerigen Lemmas.

Satz 4 Sei F ein Zustand und $c \in C_{\text{left}}(F)$ die beschossene Zelle. Sei außerdem $h, h' \in \mathbb{N}$. Dann gilt:

Falls $\exists I \in \text{asc}(F, c): h, h' \in I$:

$$h > h' \Leftrightarrow \text{distributions_removed}(F, c, h) > \text{distributions_removed}(F, c, h')$$

Falls $\exists I \in \text{desc}(F, c): h, h' \in I$:

$$h > h' \Leftrightarrow \text{distributions_removed}(F, c, h) < \text{distributions_removed}(F, c, h')$$

Falls $\exists I \in \text{eq}(F, c): h, h' \in I$:

$$\text{distributions_removed}(F, c, h) = \text{distributions_removed}(F, c, h')$$

Beweis. Falls $\exists I \in \text{asc}(F, c): h, h' \in I$:

$$\begin{aligned} & h > h' \\ \Leftrightarrow & P(\text{hit}(c) = h \mid F) > P(\text{hit}(c) = h' \mid F) \\ \Leftrightarrow & \text{distribution_count}(\text{hypo_fire}(F, c, h)) > \text{distribution_count}(\text{hypo_fire}(F, c, h')) \\ \Leftrightarrow & \text{distribution_count}(F) - \text{distribution_count}(\text{hypo_fire}(F, c, h)) \\ & < \text{distribution_count}(F) - \text{distribution_count}(\text{hypo_fire}(F, c, h')) \\ \Leftrightarrow & \text{distributions_removed}(F, c, h) < \text{distributions_removed}(F, c, h') \end{aligned}$$

Falls $\exists I \in \text{desc}(F, c): h, h' \in I$:

$$\begin{aligned} & h > h' \\ \Leftrightarrow & P(\text{hit}(c) = h \mid F) < P(\text{hit}(c) = h' \mid F) \\ \Leftrightarrow & \text{distribution_count}(\text{hypo_fire}(F, c, h)) < \text{distribution_count}(\text{hypo_fire}(F, c, h')) \\ \Leftrightarrow & \text{distribution_count}(F) - \text{distribution_count}(\text{hypo_fire}(F, c, h)) \\ & > \text{distribution_count}(F) - \text{distribution_count}(\text{hypo_fire}(F, c, h')) \\ \Leftrightarrow & \text{distributions_removed}(F, c, h) > \text{distributions_removed}(F, c, h') \end{aligned}$$

Falls $\exists I \in \text{eq}(F, c): h, h' \in I$:

$$\begin{aligned} \Leftrightarrow & P(\text{hit}(c) = h \mid F) = P(\text{hit}(c) = h' \mid F) \\ \Leftrightarrow & \text{distribution_count}(\text{hypo_fire}(F, c, h)) = \text{distribution_count}(\text{hypo_fire}(F, c, h')) \\ \Leftrightarrow & \text{distribution_count}(F) - \text{distribution_count}(\text{hypo_fire}(F, c, h)) \\ & = \text{distribution_count}(F) - \text{distribution_count}(\text{hypo_fire}(F, c, h')) \\ \Leftrightarrow & \text{distributions_removed}(F, c, h) = \text{distributions_removed}(F, c, h') \end{aligned}$$

□

Daraus folgt, dass abhängig von der Wahrscheinlichkeitsverteilung von $P(\text{hit}(c) = h \mid F)$ bei dem Versuch möglichst viele Schiffe zu treffen dadurch entweder auch möglichst viele Schiffsverteilungen ausgeschlossen werden können oder eben auch möglichst wenig Schiffsverteilungen ausgeschlossen werden können.

Daher ist die genaue Korrelation zwischen ausgeschlossenen Schiffsverteilungen und getroffenen Schiffen für jede Zelle individuell, da die Wahrscheinlichkeitsverteilung von $P(\text{hit}(c) = h \mid F)$ individuell ist. Für manche Zellen, können beide Strategien äquivalent sein, aber für andere Zellen kann auch eine inverse Korrelation bestehen und mit mehr Treffern weniger Schiffsverteilungen ausgeschlossen werden.

In den meisten Situationen lässt sich keine genaue Aussage darüber treffen, ob beide Strategien die gleichen Zellen auswählen. In der folgenden, etwas spezielleren Situation allerdings lässt sich eine Aussage treffen:

Satz 5 *Sei F ein Zustand. Sei außerdem $c = \text{strat}_{\text{greedy-hit}}(F)$ und $\exists c_{\text{other}} \in C_{\text{left}}(F) \setminus \{c\} : \mathbb{E}(\text{hit}(c_{\text{other}}) \mid F) < \mathbb{E}(\text{hit}(c) \mid F)$, d.h. c ist ein echtes Maximum. Dann gilt:*

$$[0, \text{ship_count}] \in \text{asc}(F, c) \Rightarrow c \neq \text{strat}_{\text{greedy-distribution}}(F)$$

In anderen Worten, falls die Wahrscheinlichkeitsverteilung der nach der greedy-hit-Strategie beschossenen Zelle streng monoton steigend ist, muss die greedy-distribution-Strategie eine andere Zelle beschießen, da die sonst nicht die maximale Anzahl an Schiffsverteilungen ausschliesst. Diese andere Zelle existiert nach Voraussetzung.

4 Optimale-Strategie

Um eine optimale Schuss-Strategie zu definieren, werden ein paar Definitionen benötigt, um den Begriff ‘Optimal’ zu definieren.

4.1 Definition von Optimalität

Definition 25. *Sei L die gewählte Schiffsverteilung und F der momentane Zustand und sei $\text{determined}(L, F)$ wahr. Dann ist*

$$\text{shots_to_finish}(L, F) = \sum_{l \in L_a} |\text{cells}(l) \setminus C_{\text{shot}}(F)|$$

die Anzahl an Schüssen, die noch benötigt werden, um alle Schiffe der bereits bekannten Schiffsverteilung zu versenken.

Definition 26. *Sei L die gewählte Schiffsverteilung und F der momentane Zustand. Sei außerdem $\text{strat} \in \text{all_strategies}$ die verwendete Schuss-Strategie. Dann ist*

$$\text{shots_left}(L, F, \text{strat}) = \begin{cases} \text{shots_to_finish}(L, F) & , \text{determined}(L, F) \\ \text{shots_left}(L, \text{fire}(L, F, \text{strat}(F)), \text{strat}) + 1 & , \text{sonst} \end{cases}$$

die Anzahl an Schüssen, die benötigt werden, um alle Schiffe der bereits bekannten Schiffsverteilung mit der Schuss-Strategiefunktion $strat$ zu versenken.

Satz 6 Sei F der momentane Zustand und $strat \in all_strategies$ die verwendete Schuss-Strategie. Dann ist

$$\mathbb{E}(shots_left(F, strat)) = \frac{1}{distribution_count(F)} * \sum_{L \in distributions(F)} shots_left(L, F, strat)$$

die erwartete Anzahl an Schüssen, die benötigt werden, um alle Schiffe zu versenken.

Beweis.

$$\mathbb{E}(shots_left(F, strat)) = \quad (4.1)$$

$$\sum_{L \in distributions(F)} P(L_a = L \mid F) * shots_left(L, F, strat) = \quad (4.2)$$

$$\frac{1}{distribution_count(F)} * \sum_{L \in distributions(F)} shots_left(L, F, strat) \quad (4.3)$$

□

Nun kann definiert werden, was es für eine Strategie heißt, optimal zu sein:

Definition 27. Eine Strategiefunktion $strat \in all_strategies$ heißt optimal, falls für alle $strat_{alt} \in all_strategies$ gilt:

$$\mathbb{E}(shots_left(F, strat)) \leq \mathbb{E}(shots_left(F, strat_{alt}))$$

4.2 Herleitung

Lemma 3. Sei F der momentane Zustand, c die Zelle, auf die geschossen werden soll und $strat \in all_strategies$ die verwendete Schuss-Strategie. Dann ist

$$\mathbb{E}(shots_left(F, c, strat)) = \mathbb{E}(shots_left(fire(L_a, F, c), strat)) + 1$$

die erwartete Anzahl an Schüssen, die benötigt werden, um alle Schiffe zu versenken, falls der nächste Schuss auf Zelle c abgefeuert wird.

Beweis. Trivial

Definition 28. Sei F der momentane Zustand und $strat \in all_strategies$ die verwendete Schuss-Strategie. Dann ist

$$min_expected_shots(F, strat) = \min_{c \in C_{left}(F)} \mathbb{E}(shots_left(F, c, strat))$$

die minimale erwartete Anzahl an Schüssen um alle Schiffe zu versenken, falls die nächste beschossene Zelle frei gewählt werden kann und danach die Schuss-Strategie $strat$ verwendet wird.

Definition 29. Sei F ein Zustand. Dann ist

$$strat_{opt}(F) = c \in C_{left}(F) \mid min_expected_shots(F, strat_{opt}) = \mathbb{E}(shots_left(F, c, strat_{opt}))$$

die optimale Strategiefunktion. Diese wählt für den Zustand F eine Zelle aus, welche nach Beschuss die Anzahl an benötigten Schüssen um alle Schiffe zu versenken, minimiert.

Satz 7 (Optimalität) Sei $strat_{opt}$ die optimale Strategiefunktion und $strat_{alt} \in all_strategies$ eine andere Strategiefunktion. Dann gilt:

$$\mathbb{E}(shots_left(F, strat_{opt})) \leq \mathbb{E}(shots_left(F, strat_{alt}))$$

Beweis.

$$\mathbb{E}(shots_left(F, strat_{opt})) = \mathbb{E}(shots_left(F, strat_{opt}(F), strat_{opt})) = \quad (4.4)$$

$$\min_{c \in C_{left}(F)} \mathbb{E}(shots_left(F, c, strat)) \leq \mathbb{E}(shots_left(F, c_{other}, strat)), c_{other} \in C_{left}(F) \quad (4.5)$$

\Rightarrow

$$\quad (4.6)$$

$$\mathbb{E}(shots_left(F, strat_{opt})) \leq \mathbb{E}(shots_left(F, strat_{alt})) \quad (4.7)$$

□

5 Optimierungen

Wir nähern uns langsam der Implementation der Schuss-Strategien. Um die Laufzeit der Implementation möglichst klein zu halten, werden in diesem Kapitel eine Reihe von Optimierungsmethoden vorgestellt.

5.1 Berechnung von Schiffpositionen

[Dennis]

Definition 30. Sei C_{pos} eine Menge an Zellen, die belegt werden sollen. Sei außerdem C_{neg} eine Menge an Zellen, die nicht belegt werden sollen.

Dann ist

$$location_count(C_{pos}, C_{neg})$$

die Anzahl der möglichen Schiffpositionen $l \in L_{all}$ für die gilt: $C_{pos} \subseteq cells(l)$ und $C_{neg} \cap cells(l) = \emptyset$.

5.2 Berechnung der Verteilungsanzahl mit Verteilungsfunktionen

[Christopher]

In allen vorherigen Berechnungen wird nur die Anzahl an Schiffsverteilungen zum Zustand F , $distribution_count(F)$, benötigt, da es zu jedem Zustand eine Gleichverteilung an Schiffsverteilungen gibt.

Der naive Ansatz diesen Wert zu berechnen, wäre alle Verteilungen zu konstruieren und dann zusammen zu zählen. Mithilfe den sogenannten Verteilungsfunktionen und ein bisschen Kombinatorik lässt sich die Anzahl einfacher berechnen.

Definition 31. Sei $cell_combinations(F) = \mathcal{P}(C_{shot}(F))$.

Definition 32. Sei $comb \in cell_combinations(F)$. Dann ist

$$location_count(F, comb) = location_count(comb, C_{shot}(F) \setminus comb)$$

eine Kurzform.

Definition 33. Sei $comb \in cell_combinations(F)$. Dann ist

$$max_shared_ship_count(F, comb) = \min\{location_count(F, comb)\} \cup \{h \mid (c, h) \in F \wedge c \in comb\}$$

Definition 34. Sei

$$hit_count(F, c, s) = \sum_{comb \in \{x \in cell_combinations(F) \mid c \in x\}} s(comb)$$

Definition 35. *Sei*

$$\begin{aligned} \text{share_functions}(F) = \{ & s: \text{cell_combinations}(F) \rightarrow \mathbb{N}_0 \mid \\ & \forall (c, h) \in F: \text{hit_count}(F, c, s) = h \wedge \\ & \forall \text{comb} \in \text{cell_combinations}(F): 0 \leq s(\text{comb}) \leq \text{max_shared_ship_count}(F, \text{comb}) \\ & \} \end{aligned}$$

die Menge an Share-Funktionen, die das LGS (...) lösen.

Satz 8 *Sei F der momentane Zustand und $s \in \text{share_functions}(F)$ eine Verteilungsfunktion. Dann ist*

$$\text{distribution_count}(F, s) = \left(\prod_{\text{comb} \in \text{cell_combinations}(F)} \binom{\text{location_count}(F, \text{comb})}{s(\text{comb})} \right)$$

die Anzahl an möglichen Schiffsverteilungen für die Verteilungsfunktion s .

Beweis. Für jede Teilkombination $\text{comb} \in \text{cell_combinations}_F$ gibt es $\text{location_count}_F(\text{comb})$ Positionen, die genau die Zellen der Teilkombination belegen. Aus diesen Positionen werden dann mit der Funktion s_F genau $s_F(\text{comb})$ Positionen für Schiffe ausgewählt. Daher gibt es für jede Teilkombination $\binom{\text{location_count}_F(\text{comb})}{s_F(\text{comb})}$ verschiedene Schiffspositionen.

Die gesamte Anzahl an möglichen Schiffsverteilungen ergibt sich dann einfach aus der Multiplikation der Anzahl an Schiffspositionen der einzelnen Teilkombinationen.

Definition 36. *Sei*

$$\text{distribution_count}(F) = \sum_{s \in \text{share_functions}(F)} \text{distribution_count}(s)$$

Joel und Samuel:

6 Implementierung der Strategien

7 Vergleich der Strategien