Kommunikationskomplexität

Seminar über Algorithmen, Prof. Dr. Alt, Sommersemester 2010, Freie Universität Berlin

Matthias Rost

13. Juli 2010

Inhaltsverzeichnis

Ursprünge der Kommunikationskomplexität

- Yao [1] hat Ende der 70er Jahre den ersten Artikel über Kommunikationskomplexität an sich veröffentlicht
- Zuvor wurde die Kommunikationskomplexität vor allem im Bereich des VLSI-Chip-Designs implizit verwandt.

VLSI

- VLSI $\triangleq$ Very-Large-Scale Integration
- Ziel
  - Möglichst viel Logik auf kleinem Chip (Kosten)
  - Möglichst schnelle Berechnung (wenig Taktzyklen)

AT^2-Schranke für DFT

- Ende der 70er Jahre: erste $AT^2 - Schranken$
  - $A \triangleq$ Fläche des Chips
  - $T \triangleq$ Anzahl der Taktzyklen
- Erster Beweis mit impliziter Kommunikationskomplexität über diskrete Fourier-Transformation [2]
  - $AT^2 \geq \frac{n^2}{16}$, wobei n die Länge der (binären) Eingabe ist
  - sofern sich die Länge der Eingabe verdoppelt, ...
    - muss der Platz verdoppelt werden, oder
    - die Berechnung dauert doppelt so lange.

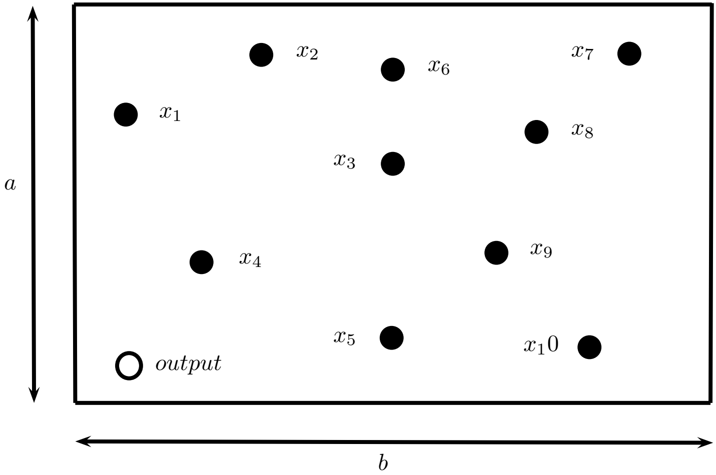
Annahmen

1. Abmessungen $a \times b$, mit $a \leq b$.
   1. Abmessungen werden in λ gemessen
   2. $\lambda \triangleq$ minimaler Abstand zweier Leitungen
   3. $A \triangleq$Fläche des Chips
2. Dem Chip werden n Eingabebits übergeben.
3. Pro Taktzyklus kann über eine Leitung des Chips nur ein Bit übertragen werden.

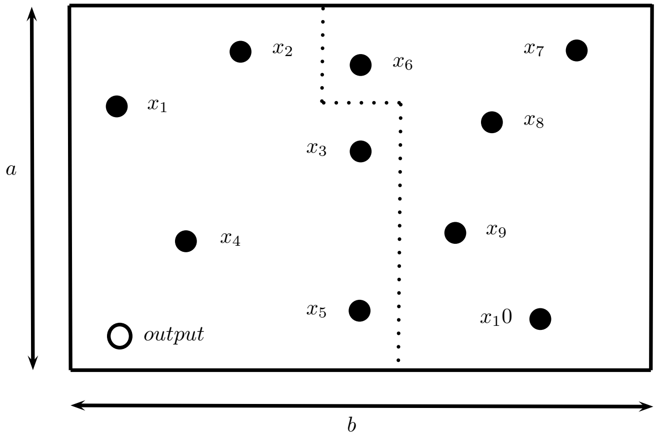
Herleitung von AT^2-Schranken

1. Wählen Schnitt w
   1. Partitionierung der Eingabebits in zwei gleich große Mengen
   2. Für rechteckigen Chip liegt die Länge von $|w|$ in $O(b)$.
2. mindestens αn viele Bits müssen w überqueren
3. Innerhalb von T Taktzyklen: Maximal $|w|T$ viele Bits können den Schnitt überqueren
4. Somit gilt $\alpha n \leq |w|T$.
5. $|w| \in O(b) \Rightarrow |w|^2 \in O(A)$
6. Wir erhalten $\alpha^2 n^2 \leq |w|^2 T^2$ bzw. $n^2 \in O(AT^2)$

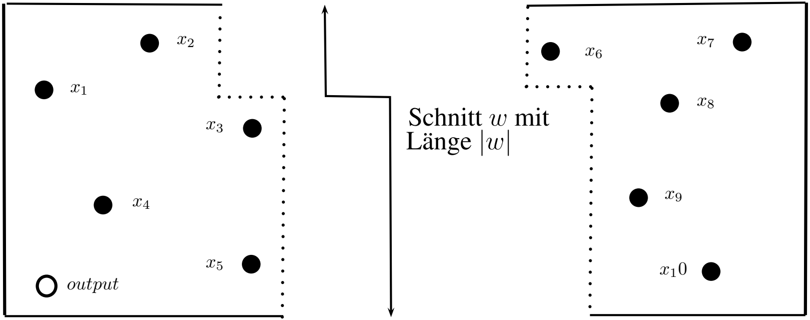
Herleitung von AT^2-Schranken



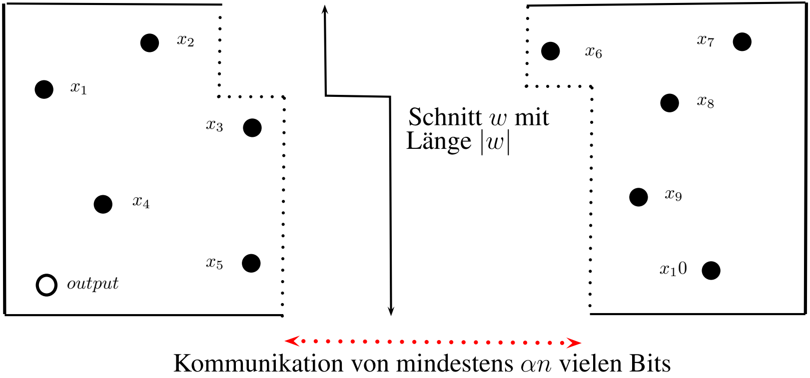
Herleitung von AT^2-Schranken



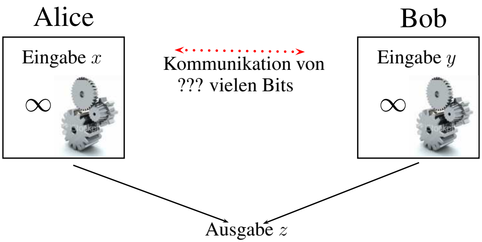
Herleitung von AT^2-Schranken



Herleitung von AT^2-Schranken



Verallgemeinerung: Kommunikationskomplexität



Problem

Alice und Bob sollen die Funktion $f : X \times Y \to Z$ für ein $(x, y) \in X \times Y$ berechnen. Dabei kennt Alice nur x und Bob ausschließlich y. *Wie viel Kommunikation ist zwischen Alice und Bob mindestens erforderlich?*

Ein Hinweis zur Literatur

- Für dieses Seminar ist eigentlich [3] die Standardliteratur
- Ich verwende hingegen [4], da die Vorgehensweise in diesem strukturierter ist.

Protokoll

Definition 1

Protokoll

Ein Protokoll P auf der Menge $X \times Y$ von Eingaben und dem Bild Z ist ein binärer Baum, in dem jeder interner Knoten v mit einer Funktion $a_v : X \to \{0, 1\}$ oder einer Funktion $b_v : Y \to \{0, 1\}$ beschriftet ist und jedes Blatt ein Element aus Z ist.

Kosten eines Protokolls

Definition 2

Kosten eines Protokolls
Die Kosten eines Protokolls bei Eingabe $(x, y) \in X \times Y$ ist die Länge des Pfades von der Wurzel bis zum Blatt. Die Kosten des Protokolls ohne spezifische Eingabe ist die Höhe des Protokoll-Baums.

Deterministische Kommunikationskomplexität

Definition 3

Deterministische Kommunikationskomplexität
Sei eine Funktion $f : X \times Y \to Z$ gegeben. Die deterministische Kommunikationskomplexität $D(f)$ Funktion f ist das Minimum über alle Kosten derjeniger Protokolle, die f berechnen.

Kombinatorisches Rechteck

Definition 4

Kombinatorisches Rechteck
Ein kombinatorisches Rechteck aus $X \times Y$ ist eine Teilmenge $R \subseteq X \times Y$, so dass $R = A \times B$ mit $A \subseteq X$ und $B \subseteq X$.

Theorem 1

Theorem 1

$R \subseteq X \times Y$ ist ein (kombinatorisches) Rechteck $\Leftrightarrow$ $(x_1, y_1) \in R \wedge (x_2, y_2) \in R \Rightarrow (x_1, y_2) \in R$ gilt.

$$\Rightarrow \quad \begin{tikzcd} (x_1, y_1) \in R \arrow[dr, dashed] & & (x_2, y_2) \in R \arrow[dl, dashed] \\ & (x_1, y_2) \in R & \end{tikzcd}$$

$$\Rightarrow \quad \begin{tikzcd} (x_1, y_1) \in R \arrow[dr, dashed] & & (x_2, y_2) \in R \arrow[dl, dashed] \\ & (x_2, y_1) \in R & \end{tikzcd}$$

Beweis Theorem 1

Beweis.

$\Leftarrow$: Seien $A = \{x | \exists y' : (x, y') \in R\}$ und $B = \{y | \exists x' : (x', y) \in R\}$. Wir zeigen, dass $R = A \times B$ ist.

Beweis Theorem 1

Beweis.

$\Leftarrow$: Seien $A = \{x | \exists y' : (x, y') \in R\}$ und $B = \{y | \exists x' : (x', y) \in R\}$. Wir zeigen, dass $R = A \times B$ ist.

- $R \subseteq A \times B$: Da $(x, y) \in R$ gilt nach Definition von A bzw B auch $x \in A$ und $y \in B$.

Beweis Theorem 1

Beweis.

$\Leftarrow$: Seien $A = \{x | \exists y' : (x, y') \in R\}$ und $B = \{y | \exists x' : (x', y) \in R\}$. Wir zeigen, dass $R = A \times B$ ist.

- $R \subseteq A \times B$: Da $(x, y) \in R$ gilt nach Definition von A bzw B auch $x \in A$ und $y \in B$.
- $A \times B \subseteq R$: Sei $(x, y) \in A \times B$.
  - Da $x \in A$ muss es ein y' geben, so dass $(x, y') \in R$.

Beweis Theorem 1

Beweis.

$\Leftarrow$: Seien $A = \{x|\exists y' : (x, y') \in R\}$ und $B = \{y|\exists x' : (x', y) \in R\}$. Wir zeigen, dass $R = A \times B$ ist.

- $R \subseteq A \times B$: Da $(x, y) \in R$ gilt nach Definition von A bzw B auch $x \in A$ und $y \in B$.
- $A \times B \subseteq R$: Sei $(x, y) \in A \times B$.
  - Da $x \in A$ muss es ein y' geben, so dass $(x, y') \in R$.
  - Analog muss es ein x' geben, so dass $(x', y) \in R$, da $y \in R$.

Beweis Theorem 1

Beweis.

$\Leftarrow$: Seien $A = \{x | \exists y' : (x, y') \in R\}$ und $B = \{y | \exists x' : (x', y) \in R\}$. Wir zeigen, dass $R = A \times B$ ist.

- $R \subseteq A \times B$: Da $(x, y) \in R$ gilt nach Definition von A bzw B auch $x \in A$ und $y \in B$.
- $A \times B \subseteq R$: Sei $(x, y) \in A \times B$.
  - Da $x \in A$ muss es ein y' geben, so dass $(x, y') \in R$.
  - Analog muss es ein x' geben, so dass $(x', y) \in R$, da $y \in R$.
  - Gemäß der Annahme $(x_1, y_1) \in R \wedge (x_2, y_2) \in R \Rightarrow (x_1, y_2) \in R$ ist daher auch $(x, y) \in R$.

$\square$

Beweis Theorem 1

Beweis.

$\Rightarrow$: Sei $R = A \times B$ ein Rechteck. Wenn $(x_1, y_1) \in R$ und $(x_2, y_2) \in R$ so gilt $x_1 \in A$ und $y_2 \in B$. Da $R = A \times B$ muss daher auch $(x_1, y_2) \in R$ sein. $\square$

R_v

Definition 5

Sei P ein Protokoll mit Definitionsbereich $X \times Y$ und v ein Knoten innerhalb des Protokoll-Baums. Wir definieren als R_v die Menge von Eingaben $(x, y) \in X \times Y$ welche bei Traversierung des Baums den Knoten v erreichen.

$\{R_{l \in L}\}$ ist eine Partition

Korollar 1

Wenn L die Menge an Blättern des Protokoll-Baums ist, so ist $\{R_{l \in L}\}$ eine Partition des Eingaberaums.

Beweis.

- Annahme: $R_v \neq \emptyset$

$\{R_{l \in L}\}$ ist eine Partition

Korollar 1

Wenn L die Menge an Blättern des Protokoll-Baums ist, so ist $\{R_{l \in L}\}$ eine Partition des Eingaberaums.

Beweis.

- Annahme: $R_v \neq \emptyset$
- Eingabe $(x, y) \in X \times Y$ führt zu genau einem Blatt

$\square$

R_v ist ein Rechteck

Theorem 2

Sei P ein Protokoll und v ein Knoten des entsprechenden Protokollbaums, so ist R_v ein Rechteck.

Beweis.

- Induktion über die Tiefe des Knotens v

R_v ist ein Rechteck

Theorem 2

Sei P ein Protokoll und v ein Knoten des entsprechenden Protokollbaums, so ist R_v ein Rechteck.

Beweis.

- Induktion über die Tiefe des Knotens v
- $R_{wurzel} = X \times Y$

R_v ist ein Rechteck

Theorem 2

Sei P ein Protokoll und v ein Knoten des entsprechenden Protokollbaums, so ist R_v ein Rechteck.

Beweis.

- Induktion über die Tiefe des Knotens v
- $R_{wurzel} = X \times Y$
- Sei w der Elternknoten von v, v linkes Kindelement von w, Alice spricht in w

R_v ist ein Rechteck

Theorem 2

Sei P ein Protokoll und v ein Knoten des entsprechenden Protokollbaums, so ist R_v ein Rechteck.

Beweis.

- Induktion über die Tiefe des Knotens v
- $R_{wurzel} = X \times Y$
- Sei w der Elternknoten von v, v linkes Kindelement von w, Alice spricht in w
- $R_v = R_w \cap \{(x, y) | a_w(x) = 0\}$

R_v ist ein Rechteck

Theorem 2

Sei P ein Protokoll und v ein Knoten des entsprechenden Protokollbaums, so ist R_v ein Rechteck.

Beweis.

- Induktion über die Tiefe des Knotens v
- $R_{wurzel} = X \times Y$
- Sei w der Elternknoten von v, v linkes Kindelement von w, Alice spricht in w
- $R_v = R_w \cap \{(x,y) | a_w(x) = 0\}$
- Da $R_w = A_w \times B_w$ folgt $R_v = (A_w \cap \{x | a_w(x) = 0) \times B_w$.

$\square$

Monochromatische Rechtecke

Definition 6

Eine Teilmenge $R \subseteq X \times Y$ wird als f-monochromatisch bezeichnet, falls die Funktion f auf R konstant ist.

Ein Protokoll induziert eine f-monochromatische Partition

Lemma 1

Jedes Protokoll P einer Funktion f induziert eine Partition von $X \times Y$ in f-monochromatische Rechtecke. Die Anzahl an Rechtecken ist die Anzahl der Blätter von P.

Untere Schranke über die Anzahl der monochromatischen Rechtecke

Korollar 2

Wenn jede Partition von $X \times Y$ in f-monochromatische Rechtecke mindestens t Rechtecke erfordert, gilt $D(f) \geq log_2 t$.

Beweis.

- Lemma 1: Blätter induzieren Partition in f-monochromatische Rechtecke

Untere Schranke über die Anzahl der monochromatischen Rechtecke

Korollar 2

Wenn jede Partition von $X \times Y$ in f-monochromatische Rechtecke mindestens t Rechtecke erfordert, gilt $D(f) \geq log_2 t$.

Beweis.

- Lemma 1: Blätter induzieren Partition in f-monochromatische Rechtecke
- $\Rightarrow$Jedes Protokoll hat immer mindestens t viele Blätter

Untere Schranke über die Anzahl der monochromatischen Rechtecke

Korollar 2

Wenn jede Partition von $X \times Y$ in f-monochromatische Rechtecke mindestens t Rechtecke erfordert, gilt $D(f) \geq log_2 t$.

Beweis.

- Lemma 1: Blätter induzieren Partition in f-monochromatische Rechtecke
- $\Rightarrow$Jedes Protokoll hat immer mindestens t viele Blätter
- $\Rightarrow$ Höhe des Baums $\geq log_2 t$

□

Fooling Set

Definition 7

Sei $f : X \times Y \to \{0, 1\}$. Eine Menge $S \subset X \times Y$ wird als Fooling Set bezeichnet, wenn es einen Wert $z \in \{0, 1\}$ gibt, so dass

- Für jedes $(x, y) \in S$ gilt $f(x, y) = z$.

Fooling Set

Definition 7

Sei $f : X \times Y \to \{0, 1\}$. Eine Menge $S \subset X \times Y$ wird als Fooling Set bezeichnet, wenn es einen Wert $z \in \{0, 1\}$ gibt, so dass

- Für jedes $(x, y) \in S$ gilt $f(x, y) = z$.
- Für alle paarweise verschiedenen Elemente (x_1, y_1) und (x_2, y_2) aus S gilt entweder $f(x_1, y_2) \neq z$ oder $f(x_2, y_1) \neq z$.

Theorem 3

Theorem 3

Wenn die Funktion f ein Fooling Set S der Größe t hat, gilt $D(f) \geq log_2 t$.

Beweis.

- Widerspruchsbeweis: Keine zwei Elemente aus S können im gleichen monochromatischen Rechteck liegen

Theorem 3

Theorem 3

Wenn die Funktion f ein Fooling Set S der Größe t hat, gilt $D(f) \geq log_2 t$.

Beweis.

- Widerspruchsbeweis: Keine zwei Elemente aus S können im gleichen monochromatischen Rechteck liegen
- Annahme: Zwei Elemente (x_1, y_1), $(x_2, y_2) \in S$ sind Teil des gleichen monochromatischen Rechtecks R

Theorem 3

Theorem 3

Wenn die Funktion f ein Fooling Set S der Größe t hat, gilt $D(f) \geq log_2 t$.

Beweis.

- Widerspruchsbeweis: Keine zwei Elemente aus S können im gleichen monochromatischen Rechteck liegen
- Annahme: Zwei Elemente (x_1, y_1), $(x_2, y_2) \in S$ sind Teil des gleichen monochromatischen Rechtecks R
- Theorem 1: $\Rightarrow (x_1, y_2),(x_2, y_1) \in R$

Theorem 3

Theorem 3

Wenn die Funktion f ein Fooling Set S der Größe t hat, gilt $D(f) \geq log_2 t$.

Beweis.

- Widerspruchsbeweis: Keine zwei Elemente aus S können im gleichen monochromatischen Rechteck liegen
- Annahme: Zwei Elemente (x_1, y_1), $(x_2, y_2) \in S$ sind Teil des gleichen monochromatischen Rechtecks R
- Theorem 1: $\Rightarrow (x_1, y_2), (x_2, y_1) \in R$
- Es gilt jedoch $f(x_1, y_2) \neq f(x_1, y_1)$ oder $f(x_2, y_1) \neq f(x_1, y_1)$. Widerspruch.
- Theorem 3 folgt aus Korollar 2.

Beispiel EQ

EQ

Die Funktion $EQ : \{0,1\}^n \times \{0,1\}^n \rightarrow \{0,1\}$ ist definiert als $EQ(x,y) = \begin{cases} 1 & , x = y \\ 0 & , sonst \end{cases}$. Als Fooling Set wählen wir $S = \{(i,i) | i \in \{0,1\}^n\}$.

Dies ist ein Fooling Set, da

- $\forall s \in S : EQ(s) = 1$
- $\forall (x_1,y_1),(x_2,y_2) \in S \wedge (x_1,y_1) \neq (x_2,y_2) : EQ(x_1,y_2) = 0$.

Gemäß Theorem 3 gilt daher $D(EQ) \geq log_2(|S|) = log_2(2^n) = n$.

Beispiel DISJ

DISJ

Die Funktion $DISJ : X \times Y \to \{0,1\}$, mit $X, Y \subseteq 2^{\{0,1\}^n}$ ist definiert als $DISJ(x,y) = \begin{cases} 1 & , x \cap y = \emptyset \\ 0 & , sonst \end{cases}$. Als Fooling Set wählen wir $S = \{(A, \bar{A}) | A \subseteq 2^{\{0,1\}^n}\}$.

Dies ist ein Fooling Set, da

- $\forall s \in S : DISJ(s) = 1$
- $\forall (A_1, \bar{A}_1), (A_2, \bar{A}_2) \in S \wedge (A_1, \bar{A}_1) \neq (A_2, \bar{A}_2) :$ $DISJ(A_1, \bar{A}_2) = 0 \vee DISJ(A_2, \bar{A}_1) = 0$.

Gemäß Theorem 3 gilt daher $D(DISJ) \geq log_2(|S|) = log_2(2^{2^n}) = 2^n$.

M_f

Definition 8

M_f

Sei $f : X \times Y \to \{0, 1\}$ eine Funktion. Wir assoziieren mit der Funktion f die Matrix M_f, in welcher der Funktionswert von f abgespeichert ist. Die Matrix sei über die Eingabewerte (x, y) indizierbar.

Theorem 4

Theorem 4

Für eine Funktion $f : X \times Y \to \{0, 1\}$ *gilt* $D(f) \geq log_2 rang(f)$.

Beweis.

- Hilfsvariablen
  - $B_1 \triangleq$ Menge an Blättern, welche das Ergebnis 1 repräsentieren

Theorem 4

Theorem 4

Für eine Funktion $f : X \times Y \to \{0, 1\}$ gilt $D(f) \geq log_2 rang(f)$.

Beweis.

- Hilfsvariablen
  - $B_1 \triangleq$ Menge an Blättern, welche das Ergebnis 1 repräsentieren
  - MatrixM_b für $b \in B_1 : M_b(x, y) = 1 \Leftrightarrow (x, y) \in R_b$ und $M_b(x, y) = 0 \Leftrightarrow (x, y) \notin R_b$.

$\square$

Beweis Theorem 4

Beweis.

Für jedes (x, y) der Eingabe gilt:

- Es gilt $M_f = \sum_{b \in B_1} M_b$.

Beweis Theorem 4

Beweis.

Für jedes (x, y) der Eingabe gilt:

- Es gilt $M_f = \sum_{b \in B_1} M_b$.
  - $f(x, y) = 1 \Rightarrow \exists! b \in B_1 : M_b(x, y) = 1$
  - $f(x, y) = 0 \Rightarrow \forall b \in B_1 : M_b(x, y) = 0$

Beweis Theorem 4

Beweis.

Für jedes (x, y) der Eingabe gilt:

- Es gilt $M_f = \sum_{b \in B_1} M_b$.
  - $f(x, y) = 1 \Rightarrow \exists! b \in B_1 : M_b(x, y) = 1$
  - $f(x, y) = 0 \Rightarrow \forall b \in B_1 : M_b(x, y) = 0$
- Es gilt $rang(A + B) \leq rang(A) + rang(B)$

Beweis Theorem 4

Beweis.

Für jedes (x, y) der Eingabe gilt:

- Es gilt $M_f = \sum_{b \in B_1} M_b$.
  - $f(x, y) = 1 \Rightarrow \exists! b \in B_1 : M_b(x, y) = 1$
  - $f(x, y) = 0 \Rightarrow \forall b \in B_1 : M_b(x, y) = 0$
- Es gilt $rang(A + B) \leq rang(A) + rang(B)$
  - $\Rightarrow rang(M_f) = rang(\sum_{b \in B_1} M_b) \leq \sum_{b \in B_1} rang(M_b)$.

Beweis Theorem 4

Beweis.

Für jedes (x, y) der Eingabe gilt:

- Es gilt $M_f = \sum_{b \in B_1} M_b$.
  - $f(x, y) = 1 \Rightarrow \exists! b \in B_1 : M_b(x, y) = 1$
  - $f(x, y) = 0 \Rightarrow \forall b \in B_1 : M_b(x, y) = 0$
- Es gilt $rang(A + B) \leq rang(A) + rang(B)$
  - $\Rightarrow rang(M_f) = rang(\sum_{b \in B_1} M_b) \leq \sum_{b \in B_1} rang(M_b)$.
- Es gilt $rang(M_b) = 1 \Rightarrow rang(M_f) \leq |B_1| \leq |B|$

Beweis Theorem 4

Beweis.

Für jedes (x, y) der Eingabe gilt:

- Es gilt $M_f = \sum_{b \in B_1} M_b$.
  - $f(x, y) = 1 \Rightarrow \exists! b \in B_1 : M_b(x, y) = 1$
  - $f(x, y) = 0 \Rightarrow \forall b \in B_1 : M_b(x, y) = 0$
- Es gilt $rang(A + B) \leq rang(A) + rang(B)$
  - $\Rightarrow rang(M_f) = rang(\sum_{b \in B_1} M_b) \leq \sum_{b \in B_1} rang(M_b)$.
- Es gilt $rang(M_b) = 1 \Rightarrow rang(M_f) \leq |B_1| \leq |B|$
- Das Theorem folgt aus Korollar 2.

$\square$

Beispiel IP

IP

Die Funktion $IP : \{0,1\}^n \times \{0,1\}^n \rightarrow \{0,1\}$ ist definiert als $IP(x,y) = \sum x_i \cdot y_i (mod\, 2)$. Sie berechnet somit, ob die Anzahl an übereinstimmenden 1en gerade oder ungerade ist.

Wir betrachten beispielhaft die Matrize

$$M_{IP} = \begin{array}{ccccc} - & 00 & 01 & 10 & 11 \\ 00 & 0 & 0 & 0 & 0 \\ 01 & 0 & 1 & 0 & 1 \\ 10 & 0 & 0 & 1 & 1 \\ 11 & 0 & 1 & 1 & 0 \end{array}.$$

Beispiel IP

- $N = (M_{IP})^2 = \begin{matrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \end{matrix} \times \begin{matrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \end{matrix} = \begin{matrix} 0 & 0 & 0 & 0 \\ 0 & 2 & 1 & 1 \\ 0 & 1 & 2 & 1 \\ 0 & 1 & 1 & 2 \end{matrix}.$
- $N_{y,y'} = \sum_{z \in \{0,1\}^n} \langle y, z \rangle \cdot \langle z, y' \rangle$
- $\Rightarrow N_{y,y'} =$Anzahl an Zahlen $z \in \{0,1\}^n$, so dass $\langle y, z \rangle \cdot \langle z, y' \rangle = 1$.

Beispiel IP

- Überlegung:
  - Auf der Diagonalen ist $N_{y,y} = \sum_{z \in \{0,1\}^n} \langle y, z \rangle \cdot \langle z, y \rangle = 2^{n-1}$, da $\langle y, z \rangle = \langle z, y \rangle$ und es 2^{n-1} viele Strings der Länge n gibt, so dass die Anzahl ungerade ist.

Beispiel IP

- Überlegung:
  - Auf der Diagonalen ist $N_{y,y} = \sum_{z \in \{0,1\}^n} \langle y, z \rangle \cdot \langle z, y \rangle = 2^{n-1}$, da $\langle y, z \rangle = \langle z, y \rangle$ und es 2^{n-1} viele Strings der Länge n gibt, so dass die Anzahl ungerade ist.
  - In der ersten Spalte bzw. der ersten Zeile können nur 0en auftreten.

Beispiel IP

- Überlegung:
  - Auf der Diagonalen ist $N_{y,y} = \sum_{z \in \{0,1\}^n} \langle y, z \rangle \cdot \langle z, y \rangle = 2^{n-1}$, da $\langle y, z \rangle = \langle z, y \rangle$ und es 2^{n-1} viele Strings der Länge n gibt, so dass die Anzahl ungerade ist.
  - In der ersten Spalte bzw. der ersten Zeile können nur 0en auftreten.
  - Alle sonstigen Felder gleichen 2^{n-2}.

Beispiel IP

- Überlegung:
  - Auf der Diagonalen ist $N_{y,y} = \sum_{z \in \{0,1\}^n} \langle y, z \rangle \cdot \langle z, y \rangle = 2^{n-1}$, da $\langle y, z \rangle = \langle z, y \rangle$ und es 2^{n-1} viele Strings der Länge n gibt, so dass die Anzahl ungerade ist.
  - In der ersten Spalte bzw. der ersten Zeile können nur 0en auftreten.
  - Alle sonstigen Felder gleichen 2^{n-2}.
  - $rang(N) = 2^n - 1$

Beispiel IP

- Überlegung:
  - Auf der Diagonalen ist $N_{y,y} = \sum_{z \in \{0,1\}^n} \langle y, z \rangle \cdot \langle z, y \rangle = 2^{n-1}$, da $\langle y, z \rangle = \langle z, y \rangle$ und es 2^{n-1} viele Strings der Länge n gibt, so dass die Anzahl ungerade ist.
  - In der ersten Spalte bzw. der ersten Zeile können nur 0en auftreten.
  - Alle sonstigen Felder gleichen 2^{n-2}.
  - $rang(N) = 2^n - 1$
  - Da $rang(A \times B) \leq min(rang(A), rang(B))$, folgt $rang(M_{IP}) \geq 2^n - 1$.

Beispiel IP

- Überlegung:
  - Auf der Diagonalen ist $N_{y,y} = \sum_{z \in \{0,1\}^n} \langle y,z \rangle \cdot \langle z,y \rangle = 2^{n-1}$, da $\langle y,z \rangle = \langle z,y \rangle$ und es 2^{n-1} viele Strings der Länge n gibt, so dass die Anzahl ungerade ist.
  - In der ersten Spalte bzw. der ersten Zeile können nur 0en auftreten.
  - Alle sonstigen Felder gleichen 2^{n-2}.
  - $rang(N) = 2^n - 1$
  - Da $rang(A \times B) \leq min(rang(A), rang(B))$, folgt $rang(M_{IP}) \geq 2^n - 1$.
  - Somit gilt nach Theorem 4 $D(IP) \geq log(2^n - 1) \geq n - 1$.

Bibliographie

A. C.-C. Yao, "Some complexity questions related to distributive computing(preliminary report)," in *STOC '79: Proceedings of the eleventh annual ACM symposium on Theory of computing*, (New York, NY, USA), ACM, 1979.

C. D. Thompson, "Area-time complexity for vlsi," in *STOC '79: Proceedings of the eleventh annual ACM symposium on Theory of computing*, (New York, NY, USA), ACM, 1979.

S. Arora and B. Barak, *Computational Complexity: A Modern Approach*.
New York, NY, USA: Cambridge University Press, 2009.

E. Kushilevitz and N. Nisan, *Communication complexity*.
New York, NY, USA: Cambridge University Press, 1997.