Seminar über Algorithmen	18.02.2014

Komplexität von proportionalem Cake Cutting

Simon Tippenhauer — *Wolfgang Mulzer, Yannik Stein*

1 Einführung

1.1 Problemstellung

Gegeben ist eine Anzahl von Spielern und eine Menge von beliebig aufteilbaren Gegenständen, die jeder Spieler individuell bewertet. Die Aufgabe ist die Bestimmung einer fairen Partition, die jedem Spieler einen Anteil der Menge zuordnet. Eine Partition kann unter verschieden Aspekten als fair angesehen werden, zu denen z.B. *Proportionalität*, *Neidfreiheit* und *Gleichheit* gehören. Diese Aufgabenstellung wird als *Cake Cutting* Problem bezeichnet. In Anlehnung an diese Bezeichnung werden im Folgenden die Gegenständen zusammengefasst als ein Kuchen benannt, der unter den Spielern fair in Stücke aufgeteilt werden soll.

1.2 Motivation

Das Problem des Cake Cuttings tritt in unterschiedlichen Bereichen auf, wie z.B. bei der Auflösung von Unternehmen und der damit verbundenen Aufteilung der Besitzgüter unter den einzelnen Unternehmern. Ein weiteres Beispiel ist die Verteilung des Hab und Guts unter den Erben eines kürzlich verschiedenen Angehörigen. Das namensgebende Problem ist ein Kuchen mit unterschiedlichem Belag, der unter mehren Personen aufgeteilt werden soll und jeder die einzelnen Beläge mehr oder wenig gerne isst.

1.3 Einführende Terminologie

1.3.1 Definition des Problems

- Der Kuchen wird durch das Intervall $[0, 1]$ modelliert.
- Jeder der n Spieler $p \in P = \{1, \ldots, n\}$ hat eine eigene Bewertungsfunktion $V_p(x_1, x_2)$ für den Kuchen, die dem Teilintervall $[x_1, x_2] \subseteq [0, 1]$ einen Wert zuordnet. An die Bewertungsfunktion werden einige Voraussetzungen gestellt:
  - *Normalisierung*: $V_p(0, 1) = 1$.
  - *Teilbarkeit*: Für alle Teilintervalle $[x_1, x_2]$ und $0 \leq \lambda \leq 1$ existiert ein $y \in [x_1, x_2]$, sodass $V_p(x_1, y) = \lambda V_p(x_1, x_2)$.
  - *Nichtnegativität*: Für alle Teilintervalle $[x_1, x_2]$ gilt $V_p(x_1, x_2) \geq 0$.
  - *Additivität*: Für zwei disjunkte Teilintervalle $[x_1, y_1], [x_2, y_2]$ gilt $V_p(x_1, y_1) + V_p(x_2, y_2) = V_p([x_1, y_1] \cup [x_2, y_2])$.
- Ein *Stück* ist eine Menge von Teilintervallen. Der Wert eines Stücks ist die Summe der Werte seiner disjunkten Teilintervalle.

- Ein Protokoll soll jedem Spieler p ein Stück C_p zuordnen, sodass die Stücke eine Partition des gesamten Kuchens bilden und diese Zuordnung fair ist. Als Kriterien der Fairness betrachten wir folgende Eigenschaften:
  - *Proportionalität*: Für alle $p \in P$, $V_p(C_p) \geq \frac{1}{n}$.
  - *Neidfreiheit*: Für alle $p, q \in P$, $V_p(C_p) \geq V_p(C_q)$.
  - *Gleichheit*: Für alle $p, q \in P$, $V_p(C_p) = V_q(C_q)$.

Ein Protokoll für proportionales Cake Cutting ist *c-fair* zu jedem Spieler p, wenn der Wert von C_p bzgl. V_p mindestens den Wert $\frac{1}{cn}$ hat.

1.3.2 Das Robertson-Webb Modell

Für die Analyse von Cake Cutting Protokollen wird standardmäßig das Robertson-Webb Modell zugrunde gelegt. In diesem Modell sind zwei Anfragen definiert, die an die beteiligten Spieler individuell gerichtet werden können.

- $Eval_p(x_1, x_2)$ ist eine Bewertungsanfrage für einen Spieler p, die den Wert des Intervalls $[x_1, x_2]$ bzgl. des Spielers p liefert. Es gilt $Eval_p(x_1, x_2) = V_p(x_1, x_2)$.

- $Cut_p(x_1, \alpha)$ ist eine Schnittanfrage an einen Spieler p, die ein $x_2 \geq x_1$ zurückgibt, sodass das Intervall $[x_1, x_2]$ den Wert α für den Spieler p bzgl. seiner Bewertungsfunktion V_p hat.

Die Komplexität eines Protokolls ist die Anzahl von Anfragen im Robertson-Webb Modell, die zur fairen Aufteilung des Kuchens benötigt werden. Der Worst Case ist die maximale Anzahl benötigter Anfragen über alle möglichen Bewertungsfunktionen.

1.4 Zwei Protokolle für proportionales Cake Cutting

Die beiden Protokolle von Dubins & Spanier und Even & Paz verdeutlichen den Unterschied zwischen einem kontinuierlichen und einem diskreten Protokoll für das Cake Cutting Problem. Sei n die Anzahl der Spieler.

1.4.1 Moving Knife Protokoll - *Dubins-Spanier*

Das Protokoll von Dubins & Spanier besteht aus $n-1$ Runden, in denen ein Messer kontinuierlich über den Kuchen bewegt wird. In jeder Runde $i = 1, 2, \ldots, n-1$ wird ein Schnitt cut_i ausgeführt, sobald für einen Spieler das Stück $[cut_{i-1}, cut_i]$ mit $cut_0 = 0$, den Wert $\frac{1}{n}$ erreicht. Der letzte verbleibende Spieler bekommt das Stück $[cut_{n-1}, 1]$. Es ist leicht zu zeigen, dass das Protokoll proportional ist, aber nicht in allen Fällen neidfrei.

1.4.2 Teile-und-Herrsche Protokoll - *Even-Paz*

Das Protokoll von Evan & Paz erfüllt die selber Eigenschaften wie das von Dubins & Spanier. Für eine Teilmenge $N = \{1, 2, \ldots, k\}$ der Spieler und ein Stück $[x, z]$ wird jedem Spieler $p \in N$ eine Anfrage $Cut_p(x, \frac{1}{2}) = y_p$ gestellt. Seien $y_{p_1} \leq \cdots \leq y_{p_k}$ die Antworten der Spieler. Anschließend wird rekursiv mit der Teilmenge $\{1, \ldots, \frac{k}{2}\}$ und dem Stück $[x, y_{\frac{k}{2}}]$, respektive $\{\frac{k}{2} + 1, \ldots, k\}$ und $[y_{\frac{k}{2}}, z]$, fortgefahren. Sobald nur noch ein Spieler übrig ist, wird ihm das Stück zugeordnet. Das Protokoll wird anfangs mit den Spielern $N = \{1, \ldots, n\}$ und dem gesamten Kuchen $[0, 1]$ aufgerufen.

1.5 Diskussion von Vorarbeiten

Ein deterministisches Protokoll mit $O(n^2)$ Schnitten im Worst Case haben in den 1940er Jahren die polnischen Mathematiker Banach und Knaster entwickelt, beschrieben 1948 von Steinhaus [5]. Im Jahre 1984 Jahre stellten Even und Paz [2] einen deterministischen Teile-und-Herrsche Algorithmus vor, der die obere Schranke auf $O(n \log n)$ verbesserte.

In den darauffolgenden Jahren wurde versucht zu zeigen, dass die faire Kuchenaufteilung von unten durch $\Omega(n \log n)$ beschränkt ist. Dabei wurde der Ansatz verfolgt, Sortieren auf Cake Cutting zu reduzieren. Erste Ergebnisse erzielten Magdon-Ismail, Busch und Krishnamoothy [3], die zeigen konnten, dass jedes Protokoll $\Omega(n \log n)$ Vergleiche benötigt. Jedoch ist dies keine untere Schranke für die Anzahl von Anfragen im Robertson-Webb Modell. Unter der Einschränkung, dass jedem Spieler nur ein zusammenhängendes Stück zugeteilt wird, haben Sgall und Woeginger [4] die untere Schranke $\Omega(n \log n)$ beweisen können.

1.6 Ergebnisse der Autoren

Edmonds und Pruhs [1] konnten die untere Schranke $\Omega(n(\log n - \log c))$ für die Komplexität jedes deterministischen c–fairen, proportionalen Cake Cutting Protokolls im Robertson-Webb Modell zeigen. Deren Ansatz basiert nicht auf einer Reduktion vom Sortieren sondern auf der Feststellung, dass Spielern Stücke zugeteilt werden müssen, die sowohl reich an Wert als auch dünn sind.

2 Skizzierung des Beweises für die untere Schranke

2.1 Beweisführung

Satz 1. *Die Komplexität jedes deterministischen Cake Cutting Protokolls ist $\Omega(n(\log n - \log c))$ im Robertson-Webb Modell.*

Die untere Schranke für Cake Cutting folgt aus dem Zusammenhang mit der Komplexität des *Dünn-Reich Spiels*. Hierbei soll für einen einzigen Spieler ein dünnes, reiches Stück des Kuchens bestimmt werden. Ein Stück wird als *dünn* bezeichnet, wenn es höchstens $\frac{2}{n}$ breit ist und als *reich*, wenn es mindestens den Wert $\frac{1}{cn}$ hat.

Lemma 2 (Zusammenhang)**.** *Wenn die deterministische Komplexität des Dünn-Reich Spiels $T(n)$ ist, dann ist die deterministische Komplexität von Cake Cutting $\Omega(n \cdot T(n))$.*

Aus dem Lemma folgt, dass Cake Cutting von unten beschränkt ist mit $\Omega(n(\log n - \log c))$, wenn für das Dünn-Reich Spiel $\Omega((\log n - \log c))$ gilt. Um dies zu zeigen, werden spezielle Bewertungsfunktionen betrachtet, die aus sogenannten Wertbäumen abgeleitet werden.

Lemma 3. *Wenn ein deterministisches Protokoll A für das Dünn-Reich Spiel mit Komplexität $T(n)$ existiert, dann existiert auch ein deterministisches Protokoll B mit Komplexität $O(T(n))$ für aus Wertbäumen abgeleitete Wertverteilungen, dass ein Blatt des zugrunde liegenden Wertbaums zurückgibt.*

Somit genügt es Protokolle für das Dünn-Reich Spiel zu betrachten, bei denen Wertbäume für die Bewertungsfunktion verwendet werden und die ein Blatt des Wertbaums berechnen.

Lemma 4. *Die deterministische Komplexität vom Dünn-Reich Spiel ist $\Omega(\log n - \log c)$.*

Aus der Korrektheit der Lemmata 2–4 folgt direkt die untere Schranke $\Omega(n(\log n - \log c))$ für das Cake Cutting Problem.

2.2 Beweise zu Lemma 2 (Zusammenhang)

Beweis. Für jedes beliebige Protokoll für das Cake Cutting Problem können die einzelnen Spieler als Black Box betrachtet werden, die vom Protokoll gestellte Eval- und Cut-Anfragen beantworten. Die nächste Aktion basiert nur auf den Informationen aus vergangenen Anfragen. Zum Schluss teilt das Protokoll jedem Spieler ein Stück des Kuchens mit.

Aus Sicht eines Spielers p liegt das Dünn-Reich Spiel vor. Nun wird angenommen, dass dieser Spieler weniger als $T(n)$ Anfragen erhält. Dann existiert eine Wertverteilung des Kuchens, sodass das Stück C_p nicht dünn oder reich ist. Wenn ein Cake Cutting Protokoll nun weniger als $\frac{1}{2}nT(n)$ Anfragen stellt, haben mehr als die Hälfte der Spieler ein solches Stück. Das Protokoll schlägt fehl, wenn einer der Spieler kein reiches Stück bekommt. Daraus folgt, dass mehr als die Hälfte kein dünnes Stück bekommen müssten. Dadurch können die Stücke nicht mehr disjunkt sein und das Protokoll ist nicht korrekt. □

3 Details und Beweis zu Lemma 4

3.1 Wertbäume

Sei o.B.d.A $\frac{n}{2}$ eine Dreierpotenz. Ein *Wertbaum* ist ein gewurzelter, vollständiger, ternärer Baum, dessen Blätter disjunkte Intervalle der Breite $\frac{2}{n}$ des Kuchens repräsentieren. Die Kanten zu zwei Kindern eines innere Knotens u sind mit $\frac{1}{4}$ (*leichte Kante*) und die dritte Kante ist mit $\frac{1}{2}$ (*schwere Kante*) markiert. Der Wert $V(u)$ eines Knotens ist das Produkt der Kantenmarkierungen auf dem Weg von der Wurzel zu dem Knoten. Der Wert eines Blatts ist auf dem zugehörigen Intervall gleich verteilt.

Damit ein Blatt v einen Wert von mindestens $\frac{1}{cn}$ hat und ein gültiges Stück für das Dünn-Reich Spiels ist, müssen auf dem Weg von der Wurzel genügend schwere Kanten liegen. Zu einem Knoten u bezeichnet

- $l(u)$ die Anzahl Kanten auf dem Weg von der Wurzel zu u und
- $q(u)$ die Anzahl der schweren Kanten auf diesem Weg.

Daraus folgt, dass $V(u) = \left(\frac{1}{2}\right)^{q(u)} \left(\frac{1}{2}\right)^{l(u)-q(u)}$ und weiter, dass die Anzahl schwerer Kanten $q(v) \geq 2l(u) - \log cn = 2\log_3 \frac{n}{2} - \log_2 cn$ sein muss.

3.2 Adversary Methode für untere Schranken

Unter Verwendung der Adversary Methode soll für das Dünn-Reich Spiel bzgl. Wertbäumen die Mindestanzahl von Anfragen bestimmt werden, die für ein erfolgreiches Protokoll notwendig sind. Dazu wird das Modell in der Hinsicht verändert, dass der Widersacher bei Eval- und Cut-Anfragen mindestens so viele Knoten des zugrundeliegenden Wertbaums *aufdeckt*, dass das Protokoll die Antworten berechnen kann.

Definition 5. *Sei $P = u_0, u_1, \dots, u_k$ ein Weg von der Wurzel u_0 zu einem Knoten u_k eines Wertbaums. Dann ist u_k* aufgedeckt*, wenn alle Kantenmarkierungen zu Kindern von u_i für $i = 0, 1, \dots, k-1$ bekannt sind.*

Der Widersacher versucht nun während des Protokolls die Kanten so zu markieren, dass möglichst wenig schwere Kanten auf allen Wurzel-Blatt Wegen aufgedeckt werden. Die Knoten müssen vom Widersacher konsistent aufgedeckt werden, d.h. der Wert einer einmal bekannten Kante, darf sich während des Protokolls nicht ändern. Sobald genügend schwere Kanten auf einem Weg von der Wurzel zu einem Blatt u bekannt sind, kann das Protokoll als gültige Antwort u zurückgeben.

Die zusätzlichen Informationen des Widersachers vereinfachen das Cake Cutting Problem im Vergleich zum ursprünglichen Modell. Kann nun die untere Schranke von $\Omega(n(\log n - \log c))$ im neuen Modell gezeigt werden, folgt daraus ebenfalls Lemma 4.

3.3 Beweis

Bezüglich des Widersachers für die untere Schranke ergeben sich folgende Eigenschaften über Wertbäume, die für den Beweis der unteren Schranke des Dünn-Reich Spiels notwendig sind.

Lemma 6.

- *Für jeden aufgedeckten Knoten u kann der Wert $V(u)$ des Intervalls des Kuchens unter ihm berechnet werden.*
- *Seien u ein aufgedeckter Knoten, x der Punkt ganz links und y der Punkt ganz rechts in u. Dann kann $V(0, x)$ und $V(0, y)$ berechnet werden.*
- *Sei x ein Punkt in einem aufgedeckten Blatt u und sei $y \geq x$ ein Punkt in einem aufgedeckten Blatt v. Dann kann $V(0, x)$ und $V(x, y)$ berechnet werden.*
- *Seien u ein aufgedecktes Blatt, x_1 ein Punkt in u und α ein Kuchenwert. Damit kann der kleinste gemeinsame Nachfolger von u und dem Knoten v, der den Punkt x_2 mit $V(x_1, x_2) = \alpha$ enthält, berechnet werden.*

Beweis zu Lemma 4. Sei A ein korrektes, deterministisches Protokoll für das Dünn-Reich Spiel für Wertverteilungen, abgeleitet aus Wertbäumen, dass ein Blatt liefert. Es wird eine Induktion über die Anzahl k von Anfragen an den Widersacher geführt. Während des Protokolls werden vier Invarianten eingehalten.

1. Die einzigen Informationen über den zugrundeliegenden Wertbaum sind die aufgedeckten Knoten des Widersachers.
2. Entweder sind alle Kanten oder keine Kante ausgehend von einem Knoten bekannt.
3. Die aufgedeckten Knoten bilden eine Zusammenhangskomponente mit der Wurzel.
4. Nach k Anfragen sind höchstens $2k$ schwere Kanten auf jedem Weg von der Wurzel zu einem Blatt bekannt.

Für den Induktionsanfang sind die Invarianten erfüllt. Nun stellt das Protokoll seine k-te Anfrage und nach Induktionsvoraussetzung sind die Invarianten nach $k-1$ Anfragen erfüllt. Sei L die Höhe des Wertbaums.

Fall 1: $Eval(x_1, x_2)$. Sei $u_0, \ldots, u_L$ ein Weg von der Wurzel u_0 zum Blatt u_L, dass x_1 enthält und sei u_i der niedrigste aufgedeckte Knoten auf diesem Weg. Dann werden alle Kanten vom Widersacher auf dem Pfad als leicht und die anderen beliebig aufgedeckt. Analog wird mit x_2 verfahren.

Fall 2: $Cut(x_1, \alpha) = x_2$. Für x_1 wird wie bei einer Eval-Anfrage verfahren. Sei u_i der kleinste gemeinsame Knoten, der die Blätter mit x_1 und x_2 enthält. Sei v der letzte aufgedeckte Knoten auf dem Weg von u_i nach x_2 und sei β der Wert von v, dem α zugeschrieben wird. Sowohl v als auch β sind nach Lemma 6 wohldefiniert.

(i) $\frac{\beta}{V(v)} \leq \frac{1}{2}$: Kanten von v werden mit $[\frac{1}{4}, \frac{1}{4}, \frac{1}{2}]$ aufgedeckt.

(ii) $\frac{\beta}{V(v)} > \frac{1}{2}$: Kanten von v werden mit $[\frac{1}{2}, \frac{1}{4}, \frac{1}{4}]$ aufgedeckt.

In beiden Fällen wird auf dem Weg eine leichte Kante aufgedeckt. Dieser Vorgang wird bis ins Blatt fortgesetzt und A hat genügend Informationen um x_2 zu bestimmen.

Jede Anfrage erfüllt weiterhin alle vier Invarianten.

Angenommen das Protokoll terminiert nach weniger als $(2L - \log_2 cn)/2$ Anfragen und gibt ein Blatt u zurück, dass reich sein soll. Dann können nach der 4. Invariante höchstens $(2L - \log_2 cn)$ Kanten auf dem Weg von der Wurzel zu u schwer sein. Werden nun die verbleibenden unaufgedeckten Kanten auf leicht gesetzt, ist $V(u)$ kleiner als $\frac{1}{cn}$. Dies steht im Widerspruch zu der Voraussetzung, das A korrekt ist. Daraus folgt, dass mindestens $(2L - \log_2 cn) = \Omega(\log n - \log c)$ Anfragen nötig sind, um das Dünn-Reich Spiel zu lösen. $\square$

Literatur

[1] J. EDMONDS AND K. PRUHS (2006). *Cake Cutting Really is Not a Piece of Cake*. Proceedings of the Seventeenth Annual ACM-SIAM Symposium on Discrete Algorithm.

[2] S. EVEN AND A. PAZ (1984). *A note on cake cutting*. Discrete Applied Mathematics 7, 285–296.

[3] M. MAGDON-ISMAIL, C. BUSCH, AND M.S. KRISHNAMOORTHY (2003). *Cake cutting is not a piece of cake*. Proceedings of the 20th Annual Symposium on Theoretical Aspects of Computer Science (STACS'2003), LNCS 2607, Springer Verlag, 596–607.

[4] J. SGALL AND G. J. WOEGINGER (2003). *A lower bound for cake cutting*. LNCS 2461, Springer Verlag, 896–901. Proc. of the 11th Ann. European Symp. on Algorithms (ESA), Lecture Notes in Comput. Sci. 2832, Springer, 459–469.

[5] H. STEINHAUS (1948). *The problem of fair division*. Econometrica 16, 101–104.