



Zur impliziten Darstellung von Graphen durch OBDDs

Robin Nunkesser

23. Juni 2004



1. Überblick

Überblick

1. Überblick

2. Motivation

3. Grundlagen

4. Ergebnisse

5. Eigene Ergebnisse

6. Ausblick



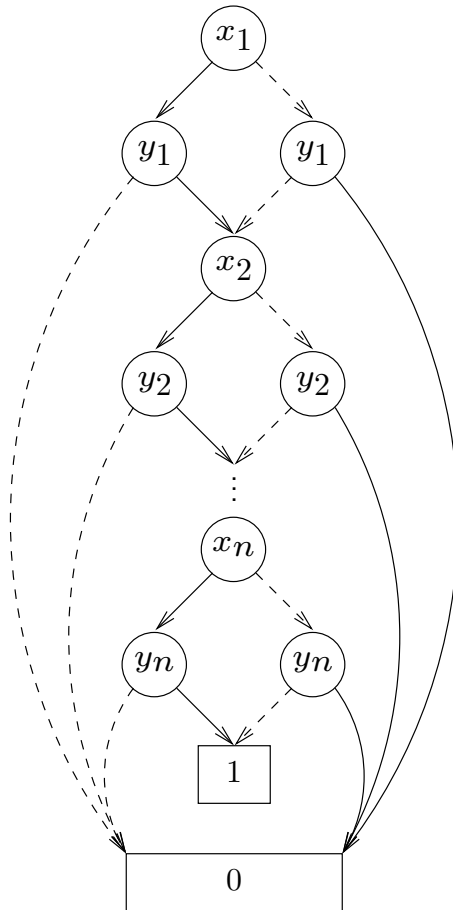
Motivation

	Adjazenzmatrix	Adjazenzliste	Inzidenzmatrix	OBDDs
Platzbedarf	$\Theta(n^2)$	$\mathcal{O}(n + m)$	$\Theta(nm)$?
Adjazenzabfrage	$\mathcal{O}(1)$	$\mathcal{O}(n)$	$\mathcal{O}(1)$	$\mathcal{O}(\log n)$

- Implizite Darstellung von Graphen ermöglicht Speicherung sehr großer Graphen
- Implizite Algorithmen können Strukturinformationen direkt nutzen



Grundlagen

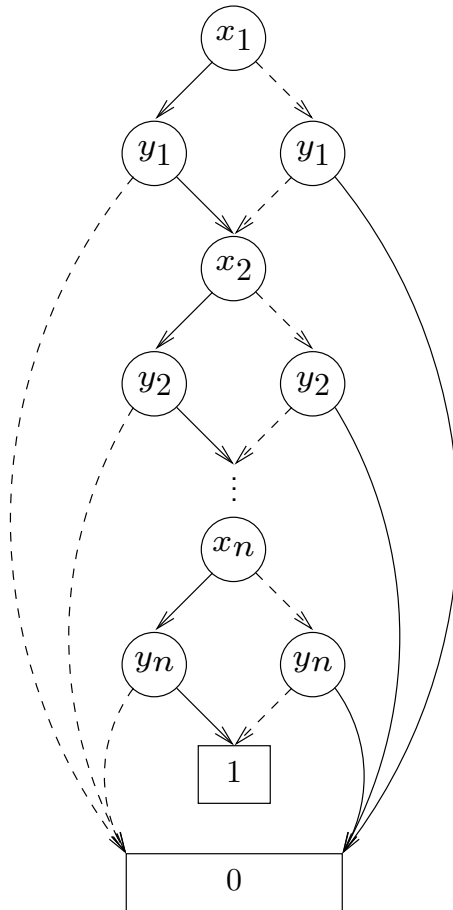


- OBDDs repräsentieren boolesche Funktionen durch einen Graphen
- OBDDs sind gerichtete kreisfreier Graphen mit Quelle und Senken, innere Knoten haben zwei ausgehende Kanten
- OBDD in Abbildung repräsentiert Gleichheitstest: Für $x, y \in \{0, 1\}^n$ ist die Funktion $\text{EQUAL}(x, y) \in B_{2n}$ definiert als:

$$\text{EQUAL}(x, y) = 1 \iff x = y.$$



Grundlagen



Definition 1 Graph G wird implizit durch OBDD dargestellt, wenn das OBDD die charakteristische Funktion der Kanten von G

$$\chi_{E(G)} : U \times U \rightarrow \{0, 1\} \text{ für } U \supseteq V(G)$$

repräsentiert:

$$\chi_{E(G)}(v, w) = 1 \iff \{v, w\} \in E(G).$$



Die Knoten werden dazu binär kodiert.



Ergebnisse

- Daniel Sawitzki ([Saw02]): Obere Schranke für die Größe einer impliziten Darstellung für quadratische Gitter mit $\chi_V \equiv 1$ von

$$21 \log n - 12$$

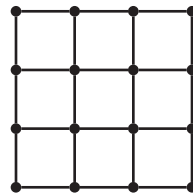


Abbildung 1: Beispiel eines quadratischen Gitters

- Philipp Wölfel ([Wöl03]): Obere Schranke für die Größe einer impliziten Darstellung für Cographen von

$$4n \log n + 8n$$



Eigene Ergebnisse

Graphklasse	Obere Schranke für OBDD-Größe
Beliebigdimensionale Gitter mit $\chi_V \equiv 1$	$14 \log n$
Beliebigdimensionale Tori mit $\chi_V \equiv 1$	$22 \log n$
Beliebigdimensionale Gitter	$21,04 \log n$
Beliebigdimensionale Tori	$28,25 \log n$
Cograph (P_4 -frei)	$3n \log n + 2n - \log n + 3,5$
P_4 -reduzierbar	$3,25n \log n + 2n - 2 \log n + 2,5$
P_4 -spärlich	$3,5n \log n + 2n - 2 \log n + 3,5$
P_4 -erweiterbar	$3,25n \log n + 2n - 2 \log n + 2,5$
Einheitsintervallgraphen	$10n + 1$



Graphen mit wenigen P_4 s - Cographen

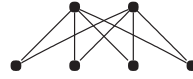


Abbildung 2: Beispiel eines Cographen

Definition 2 Ein Graph ist genau dann ein *Cograph*, wenn er keinen P_4 als induzierten Teilgraphen enthält. \diamond

- Aus Einzelknotengraphen durch Anwendung zweier Operationen konstruierbar ([Ler72]):

Vereinigung:

$$G_1 \cup G_2 = (V(G_1) \cup V(G_2), E(G_1) \cup E(G_2))$$

***-Produkt:**

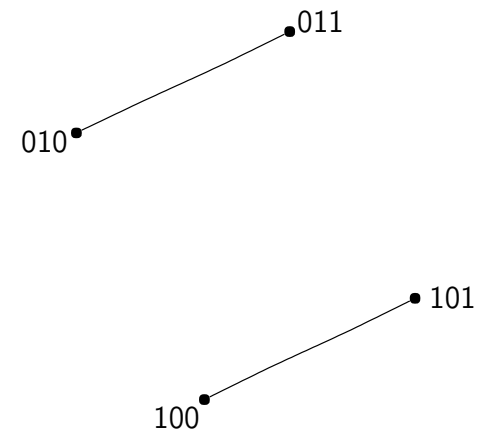
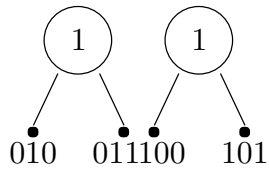
$$G_1 * G_2 = (V(G_1) \cup V(G_2), E(G_1) \cup E(G_2) \cup \{\{v, w\} \mid v \in V(G_1), w \in V(G_2)\})$$

- Eindeutig durch einen Baum darstellbar ([Ler71])



5. Eigene Ergebnisse

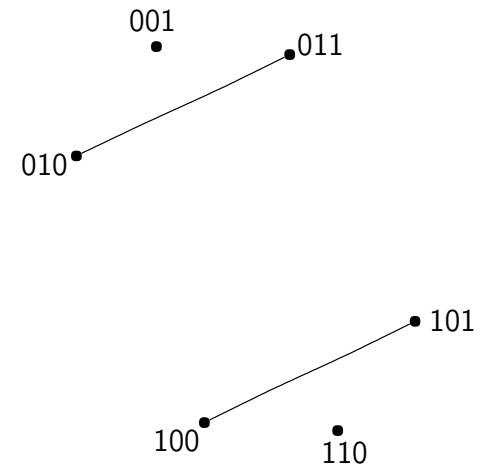
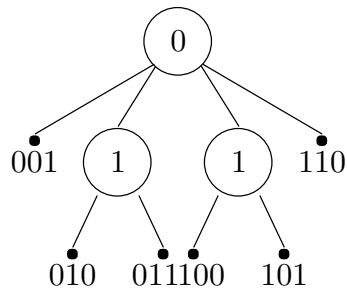
Cobäume





5. Eigene Ergebnisse

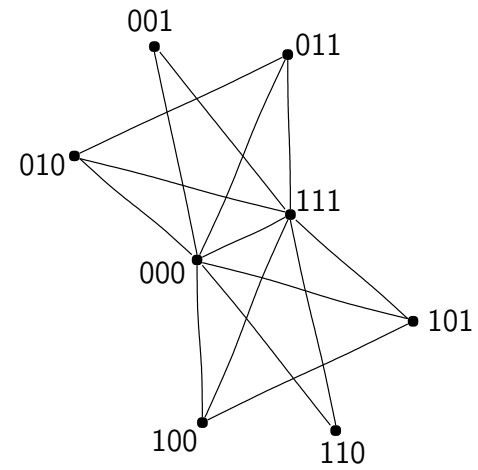
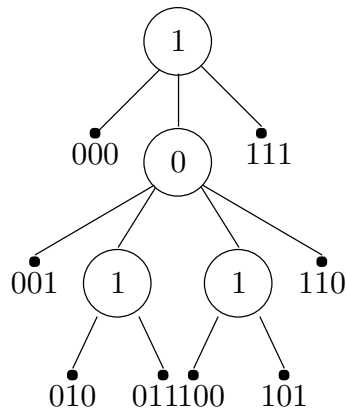
Cobäume





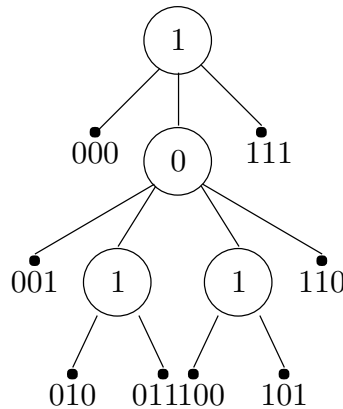
5. Eigene Ergebnisse

Cobäume





Algorithmus zur Berechnung von $\chi_{E(G)}(x, y)$



- Tiefster gemeinsamer Vorgängerknoten zweier Blätter gibt an, ob die zugehörigen Knoten des Cographen adjazent sind
- Kodierung der Knoten gemäß Preorder-Traversierung des Cobaums

1. Volles Lesen von x

Kosten: $2n - \frac{n}{2^{\lceil \log n \rceil}} + \lceil \log n \rceil + 1$

(Abschätzung für $\sum_{i=0}^{\lceil \log n \rceil} \lceil \frac{n}{2^i} \rceil$)

2. Bei jedem Leseschritt von y muss tiefster möglicher gemeinsamer Vorgängerknoten und $x < y$, $x > y$ oder $x = y$ gespeichert werden

Kosten: $(\lceil \log n \rceil - 1)(n + 2(n - 1))$

Gesamt: $3n \log n + 2n - \log n + 3,5$



Graphen mit wenigen P_4 s ([JO89, Hoá85, JO91])



Abbildung 3: Ein P_4 -reduzierbarer Graph

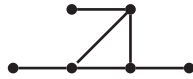


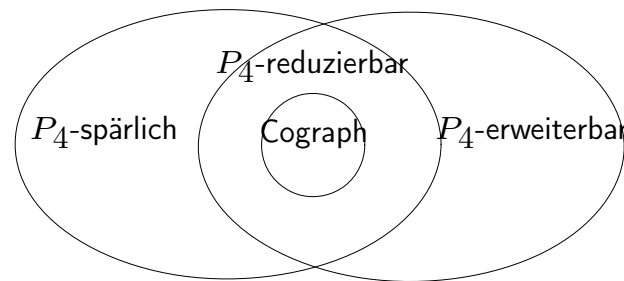
Abbildung 4: Ein P_4 -spärlicher Graph



Abbildung 5: Ein P_4 -erweiterbarer Graph

Definition 3

1. Ein G ist P_4 -reduzierbar, wenn jeder Knoten zu höchstens einem P_4 des G gehört.
2. Ein Graph ist P_4 -spärlich, wenn jede beliebige Menge von fünf Knoten des Graphen höchstens einen P_4 induziert.
3. Ein Graph ist P_4 -erweiterbar, wenn es zu jedem induzierten P_4 höchstens einen weiteren Knoten gibt, der mit drei Knoten des P_4 s ebenfalls einen P_4 induziert. \diamond





P_4 -reduzierbare Graphen

Werden durch eine zusätzliche Operation erzeugt:

Definition 4 ([JO89]) Seien $G_1 = (V_1, E_1)$ und $G_2 = (V_2, E_2)$ Graphen mit disjunkten Knotenmengen und den Eigenschaften:

1. $V_1 = \{a, d\}, E_1 = \emptyset$.
2. Es gibt Knoten $b, c \in V_2$ mit $\{\{b, v\} \mid v \in V_2, v \neq b\} \cup \{\{c, v\} \mid v \in V_2, v \neq c\} \subseteq E_2$.

Dann ist

$$G_1 \text{ (2) } G_2 := (V_1 \cup V_2, \{\{a, b\}, \{c, d\} \cup E_2\}).$$



Diese Operation ist unhandlich für den Algorithmus.



P_4 -reduzierbare Graphen

Definition 5

1. Seien $G_1 = (\{a\}, \emptyset)$, $G_2 = (\{b\}, \emptyset)$, $G_3 = (\{c\}, \emptyset)$ und $G_4 = (\{d\}, \emptyset)$ vier Graphen mit disjunkten Knotenmengen. Dann ist die Operation P_4 -Erzeugung $\overset{4}{\dashv}$ definiert als:

$$\overset{4}{\dashv} (G_1, G_2, G_3, G_4) = (\{a, b, c, d\}, \{\{a, b\}, \{b, c\}, \{c, d\}\})$$

2. Seien $G_1 = (V_1 := \{a, b, c, d\}, E_1 := \{\{a, b\}, \{b, c\}, \{c, d\}\})$ ein P_4 und $G_2 = (V_2, E_2)$ ein beliebiger Graph ($V_2 = E_2 = \emptyset$ möglich) mit disjunkten Knotenmengen. Dann ist die Operation *Flügel-Addition* Υ definiert als:

$$G_1 \Upsilon G_2 := (V_1 \cup V_2, E_1 \cup E_2 \cup \{\{b, v\} \mid v \in V_2\} \cup \{\{c, w\} \mid v \in V_2\})$$



Neben tiefstem möglichem gemeinsamen Vorgängerknoten und $x < y$, $x > y$ oder $x = y$ muss noch zwischenzeitlich „ c -ter Knoten P_4 “, „innerer Knoten P_4 “, „äußerer Knoten P_4 “ gespeichert werden. \rightarrow Geschieht durch Wahl eines isomorphen Operationsbaums.



Intervallgraphen

Definition 6 ([Haj57, Rob69])

1. Sei M eine Menge von Objekten, für die eine Schnittoperation \cap definiert ist, dann ist der *Schnittgraph* $G_M = (V, E)$ definiert durch:

$$V = M$$

$$E = \{\{v, w\} \mid v, w \in M, v \cap w \neq \emptyset\}.$$

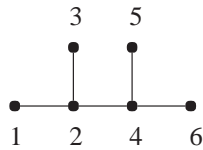


Abbildung 6: Beispiel eines Intervallgraphen und seiner Intervalle

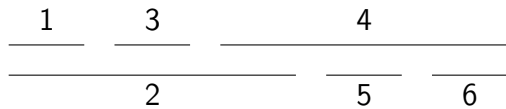


Abbildung 7: Beispiel für die Intervalle des Graphen aus Abbildung 6



Abbildung 8: Klaue

2. Ein *Intervallgraph* ist ein Schnittgraph auf einer endlichen Menge von Intervallen in den reellen Zahlen.
3. Ein *Einheitsintervallgraph* ist ein Intervallgraph, dessen Intervalle alle geschlossen sind und Länge 1 haben. \diamond



Einheitsintervallgraphen

Satz 7 Sei M eine endliche Menge von geschlossenen Intervallen der Länge 1 und G_M der entsprechende Einheitsintervallgraph. Der Graph $G_{M_1} = (V_{M_1}, E_{M_1})$ mit

$$V_{M_1} = \{a \mid [a; b] \in V(G_M)\}$$

$$E_{M_1} = \{\{v, w\} \mid v, w \in V_{M_1}, |v - w| \leq 1\}$$

ist isomorph zu G_M .

Wähle mit $n = \lceil \log |V_{M_1}| \rceil$ die binären Kodierungen $x = x_n x_{n-1} \dots x_1$ und $y = y_n y_{n-1} \dots y_1$ zweier Knoten $v, w \in V_{M_1}$ so, dass gilt:

$$0^{10n} \leq x, y \leq (|V_{M_1}| - 1)^{10n}$$

$$v \leq w \iff x \leq y.$$

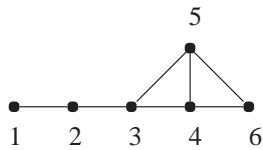


Abbildung 9: Beispiel eines Einheitsintervallgraphen.

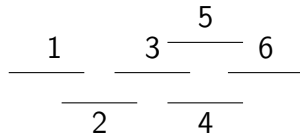


Abbildung 10: Beispiel für die Intervalle des Graphen aus Abbildung 9.



Algorithmus zur Berechnung von $\chi_{E(G)}(x, y)$

1. Abwechselndes Lesen der Bits von x und y
2. Berechnung des möglichen Abstands der durch x und y kodierten Knoten:
 - (a) Sicher kleiner gleich 1: Es existiert Kante zwischen x und y
 - (b) Sicher größer 1: Es existiert keine Kante zwischen x und y
 - (c) Unsicher: Gelesenes speichern

Platzbedarf: $10n + 1$



Beweis der Platzbedarfsschranke

Sei $N := \left\lceil \frac{n}{2^{\lceil \log n \rceil - k}} \right\rceil$, dann ergibt die Abstandsbestimmung von $x_1 x_2 \dots x_k$ und $y_1 y_2 \dots y_l$ höchstens

$$\begin{cases} 1 & , \text{ falls } N_k = 1 \\ 2 & , \text{ falls } N_k = 2, l = k - 1 \\ 4 & , \text{ falls } N_k = 2, l = k \\ 2N_k + 2N_{k+1} - 4 & , \text{ falls } N_k > 2, l = k - 1 \\ 4N_k - 4 & , \text{ falls } N_k > 2, l = k - 1 \end{cases}$$

Mal einen unsicheren Wert. Insgesamt kann der Algorithmus so höchstens

$$\sum_{i=2}^{\lceil \log n \rceil} \left(2 \left\lceil \frac{n}{2^{\lceil \log n \rceil - i}} \right\rceil + 2 \left\lceil \frac{n}{2^{\lceil \log n \rceil - i + 1}} \right\rceil - 4 \right) + \sum_{i=2}^{\lceil \log n \rceil - 1} \left(4 \left\lceil \frac{n}{2^{\lceil \log n \rceil - i}} \right\rceil - 4 \right) + 9 \leq 10n + 1$$

Knoten erzeugen.



Ausblick

1. Beweis unterer Schranken für den worst-case durch Abzählargumente
2. Beweis unterer Schranken unabhängig von der Knotenkodierung ohne Abzählargumente
3. Betrachtung allgemeiner Intervallgraphen



Danke für eure Aufmerksamkeit!



Noch Zeit?!



Gitter und Tori

Definition 8 Sei $G = (V, E)$ ein Graph.

1. G heißt r -dimensionales Gitter (Gr^r), falls $V = V_1 \times V_2 \times \dots \times V_r$ und

$$E(Gr^r) = \left\{ \{(v_1, v_2, \dots, v_r), (w_1, w_2, \dots, w_r)\} \in \mathcal{P}_2(V) \mid \sum_{i=1}^r |v_i - w_i| = 1 \right\}.$$

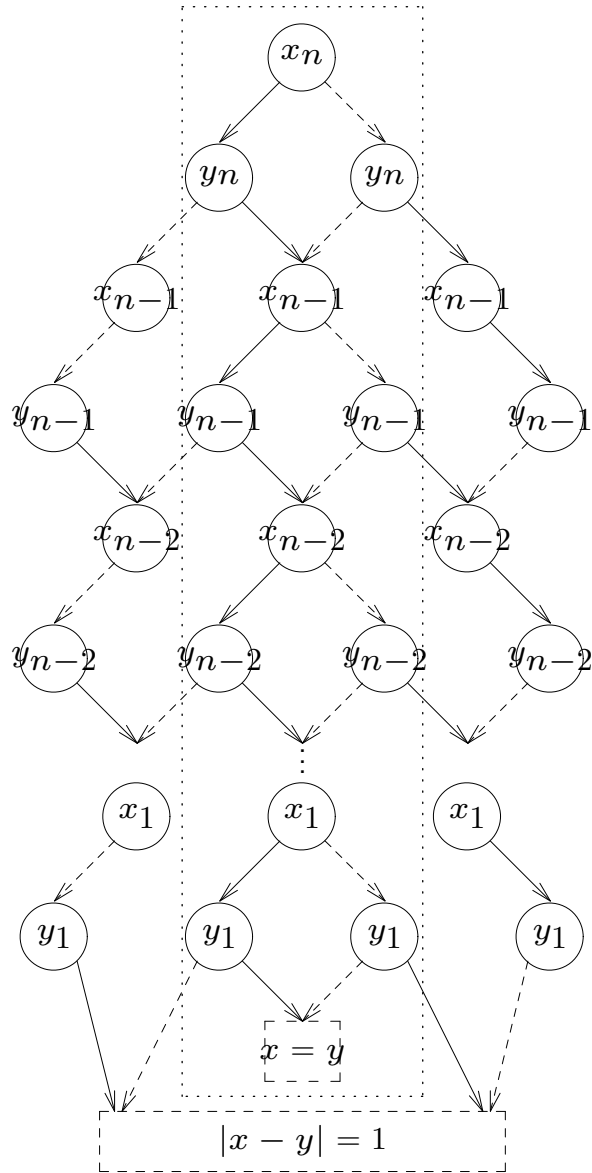
2. G heißt r -dimensionaler Torus (To^r), falls G ein Gitter mit zusätzlichen Kanten ist:

$$\begin{aligned} E(To^r) &= E(Gr^r) \cup \\ &\bigcup_{i=1}^r \{ \{(v_1, \dots, v_r), (w_1, \dots, w_r)\} \in \mathcal{P}_2(V) \mid |v_i - w_i| = |V_i| - 1 \\ &\wedge \sum_{\substack{j=1 \\ j \neq i}}^r |v_j - w_j| = 0 \}. \end{aligned}$$

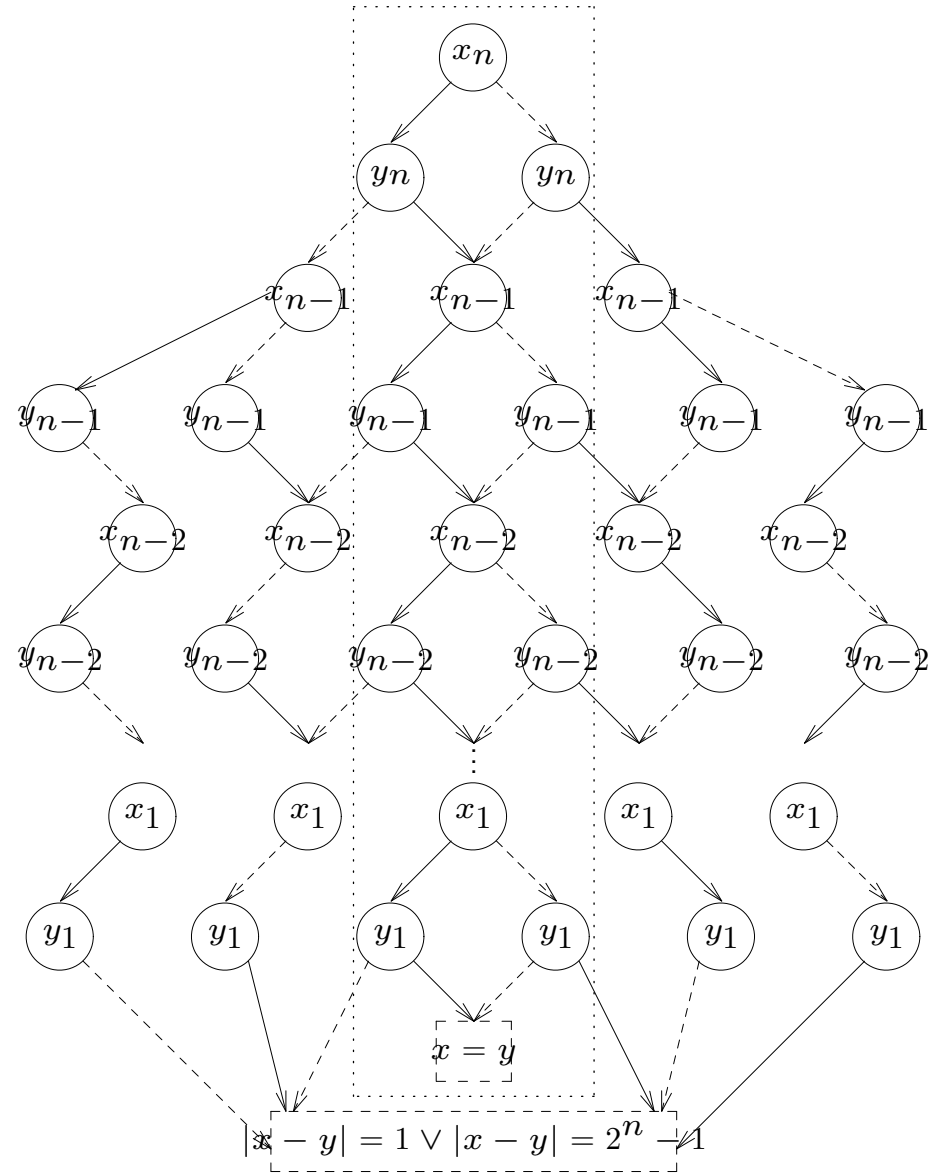




Gitter und Tori



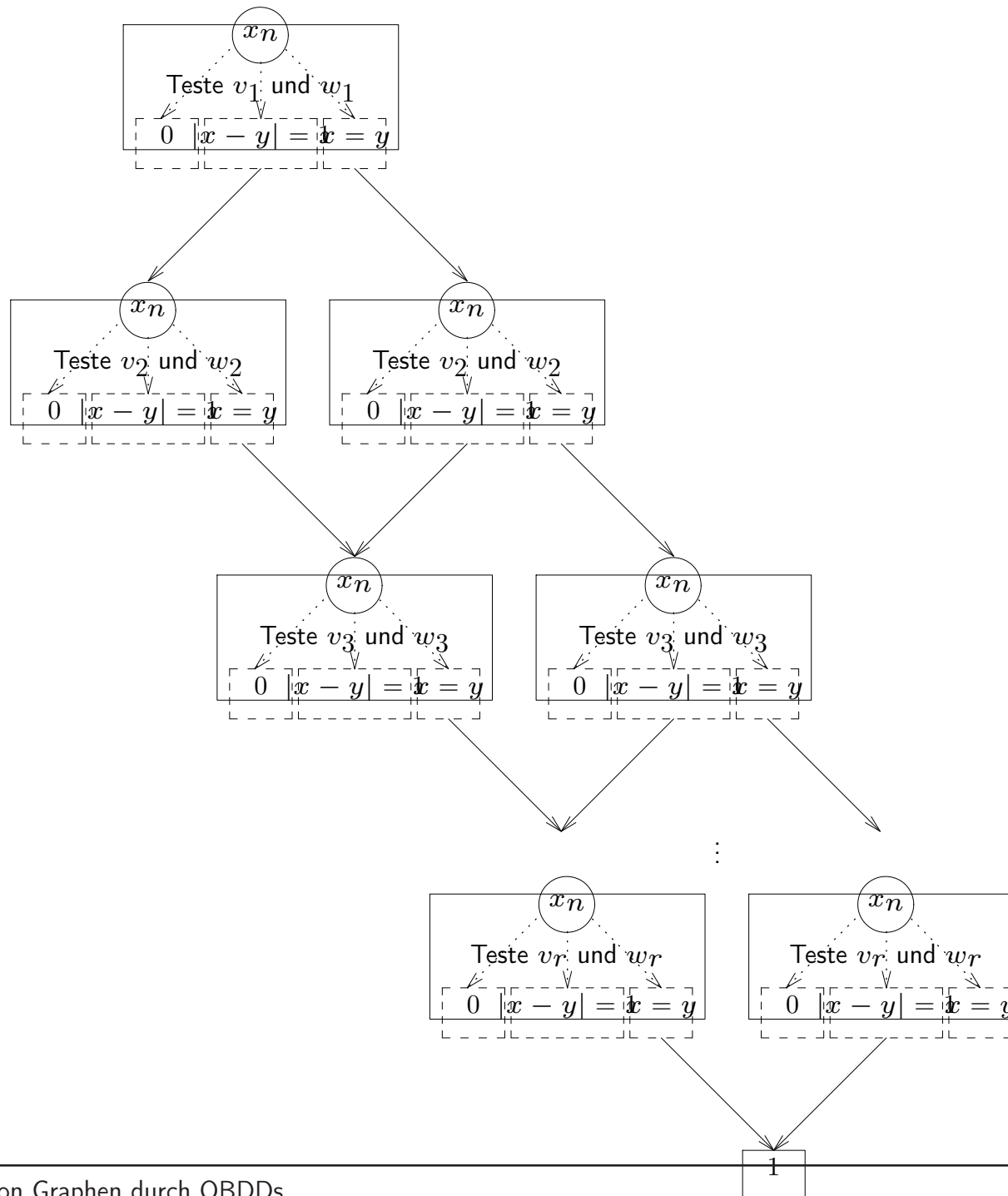
Größe: $7n - 4$



Größe: $11n - 10$



5. Eigene Ergebnisse





Gitter und Tori

Da wir für den ersten Test nur einen OBDD-Abschnitt benötigen, testen wir dort die Dimension mit der höchsten Knotenzahl. Der OBDD-Abschnitt hat also Größe $7 \max_{1 \leq i \leq r} \{\log |V_i|\} - 4$.

Für alle weiteren Tests einer Dimension i mit $1 \leq i \leq r$ benötigen wir zwei OBDD-Abschnitte der Größe $7 \log |V_i| - 4$. Dazu kommen noch zwei Senken:

$$\sum_{i=1}^r 2(7 \log |V_i| - 4) - \left(7 \max_{1 \leq i \leq r} \{\log |V_i|\} - 4\right) + 2 = 14 \sum_{i=1}^r \log |V_i| - 7 \max_{1 \leq i \leq r} \log |V_i| - 8r + 6$$

Im Verhältnis zur Knotenzahl $n = \prod_{i=1}^r |V_i|$ ist dies:

$$\begin{aligned} 14 \sum_{i=1}^r \log |V_i| - 7 \max_{1 \leq i \leq r} \log |V_i| - 8r + 6 &= 14 \log n - 7 \max_{1 \leq i \leq r} \log |V_i| - 8r + 6 \\ &< 14 \log n \end{aligned}$$



Danke für eure Aufmerksamkeit!



Literatur

- [Haj57] G. Hajós. Über eine Art von Graphen. *Internationale Mathematische Nachrichten*, 11, 1957. Problem 65.
- [Hoá85] C. Hoáng. PhD thesis, McGill University, Montreal, Quebec, 1985.
- [JO89] B. Jamison and S. Olariu. p_4 -reducible graphs - a class of uniquely tree representable graphs. *Studies in Applied Mathematics*, 81:79–87, 1989.
- [JO91] B. Jamison and S. Olariu. On a unique tree representation for p_4 -extendible graphs. *Discrete Applied Mathematics*, 34:151–164, 1991.
- [Ler71] H. Lerchs. On cliques and kernels. Technical report, Department of Computer Science, University of Toronto, 1971.



- [Ler72] H. Lerchs. On the clique-kernel structure of graphs. Technical report, Department of Computer Science, University of Toronto, 1972.
- [Rob69] F. S. Roberts. Indifference graphs. In F. Harary, editor, *Proof Techniques in Graph Theory*, pages 139–146. Academic Press, New York, 1969.
- [Saw02] D. Sawitzki. Implizite Algorithmen für Graphprobleme. Diplomarbeit, 2002.
- [Wöl03] P. Wölfel. The obdd-size of cographs. Internal Report, FB Informatik 2, Universität Dortmund, 2003.