18.05.2004

<u>Beispiel:</u>

Die Spieler wählen Zahlen aus $\{1,\ldots,K\}$. Ein fixer Preis wird zu gleichen Teilen zwischen den Spielern aufgeteilt, die dem Wert $\frac{2}{3|N|}\sum_{l=1}^{|N|} a_l$ am nächsten kommen (Anm: $\frac{2}{3}$ des Durchschnitts) .

Es existiert ein reines Nash-Gleichgewicht bei $a_l = 1$ für alle $l \in \{1,\ldots,|N|\}$.

Nash-Gleichgewicht in gemischten Strategien?

1) Annahme: es gibt ein weiteres Nash-Gleichgewicht α (in gemischten Strategien)

2) Dann existiert ein maximales $k^* > 1$, das von mindestens einem Spieler mit Wahrscheinlichkeit größer 0 gespielt wird.

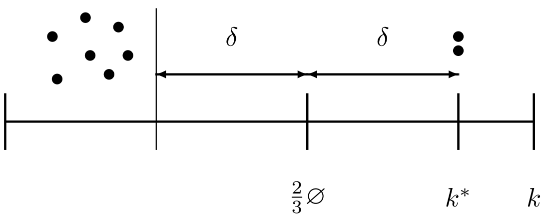
3) Annahme: Spieler i spielt k^* ($\hat{=}$ hat k^* in Unterstützermenge)

4) $U_i(k^*, \alpha_{-i}) > 0$, da k^* so viel wie alle anderen Strategien in der Unterstützungsmenge erbringen muss.

5) Sei a eine Realisierung von α mit $a_i = k^*$ und $u_i(\alpha) > 0$. Dann muss mindestens ein anderer Spieler j auch k^* spielen, denn nicht alle anderen Spieler können $a_l \leq \frac{2}{3|N|}\sum_h a_h$ spielen.

   Auch der Fall $\frac{2}{3|N|}\sum_h a_h \leq a_j < k^*$ kann ausgeschlossen werden, da i nicht gewonnen hätte!

   Kein Spieler kann höher als k^* spielen, da k^* maximal ist.



6) In dieser Situationen könnte i sich verbessern, wenn er $k^* - 1$ spielt (für $k^* \geq 2$).

   (Anm: er kommt dem Durchschnitt dadurch um $\frac{1}{|N|}$ näher. Darum ...)

7) D.h. es ist immer besser $k^* - 1$ zu spielen, d.h. k^* kann nicht in der Unterstützungsmenge liegen, d.h. α kann kein Nash-Gleichgewicht sein.

$\rightsquigarrow$ Es gibt nur das Nash-Gleichgewicht in reinen Strategien: $(1,\ldots,1)$.

3.1. Algorithmen für die Nash-Gleichgewichts-Bestimmung in Nullsummenspielen

<u>Exkurs:</u> Lineare Programmierung/Lineare Optimierung

[„Lineare Programmierung“ geprägt in den 30er Jahren - bevor Computer programmiert wurden, damals bedeutete Programmierung Planung (vgl. dynamische Programmierung)]

(Anm: vgl. in Kapitel 6)
Worum geht es: Lineare Ungleichungen über n reellwertige Variablen und eine lineare Zielfunktion, die man maximieren möchte.

Beispiel („Sortimentproblem“):

Es werden zwei Sortimente eines Artikels produziert, die alle verkauft werden können.

Sortiment 1:	25 Min Schneiden, 60 Minuten Zusammenbauen, 68 Minuten Postprocessing.
	30 Euro pro Artikel
Sortiment 2:	75 Minuten Schneiden, 60 Minuten Zusammenbau und 34 Minuten Postprocessing
	40 Euro Gewinn.
Pro Tag:	450 Minuten Zuschneiden, 480 Minuten Zusammenbau, 476 Minuten Postprocessing

Wie maximiert man den Gewinn?

x: Anzahl Artikel in Sortiment 1

y: Anzahl Artikel in Sortiment 2

1. $x \geq 0$, $y \geq 0$ und
2. $25x + 75y \leq 450 \Rightarrow y \leq \underbrace{\frac{450}{75}}_{=6} - \underbrace{\frac{25x}{75}}_{1/3} \Rightarrow y \leq 6 - \frac{1}{3}x$
3. $60x + 60y \leq 480 \Rightarrow y \leq 8 - x$
4. $68x + 34y \leq 476 \Rightarrow y \leq 14 - 2x$
5. Maximiere $z = 30x + 40y$

Ungleichungen (1)-(4) beschreiben zulässige Lösungen.

(5) ist die *Zielfunktion* (objective function).

(1)-(4) beschreiben *konvexe* Menge in $\mathbb{R}^2$.

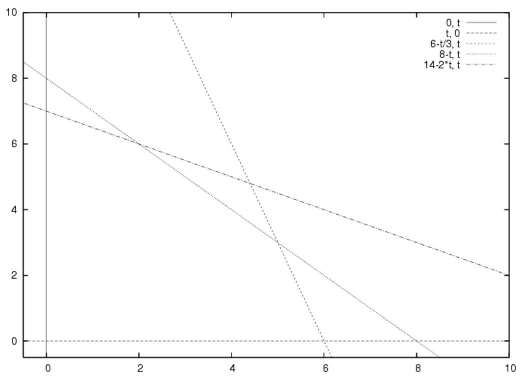


Abbildung 3.1: $\rightsquigarrow$ zulässige Lösungen (feasible solution)

$$y = \frac{z}{40} - \frac{3x}{4}$$

Die Zielfunktion gibt für jeden Punkt eine Qualität:

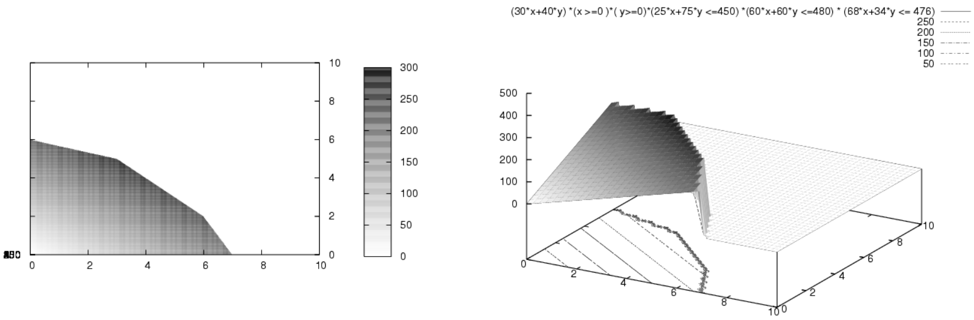


Abbildung 3.2: Isolinien für verschiedene z-Werte.

Allgemein (Standardform):

n Variablen x_i (reellwertig)
m Koeffizienten b_j
m Konstanten c_j
$m \times n$ Koeffizienten a_{ij} für Gleichungen
m Gleichungen
$c_j = \sum_{i=1}^{n} a_{ij} x_i$
Zielfunktion:
$\sum_{i=1}^{n} b_i x_i$(für $x_i \geq 0$) zu minimieren.

(Anm: man kann statt Ungleichungen Gleichungen wählen, da:
$x + y \leq c \Leftrightarrow x + y + \underbrace{z}_{\textbf{Slack-Variable}} = c$ für ein $z \geq 0$)

Simplex-Methode: Standardlösungsmethode, um LPs zu lösen. Diese Methode ist im schlechtesten Fall exponentiell. In allen praktischen Fällen ist es aber „gutmütig“!

Anwendung auf Nullsummenspiele:

Satz von Nash:

Nash-Gleichgewichte existieren in gemischten Strategien.

Maximin-Satz:

Falls Nash-Gleichgewicht existiert, dann ist es ein Paar von Maximinimierern.
$A_1 = \{a_{11}, \ldots, a_{1n}\}$, $A_2 = \{a_{21}, \ldots, a_{2m}\}$

Spieler 1 sucht gemischte Strategie α_1.

$$\sum_{j=1}^{n} \alpha_1(a_{11}) \cdot u_1(a_{11}, a_{2i}) \geq u \qquad \text{für alle } i \in \{1, \ldots, m\}$$

$$\sum_{j=1}^{n} \alpha_1(a_{1j}) = 1$$

$$\alpha_1(a_{1j}) \geq 0 \qquad \text{für alle } j \in \{1, \ldots, n\}$$

Maximiere u.
Die Lösung dieses LPs ist ein Maximinimierer für Spieler 1.
Analog für Spieler 2.