

# Online-Optimierung als Zwei-Personen-Nullsummenspiel und Yaos Prinzip

Lukas Graf

15. Oktober 2015

## Einführung

Stellen wir uns vor wir haben ein bestimmtes Online-Optimierungsproblem gegeben und fragen uns wie „schwer“ dieses ist. Eine natürliche Weise dies zu beurteilen wäre dann sich zu fragen, wie gut ein bester Online-Algorithmus für dieses Problem ist, d.h. welche Kompetitivität wir mit diesem besten Algorithmus erreichen können.

Da wir letzteren im Allgemeinen aber wohl nicht von Vornherein kennen werden, wäre es etwas viel verlangt diese Kompetitivität exakt zu bestimmen. Stattdessen wollen wir versuchen, obere und untere Schranken für die Kompetitivität des bestmöglichen Online-Algorithmus zu finden.

Den einfachen Fall bilden hierbei offenbar die oberen Schranken: Dazu müssen wir uns nur einen beliebigen Online-Algorithmus vorgeben und dessen Güte bestimmen. Die Kompetitivität eines besten Online-Algorithmus kann dann höchstens so groß sein, wie die des soeben gewählten Algorithmus. Schwieriger ist dagegen das Finden unterer Schranken: Wie sollen wir Aussagen darüber treffen, welche Kompetitivität kein Online-Algorithmus unterschreiten kann (ohne dazu alle denkbaren Online-Algorithmen auszuprobieren)?

Eine Antwort darauf liefert ein von Andrew Yao entwickeltes Prinzip ([Yao77, Theorem 1]), welches auf der Idee basiert Online-Optimierungsprobleme als Nullsummenspiel zwischen zwei Personen aufzufassen: Dem Online-Spieler, der versucht einen möglichst guten Online-Algorithmus zu finden, und dem Offline-Spieler, dessen Ziel es ist eine Anfragesequenz zu finden, auf der der gewählte Online-Algorithmus im Verhältnis zum optimalen Offline-Algorithmus möglichst schlecht abschneidet.

Dank dieser Betrachtungsweise lassen sich dann Erkenntnisse aus der Spieltheorie (und hier insbesondere von Neumanns Minimax-Theorem) auf die Online-Optimierung übertragen. Dadurch werden wir nicht nur tatsächlich einen Weg finden untere Schranken

für die Kompetitivität zu bestimmen, sondern wir können sogar zeigen, dass auf diese Weise (wenigstens theoretisch) immer eine solche untere Schranke gefunden werden kann, die mit der stärksten obersten Schranke zusammenfällt (und somit gerade die Kompetitivität eines besten Online-Algorithmus festlegt).

Die vorliegende Ausarbeitung orientiert sich im Wesentlichen an Kapitel 8 des Buches *Online Computation and Competitive Analysis* von Borodin und El-Yaniv ([BEY98]). Der Beweis von Yaos Prinzip verwendet darüber hinaus einige Vereinfachungen aus Kapitel 5 des Skriptes „Online Optimierung“ von Krumke und Rambau ([KR05]).

## 1 Zwei-Personen-Nullsummenspiel

Ein Zwei-Personen Spiel ist im Wesentlichen eine (endliche) Abfolge von Entscheidungen zweier Spieler (man denke beispielsweise an Schere-Stein-Papier, Tic-Tac-To, Schach, ...). Um daraus ein Nullsummenspiel zu machen, müssen wir dann außerdem für jeden denkbaren Spielausgang einen gewissen Betrag festlegen, den Spieler 1 an Spieler 2 zahlen muss (oder umgekehrt - letzteres werden wir im Folgenden der Einfachheit halber immer als negative Auszahlung von Spieler 1 an Spieler 2 auffassen).

**Beispiel 1.1.** Ein einfaches Spiel könnte beispielsweise so aussehen:

Unser Gegenspieler denkt sich eine endliche Folge natürlicher Zahlen aus. Dann zählt er nacheinander die Zahlen dieser Folge auf, bis wir „Stopp“ sagen (oder die Folge zu Ende ist). Danach verrät unser Gegenspieler, was die kleinste Zahl in seiner Folge war und wir zahlen ihm die Differenz der Zahl, bei der wir „Stopp“ gesagt haben, und dem Eineinhalbfachen der kleinsten Zahl der Folge.

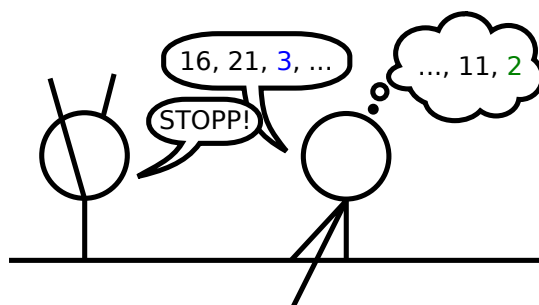


Abbildung 1: Finde die kleinste Zahl!

Unser Ziel ist es folglich bei einer Zahl „Stopp“ zu sagen, die möglichst nahe an der kleinsten Zahl der Folge liegt (idealerweise höchstens 1,5-mal so groß ist, damit wir auch Gewinn machen).

Eine naheliegende Fragestellung wäre dann, welche Strategie in diesem Spiel besonders erfolgversprechend ist, oder auch, ob es überhaupt eine Strategie gibt, mit der wir einen sicheren Gewinn machen können.

Um nun leichter verschiedene Spielabläufe vergleichen zu können, wollen wir außerdem davon ausgehen, dass die beiden Spieler ihre Entscheidungen nicht erst unter dem Spiel „spontan“ treffen, sondern bereits zu Beginn des Spiels für jede denkbare Entscheidungssituation<sup>1</sup> festlegen, welche Entscheidung sie in dieser treffen würden. Alle diese Festlegungen zusammen bezeichnen wir dann als *reine Strategien* der beiden Spieler.

Damit lässt sich ein (endliches) Zwei-Personen-Nullsummenspiel wie folgt beschreiben:

**Definition 1.2.** Ein (*endliches*) *Zwei-Personen-Nullsummenspiel* ist gegeben durch

- eine Menge von *reinen Strategien*  $\{s_i^1 \mid i \in I\}$  für Spieler 1 (indiziert durch eine endliche Menge  $I$ ),
- eine Menge von *reinen Strategien*  $\{s_j^2 \mid j \in J\}$  für Spieler 2 (indiziert durch eine endliche Menge  $J$ )
- und eine *Auszahlungsfunktion*  $H : I \times J \rightarrow \mathbb{R}$  für Spieler 1.

Wählen also Spieler 1 und 2 die reinen Strategien  $s_i^1$  bzw.  $s_j^2$ , so zahlt Spieler 1 nach Ende des Spiels  $H(i, j)$  an Spieler 2. Offenbar ist nun das Ziel von Spieler 1 die Auszahlungsfunktion  $H$  durch die Wahl seiner Strategie zu minimieren (Spieler 1 ist *Kostenminimierer*), während Spieler 2 versucht  $H$  zu maximieren (Spieler 2 ist *Gewinnmaximierer*).

*Bemerkung 1.3.* Die Kostenfunktion (und damit effektiv das gesamte Spiel) lässt sich gut als Matrix darstellen, deren Zeilen den reinen Strategien von Spieler 1 und deren Spalten den reinen Strategien von Spieler 2 entsprechen:

	...	$s_j^2$	...
$\vdots$			
$s_i^1$		$H(i, j)$	
$\vdots$			

Entsprechend wird Spieler 1 dann als „Zeilenspieler“ und Spieler 2 als „Spaltenspieler“ bezeichnet. Die Wahl einer reinen Strategie entspricht sodann der Wahl einer Zeile  $i$  bzw. Spalte  $j$ . Daher werden wir im Folgenden oft nicht mehr zwischen der Strategie  $s_i^1$  und dem Index dieser Strategie  $i$  unterscheiden und direkt von der „Strategie  $i$ “ sprechen.

**Definition 1.4.** Ein Online-Optimierungsproblem (mit blindem Offline-Gegenspieler) können wir nun wie folgt als Zwei-Personen-Nullsummenspiel auffassen:

- Spieler 1 ist der Online-Spieler und seine reinen Strategien entsprechen den zur Verfügung stehenden deterministischen Algorithmen  $\{\text{ALG}_i \mid i \in I\}$ .
- Spieler 2 ist der Offline-Gegenspieler und seine reinen Strategien entsprechend den zulässigen endlichen Anfragesequenzen  $\{\sigma_j \mid j \in J\}$ .

---

<sup>1</sup>Jede solche Entscheidungssituation wird dabei festgelegt durch alle Informationen, die dem jeweiligen Spieler zu diesem Zeitpunkt bekannt sind (also bspw. zuvor bereits getroffene Entscheidungen)

- Als Auszahlungsfunktion wählen wir für eine fest gewählte Konstante  $c \geq 1$ :

$$H_c(i, j) := \text{ALG}_i(\sigma_j) - c \cdot \text{OPT}(\sigma_j)$$

*Bemerkung 1.5.* Damit wir auf diesem Weg tatsächlich ein *endliches* Zwei-Personen-Nullsummenspiel erhalten, müssen wir also voraussetzen, dass auch das ursprüngliche Optimierungsproblem in einem gewissen Sinne endlich ist: Es darf nur endlich viele Anfragesequenzen geben und können nur endlich viele deterministische Algorithmen zur Auswahl stehen.

Bis einschließlich Abschnitt 4 werden wir nun ausschließlich solche endlichen Online-Optimierungsprobleme betrachten.

*Beobachtung 1.6.* Diese Auszahlungsfunktion erlaubt uns folgende Interpretation: Ein Algorithmus  $\text{ALG}_i$  ist genau dann  $c$ -kompetitiv, wenn  $H_c$  für alle möglichen Anfragesequenzen kleiner oder gleich 0 ist.<sup>2</sup>

**Beispiel 1.7.** Betrachten wir noch einmal das Spiel aus Beispiel 1.1, so sehen wir, dass dieses im Grunde bereits die Modellierung eines Online-Optimierungsproblems ist:

Die Anfragesequenzen<sup>3</sup> sind die Folgen natürlicher Zahlen und unsere Aufgabe ist es die kleinste Zahl (bzw. zumindest eine möglichst kleine) darin zu finden. Es handelt sich hierbei auch tatsächlich um ein Online-Problem, da wir die Sequenz nur stückweise erfahren und immer sofort eine Entscheidung treffen müssen (ist das die kleinste Zahl oder warten wir noch auf eine kleinere?).

Das beschriebene Auszahlungsverfahren (wir zahlen unsere Zahl -  $1,5 \times$  die kleinste Zahl an Spieler 2) entspricht gerade der Wahl  $c = 1,5$  in der oben beschriebenen Auszahlungsfunktion (der optimale Offline-Algorithmus ist sicher immer in der Lage die tatsächlich kleinste Zahl der Folge zu finden).

Unsere Suche nach einer guten Spielstrategie übersetzt sich damit in die Suche nach einem guten Online-Algorithmus. Die Frage, ob wir erwarten können in diesem Spiel Gewinn zu machen, wird zur Frage, ob es für dieses Problem einen  $1,5$ -kompetitiven Online-Algorithmus gibt.

Anstatt vor jedem Spiel eine neue Strategie wählen zu müssen, wollen wir den nun Spielern auch erlauben ihre Strategie zu Beginn des Spiels zufällig zu bestimmen. Dazu legen die Spieler eine Wahrscheinlichkeitsverteilung über den ihnen zur Verfügung stehenden

---

<sup>2</sup>Für endliche Online-Optimierungsprobleme betrachten wir hier ausschließlich *strikte*  $c$ -Kompetitivität. Durch Zulassen einer additiven Konstante  $\alpha$  könnten wir sonst trivialerweise immer jede beliebige Kompetitivität erreichen (durch Setzen von  $\alpha$  auf die maximalmöglichen Kosten des gerade betrachteten Online-Algorithmus).

<sup>3</sup>Um die Endlichkeitsbedingung zu erfüllen sollten wir die Menge der Anfragesequenzen hier noch weiter einschränken - etwa indem wir fordern, dass diese Länge  $k$  haben müssen und nur Zahlen aus  $\{1, \dots, n\}$  enthalten dürfen (für feste  $k$  und  $n$ ). Die Menge aller deterministischen Algorithmen ist dann automatisch endlich (unter der - sicher sinnvollen - Annahme, dass wir zwei Algorithmen nur als verschieden ansehen wollen, wenn sie auf wenigstens einer Folge zu unterschiedlichen Ergebnissen führen)

reinen Strategien fest. Die tatsächliche Strategie wird dann zu Beginn des Spiels gemäß der Verteilung zufällig ausgewählt.

Da die Wahl der Strategien und damit auch die Spielabläufe nun ein zufälliges Element beinhalten, können wir dann nicht mehr von „der Auszahlung“ sprechen, sondern sollten die erwartete Auszahlung betrachten, d.h. den Erwartungswert gebildet über der Auszahlungsfunktion.

**Definition 1.8.** Für ein Zwei-Personen-Nullsummenspiel definieren wir:

- Eine *gemischte Strategie für Spieler 1* ist eine Wahrscheinlichkeitsverteilung  $p$  über den ihm zur Verfügung stehenden reinen Strategien. Wir können damit  $p$  als durch  $I$  indizierten Vektor  $(p_i)_{i \in I}$  auffassen, wobei  $p_i$  die Wahrscheinlichkeit bezeichnet, dass die Strategie  $s_i^1$  ausgewählt wird.
- Analog definieren wir eine *gemischte Strategie für Spieler 2* als  $(q_j)_{j \in J}$ .
- Für zwei gemischte Strategien  $p$  und  $q$  ist die *erwartete Auszahlung* dann:

$$\mathbf{H}(p, q) := \mathbf{E}_{p,q}[H(i, j)] := \sum_{i \in I, j \in J} p_i q_j H(i, j)$$

*Bemerkung 1.9.* Man könnte nun vermuten, dass die Einschränkung nur einmal zu Beginn den Zufall entscheiden zu lassen, und danach deterministisch zu spielen eine recht starke Einschränkung ist. Insbesondere dann, wenn wir gerade ein Online-Optimierungsproblem modellieren wollen - man kann sich schließlich leicht Algorithmen vorstellen, die auch zur Ausführungszeit noch weitere Zufallsentscheidungen treffen. Im Allgemeinen ist dies auch so.

Es gibt aber bestimmte Klassen von Spielen, für die das nicht so ist (in denen es also zu jeder denkbaren Strategie auch eine mindestens ebenso gute gemischte Strategie nach obiger Definition gibt). Eine solche Klasse bilden die Spiele mit perfektem Gedächtnis (siehe [BEY98, Theorem 6.3]). Für Online-Optimierungsprobleme ist die entsprechende Bedingung dabei schon dann erfüllt, wenn wir davon ausgehen, dass die Algorithmen (sowohl Online als auch Offline) über unbegrenzten Speicher verfügen können - was wir im Folgenden auch tun wollen.

*Bemerkung 1.10.* Auf die als Nullsummenspiele modellierten Online-Optimierungsprobleme überträgt sich die Einführung des Zufalls wie folgt:

- Eine gemischte Strategie  $p$  von Spieler 1 entspricht hier einem randomisierten Algorithmus  $\text{ALG}_p$ , der zu Beginn einmal anhand der Verteilung  $p$  ein zufälliges  $i$  aus  $I$  zieht und dann den entsprechenden deterministischen Algorithmus  $\text{ALG}_i$  ausführt.<sup>4</sup>

---

<sup>4</sup>Unter der in Bemerkung 1.9 beschriebenen Annahme erhalten wir auf diesem Weg effektiv *alle* randomisierten Online-Algorithmen (bzw. zumindest jeweils äquivalente). Wir können mit dieser Modellierung daher tatsächlich Aussagen über alle randomisierten Online-Algorithmen treffen.

Umgekehrt heißt dies allerdings auch: Untersuchen wir ein Online-Optimierungsproblem, in dem besagte Annahme verletzt ist (in dem also der interne Speicherplatz der Algorithmen begrenzt ist), so ist die Gültigkeit der hier gezeigten Aussagen nicht mehr gewährleistet!

- Einer gemischten Strategie von Spieler 2 entspricht, dass die Anfragesequenz zu Beginn zufällig gemäß einer Wahrscheinlichkeitsverteilung über allen möglichen Anfragesequenzen  $\{\sigma_j \mid j \in J\}$  gewählt wird.
- Die erwartete Auszahlungsfunktion ist dann gegeben durch:

$$\mathbf{H}_c(p, q) := \mathbf{E}_{p,q}[\text{ALG}_i(\sigma_j) - c \cdot \text{OPT}(\sigma_j)] = \mathbf{E}_{p,q}[\text{ALG}_i(\sigma_j)] - c \cdot \mathbf{E}_q[\text{OPT}(\sigma_j)]$$

Beachte, dass das Ergebnis des optimalen Offline-Algorithmus natürlich unabhängig von der Zufallsentscheidung des Online-Algorithmus ist.

*Beobachtung 1.11.* Die gemischten Strategien beinhalten insbesondere auch wieder alle reinen Strategien. Denn wir können jede reine Strategie  $k$  mit der Wahrscheinlichkeitsverteilung  $e^{(k)} := (\delta_{ki})_{i \in I}$  identifizieren, die zu 100% die reine Strategie  $k$  auswählt.

*Beobachtung 1.12.* Auch die erwartete Auszahlungsfunktion lässt sich wieder im Rahmen der Online-Optimierung interpretieren: Ein randomisierter Algorithmus  $\text{ALG}_p$  ist genau dann  $c$ -kompetitiv, wenn  $\mathbf{H}_c$  für alle möglichen (deterministischen!) Anfragesequenzen kleiner oder gleich 0 ist, d.h.

$$\text{ALG}_p \text{ ist } c\text{-komp.} \iff 0 \geq \max_{j \in J} \mathbf{H}_c(p, e^{(j)}) = \max_{j \in J} (\mathbf{E}_p[\text{ALG}_i(\sigma_j)] - c \cdot \text{OPT}(\sigma_j)) \quad (1)$$

Für einen gegebenen Algorithmus  $\text{ALG}_p$  ist damit

$$c := \max_{j \in J} \frac{\mathbf{E}_p[\text{ALG}_i(\sigma_j)]}{[\text{OPT}(\sigma_j)]}$$

gerade dessen Kompetitivitätsverhältnis.

Eine untere Schranke für die Kompetitivität aller Algorithmen in einem gegebenen Online-Optimierungsproblem zu finden ist damit äquivalent dazu ein  $c \geq 1$  zu finden, für das wir zeigen können, dass  $\max_{j \in J} (\mathbf{E}_p[\text{ALG}_i(\sigma_j)] - c \cdot \text{OPT}(\sigma_j))$  für jede Verteilung  $p$  (also jeden randomisierten Online-Algorithmus  $\text{ALG}_p$ ) größer oder gleich 0 ist. Anders gesagt: Wir wollen ein  $c \geq 1$  finden, sodass gilt:

$$\min_p \max_{j \in J} (\mathbf{E}_p[\text{ALG}_i(\sigma_j)] - c \cdot \text{OPT}(\sigma_j)) \geq 0 \quad (2)$$

Dazu wollen wir nun zunächst das analoge Problem im allgemeinen spieltheoretischen Umfeld betrachten:

## 2 Sattelpunkte und Wert eines Spiels

Gegeben sei ein beliebiges 2-Personen Nullsummenspiel. Vorausgesetzt beide Spieler spielen absolut rational, welche Auszahlung sollten wir dann erwarten? Oder, etwas schwächer formuliert: Mit welcher Auszahlung können die beiden Spieler bei ausreichend guter Spielweise mindestens rechnen?

Um zu versuchen diese Frage zu beantworten, wollen wir das Spiel nun aus folgender „pessimistischen“ Sichtweise der beiden Spieler betrachten:

- Spieler 1 versucht seine maximalen Kosten zu minimieren. Daher betrachtet er nun für jede seiner Strategien  $p$  den schlimmstmöglichen Fall, also die Auszahlung  $\max_q \mathbf{H}(p, q)$ . Seine Strategie wählt er daher so, dass diese größtmögliche Auszahlung möglichst klein wird. Der daraus resultierende Wert

$$\min_p \max_q \mathbf{H}(p, q)$$

ist damit offenbar eine obere Schranke für die kleinste Auszahlung, die Spieler 1 sicherstellen kann.

- Analog versucht Spieler 2 die minimalen Kosten für Spieler 1 zu maximieren. Er wählt seine Strategie  $q$  daher so, dass die kleinstmögliche Auszahlung  $\min_p \mathbf{H}(p, q)$  möglichst groß wird. Der Wert

$$\max_q \min_p \mathbf{H}(p, q)$$

wird somit zu einer unteren Schranke für die höchste Auszahlung, die Spieler 2 sicherstellen kann.

*Bemerkung 2.1.* Alle oben auftauchenden Minima bzw. Maxima sind wohldefiniert, denn die Menge der Wahrscheinlichkeitsverteilungen über den endlichen Mengen  $I$  bzw.  $J$  ist kompakt und  $\mathbf{H}$  stetig. Also nimmt  $\mathbf{H}$  beide Extremwerte an.

*Beobachtung 2.2.* Es ist nun leicht zu sehen, dass allgemein gilt

$$\min_p \max_q \mathbf{H}(p, q) \geq \max_q \min_p \mathbf{H}(p, q),$$

d.h. die worst-case Kosten, mit denen Spieler 1 rechnet, sind mindestens so hoch, wie die worst-case Kosten, mit denen Spieler 2 rechnet.

*Beweis.* Für alle Strategien  $\tilde{p}$  und  $\tilde{q}$  gilt sicher, dass die beste Konter-Strategie  $q$  von Spieler 2 gegen  $\tilde{p}$  eine mindestens so hohe Auszahlung liefert wie  $\tilde{q}$ . Ebenso führt die beste Konter-Strategie  $p$  von Spieler 1 gegen  $\tilde{q}$  zu einer höchstens so hohen Auszahlung wie  $\tilde{p}$ . Zusammen erhalten wir also für alle Strategien  $\tilde{p}$  und  $\tilde{q}$ :

$$\max_q \mathbf{H}(\tilde{p}, q) \geq \mathbf{H}(\tilde{p}, \tilde{q}) \geq \min_p \mathbf{H}(p, \tilde{q})$$

Insbesondere gilt also für alle Strategien  $\tilde{p}$  und  $\tilde{q}$

$$\max_q \mathbf{H}(\tilde{p}, q) \geq \mathbf{H}(\tilde{p}, \tilde{q}) \geq \min_p \mathbf{H}(p, \tilde{q})$$

und daher auch (bei Wahl der entsprechend besten Strategien)

$$\min_p \max_q \mathbf{H}(p, q) \geq \max_q \min_p \mathbf{H}(p, q).$$

□

Interessant wäre es nun zu wissen, wann hier sogar Gleichheit gilt - wann also die von Spieler 1 und Spieler 2 durch ihre worst-case-Analysen gefundenen Schranken übereinstimmen. Ist dies nämlich der Fall, so ist der dadurch erhaltene Wert wohl ein guter Kandidat für die Auszahlung, die wir bei optimaler Spielweise beider Spieler erwarten sollten. Wir definieren daher:

**Definition 2.3.** Gilt in einem Spiel  $\min_p \max_q \mathbf{H}(p, q) = \max_q \min_p \mathbf{H}(p, q) =: v$ , so bezeichnen wir  $v$  als *den Wert des Spiels*.

Um nun die Frage zu beantworten, wann diese Gleichheit tatsächlich auftritt, wollen wir noch eine alternative Spielweise betrachten: Spieler 1 wählt zunächst eine beliebige Strategie und verrät diese Spieler 2. Dieser wählt dann seine beste Konterstrategie. Daraufhin (also zum Beispiel im nächsten Spiel) wählt Spieler 1 die Strategie, die den besten Konter zu Spieler 2s Konterstrategie darstellt. Als nächstes passt wieder Spieler 2 seine Strategie an usw.

Eine naheliegende Frage ist dann, ob sich dieses Verfahren jemals stabilisieren kann, d.h. ob es eine Kombination von gemischten Strategien gibt, sodass sich für keinen der beiden Spieler ein weiterer (einseitiger) Strategiewechsel lohnt. Gibt es eine solche Kombination, wollen wir diese als Sattelpunkt bezeichnen:

**Definition 2.4.** Ein Tupel  $(\tilde{p}, \tilde{q})$  von gemischten Strategien heißt *Sattelpunkt*, wenn gilt:

$$\text{Für alle gemischte Strategien } p, q \text{ gilt : } \mathbf{H}(\tilde{p}, q) \leq \mathbf{H}(\tilde{p}, \tilde{q}) \leq \mathbf{H}(p, \tilde{q})$$

D.h. in einem Sattelpunkt gilt: Spieler 1 kann die Auszahlung durch einen Strategiewechsel höchstens erhöhen, während Spieler 2 die Auszahlung durch einen Strategiewechsel höchstens senken kann. Eine solche Situation bezeichnet man auch als *Nash-Gleichgewicht*.

Offenkundig wäre nun auch die Auszahlung  $\mathbf{H}(\tilde{p}, \tilde{q})$  in einem Sattelpunkt ein naheliegender Kandidat für die Auszahlung, die wir bei vernünftiger Spielweise beider Spieler erwarten können. Und tatsächlich stimmen diese beiden Werte sogar überein, wie folgendes Lemma zeigen wird:

**Lemma 2.5.** *Für jedes endliche Zwei-Personen-Nullsummenspiel sind äquivalent:*

- (a) *Das Spiel hat einen Sattelpunkt  $(\tilde{p}, \tilde{q})$*
- (b) *Das Spiel hat einen Wert, d.h.  $\min_p \max_q \mathbf{H}(p, q) = \max_q \min_p \mathbf{H}(p, q)$*

*Beweis.* „(a) $\Rightarrow$ (b)“: Ist  $(\tilde{p}, \tilde{q})$  ein Sattelpunkt, so gilt nach Definition:

$$\begin{aligned} & \forall p, q : \quad \mathbf{H}(\tilde{p}, q) \leq \mathbf{H}(\tilde{p}, \tilde{q}) \leq \mathbf{H}(p, \tilde{q}) \\ \Rightarrow & \quad \max_q \mathbf{H}(\tilde{p}, q) \leq \mathbf{H}(\tilde{p}, \tilde{q}) \leq \min_p \mathbf{H}(p, \tilde{q}) \\ \Rightarrow & \quad \min_p \max_q \mathbf{H}(p, q) \leq \mathbf{H}(\tilde{p}, \tilde{q}) \leq \min_p \mathbf{H}(p, \tilde{q}) \\ \Rightarrow & \quad \min_p \max_q \mathbf{H}(p, q) \leq \mathbf{H}(\tilde{p}, \tilde{q}) \leq \max_q \min_p \mathbf{H}(p, q) \end{aligned}$$



Zusammen mit Beobachtung 2.2 folgt daraus

$$\min_p \max_q \mathbf{H}(p, q) = \mathbf{H}(\tilde{p}, \tilde{q}) = \max_q \min_p \mathbf{H}(p, q)$$

und daher gilt (b).

„(b) $\Rightarrow$ (a)“: Es gelte  $\min_p \max_q \mathbf{H}(p, q) = \max_q \min_p \mathbf{H}(p, q) =: v$ . Dann gibt es Strategien  $\tilde{p}$  und  $\tilde{q}$ , an denen das „äußere“ Minimum bzw. Maximum angenommen wird:

$$\max_q \mathbf{H}(\tilde{p}, q) = v = \min_p \mathbf{H}(p, \tilde{q})$$

Daraus wiederum erhalten wir für beliebige Strategien  $p$  und  $q$  die folgende Ungleichung:

$$\mathbf{H}(\tilde{p}, q) \leq v \leq \mathbf{H}(p, \tilde{q})$$

Insbesondere gelten also (für alle  $p$  und  $q$ ):

$$\mathbf{H}(\tilde{p}, q) \leq v \leq \mathbf{H}(\tilde{p}, \tilde{q}) \text{ und } \mathbf{H}(\tilde{p}, \tilde{q}) \leq v \leq \mathbf{H}(p, \tilde{q})$$

und damit ist  $(\tilde{p}, \tilde{q})$  ein Sattelpunkt. □

*Beobachtung 2.6.* Aus dem Beweis zu „(a) $\Rightarrow$ (b)“ folgt insbesondere, dass die Auszahlung am Sattelpunkt gerade dem Wert des Spiels entspricht.

*Bemerkung 2.7.* Beachte, dass alle bisherigen Definition und Aussagen auch genauso für den Fall gelten, dass wir nur reine Strategien erlauben.

**Beispiel 2.8.** Wir schränken nun Beispiel 1.7 noch weiter ein, indem wir unserem Gegenspieler nur noch erlauben Folgen der Länge 2 aus den Zahlen 1, 2 und 3 zu bilden. Ein deterministischer Online-Algorithmus ist dann bereits dadurch eindeutig bestimmt, wann er sich für die erste Zahl entscheidet (und wann für die zweite). Bspw. sei also  $\text{ALG}_{\{1,3\}}$  der Algorithmus, der sich für die erste Zahl entscheidet, falls diese 1 oder 3 ist und sonst auf die zweite Zahl wartet.

Schreiben wir nun alle möglichen Anfragesequenzen als Spalten und alle deterministischen Algorithmen als Zeilen, so erhalten wir die folgende Tabelle:

	1, 2	1, 3	2, 1	2, 3	3, 1	3, 2
$\text{ALG}_{\emptyset}$	2/1	3/1	1/1	3/2	1/1	2/2
$\text{ALG}_{\{1\}}$	1/1	1/1	1/1	3/2	1/1	2/2
$\text{ALG}_{\{2\}}$	2/1	3/1	2/1	2/2	1/1	2/2
$\text{ALG}_{\{3\}}$	2/1	3/1	1/1	3/2	3/1	3/2
$\text{ALG}_{\{1,2\}}$	1/1	1/1	2/1	2/2	1/1	2/2
$\text{ALG}_{\{1,3\}}$	1/1	1/1	1/1	3/2	3/1	3/2
$\text{ALG}_{\{2,3\}}$	2/1	3/1	2/1	2/2	3/1	3/2
$\text{ALG}_{\{1,2,3\}}$	1/1	1/1	2/1	2/2	3/1	3/2

Zur Notation der Auszahlung verwenden wir hierbei folgende Konvention: Anstatt den tatsächlichen Wert der Auszahlungsfunktion (also  $H_{1,5}(i, j) = \text{ALG}_i(\sigma_j) - 1,5 \cdot \text{OPT}(\sigma_j)$ ) in die Zellen der Matrix zu schreiben, notieren wir dort jeweils den Wert  $\frac{\text{ALG}_i(\sigma_j)}{\text{OPT}(\sigma_j)}$ . Offenbar machen wir in einem Spiel also genau dann Gewinn, wenn in der entsprechenden Zelle ein Eintrag kleiner 1,5 steht.

Diese Tabelle können wir nun deutlich vereinfachen, indem wir alle Algorithmen weglassen, für die es einen anderen Algorithmus gibt, der auf allen Anfragesequenzen mindestens genauso gut ist (dadurch verlieren wir offenkundig keine nützlichen Strategien):

- $\text{ALG}_{\{1\}}$  ist besser als  $\text{ALG}_{\emptyset}$ ,  $\text{ALG}_{\{3\}}$  und  $\text{ALG}_{\{1,3\}}$
- $\text{ALG}_{\{1,2\}}$  ist besser als  $\text{ALG}_{\{2\}}$ ,  $\text{ALG}_{\{1,2,3\}}$  und  $\text{ALG}_{\{2,3\}}$

Somit verbleibt folgende Tabelle:

	1, 2	1, 3	2, 1	2, 3	3, 1	3, 2
$\text{ALG}_{\{1\}}$	1/1	1/1	1/1	3/2	1/1	2/2
$\text{ALG}_{\{1,2\}}$	1/1	1/1	2/1	2/2	1/1	2/2

Ganz analog können wir jetzt „schlechte“ Anfragesequenzen weglassen (schlecht aus Sicht des Gegenspielers!) und erhalten dann:

	2, 1	2, 3
$\text{ALG}_{\{1\}}$	1/1	3/2
$\text{ALG}_{\{1,2\}}$	2/1	2/2

Wollen wir nun unsere Strategie nach dem Minimax-Vorgehen auswählen, so können wir das also auch in dieser vereinfachten Tabelle tun und sehen hier leicht, dass wir damit auf einen Worst-Case-Wert von 3/2 kommen (für den ersten Algorithmus).

Das entsprechende Maximin-Vorgehen aus Sicht unseres Gegenspielers hingegen führt zu einem Wert von 1/1 (für beide Anfragesequenzen). Wir sehen also, dass dieses Spiel bei ausschließlicher Verwendung reiner Strategien keinen Wert hat.

Ebenfalls ist nun leicht zu sehen, dass dieses Spiel auch keinen Sattelpunkt aus reinen Strategien besitzt. Ob sich dies ändert wenn wir auch wieder gemischte Strategien zulassen, werden wir in Abschnitt 3 sehen.

Zuvor wollen wir jedoch noch den folgenden Zusammenhang zwischen gemischten und reinen Strategien beweisen:

*Beobachtung 2.9.* Kennt Spieler 1 die von seinem Gegenspieler gewählte gemischte Strategie  $q$ , so ist unter seinen besten Konterstrategien auch mindestens eine reine Strategie. Es gilt also:

$$\min_p \mathbf{H}(p, q) = \min_i \mathbf{H}(e^{(i)}, q)$$

Analog gilt dies für Spieler 2: Für jede gemischte Strategie  $p$  von Spieler 1 ist

$$\max_q \mathbf{H}(p, q) = \max_j \mathbf{H}(p, e^{(j)}).$$

*Beweis.* Da jede gemischte Strategie  $p$  aus reinen Strategien zusammengesetzt ist, gilt einerseits sicher

$$\mathbf{H}(p, q) = \sum_{k \in I} p_k \mathbf{H}(e^{(k)}, q) \geq \sum_{k \in I} p_k \min_i \mathbf{H}(e^{(i)}, q) = \min_i \mathbf{H}(e^{(i)}, q)$$

und daher auch

$$\min_p \mathbf{H}(p, q) \geq \min_i \mathbf{H}(e^{(i)}, q).$$

Da andererseits jede reine Strategie insbesondere auch eine gemischte Strategie ist, gilt ebenso

$$\min_p \mathbf{H}(p, q) \leq \min_i \mathbf{H}(e^{(i)}, q)$$

und daher insgesamt

$$\min_p \mathbf{H}(p, q) = \min_i \mathbf{H}(e^{(i)}, q)$$

Völlig analog ergibt sich die Aussage für Spieler 2. □

Wie die folgende Proposition zeigt, macht es uns diese Beobachtung einfacher die Existenz eines Sattelpunkts zu zeigen:

**Proposition 2.10.** *Ein Tupel  $(\tilde{p}, \tilde{q})$  ist genau dann ein Sattelpunkt, wenn für alle reinen Strategien  $i$  und  $j$  gilt:*

$$\mathbf{H}(\tilde{p}, e^{(j)}) \leq \mathbf{H}(\tilde{p}, \tilde{q}) \leq \mathbf{H}(e^{(i)}, \tilde{q})$$

*Beweis.* Es gilt:  $(\tilde{p}, \tilde{q})$  ist Sattelpunkt

$$\begin{aligned} &\iff \forall p, q : \quad \mathbf{H}(\tilde{p}, q) \leq \mathbf{H}(\tilde{p}, \tilde{q}) \leq \mathbf{H}(p, \tilde{q}) \\ &\iff \max_q \mathbf{H}(\tilde{p}, q) \leq \mathbf{H}(\tilde{p}, \tilde{q}) \leq \min_p \mathbf{H}(p, \tilde{q}) \\ &\stackrel{\text{Beobachtung 2.9}}{\iff} \max_j \mathbf{H}(\tilde{p}, e^{(j)}) \leq \mathbf{H}(\tilde{p}, \tilde{q}) \leq \min_i \mathbf{H}(e^{(i)}, \tilde{q}) \\ &\iff \forall i, j : \quad \mathbf{H}(\tilde{p}, e^{(j)}) \leq \mathbf{H}(\tilde{p}, \tilde{q}) \leq \mathbf{H}(e^{(i)}, \tilde{q}) \end{aligned}$$

□

Damit können wir uns nun endlich der Frage zuwenden, wann überhaupt ein Sattelpunkt existiert. Eine Frage, die das zentrale Theorem des folgenden Kapitels beantworten wird.

### 3 Das Minimax-Theorem

**Satz 3.1** (von Neumann). *Jedes endliche Zwei-Personen-Nullsummenspiel hat einen Wert, d.h. es gilt:*

$$\min_p \max_q \mathbf{H}(p, q) = \max_q \min_p \mathbf{H}(p, q)$$

Für den Beweis dieses Theorems werden wir ohne eigenen Beweis die folgende Version des Brouwerschen Fixpunktsatzes verwenden:

**Satz 3.2.** *Sei  $X \subseteq \mathbb{R}^n$  eine kompakte und konvexe Menge. Dann hat jede stetige Abbildung von  $X$  auf sich selbst einen Fixpunkt,*

*d.h. für jede stetige Abbildung  $\phi : X \rightarrow X$  gibt es einen Punkt  $x \in X$  mit  $\phi(x) = x$ .*

*Beweis des Minimax-Theorems.* Wegen Lemma 2.5 genügt es zu zeigen, dass jedes Spiel einen Sattelpunkt besitzt und wegen Proposition 2.10 reicht es dazu zwei gemischte Strategien  $\tilde{p}$  und  $\tilde{q}$  zu finden, sodass für alle reinen Strategien  $i$  und  $j$  gilt:

$$\mathbf{H}(\tilde{p}, e^{(j)}) \leq \mathbf{H}(\tilde{p}, \tilde{q}) \leq \mathbf{H}(e^{(i)}, \tilde{q})$$

Oder äquivalent dazu:

$$\begin{aligned} D_i^1(\tilde{p}, \tilde{q}) &:= \mathbf{H}(\tilde{p}, \tilde{q}) - \mathbf{H}(e^{(i)}, \tilde{q}) \leq 0 && \text{für alle } i \in I \text{ und} \\ D_j^2(\tilde{p}, \tilde{q}) &:= \mathbf{H}(\tilde{p}, e^{(j)}) - \mathbf{H}(\tilde{p}, \tilde{q}) \leq 0 && \text{für alle } j \in J \end{aligned} \quad (3)$$

Seien nun  $X \subset \mathbb{R}^I$ ,  $Y \subset \mathbb{R}^J$  die Mengen aller gemischten Strategien. Diese sind kompakt und konvex ( $I$  und  $J$  sind endlich!), also ebenso  $X \times Y$ . Auf dieser Menge definieren wir die folgende Abbildung

$$\phi : X \times Y \rightarrow X \times Y : (p, q) = ((p_k)_{k \in I}, (q_l)_{l \in J}) \mapsto (r, s) := ((r_k)_{k \in I}, (s_l)_{l \in J})$$

wobei die Einträge von  $r \in X$  und  $s \in Y$  wie folgt definiert sind:

$$\begin{aligned} r_k &:= \frac{p_k + \max\{D_k^1(p, q), 0\}}{1 + \sum_{i \in I} \max\{D_i^1(p, q), 0\}} \\ s_l &:= \frac{q_l + \max\{D_l^2(p, q), 0\}}{1 + \sum_{j \in J} \max\{D_j^2(p, q), 0\}} \end{aligned}$$

Diese Abbildung ist

- wohldefiniert (d.h.  $r$  und  $s$  sind Wahrscheinlichkeitsverteilungen über  $I$  bzw.  $J$ ), denn alle  $r_k$  sind offensichtlich nicht-negativ und es gilt

$$\sum_{k \in I} r_k = \frac{\sum_{k \in I} p_k + \sum_{k \in I} \max\{D_k^1(p, q), 0\}}{1 + \sum_{i \in I} \max\{D_i^1(p, q), 0\}} = 1$$

(analog für  $s$ )

- und stetig als Verknüpfung stetiger Abbildungen (Maximum, Addition, Inversenbildung und Produkt).

Also ist der Brouwersche Fixpunktsatz anwendbar und es gibt ein Tupel  $(\tilde{p}, \tilde{q})$  mit  $\phi((\tilde{p}, \tilde{q})) = (\tilde{p}, \tilde{q})$ .

Wir wollen nun zeigen, dass dieses Tupel ein Sattelpunkt ist. Dazu betrachten wir die Gleichung  $\phi((\tilde{p}, \tilde{q})) = (\tilde{p}, \tilde{q})$  komponentenweise, d.h. für  $k \in I$  gilt

$$\tilde{p}_k + \max\{D_k^1(\tilde{p}, \tilde{q}), 0\} = \tilde{p}_k \cdot \left(1 + \sum_{i \in I} \max\{D_i^1(\tilde{p}, \tilde{q}), 0\}\right),$$

also

$$\max\{D_k^1(\tilde{p}, \tilde{q}), 0\} = \tilde{p}_k \cdot \sum_{i \in I} \max\{D_i^1(\tilde{p}, \tilde{q}), 0\}.$$

Falls wir nun ein  $k \in I$  finden können, dass  $D_k^1(\tilde{p}, \tilde{q}) \leq 0$  und  $\tilde{p}_k > 0$  gelten (\*), dann ist

$$0 = \max\{D_k^1(\tilde{p}, \tilde{q}), 0\} = \underbrace{\tilde{p}_k}_{>0} \cdot \sum_{i \in I} \underbrace{\max\{D_i^1(\tilde{p}, \tilde{q}), 0\}}_{\geq 0}.$$

Also muss für alle  $i \in I$  gelten

$$0 \geq D_i^1(\tilde{p}, \tilde{q}) = \mathbf{H}(\tilde{p}, \tilde{q}) - \mathbf{H}(e^{(i)}, \tilde{q}),$$

was (zusammen mit einem analogen Beweis für  $D_j^2(\tilde{p}, \tilde{q})$ ) gerade zeigt, dass  $(\tilde{p}, \tilde{q})$  ein Sattelpunkt ist (siehe Gleichung (3)).

Es bleibt also noch zu zeigen, dass es tatsächlich ein  $k \in I$  mit  $D_k^1(\tilde{p}, \tilde{q}) \leq 0$  und  $\tilde{p}_k > 0$  gibt. Dazu nehmen wir an es gäbe kein solches  $k$ , d.h. wann immer  $\tilde{p}_k > 0$  gilt, so folgt bereits  $D_k^1(\tilde{p}, \tilde{q}) > 0$ , also  $\mathbf{H}(\tilde{p}, \tilde{q}) > \mathbf{H}(e^{(k)}, \tilde{q})$ . Dann würde jedoch auch gelten:

$$\mathbf{H}(\tilde{p}, \tilde{q}) = \sum_{k \in I} \tilde{p}_k \cdot \mathbf{H}(e^{(k)}, \tilde{q}) < \sum_{k \in I} \tilde{p}_k \cdot \mathbf{H}(\tilde{p}, \tilde{q}) = \mathbf{H}(\tilde{p}, \tilde{q}) \cdot \sum_{k \in I} \tilde{p}_k = \mathbf{H}(\tilde{p}, \tilde{q}) \quad \not\leq$$

Die Annahme war also falsch und es muss tatsächlich ein  $k$  mit den in (\*) gewünschten Eigenschaften geben.  $\square$

**Beispiel 3.3.** In Beispiel 2.8 haben wir gesehen, dass unser eingeschränktes „kleinste Zahl Finden“-Spiel keinen Sattelpunkt besitzt, wenn wir nur reine Strategien zulassen. Erlauben wir hingegen auch gemischte Strategien, so muss es nach obigem Theorem einen solchen Sattelpunkt geben.

Wir wissen bereits, dass die Folgen (2, 1) und (2, 3) sowie die Algorithmen  $\text{ALG}_{\{1\}}$  und  $\text{ALG}_{\{1,2\}}$  die einzigen „interessanten“ sind. Es bietet sich daher an auch unsere gemischten Strategien jeweils nur aus diesen beiden reinen Strategien zusammenzusetzen.

Jede derartige gemischte Strategie können wir daher durch die Wahl einer einzigen Wahrscheinlichkeit  $p$  bzw.  $q$  wie folgt festlegen:

$$ALG_p = p \cdot ALG_{\{1\}} + (1 - p) \cdot ALG_{\{1,2\}}$$

$$\sigma_q = q \cdot (2, 1) + (1 - q) \cdot (2, 3)$$

Zeichnen wir nun erneut die entsprechende „Tabelle“, so sieht diese wie folgt aus (erneut mit der Konvention, dass wir in die Zellen das Verhältnis der Erwartungswerte  $\frac{ALG_p(\sigma_q)}{OPT(\sigma_q)}$  schreiben):

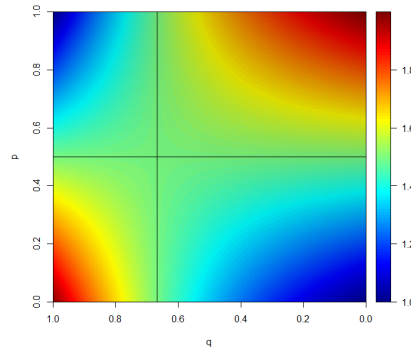


Abbildung 2: Die Werte  $\frac{ALG_p(\sigma_q)}{OPT(\sigma_q)}$  - kleine Werte sind blau, große sind rot (siehe Legende rechts)

Wir „sehen“ nun, dass unser Spiel tatsächlich einen Sattelpunkt hat - nämlich den Punkt für den weder eine Verschiebung in vertikaler Richtung (Strategiewechsel von Spieler 1) noch eine in horizontaler Richtung (Strategiewechsel von Spieler 2) zu einer Veränderung des resultierenden Wertes führt.

An dieser Stelle nimmt das Verhältnis  $\frac{ALG_p(\sigma_q)}{OPT(\sigma_q)}$  den Wert 1,5 an - und dementsprechend ist die erwartete Auszahlung bei Wahl dieser beiden Strategien gerade 0.

Fassen wir die Werte in Abbildung 2 als Höhenangaben auf, so ist dieser Punkt auch tatsächlich ein „anschaulicher“ Sattel, wie wir an folgendem Plot sehen können:

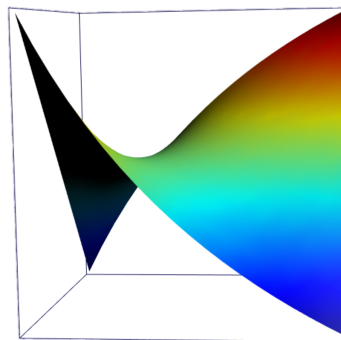


Abbildung 3: Der Sattelpunkt dieses Spiels

Wenn unsere Modellierung des Online-Optimierungsproblems als Nullsummenspiel „vernünftig“ war, sollten wir nun erwarten, dass wir mit dem Sattelpunkt auch einen optimalen Algorithmus (nämlich  $\text{ALG}_p$ ) und eine diese Optimalität beweisende randomisierte Anfragesequenz (nämlich  $\sigma_q$ ) gefunden haben.

Dass dies tatsächlich so ist - sowohl hier als auch im Allgemeinen -, wollen wir nun noch zeigen. Dazu werden wir folgende Konsequenz aus dem Minimax-Theorem benötigen:

**Lemma 3.4** (Loomi). *Für jedes endliche 2-Personen Nullsummenspiel gilt*

$$\max_q \min_i \mathbf{H}(e^{(i)}, q) = \min_p \max_j \mathbf{H}(p, e^{(j)})$$

*Beweis.* Aus Beobachtung 2.9 wissen wir, dass für alle Strategien  $p$  und  $q$  gilt:

$$\min_i \mathbf{H}(e^{(i)}, q) = \min_p \mathbf{H}(p, q) \text{ und } \max_q \mathbf{H}(p, q) = \max_i \mathbf{H}(p, e^{(i)})$$

Insbesondere gilt dies auch für die Strategien, an denen der maximale bzw. minimale Wert angenommen wird:

$$\max_q \min_i \mathbf{H}(e^{(i)}, q) = \max_q \min_p \mathbf{H}(p, q) \text{ und } \min_p \max_q \mathbf{H}(p, q) = \min_p \max_j \mathbf{H}(p, e^{(j)})$$

Die Gleichheit der mittleren beiden Werte (und damit die Aussage des Lemmas) folgt nun direkt aus dem Minimax-Theorem.  $\square$

Mit Loomis Lemma wiederum erhalten wir die folgende Abschätzung:

**Korollar 3.5** (Yaos Ungleichung). *Für jede gemischte Strategie  $q$  gilt*

$$\min_i \mathbf{H}(e^{(i)}, q) \leq \min_p \max_j \mathbf{H}(p, e^{(j)})$$

*und für wenigstens eine gemischte Strategie  $q$  gilt dabei auch Gleichheit.*

Anschaulich besagt diese Abschätzung das Folgende: Angenommen Spieler 2 spielt nur reine Strategien. Dann kann Spieler 1 die Kosten seiner bestmöglichen gemischten Strategie (die er möglicherweise noch gar nicht kennt) wie folgt nach unten abschätzen: Er betrachtet eine gemischte Strategie für Spieler 2 (obwohl dieser ja gar keine gemischten Strategien spielt!) und testet jede seiner reinen Strategien gegen diese. Die Kosten seiner besten reinen Strategie gegen die gewählte gemischte Strategie seines Gegners sind dann eine untere Schranke für die Kosten seiner bestmöglichen gemischten Strategie gegen die schlimmste reine Strategie seines Gegners.

Darüber hinaus kann Spieler 1 (zumindest theoretisch) durch geschickte Wahl von  $q$  sogar eine scharfe untere Schranke erhalten. Allerdings ist im Allgemeinen nicht klar, wie eine solches geschicktes  $q$  gefunden werden kann (da unser Beweis zum Minimax-Theorem nicht konstruktiv ist).

Blicken wir nun zurück an den Ausgangspunkt unseres Exkurses in die Spieltheorie, so stellen wir fest, dass wir jetzt in der Lage sind genau die Abschätzung zu treffen, die wir brauchen um untere Schranken für die Kompetitivität zu finden (siehe Gleichung (2)). Es wird also Zeit zur Online-Optimierung zurückzukehren!

## 4 Yaos Prinzip

*Erinnerung 4.1.* Um Online-Optimierungsprobleme als 2-Personen Nullsummenspiele auffassen zu können, hatten wir für eine feste Konstante  $c \geq 1$  die folgende erwartete Auszahlung definiert (für Verteilungen  $p$  über den deterministischen Algorithmen  $\text{ALG}_i$  und  $q$  über den Anfragesequenzen  $\sigma_j$ ):

$$\mathbf{H}_c(p, q) := \mathbf{E}_{p,q} [\text{ALG}_i(\sigma_j) - c \cdot \text{OPT}(\sigma_j)]$$

Außerdem hatten wir gesehen, dass ein durch eine Verteilung  $p$  gegebener Online-Algorithmus genau dann  $c$ -kompetitiv ist, wenn  $\max_{j \in J} \mathbf{H}_c(p, e^{(j)}) \leq 0$  ist (siehe Gleichung (1)).

Setzen wir diese Auszahlungsfunktion in Yaos Ungleichung (Korollar 3.5) ein, so erhalten wir:

$$\min_i \mathbf{E}_q [\text{ALG}_i(\sigma_j) - c \cdot \text{OPT}(\sigma_j)] \leq \min_p \max_j \mathbf{E}_p [\text{ALG}_i(\sigma_j) - c \cdot \text{OPT}(\sigma_j)]$$

oder etwas umgeformt:

$$\min_i \mathbf{E}_q [\text{ALG}_i(\sigma_j)] - c \cdot \mathbf{E}_q [\text{OPT}(\sigma_j)] \leq \min_p \max_j (\mathbf{E}_p [\text{ALG}_i(\sigma_j)] - c \cdot \text{OPT}(\sigma_j))$$

Auf der rechten Seite dieser Gleichung steht nun gerade der Wert, um den der beste randomisierte Online-Algorithmus auf der schlimmstmöglichen Anfragesequenz schlechter ist als  $c$ -mal der optimale Algorithmus. Dieser Wert ist folglich gerade dann kleiner oder gleich 0, wenn der beste Online-Algorithmus wenigstens  $c$ -kompetitiv ist.

Umgekehrt ist die rechte Seite genau dann größer oder gleich 0, wenn der bestmögliche Online-Algorithmus höchstens  $c$ -kompetitiv ist. Ein  $c$ , für das die rechte Seite der Gleichung größer oder gleich 0 ist, ist daher automatisch eine untere Schranke für die Kompetitivität aller Online-Algorithmen. Offensichtlich gilt dies beispielsweise für das  $c$ , welches die linke Seite der Ungleichung zu 0 macht, also

$$c := \frac{\min_i \mathbf{E}_q [\text{ALG}_i(\sigma_j)]}{\mathbf{E}_q [\text{OPT}(\sigma_j)]}.$$

Um eine untere Schranke für die Kompetitivität aller Online-Algorithmen zu finden können wir daher wie folgt vorgehen:

- (a) Wähle eine beliebige Verteilung  $q$  über den Anfragesequenzen.



- (b) Berechne dafür die erwarteten Kosten des besten deterministischen Online-Algorithmus sowie des optimalen Algorithmus.
- (c) Das Verhältnis dieser beiden Werte ist eine untere Schranke für die Kompetitivität aller Online-Algorithmen.

Dieses Vorgehen bezeichnet man auch als Yaos Prinzip:

**Satz 4.2** (Yaos Prinzip für Kostenminimierung in endlichen Spielen). *Sei ein endliches Online-Optimierungsproblem gegeben durch endlich viele deterministische Algorithmen  $\{\text{ALG}_i \mid i \in I\}$  und endlich viele endliche Anfragesequenzen  $\{\sigma_j \mid j \in J\}$ . Ist  $q$  eine beliebige Verteilung über den Anfragesequenzen, so ist*

$$\frac{\min_i \mathbf{E}_q [\text{ALG}_i(\sigma_j)]}{\mathbf{E}_q [\text{OPT}(\sigma_j)]}$$

*eine untere Schranke für die Kompetitivität aller randomisierten Online-Algorithmen.*

*Außerdem gibt es eine Verteilung  $\tilde{q}$ , sodass*

$$\frac{\min_i \mathbf{E}_{\tilde{q}} [\text{ALG}_i(\sigma_j)]}{\mathbf{E}_{\tilde{q}} [\text{OPT}(\sigma_j)]}$$

*die Kompetitivität des besten randomisierten Online-Algorithmus beschreibt.*

*Beweis.* Die Korrektheit der ersten Aussage ist bereits durch die Vorüberlegungen gezeigt.

Die zweite Aussage ergibt sich, wenn wir für  $\tilde{q}$  gerade die Verteilung wählen, für die in Yaos Ungleichung (Korollar 3.5) Gleichheit gilt.  $\square$

Obwohl es natürlich gut zu wissen ist, dass mit Yaos Prinzip grundsätzlich immer eine scharfe untere Schranke gefunden werden kann, wird dieser Teil des Satzes oft weggelassen und schon allein die erste Aussage als Yaos Prinzip bezeichnet.<sup>5</sup> Tatsächlich ist die zweite Aussage (zumindest mit dem hier vorgestellten Beweis) eine reine Existenzaussage und es ist im Allgemeinen nicht klar wie eine entsprechend „geschickte“ Verteilung  $q$  gefunden werden kann.

Insofern ist es für die praktische Anwendung sicher sinnvoll sich vor allem auf die erste Aussage zu konzentrieren. Möchte man nur diese zeigen, gibt es zudem einen etwas einfacheren Beweis ohne Verwendung des Minimax-Theorems und daher auch ohne den den Umweg über die Spieltheorie (nichtsdestotrotz ist es für die Anschauung sicher hilfreich das Problem weiterhin als 2-Personen Spiel aufzufassen):

---

<sup>5</sup>Siehe bspw. [BEY98] und [Wikipedia: Yao's Principle](#).

In der ursprünglichen Arbeit von Andrew Yao hat die entsprechende Aussage ([Yao77, Theorem 1]) mehr die Form von Lemma 3.4 und umfasst damit insbesondere auch die Aussage, dass immer eine Verteilung gefunden werden kann, für die die Schranke scharf ist.

*Alternativer Beweis (Yaos Prinzip - 1. Teil).* Sei  $q$  die gegebene Wahrscheinlichkeitsverteilung über den Anfragesequenzen  $\sigma_j$  und  $\text{ALG}_p$  ein beliebiger randomisierter Online-Algorithmus. Ist dieser  $c$ -kompetitiv, so gilt also für alle  $j$ :

$$c \cdot \text{OPT}(\sigma_j) \geq \mathbf{E}_p [\text{ALG}_i(\sigma_j)]$$

Damit gilt diese Ungleichung auch, wenn wir die Anfragesequenzen gemäß der Verteilung  $q$  gewichten:

$$c \cdot \mathbf{E}_q [\text{OPT}(\sigma_j)] = \sum_{j \in J} q_j \cdot c \cdot \text{OPT}(\sigma_j) \geq \sum_{j \in J} q_j \cdot \mathbf{E}_p [\text{ALG}_i(\sigma_j)] = \mathbf{E}_q \mathbf{E}_p [\text{ALG}_i(\sigma_j)] \quad (4)$$

Wir betrachten nun ein Spiel, in dem der Online-Spieler seinen Algorithmus zufällig gemäß  $p$  bestimmt, während der Offline-Spieler seine Anfragesequenz zufällig gemäß  $q$  bestimmt. Die erwarteten Kosten sind dann  $\mathbf{E}_q [\text{OPT}(\sigma_j)]$  und  $\mathbf{E}_q \mathbf{E}_p [\text{ALG}_i(\sigma_j)]$ .

Da die beiden Entscheidungen sicher (stochastisch) unabhängig voneinander getroffen werden, dürfen wir die Erwartungswerte vertauschen und es gilt:

$$\begin{aligned} \mathbf{E}_q \mathbf{E}_p [\text{ALG}_k(\sigma_j)] &= \mathbf{E}_p \mathbf{E}_q [\text{ALG}_k(\sigma_j)] = \sum_{k \in I} p_k \cdot \mathbf{E}_q [\text{ALG}_k(\sigma_j)] \geq \sum_{k \in I} p_k \cdot \min_i \mathbf{E}_q [\text{ALG}_i(\sigma_j)] \\ &= \min_i \mathbf{E}_q [\text{ALG}_i(\sigma_j)] \cdot \sum_{k \in I} p_k = \min_i \mathbf{E}_q [\text{ALG}_i(\sigma_j)] \cdot 1 \end{aligned}$$

Zusammen mit Gleichung (4) erhalten wir

$$c \cdot \mathbf{E}_q [\text{OPT}(\sigma_j)] \geq \min_i \mathbf{E}_q [\text{ALG}_i(\sigma_j)]$$

und somit

$$c \geq \frac{\min_i \mathbf{E}_q [\text{ALG}_i(\sigma_j)]}{\mathbf{E}_q [\text{OPT}(\sigma_j)]}.$$

Dies ist also tatsächlich eine untere Schranke für die Kompetitivität des Online-Algorithmus  $\text{ALG}_p$ . Und da dieser zu Beginn beliebig gewählt worden war, ist dies insbesondere eine untere Schranke für die Kompetitivität *aller* randomisierten Online-Algorithmen.  $\square$

Man beachte, dass in diesem Beweis nicht ersichtlich ist, ob es auch eine Verteilung  $q$  gibt, für die die erhaltene untere Schranke scharf ist. Mit diesem Beweis lässt sich also wirklich nur der erste Teil von Yaos Prinzip zeigen. Im endlichen Fall ist dieser Beweis also schwächer als der über das Minimax-Theorem. Wie wir im letzten Kapitel sehen werden, hat der direkte Beweis jedoch den Vorteil, dass er sich leichter auf unendliche Online-Optimierungsprobleme verallgemeinern lässt.

## 5 Yaos Prinzip für unendliche Spiele

Nachdem wir uns in den bisherigen Kapiteln auf den Fall eingeschränkt haben, in dem nur endlich viele Anfragesequenzen und endlich viele deterministische Algorithmen zur Auswahl standen, wollen wir nun abschließend noch den Fall betrachten, in dem es unendlich viele Anfragesequenzen gibt und uns evtl. auch beliebig viele deterministische Algorithmen zur Verfügung stehen.

Blicken wir dazu zurück auf den ersten Beweis von Yaos Prinzip, so sehen wir, dass die Endlichkeit nur an einer Stelle eingeht - nämlich als Voraussetzung für das Minimax-Theorem. Eine Möglichkeit Yaos Prinzip zu verallgemeinern wäre also allgemeinere Versionen dieses Theorems zu verwenden, die schwächere Voraussetzungen als die Endlichkeit von  $I$  und  $J$  fordern. Einige Möglichkeiten dafür sind (siehe [BEY98, Abschnitt 8.3]):

- $H$  ist stetig (in  $I$  und  $J$ !)
- $I, J \subseteq \mathbb{R}$  und  $H$  ist konkav in  $I$  und konvex in  $J$ .

Wir wollen nun aber stattdessen den zweiten Teil von Yaos Prinzip fallen lassen, sodass wir den direkten Beweis verwenden können. Da in diesem die Endlichkeit nirgends benötigt wird, lässt sich dieser leicht auf den unendlichen Fall übertragen. Lediglich die zusätzliche Konstante  $\alpha$  in der Definition der Kompetitivität macht eine kleine Zusatzvoraussetzung notwendig (wollen wir nur über strikte Kompetitivität reden, könnten wir diese zusätzliche Bedingung also weglassen):

**Satz 5.1** (Yaos Prinzip für Kostenminimierung in unendlichen Spielen). *Gegeben sei ein unendliches Online-Optimierungsproblem mit den deterministischen Algorithmen  $\{\text{ALG}_i \mid i \in I\}$  und den Anfragesequenzen  $\Sigma := \{\sigma_j \mid j \in J\}$ . Für jedes  $n \in \mathbb{N}$  sei ferner  $\Sigma^n := \{\sigma_j^n \in \Sigma \mid \sigma_j^n \text{ hat Länge } n\} \subseteq \Sigma$  die Menge aller Anfragesequenzen der Länge  $n$  und zu jeder Verteilung  $q$  über  $J$  (bzw.  $\Sigma$ ) sei  $q^n$  die Einschränkung von  $q$  auf  $\Sigma^n$  (mit passender Normierung).*

*Ist nun  $q$  eine Verteilung über den Anfragesequenzen für die die erwarteten Kosten des optimalen Offline-Algorithmus gegen unendlich gehen, d.h.*

$$\lim_{n \rightarrow \infty} \mathbf{E}_q^n [\text{OPT}(\sigma_j^n)] = \infty,$$

*so ist*

$$\limsup_{n \rightarrow \infty} \frac{\inf_i \mathbf{E}_{q^n} [\text{ALG}_i(\sigma_j^n)]}{\mathbf{E}_{q^n} [\text{OPT}(\sigma_j^n)]}$$

*eine untere Schranke für die Kompetitivität aller randomisierten Online-Algorithmen.*<sup>6</sup>

---

<sup>6</sup>Diese Formulierung des Satzes entspricht im Wesentlichen der in [BLS92]. In [BEY98] wird (ohne Beweis) eine etwas schwächere Schranke (Limes inferior statt Limes superior) unter einer leicht schwächeren Voraussetzung (Limes superior statt Limes) angegeben.

*Beweis.* Der Beweis erfolgt ganz analog zum direkten Beweis im endlichen Fall und folgt dem in [BLS92, Lemma 7.2].

Sei  $q$  die gegebene Wahrscheinlichkeitsverteilung über den Anfragesequenzen  $\sigma_j$  und  $\text{ALG}_p$  ein beliebiger randomisierter Online-Algorithmus. Ist dieser  $c$ -kompetitiv, so gibt es eine Konstante  $\alpha \geq 0$ , sodass für alle Anfragesequenzen  $\sigma_j^n$  gilt:

$$\mathbf{E}_p [\text{ALG}_i(\sigma_j^n)] \leq c \cdot \text{OPT}(\sigma_j^n) + \alpha$$

Insbesondere gilt dann auch für die zu Beginn gewählte Verteilung  $q$  und alle  $n$ :

$$\mathbf{E}_{q^n} \mathbf{E}_p [\text{ALG}_i(\sigma_j^n)] \leq c \cdot \mathbf{E}_{q^n} [\text{OPT}(\sigma_j^n)] + \alpha$$

Genau wie im endlichen Fall folgt mit Vertauschen der Erwartungswerte und Abschätzen der Kosten des randomisierten Online-Algorithmus durch die Kosten des besten deterministischen Algorithmus (gegen die gegebene Verteilung  $q^n$ ):

$$c \cdot \mathbf{E}_{q^n} [\text{OPT}(\sigma_j^n)] + \alpha \geq \mathbf{E}_{q^n} \mathbf{E}_p [\text{ALG}_i(\sigma_j^n)] = \mathbf{E}_p \mathbf{E}_{q^n} [\text{ALG}_i(\sigma_j^n)] \geq \inf_i \mathbf{E}_{q^n} [\text{ALG}_i(\sigma_j^n)]$$

Umstellen der Gleichung liefert:

$$c \geq \frac{\inf_i \mathbf{E}_{q^n} [\text{ALG}_i(\sigma_j^n)]}{\mathbf{E}_{q^n} [\text{OPT}(\sigma_j^n)]} - \frac{\alpha}{\mathbf{E}_{q^n} [\text{OPT}(\sigma_j^n)]}$$

und damit

$$\begin{aligned} c &\geq \limsup_{n \rightarrow \infty} \left( \frac{\inf_i \mathbf{E}_{q^n} [\text{ALG}_i(\sigma_j^n)]}{\mathbf{E}_{q^n} [\text{OPT}(\sigma_j^n)]} - \frac{\alpha}{\mathbf{E}_{q^n} [\text{OPT}(\sigma_j^n)]} \right) \\ &= \limsup_{n \rightarrow \infty} \frac{\inf_i \mathbf{E}_{q^n} [\text{ALG}_i(\sigma_j^n)]}{\mathbf{E}_{q^n} [\text{OPT}(\sigma_j^n)]} - \lim_{n \rightarrow \infty} \frac{\alpha}{\mathbf{E}_{q^n} [\text{OPT}(\sigma_j^n)]} \\ &= \limsup_{n \rightarrow \infty} \frac{\inf_i \mathbf{E}_{q^n} [\text{ALG}_i(\sigma_j^n)]}{\mathbf{E}_{q^n} [\text{OPT}(\sigma_j^n)]}, \end{aligned}$$

wobei in den letzten beiden Schritten die Voraussetzung eingeht, dass die Kosten des optimalen Offline-Algorithmus für große  $n$  gegen unendlich gehen.

Also ist  $\limsup_{n \rightarrow \infty} \frac{\inf_i \mathbf{E}_{q^n} [\text{ALG}_i(\sigma_j^n)]}{\mathbf{E}_{q^n} [\text{OPT}(\sigma_j^n)]}$  tatsächlich eine untere Schranke für die Kompetitivität aller Online-Algorithmen.  $\square$

**Beispiel 5.2.** Diesen Satz wollen wir nun zum Abschluss noch verwenden, um eine Aussage über das eingangs vorgestellte Spiel des „kleinste Zahl Findens“ (Beispiel 1.1) zu beweisen. Genauer gesagt wollen wir die Frage beantworten, ob es eine Strategie geben

kann, die uns einen echten Gewinn sicherstellt. Oder, anders formuliert, ob es einen Online-Algorithmus für dieses Problem geben kann, der besser als 1,5-kompetitiv ist.

Betrachte dazu Anfragesequenzen für  $n \in \mathbb{N}$  und  $m \leq n$  die folgenden Anfragesequenzen der Länge  $n$

$$\sigma_m^n := (n-1, n-2, \dots, m, \underbrace{n, n, \dots, n}_{m\text{-mal}})$$

und definiere die Verteilung  $q$  so, dass jede dieser Sequenzen gleich wahrscheinlich ist.

Für jedes  $n \in \mathbb{N}$  haben wir dann genau  $n$  verschiedene Anfragesequenzen dieser Länge (die alle gleichwahrscheinlich sind), nämlich gerade die Menge

$$\{\sigma_m^n \mid m = 1, \dots, n\}$$

Für den optimalen Offline-Algorithmus gilt dann

$$\mathbf{E}_{q^n} [\text{OPT}(\sigma_j^n)] = \frac{\sum_{m=1}^n m}{n} = \frac{n(n+1)}{2n} = \frac{n+1}{2},$$

womit insbesondere die Bedingung  $\lim_{n \rightarrow \infty} \mathbf{E}_{q^n} [\text{OPT}(\sigma_j^n)] = \infty$  bereits erfüllt ist.

Für jeden festen deterministischen Online-Algorithmus gilt, dass er auf allen Sequenzen der Länge  $n$  im Wesentlichen gleich abläuft. Denn bis zum ersten Auftreten eines  $n$  sehen die Sequenzen dieser Länge für den Online-Algorithmus immer gleich aus (und müssen daher zur gleichen Entscheidung führen). Nach dem ersten  $n$  spielen dann weitere Entscheidungen des Algorithmus keine Rolle mehr (da ab dann sowieso nur noch gleiche Zahlen auftreten).

Wir können also ohne Einschränkung annehmen, dass (auf Sequenzen der Länge  $n$ ) jeder deterministische Online-Algorithmus eindeutig durch eine Zahl  $r \leq n$  bestimmt ist, die angibt, die wievielte Zahl der Folge der Algorithmus auswählt (in Folgen dieser Länge). Damit gilt für einen Algorithmus ALG, der die  $r$ -te Zahl auswählt:

$$\mathbf{E}_{q^n} [\text{ALG}(\sigma_j^n)] = \frac{\sum_{m=1}^{n-r} (n-r) + \sum_{m=n-r+1}^n m}{n} = \frac{(n-r)^2 + rn}{n} = \frac{n^2 - nr + r^2}{n}$$

Dieser Wert wird minimal für  $r = \frac{1}{2}n$  und damit sind die erwarteten Kosten des besten deterministischen Algorithmus auf Folgen der Länge  $n$  mindestens

$$\frac{n^2 - \frac{1}{2}n^2 + \frac{1}{4}n^2}{n} = \frac{3}{4}n.$$

Zusammen erhalten wir damit die folgende Abschätzung:

$$\limsup_{n \rightarrow \infty} \frac{\inf_i \mathbf{E}_{q^n} [\text{ALG}_i(\sigma_j^n)]}{\mathbf{E}_{q^n} [\text{OPT}(\sigma_j^n)]} \geq \limsup_{n \rightarrow \infty} \frac{\frac{3}{4}n}{\frac{n+1}{2}} = \limsup_{n \rightarrow \infty} \frac{3 \cdot 2 \cdot n}{4 \cdot (n+1)} = \frac{3}{2}$$

Laut Yaos Prinzip ist das nun eine untere Schranke für die Kompetitivität des besten Online-Algorithmus. Die Hoffnung einen Online-Algorithmus zu finden, der besser als 1,5-kompetitiv ist, müssen wir also aufgeben, und sollten an dem Spiel in dieser Form daher besser nicht teilnehmen.

## Literatur

- [BEY98] Allan Borodin und Ran El-Yaniv. *Online Computation and Competitive Analysis*. Cambridge University Press, 1998. ISBN: 978-0521619462. URL: <https://books.google.de/books?id=v3faI8pER6IC&lpg=PR13&ots=A6z-DL0PZX&dq=online%20computing%20and%20competitive%20analysis&hl=de&pg=PR4#v=onepage&q&f=false>.
- [BLS92] Allan Borodin, Nathan Linial und Michael E Saks. „An optimal on-line algorithm for metrical task system“. In: *Journal of the ACM (JACM)* 39.4 (1992), S. 745–763. URL: [http://www.cs.huji.ac.il/~nati/PAPERS/bls\\_online.pdf](http://www.cs.huji.ac.il/~nati/PAPERS/bls_online.pdf).
- [KR05] Sven O. Krumke und Jörg Rambau. „Online Optimierung“. 8. Apr. 2005. URL: [http://www.fsmpi.uni-bayreuth.de/skripten/online\\_optimierung.pdf](http://www.fsmpi.uni-bayreuth.de/skripten/online_optimierung.pdf).
- [Yao77] Andrew Chi-Chin Yao. „Probabilistic computations: Toward a unified measure of complexity“. In: *Foundations of Computer Science, 1977., 18th Annual Symposium on*. 1977, S. 222–227. DOI: [10.1109/SFCS.1977.24](https://doi.org/10.1109/SFCS.1977.24).