Seminar über Algorithmen 15.10.2013

Einführung in die Spieltheorie

Nadja Scharf *Wolfgang Mulzer, Yannik Stein*

1 Spiele

Ein Spiel besteht im Allgemeinen aus mindestens zwei Spielern, die in irgendeiner Form miteinander agieren. Ein Spiel kann dabei viele Formen der Interaktion modellieren. Am Ende steht ein Ergebnis, dass den Spielern Gewinne oder Verluste beschert. Während des Spiels führen die Spieler Aktionen aus, wenn sie am Zug sind. Die gewählte Aktion nennt man Strategie.

1.1 Simultane Spiele

Bei einem simultanen Spiel machen alle Spieler ihren Zug gleichzeitig. Ein solches Spiel ist zum Beispiel Stein–Schere–Papier und kann wie folgt definiert werden:

Definition 1. *Sei $N = \{1, 2, ..., n\}$ die Menge der Spieler. Für jeden Spieler i sei S_i eine Menge möglicher Strategien oder Aktionen. Wählt jeder Spieler eine Strategie, so ergibt sich ein Strategievektor $s = (s_1, s_2, ..., s_n)$ mit $s \in S = \times_{j=1}^{n} S_i$. Dann lässt sich der Gewinn für jeden Spielers i für jedes mögliche s durch Funktionen $g_i : S \to \mathbb{R}$ angeben. Diese Darstellungsform heißt **Normalform**.*

1.2 Sequentielle Spiele

Sequentielle Spiele sind zum Beispiel Schach oder Skat. Ihre Darstellung ist schwieriger. Eine Möglichkeit besteht darin, sie als simultane Spiele aufzufassen, bei dem die einzelnen Strategien ein Verhalten für jeden Status des Spiels enthalten. Diese Darstellung ist unter Umständen aber sehr groß und die Reihenfolge der Züge geht verloren. Darum wird es in anderen Vorträgen gehen.

2 Konzepte zur Lösung

Eine Lösung soll in diesem Zusammenhang Aufschluss über mögliche Ausgänge des Spiels und die unterschiedlichen Spieleigenschaften liefern. Was hierbei als Lösung gilt, hängt vom jeweiligen Konzept ab.

2.1 Lösungen in nicht–kooperativen Spielen

Zunächst soll es hierbei um simultane Spiele gehen.

2.1.1 Dominante Strategien

Die Idee dominanter Strategien beschreibt die egoistische Verhaltensweise der Spieler. So wählt jeder Spieler immer diejenige Strategie, die ihm unabhängig von der Wahl der anderen Spieler den größten Gewinn verspricht. Formal kann dies wie folgt definiert werden:

Definition 2. *Eine Strategie s_i ist für Spieler i genau dann* ***dominant****, wenn für alle $s_i' \in S_i$ und alle $s_j \in S_j$, $j \in N$, $i \neq j$ gilt*

$$g_i(s_1, ..., s_{i-1}, s_i, s_{i+1}, ..., s_n) \geq \\ g_i(s_1, ..., s_{i-1}, s_i', s_{i+1}, ..., s_n).$$

Ist die Ungleichung für alle $s_i' \neq s_i$ echt, so heißt s_i echt dominant für i.

Der Nachteil dominanter Strategien besteht darin, dass sie nicht zu einer optimalen Lösungen führen müssen. Das heißt, wenn jeder Spieler seine dominante Strategie spielt, sofern sie existiert, muss dies nicht zum größtmöglichen Gewinn führen, weder insgesamt noch für den Einzelnen.

2.1.2 Pareto–Optimalität

Den eben genannten Nachteil versucht das Pareto–Optimum durch eine globale Sicht zu beheben.

Definition 3. *Seien $s, t \in S$. Dann* ***Pareto–dominiert*** *s t, falls für alle $i \in N$ gilt $g_i(s) \geq g_i(t)$. Ist die Ungleichung echt, so herrscht starke Pareto–Dominanz.*

Definition 4. *Sei $s \in S$. Dann ist s ein* ***Pareto–Optimum****, wenn für alle $t \in S$ gilt: Falls t s Pareto–dominiert, so ist $g_i(s) = g_i(t)$ für alle $i \in N$. s ist schwach Pareto–optimal, falls kein t s stark Pareto–dominiert.*

s ist also dann optimal, wenn sich kein Spieler ohne Nachteil für die anderen durch ändern seiner Strategie verbessern kann.

2.1.3 Gleichgewichte

Ein Nash–Gleichgewicht liegt vor, wenn kein Spieler seinen Gewinn durch wechseln seiner Strategie erhöhen kann. Es stellt damit eine Verallgemeinerung dominanter Strategien dar.

Definition 5. *Sei $s = (s_1, s_2, ..., s_n) \in S$. Dann ist s ein* ***Nash–Gleichgewicht in reinen Strategien****, wenn für alle $i \in N$ und alle $s_i' \in S_i$ gilt*

$$g_i(s_1, ..., s_{i-1}, s_i, s_{i+1}, ..., s_n) \geq \\ g_i(s_1, ..., s_{i-1}, s_i', s_{i+1}, ..., s_n)$$

Dominante Strategien bilden also Nash–Gleichgewichte. Folglich sind auch Nash–Gleichgewichte nicht zwingend optimal. Nicht für jedes Spiel gibt es ein wie oben definiertes Nash–Gleichgewicht, deshalb wird ein weiteres eingeführt:

Definition 6. *Sei S_i die Menge der Strategien für Spieler i und Π_i eine Wahrscheinlichkeitsverteilung darauf. Sei $\Pi = (\Pi_1, \Pi_2, ..., \Pi_n)$. Dann ist Π ein* ***Nash–Gleichgewicht in gemischten Strategien****, wenn für alle alle $i \in N$ und alle anderen Wahrscheinlichkeitsverteilungen Π_i' gilt*

$$g_i'(\Pi_1, ..., \Pi_{i-1}, \Pi_i, \Pi_{i+1}, ..., \Pi_n) \geq \\ g_i'(\Pi_1, ..., \Pi_{i-1}, \Pi_i', \Pi_{i+1}, ..., \Pi_n),$$

wobei g_i' den erwarteten Gewinn basierend auf g_i wiedergibt.

Es ist bewiesen, dass ein solches Nash–Gleichgewicht nur bei einer endlichen Anzahl von Spielern und Strategien immer existiert.
Eine ähnliche Lösungsform stellt ein Gleichgewicht in korrelierten Strategien dar. Dabei wählt ein Spieler seine Strategie nicht selber aus, sondern bekommt diese entsprechend einer Wahrscheinlichkeitsverteilung $p : S \to [0,1]$ vorgegeben. Ein Gleichgewicht besteht dann, wenn kein Spieler von seiner vorgegebenen Strategie abweichen kann ohne einen geringeren Gewinn erwarten zu müssen.

Definition 7. *Sei $S_{-i} = S_1 \times S_2 \times ... \times S_{i-1} \times S_{i+1} \times ... \times S_n$ und (s_i, s_{-i}) mit $s_i \in S_i$, $s_{-i} \in S_{-i}$ beschreibt den entsprechend zusammengesetzten Vektor aus S. Dann ist p **Gleichgewicht in korrelierten Strategien**, wenn für alle $i \in N$ und alle $s_i, s_i' \in S_i$ gilt*

$$\sum_{s_{-i} \in S_{-i}} p(s_i, s_{-i}) \cdot g_i(s_i, s_{-i}) \geq$$
$$\sum_{s_{-i} \in S_{-i}} p(s_i, s_{-i}) \cdot g_i(s_i', s_{-i}).$$

Für sequentielle Spiele gibt es eine weitere Definition von Nash–Gleichgewicht, die im 3. Vortrag vorgestellt wird.

2.1.4 Zusammenhang

Wie schon gesehen, bildet jede Lösung dominanter Strategien ein Nash-Gleichgewicht in reinen Strategien.
Sei $s = (s_1, s_2, ..., s_n)$ ein Nash-Gleichgewicht in reinen Strategien, dann ist $\Pi = (p_1, p_2, ..., p_n)$ mit $p_i(s_i) = 1$ ein Nash-Gleichgewicht in gemischten Strategien.
Sei $\Pi = (p_1, p_2, ..., p_n)$ ein Nash-Gleichgewicht in gemischten Strategien, dann ist p mit $p(s) = p_1(s) \cdot p_2(s) \cdot ... \cdot p_n(s)$ ein Gleichgewicht in korrelierten Strategien.
Pareto-Optima lassen sich nicht in dies Folge einreihen, da ihr Konzept kein egoistisches Verhalten der Spieler berücksichtigt.

2.2 Lösungen in kooperativen Spielen

2.2.1 Starkes Nash–Gleichgewicht

In der vorherigen Betrachtung von Gleichgewichten konnte immer nur ein Spieler seine Strategie ändern. Nun dürfen sich die Spieler absprechen und Gruppen dürfen geschlossen ihre Strategien wechseln.

Definition 8. *Sei $s \in S$ und $A \subseteq N$. s lässt sich zerlegen in die Strategien der Spieler in A, bezeichnet mit s_A, und die der restlichen Spieler, bezeichnet mit s_{-A}. Seien s_A' alternative Strategien für die Spieler in A. Dann ist $s \in S$ ein **starkes Nash–Gleichgewicht**, wenn es kein A gibt, für das ein s_A' existiert mit $g_i(s_A, s_{-A}) \leq g_i(s_A', s_{-A})$ für alle $i \in A$ und für ein i ist die Ungleichung echt.*

2.2.2 Kern und Shapley–Wert

Eine weitere Version kooperativer Spiele stellt eine andere als die bisher benutzte Gewinn– bzw. Verlustfunktion dar. Hierbei erhalten Koalitionen von Spielern gemeinsam einen Gewinn oder erleiden einen Verlust, der von der Summe der individuellen Gewinne bzw. Verluste abweichen kann.

Definition 9. *Sei $N = \{1, 2, ..., n\}$ die Menge der Spieler in einem* ***kooperativen Spiel mit übertragbaren Gewinnen*** *und $C \subseteq N$ bezeichnet eine Koalition. Dann beschreibt eine Funktion $g : \mathcal{P}(N) \to \mathbb{R}^+$ die Gewinne aller Möglichen Koalitionen. Ein Ergebnis eines solchen Spiels besteht aus der Angabe der Koalitionen $N = C_1 \dot{\cup} C_2 \dot{\cup} ... \dot{\cup} C_m$ und den erhaltenen Gewinnen x_i für alle $i \in N$ mit $\sum_{i \in C_j} x_i = g(C_j)$ für alle $j = 1, 2, ..., m$.*

Definition 10. *Sei $x = (x_1, x_2, ..., x_n)$ eine mögliche Gewinnaufteilung bei nur einer großen Koalition ($m = 1$). Dann liegt x im* ***Kern****, falls für alle Möglichen Koalitionen $C \subseteq N$ gilt $\sum_{i \in C} x_i \geq g(C)$.*

Eine Gewinnaufteilung einer großen Koalition liegt also dann im Kern, wenn es keine Gruppe von Spielern gibt, die durch bilden einer eigenen Koalition profitieren würde. Sie ist also in gewisser Weise stabil.
Ein anderes Konzept befasst sich mit dem fairen Aufteilen des Gewinns.

Definition 11. *Sei $p_1, p_2, ..., p_n$ eine Permutation der Spieler aus N. Dann kann jedem Spieler p_j der Gewinn $g(\{p_1, p_2, ..., p_j\}) - g(\{p_1, p_2, ..., p_{j-1}\})$ zugewiesen werden. Der* ***Shapley–Wert*** *des Spieler $i \in N$ ist sein erwarteter Gewinn in einer zufälligen Permutation aller Spieler.*

3 Komplexität

Im zweiten Vortrag wird es um die Komplexität vom Finden von Gleichgewichten gehen. Hier wird nur kurz gezeigt, wie in einem bestimmten Fall eines simultanen Spiels ein Nash–Gleichgewicht gefunden werden kann.

Definition 12. *Sei $N = \{1, 2\}$. Es liegt dann ein* ***Nullsummenspiel mit zwei Spielern*** *vor, wenn für alle $s \in S$ $g_1(s) = -g_2(s)$ gilt.*

Für Nullsummenspiele mit zwei Spielern reicht es aus, den Gewinn eines Spielers anzugeben, um das Spiel vollständig zu beschreiben. Sei A die Matrix der Gewinne für Spieler 1. Seien p und q die Wahrscheinlichkeitsverteilungen jeweils über die Strategien der beiden Spieler. Der erwartete Gewinn des ersten Spielers und damit der Verlust des zweiten Spielers lässt sich dann berechnen durch $p^T Aq$.
Angenommen Spieler 2 kennt p. $p^T A$ gibt dann entsprechend die erwarteten Verluste für Spieler 2 als Zeilenvektor für die jeweiligen Strategien an. Spieler 2 würde also immer die Strategie mit dem zugehörigen kleinsten Eintrag in $p^T A$ wählen. Das Ziel von Spieler 1 ist es folglich dieses Minimum zu maximieren. Dies geht mit folgendem linearen Programm:

$$\begin{aligned}
&\text{Zielfunktion: } \max v \\
&\text{Nebenbedingungen:} \\
&\qquad p \geq 0 \\
&\qquad \sum_i p_i = 1 \\
&\qquad (pA)_j \geq v \text{ für alle } j
\end{aligned}$$

Sei v^* der erwartete Wert für v im Nash–Gleichgewicht. Dann muss gelten $v \leq v^*$, da Spieler 1 den minimal zu erwartenden Wert maximiert. Über das duale lineare Programm erhält man ein

q, dass für Spieler 2 sein maximal zu erwartenden Verlust minimiert. Aufgrund der Dualität ist v in beiden Fällen gleich. Daraus ergibt sich, dass $v = v^*$ ist, da Spieler 1 seinen erwarteten Gewinn nicht erhöhen kann, da dies einen größeren erwarteten Verlust für Spieler 2 bedeuten würde. Dieser hat aber q so gewählt, dass sein erwarteter Verlust höchstens v ist. Dies gilt auch umgekehrt.
Da lineare Programme in Polynomialzeit lösbar sind, lässt sich ein Nash–Gleichgewicht für Nullsummenspiele mit zwei Spielern in Polynomialzeit berechnen.

4 Markt–Gleichgewichte

Ein wichtiger Aspekt der Spieltheorie ist die Untersuchung von (freien) Märkten. Dies wird in weiteren Vorträgen genauer untersucht. Hier wird ein kurzer Algorithmus zum Finden eines Gleichgewichts in einem einfachen Markt–Modell angegeben, das heißt es werden Preise berechnet, sodass jeder Verkäufer möglichst viel Gewinn macht und jeder Käufer möglichst viel für sein Geld bekommt.

Beispiel 1. *Sei $V = \{1, 2, ..., l\}$ eine Menge von Verkäufern, die alle die selbe teilbare Ware verkaufen. Verkäufer i hat $a_i \in \mathbb{N}$ Einheiten zum Preis p_i zu verkaufen. Sei $K = \{1, 2, ..., n\}$ eine Menge von Käufern, die alle möglichst viel von der Ware kaufen wollen. Käufer j hat $m_j \in \mathbb{N}$ viel Geld zur Verfügung. Dabei kennt nicht jeder Käufer jeden Verkäufer. Sei $S \subseteq V$. Dann beschreibt $\Gamma(S)$ die Menge aller Käufer, die einen Verkäufer aus S kennen. $a(S)$ beschreibt die Summe der Waren der Verkäufer in S. Für $T \subseteq K$ beschreibt $m(T)$ die Summe des Geldes aller Käufer in T.*

Definition 13. *Ein einheitlicher Preis $x \in \mathbb{Q}^+$ für alle Waren ($\forall i \in V : p_i = x$) ist **zulässig**, falls für alle $S \subseteq V$ gilt, dass $x \cdot a(S) \leq m(\Gamma(S))$.*

Es kann bewiesen werden, dass ein x genau dann zulässig ist, wenn alle Waren verkauft werden können. Daraus ergibt sich der Algorithmus zum Festlegen der Preise, sodass Angebot und Nachfrage im Gleichgewicht sind. Das heißt, dass jeder Verkäufer alle seine Waren verkaufen kann und jeder Käufer sein gesamtes Geld ausgibt.
Der Algorithmus startet mit $x = 0$ und erhöht x solange, bis ein nichtleeres $S^* \subseteq V$ existiert, sodass alle Waren der Verkäufer aus S^* verkauft werden können und die Käufer in $\Gamma(S^*)$ ihr gesamtes Geld ausgegeben haben. Danach können diese aus dem Modell entfernt werden und der Algorithmus wiederholt das Procedere bis alle Waren verkauft sind.
S^* und das zugehörige x^* lassen sich über Flussnetzwerke berechnen. Dafür stellt jeder Käufer und jeder Verkäufer einen Knoten dar. Wenn Käufer j Verkäufer i kennt, wird eine Kante (i, j) mit Kapazität ∞ eingefügt. Zusätzlich wird eine Quelle s eingefügt mit Kanten zu allen $i \in V$ mit Kapazität $x \cdot a_i$. Umgekehrt wird eine Senke t eingefügt mit Kanten von allen Käufern $j \in K$ mit Kapazität m_j. x kann im Algorithmus so lange erhöht werden, wie $(\{s\}, V \dot\cup K \dot\cup \{t\})$ im Netzwerk ein minimaler Schnitt ist, da danach vom minimalen Schnitt mindestens eine Kante von einem Käufer zu t betroffen ist. Da keine der Kanten von Verkäufern zu Käufern aufgrund ihrer Kapazität zum Schnitt gehören können, bedeutet dies, dass es eine Menge von Verkäufern gibt, die nicht mehr alle ihre Waren verkaufen können. In Abbildung 1 soll dies verdeutlicht werden. Die Verkäufer im oberen Teil können nur an die Käufer im oberen Teil verkaufen. Überschreitet der Preis deren Budget, so kann nicht alles verkauft werden.
Mit dem Netzwerk und einem Algorithmus zum Berechnen minimaler Schnitte lässt sich x^* mit binärer Suche berechnen. Dies geht in polynomiell vielen Schritten, da $x^* = \frac{m(\Gamma(S^*))}{a(S^*)}$ und damit polynomielle Göße besitzt.

a) b)

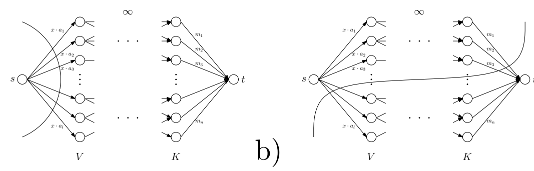

Abbildung 1: a) Schnitt, der nur von s ausgehende Kanten betrifft
b) Schnitt, der Kanten zu t betrifft

Zum Ermitteln von S^* wird zunächst ein maximaler Fluss im Netzwerk berechnet. Daraus ergibt sich ein Graph mit Restkapazitäten (vollständig ausgeschöpfte Kanten werden entfernt). Alle Verkäufer, die in diesem Graph keinen Pfad zu t mehr besitzen, gehören zu S^*. Daraus lässt sich $\Gamma(S^*)$ bestimmen und beide Knotenmengen können aus dem Netzwerk entfernt werden.
Nach höchstens $|V|$ Iterationen ist also bekannt, wieviel Einheiten zu welchem Preis an wen verkauft werden. Eine Iteration läuft in Polynomialzeit, folglich benötigt der gesamte Algorithmus Polynomialzeit.

Quellen

- Noam Nisan, Tim Roughgarden, Eva Tardos, and Vijay V. Vazirani. *Algorithmic Game Theory*. Cambridge University Press, New York, NY, USA, 2007.

- J. Rothe, D. Baumeister, C. Lindner, and I. Rothe. *Einführung in Computational Social Choice:*. Spektrum Akademischer Verlag GmbH, 2011.