On the OBDD Complexity of the Most Significant Bit of Integer Multiplication

Beate Bollig

LS2 Informatik, TU Dortmund
Germany

Dagstuhl seminar on
Computational Complexity of Discrete Problems

Outline

- Introduction
    - Integer multiplication
    - OBDDs

Outline

- Introduction
    - Integer multiplication
    - OBDDs
- Lower bounds on the OBDD complexity of the most significant bit of integer multiplication (B. (2008))

Outline

- Introduction
  - Integer multiplication
  - OBDDs
- Lower bounds on the OBDD complexity of the most significant bit of integer multiplication (B. (2008))
- Upper bounds on the OBDD complexity of the most significant bit of integer multiplication (B. and Klump (2008))

Integer multiplication

Integer multiplication is certainly one of the most important functions in computer science
→ a lot of effort in

- designing good algorithms and small circuits and
- determining its complexity

Integer multiplication

Integer multiplication is certainly one of the most important functions in computer science
→ a lot of effort in

- designing good algorithms and small circuits and
- determining its complexity

Definition

The Boolean function $\text{MUL}_{i,n} \in B_{2n}$ maps two n-bit integers $x = x_{n-1} \dots x_0$ and $y = y_{n-1} \dots y_0$ to the ith bit of their binary product $z_{2n-1} \dots z_0$, i.e., $\text{MUL}_{i,n}(x,y) = z_i$.

$\text{MUL}_{n-1,n}$: *middle bit of integer multiplication*
$\text{MUL}_{2n-1,n}$: *most significant bit of integer multiplication*

Integer multiplication

$\mathrm{MUL}_{2n-1,n}$ computes the most important bit of integer multiplication since

- z_{2n-1} is the most important bit for an approximation of the product $x \cdot y$
- lower bounds on the space complexity of nonuniform models for z_{2n-1} can be transferred to the computation of z_i, $0 \leq i \leq 2n-2$

Ordered Binary Decision Diagrams (OBDDs)

In many applications data structures for Boolean functions are necessary:

- circuit verification
- model checking
- even in graph algorithms ...

Ordered Binary Decision Diagrams (OBDDs)

In many applications data structures for Boolean functions are necessary:

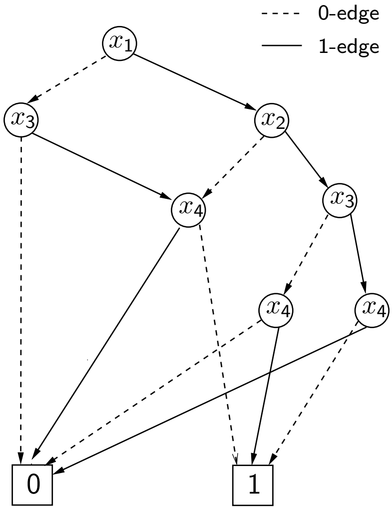
- circuit verification
- model checking
- even in graph algorithms ...

Ordered Binary Decision Diagrams (OBDDs) (Bryant (1986)):

- support all fundamental operations on Boolean functions
- are the state-of-the art data structure for Boolean functions

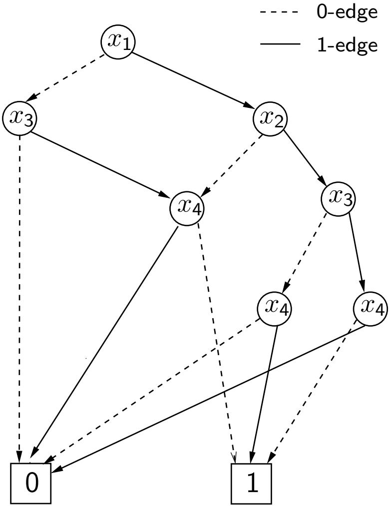
Ordered Binary Decision Diagrams (OBDDs)

- directed acyclic graph
- decision nodes:
  - marked by a variable
  - outgoing 0- and 1-edge
- one source and two sinks: 0 and 1



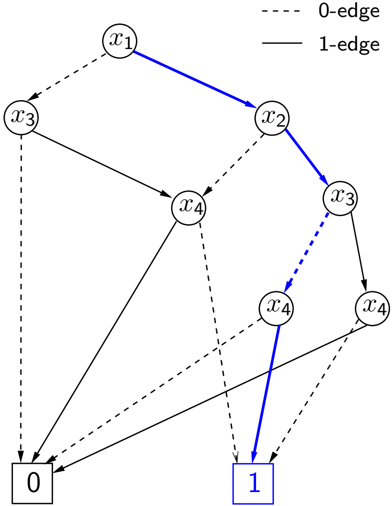
Ordered Binary Decision Diagrams (OBDDs)

- directed acyclic graph
- decision nodes:
  - marked by a variable
  - outgoing 0- and 1-edge
- one source and two sinks: 0 and 1
- variable order π on each path



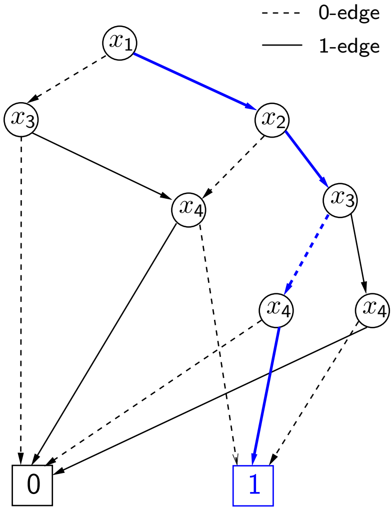
Ordered Binary Decision Diagrams (OBDDs)

- directed acyclic graph
- decision nodes:
  - marked by a variable
  - outgoing 0- and 1-edge
- one source and two sinks: 0 and 1
- variable order π on each path
- $f(b) = c \;:\Leftrightarrow$ computation path for b leads to c-sink



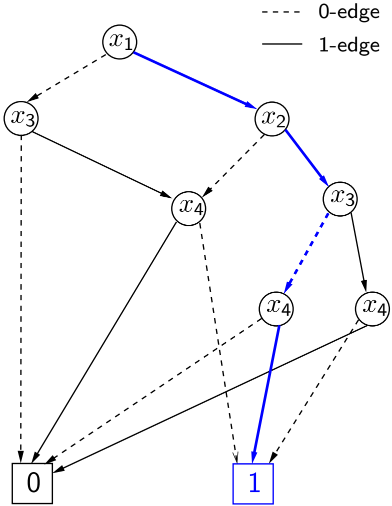
Ordered Binary Decision Diagrams (OBDDs)

- directed acyclic graph
- decision nodes:
  – marked by a variable
  – outgoing 0- and 1-edge
- one source and two sinks: 0 and 1
- variable order π on each path
- $f(b) = c \;\; :\Leftrightarrow$
  computation path for b leads to c-sink
- size of a π-OBDD: # nodes
- π-OBDD size of f: minimal size of a π-OBDD for f



Ordered Binary Decision Diagrams (OBDDs)

- directed acyclic graph
- decision nodes:
  - – marked by a variable
  - – outgoing 0- and 1-edge
- one source and two sinks: 0 and 1
- variable order π on each path
- $f(b) = c \; :\Leftrightarrow$ computation path for b leads to c-sink
- OBDD(f): minimal size of an OBDD for f



Motivation

For many functions it is not very difficult to estimate the OBDD size
But: sometimes it is not easy to prove large lower bounds for some predefined and interesting functions

Motivation

For many functions it is not very difficult to estimate the OBDD size
But: sometimes it is not easy to prove large lower bounds for some predefined and interesting functions

Despite well-known exponential lower bounds for $\text{MUL}_{n-1,n}$, until now the OBDD complexity of $\text{MUL}_{2n-1,n}$ has been unknown

Motivation

For many functions it is not very difficult to estimate the OBDD size

But: sometimes it is not easy to prove large lower bounds for some predefined and interesting functions

Despite well-known exponential lower bounds for $\mathrm{MUL}_{n-1,n}$, until now the OBDD complexity of $\mathrm{MUL}_{2n-1,n}$ has been unknown

Wegener (2000):

Is the OBDD complexity of $\mathrm{MUL}_{2n-1,n}$ exponential?

Motivation

Sawitzki (2006):

Exponential lower bounds on the space complexity of some OBDD-based algorithms for the *reachability problem* (important problem in e.g. CAD, hardware verification, model checking ...) using a lower bound of $\Omega(2^{n/6})$ on the size of π-OBDDs for $\mathrm{MUL}_{2n-1,n}$, where $\pi = (x_0, y_0, x_1, y_1, \ldots, x_{n-1}, y_{n-1})$

Motivation

Sawitzki (2006):

Exponential lower bounds on the space complexity of some OBDD-based algorithms for the *reachability problem* (important problem in e.g. CAD, hardware verification, model checking ...) using a lower bound of $\Omega(2^{n/6})$ on the size of π-OBDDs for $\mathrm{MUL}_{2n-1,n}$, where $\pi = (x_0, y_0, x_1, y_1, \ldots, x_{n-1}, y_{n-1})$

Assumption: output OBDDs use the same variable order as the input OBDDs

But: practical algorithms run reordering heuristics on intermediate OBDD results (in order to minimize their size)

Motivation

Sawitzki (2006):

Exponential lower bounds on the space complexity of some OBDD-based algorithms for the *reachability problem* (important problem in e.g. CAD, hardware verification, model checking ...) using a lower bound of $\Omega(2^{n/6})$ on the size of π-OBDDs for $\mathrm{MUL}_{2n-1,n}$, where $\pi = (x_0, y_0, x_1, y_1, \ldots, x_{n-1}, y_{n-1})$

Assumption: output OBDDs use the same variable order as the input OBDDs

But: practical algorithms run reordering heuristics on intermediate OBDD results (in order to minimize their size)

Here: OBDD($\mathrm{MUL}_{2n-1,n}$) is exponential
by-product:
simplification and improvement of Sawitzki's lower bound proof

Fooling sets and the size of OBDDs

One-way communication complexity is a standard technique to obtain lower bounds on the size of OBDDs:

Definition

Let $f \in B_n$ be a Boolean functions on the variables $X_n = X_A \cup X_B$. $S \subseteq \{0,1\}^{|X_A|} \times \{0,1\}^{|X_B|}$ is called fooling set for f if $f(a,b) = c$ for all $(a,b) \in S$ and $c \in \{0,1\}$ and for different pairs $(a_1,b_1),(a_2,b_2) \in S$ at least one of $f(a_1,b_2)$ and $f(a_2,b_1)$ is unequal to c.

Fooling sets and the size of OBDDs

One-way communication complexity is a standard technique to obtain lower bounds on the size of OBDDs:

Definition

Let $f \in B_n$ be a Boolean functions on the variables $X_n = X_A \cup X_B$. $S \subseteq \{0,1\}^{|X_A|} \times \{0,1\}^{|X_B|}$ is called fooling set for f if $f(a,b) = c$ for all $(a,b) \in S$ and $c \in \{0,1\}$ and for different pairs $(a_1, b_1), (a_2, b_2) \in S$ at least one of $f(a_1, b_2)$ and $f(a_2, b_1)$ is unequal to c.

Theorem

If $f : \{0,1\}^{|X_A|} \times \{0,1\}^{|X_B|} \rightarrow \{0,1\}$ has a fooling set of size t and π is a variable order where the X_A-variables are before the X_B-variables, the size of a π-OBDD for f is at least t.

Fooling sets and the size of OBDDs

Crucial point to prove large lower bounds on OBDD(f):
each variable order can be split into a partition of the variables such that there is a large fooling set for f

Fooling sets and the size of OBDDs

Crucial point to prove large lower bounds on OBDD(f):
each variable order can be split into a partition of the variables such that there is a large fooling set for f

Definition

The Boolean function $\overline{\mathrm{GT}}^{**}_n$: $\{0,1\}^n \times \{0,1\}^n \rightarrow \{0,1\}$ maps two n-bit integers w' and w'' to 1 iff their sum is at most $2^n - 1$.

Not difficult: $\overline{\mathrm{GT}}^{**}_n$ has a fooling set of size 2^n if X_A contains all w'-variables and X_B all w''-variables

Fooling sets and the size of OBDDs

Crucial point to prove large lower bounds on OBDD(f):
each variable order can be split into a partition of the variables such that there is a large fooling set for f

Definition

The Boolean function $\overline{\mathrm{GT}}^{**}_n$: $\{0,1\}^n \times \{0,1\}^n \rightarrow \{0,1\}$ maps two n-bit integers w' and w'' to 1 iff their sum is at most $2^n - 1$.

Not difficult: $\overline{\mathrm{GT}}^{**}_n$ has a fooling set of size 2^n if X_A contains all w'-variables and X_B all w''-variables

Here for every π:
π-OBDD for $\mathrm{MUL}_{2n-1,n}$ → π-OBDD for $\overline{\mathrm{GT}}^{**}_{n'}(w', w'')$, $n' = \Theta(n)$,
where all w'-variables are before the w''-variables in π

A general lower bound

Theorem

$\mathrm{OBDD}(\mathrm{MUL}_{2n-1,n}) = \Omega(2^{n/432})$

A general lower bound

Theorem

$\mathrm{OBDD}(\mathrm{MUL}_{2n-1,n}) = \Omega(2^{n/432})$

Sketch of proof:

Fact

- $a = 2^{n-1} + l2^{n/2}$, $0 < l \leq 2^{n/6-1}$
- $b_a := 2^n - l2^{n/2+1} + 4l^2$
  smallest integer such that $a \cdot b_a \geq 2^{2n-1}$

A general lower bound

Theorem

$\mathrm{OBDD}(\mathrm{MUL}_{2n-1,n}) = \Omega(2^{n/432})$

Sketch of proof:

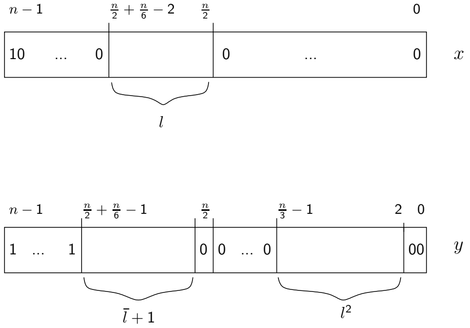
Fact

- $a = 2^{n-1} + l2^{n/2}$, $0 < l \leq 2^{n/6-1}$
- $b_a := 2^n - l2^{n/2+1} + 4l^2$
  smallest integer such that $a \cdot b_a \geq 2^{2n-1}$

Idea: replace some of the variables by constants such that $[x]_{n/2}^{n/2+n/6-2} = l$, $[y]_{n/2+1}^{n/2+n/6-1} = \bar{l} + 1$: $\rightarrow$ $([y]_2^{n/3-1} \geq l^2 \leftrightarrow x \cdot y \geq 2^{2n-1})$

- $\bar{l} := (2^{n/6-1} - 1) - l$
- $[z]_r^l$: integer whose binary representation is $(z_l, \ldots, z_r)$

Some replacements of the variables



$[x]_{n/2}^{n/2+n/6-2} = l$, $[y]_{n/2+1}^{n/2+n/6-1} = \bar{l}+1$: $\rightarrow$ $[y]_{2}^{n/3-1} \geq l^2 \leftrightarrow x \cdot y \geq 2^{2n-1}$

We make sure that $[x]_{n/2}^{n/2+n/6-2} = l \leftrightarrow [y]_{n/2+1}^{n/2+n/6-1} = \bar{l}+1$.

Closer look at l^2

Wegener (1993):

For two m-bit numbers u and w let $l := u \cdot 2^{2(m+1)} + w$. Then

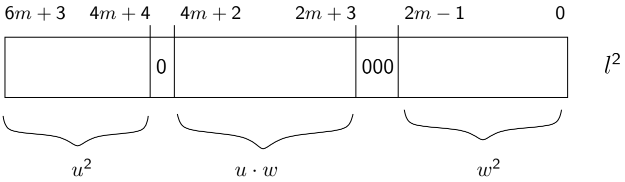
$$l^2 = u^2 \cdot 2^{4(m+1)} + uw2^{2(m+1)+1} + w^2.$$

Closer look at l^2

Wegener (1993):

For two m-bit numbers u and w let $l := u \cdot 2^{2(m+1)} + w$. Then

$$l^2 = u^2 \cdot 2^{4(m+1)} + uw2^{2(m+1)+1} + w^2.$$



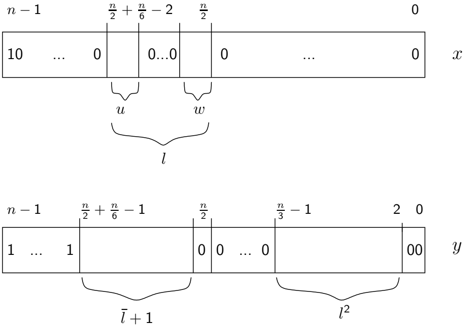
Here: $m := (n/6-3)/3$

Closer look at l^2 and further replacements

Wegener (1993):

For two m-bit numbers u and w let $l := u \cdot 2^{2(m+1)} + w$. Then

$$l^2 = u^2 \cdot 2^{4(m+1)} + uw2^{2(m+1)+1} + w^2.$$



Further replacements

Main ideas:

- choose u and w such that
  $u \cdot w \rightsquigarrow w''2^{2d+c} + (w'' + w')2^{d+c} + w'2^c$, where
    - w'' and w' are n'-bit numbers, $n' = \Theta(n)$ and $d > n'$, and
    - the w'-variables are tested before the w''-variables in π

Further replacements

Main ideas:

- choose u and w such that
  $u \cdot w \rightsquigarrow w''2^{2d+c} + (w'' + w')2^{d+c} + w'2^{c}$, where
    - w'' and w' are n'-bit numbers, $n' = \Theta(n)$ and $d > n'$, and
    - the w'-variables are tested before the w''-variables in π
- replace further y-variables (and x-variables) by constants such that
  $x \cdot y \geq 2^{2n-1} \leftrightarrow (w' + w'') \leq 2^{n'} - 1$

Further replacements

Main ideas:

- choose u and w such that
  $u \cdot w \rightsquigarrow w''2^{2d+c} + (w'' + w')2^{d+c} + w'2^c$, where
  - w'' and w' are n'-bit numbers, $n' = \Theta(n)$ and $d > n'$, and
  - the w'-variables are tested before the w''-variables in π
- replace further y-variables (and x-variables) by constants such that
  $x \cdot y \geq 2^{2n-1} \leftrightarrow (w' + w'') \leq 2^{n'} - 1$

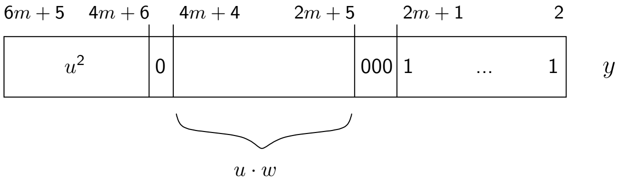
$\rightarrow$ π-OBDD for $\mathrm{MUL}_{2n-1,n}$ $\rightarrow$ π-OBDD for $\overline{\mathrm{GT}^{**}_{n'}}(w', w'')$

Further replacements

Main ideas:

- choose u and w such that
  $u \cdot w \rightsquigarrow w''2^{2d+c} + (w'' + w')2^{d+c} + w'2^{c}$, where
  - w'' and w' are n'-bit numbers, $n' = \Theta(n)$ and $d > n'$, and
  - the w'-variables are tested before the w''-variables in π
- replace further y-variables (and x-variables) by constants such that
  $x \cdot y \geq 2^{2n-1} \leftrightarrow (w' + w'') \leq 2^{n'} - 1$

$\rightarrow$ π-OBDD for $\mathrm{MUL}_{2n-1,n}$ $\rightarrow$ π-OBDD for $\overline{\mathrm{GT}^{**}_{n'}}(w', w'')$



Some remarks

- Simplification and improvement of Sawitzki's lower bound proof:

Theorem

π-OBDD($\mathsf{MUL}_{2n-1,n}$) $= \Omega(2^{n/4})$, *where*
$\pi = (x_0, y_0, x_1, y_1, \ldots, x_{n-1}, y_{n-1})$

Some remarks

- Simplification and improvement of Sawitzki's lower bound proof:

Theorem

π-OBDD($\mathrm{MUL}_{2n-1,n}$) $= \Omega(2^{n/4})$, *where*
$\pi = (x_0, y_0, x_1, y_1, \ldots, x_{n-1}, y_{n-1})$

Sketch of proof:

- $a = 2^{n-1} + l2^{n/2}$, $0 < l \leq 2^{n/4-1}$

Some remarks

- Simplification and improvement of Sawitzki's lower bound proof:

Theorem

π-OBDD($\mathrm{MUL}_{2n-1,n}$) $= \Omega(2^{n/4})$, *where*
$\pi = (x_0, y_0, x_1, y_1, \ldots, x_{n-1}, y_{n-1})$

Sketch of proof:

- $a = 2^{n-1} + l2^{n/2}$, $0 < l \leq 2^{n/4-1}$
- $b_a := 2^n - l2^{n/2+1} + 4l^2 - \lfloor \frac{4l^3}{2^{n/2-1}+l} \rfloor$
  smallest integer such that $a \cdot b_a \geq 2^{2n-1}$

Some remarks

- Simplification and improvement of Sawitzki's lower bound proof:

Theorem

π-OBDD($\text{MUL}_{2n-1,n}$) $= \Omega(2^{n/4})$, *where*
$\pi = (x_0, y_0, x_1, y_1, \dots, x_{n-1}, y_{n-1})$

Sketch of proof:

- $a = 2^{n-1} + l2^{n/2}$, $0 < l \le 2^{n/4-1}$
- $b_a := 2^n - l2^{n/2+1} + 4l^2 - \lfloor \frac{4l^3}{2^{n/2-1}+l} \rfloor$
  smallest integer such that $a \cdot b_a \ge 2^{2n-1}$
- $4l^2 - \lfloor \frac{4l^3}{2^{n/2-1}+l} \rfloor > 4(l-1)^2$

→ fooling set of size at least $2^{n/4-1}$

Some remarks

- Simplification and improvement of Sawitzki's lower bound proof:

Theorem

π-OBDD($\mathrm{MUL}_{2n-1,n}$) $= \Omega(2^{n/4})$, *where*
$\pi = (x_0, y_0, x_1, y_1, \ldots, x_{n-1}, y_{n-1})$

Sketch of proof:

- $a = 2^{n-1} + l2^{n/2}$, $0 < l \leq 2^{n/4-1}$
- $b_a := 2^n - l2^{n/2+1} + 4l^2 - \lfloor \frac{4l^3}{2^{n/2-1}+l} \rfloor$
  smallest integer such that $a \cdot b_a \geq 2^{2n-1}$
- $4l^2 - \lfloor \frac{4l^3}{2^{n/2-1}+l} \rfloor > 4(l-1)^2$

→ fooling set of size at least $2^{n/4-1}$

- OBDD($\mathrm{MUL}_{2n-1,n}$) $= \Omega(2^{n/288})$

A general upper bound

Theorem

Let π be an arbitrary variable order. The π-OBDD size of $MUL_{2n-1,n}$ is $O(2^{(4/3)n})$.

A general upper bound

Theorem

Let π be an arbitrary variable order. The π-OBDD size of $MUL_{2n-1,n}$ is $O(2^{(4/3)n})$.

Sketch of proof:

Assignments to i x- and j y-variables lead to at most $2^{2(n-i)+2(n-j)+2}$ different subfunctions of $\mathrm{MUL}_{2n-1,n}$

A general upper bound

Theorem

Let π be an arbitrary variable order. The π-OBDD size of $MUL_{2n-1,n}$ is $O(2^{(4/3)n})$.

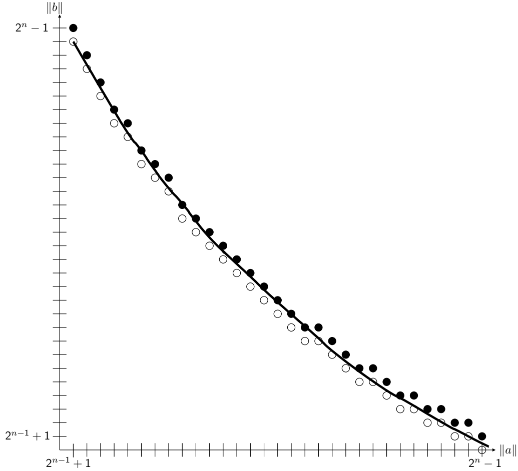
Sketch of proof:

Assignments to i x- and j y-variables lead to at most $2^{2(n-i)+2(n-j)+2}$ different subfunctions of $\mathrm{MUL}_{2n-1,n}$

- trivial upper bound of 2^{i+j}
  $\rightarrow$ upper bound of $O(2^{(4/3)n})$
- upper bound of 2^{i+7} (Amano, Maruoka (2007)) on the π-OBDD size for $\pi = (x_{n-1}, y_{n-1}, x_{n-2}, y_{n-2}, \ldots, x_0, y_0)$
  $\rightarrow$ upper bound of $O(2^{(4/5)n})$

Significant points for $\mathsf{MUL}_{2n-1,n}$

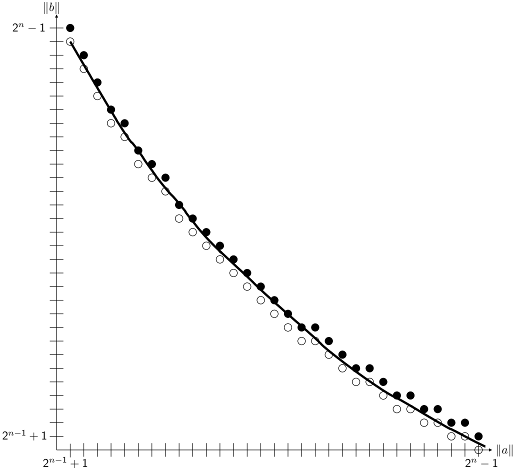
- $\| a \|$: value of the assignment a to the x-variables
- $\| b \|$: value of the assignment b to the y-variables



Significant points for $\mathrm{MUL}_{2n-1,n}$

- $\| a \|$: value of the assignment a to the x-variables
- $\| b \|$: value of the assignment b to the y-variables

Remark:
$(\| a \|, \lceil \frac{2^{2n-1}}{\|a\|} \rceil)$ significant points for $\mathrm{MUL}_{2n-1,n}$



Subfunctions of $\mathsf{MUL}_{2n-1,n}$

- $\mathsf{MUL}_{2n-1,n}$ and $f(x) := \frac{2^{2n-1}}{x}$ closely related

Subfunctions of $\mathrm{MUL}_{2n-1,n}$

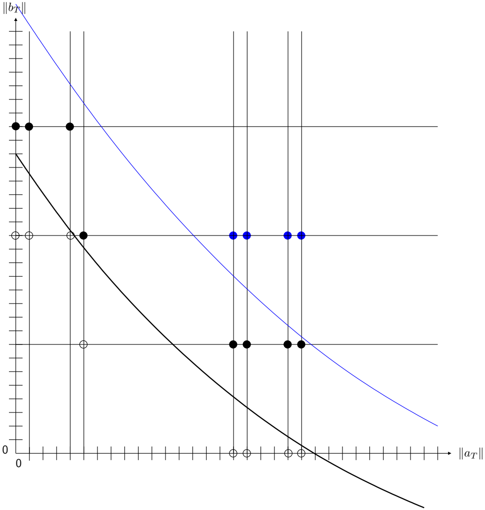
- $\mathrm{MUL}_{2n-1,n}$ and $f(x) := \frac{2^{2n-1}}{x}$ closely related
- partial assignments to x- and y-variables

  $\longrightarrow$

  $f_{c,d}(x) := \frac{2^{2n-1}}{c+x} - d$

Subfunctions of $\mathrm{MUL}_{2n-1,n}$

- $\mathrm{MUL}_{2n-1,n}$ and $f(x) := \frac{2^{2n-1}}{x}$ closely related
- partial assignments to x- and y-variables $\longrightarrow$ $f_{c,d}(x) := \frac{2^{2n-1}}{c+x} - d$
- assignments a_T and b_T to the remaining x- and y-variables $\longrightarrow$ *grid*



Representations of subfunctions of $\mathrm{MUL}_{2n-1,n}$ (1/2)

Observation

$f_{c,d}(x) := \frac{2^{2n-1}}{c+x} - d$, $x \in \mathbb{R}$, can be uniquely described by two tuples $(x', f_{c,d}(x'))$ and $(x'', f_{c,d}(x''))$, where $x', x'' \in \mathbb{R}$.

Representations of subfunctions of $\mathrm{MUL}_{2n-1,n}$ (1/2)

Observation

$f_{c,d}(x) := \frac{2^{2n-1}}{c+x} - d$, $x \in \mathbb{R}$, can be uniquely described by two tuples $(x', f_{c,d}(x'))$ and $(x'', f_{c,d}(x''))$, where $x', x'' \in \mathbb{R}$.

Problem:

representation of the tuples may be too large

Representations of subfunctions of MUL$_{2n-1,n}$ (1/2)

Observation

$f_{c,d}(x) := \frac{2^{2n-1}}{c+x} - d$, $x \in \mathbb{R}$, can be uniquely described by two tuples $(x', f_{c,d}(x'))$ and $(x'', f_{c,d}(x''))$, where $x', x'' \in \mathbb{R}$.

Problem:

representation of the tuples may be too large

Idea:

modify $f_{c,d}$ such that two tuples of assignments to the remaining x- and y-variables are (almost) sufficient in order to encode the corresponding subfunction of MUL$_{2n-1,n}$ uniquely

Representations of subfunctions of $\mathrm{MUL}_{2n-1,n}$ (2/2)

Modification of $f_{c,d}$:

- decrease parameter d until graph of the function $f_{c,d}$ hits a (significant) point of the grid for the first time $\rightarrow$ $f_{c,d'}$

Representations of subfunctions of $\mathrm{MUL}_{2n-1,n}$ (2/2)

Modification of $f_{c,d}$:

- decrease parameter d until graph of the function $f_{c,d}$ hits a (significant) point of the grid for the first time $\rightarrow f_{c,d'}$
- rotate (clockwise) the graph of the function by decreasing c and adjusting d' to hit a second point of the grid $\rightarrow f_{c'',d''}$

Representations of subfunctions of $\mathrm{MUL}_{2n-1,n}$ (2/2)

Modification of $f_{c,d}$:

- decrease parameter d until graph of the function $f_{c,d}$ hits a (significant) point of the grid for the first time $\rightarrow f_{c,d'}$
- rotate (clockwise) the graph of the function by decreasing c and adjusting d' to hit a second point of the grid $\rightarrow f_{c'',d''}$

$\rightarrow$ corresponding subfunction of $\mathrm{MUL}_{2n-1,n}$ can be uniquely reconstructed by the two points

Representations of subfunctions of $\mathrm{MUL}_{2n-1,n}$ (2/2)

Modification of $f_{c,d}$:

- decrease parameter d until graph of the function $f_{c,d}$ hits a (significant) point of the grid for the first time $\longrightarrow$ $f_{c,d'}$
- rotate (clockwise) the graph of the function by decreasing c and adjusting d' to hit a second point of the grid $\longrightarrow$ $f_{c'',d''}$

$\longrightarrow$ corresponding subfunction of $\mathrm{MUL}_{2n-1,n}$ can be uniquely reconstructed by the two points

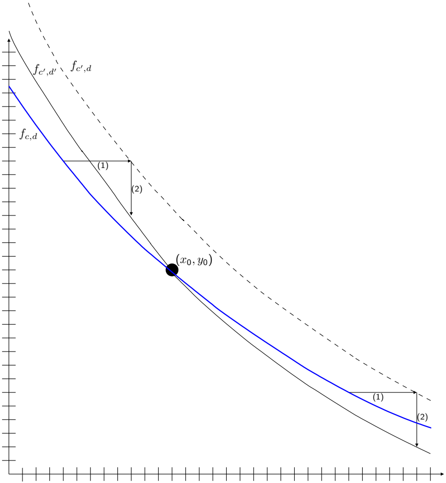
Lemma

Let $f : \mathbb{R}^+ \rightarrow \mathbb{R}^+$ *be defined as* $f(x) := \frac{2^{2n-1}}{x}$. *For arbitrary* $\Delta x, \Delta y > 0$ *there exists exactly one value* $x \in \mathbb{R}^+$ *with* $f(x) - f(x + \Delta x) = \Delta y$.

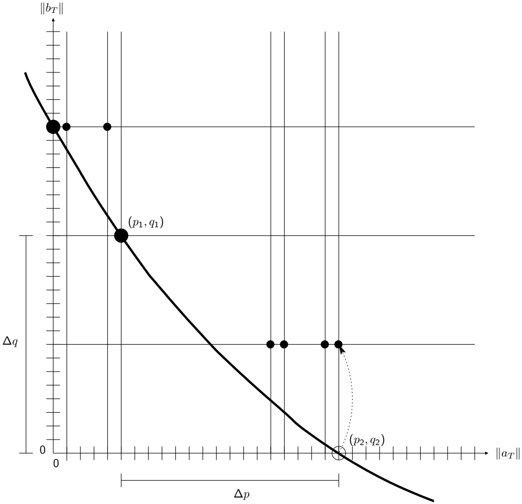
Rotation of the graph of the function

Remember:
$f_{c,d}(x) := \frac{2^{2n-1}}{c+x} - d$, where
$f : \mathbb{R} \to \mathbb{R}$

A rotation is possible by modifications of c and d



Reconstruction of a subfunction of MUL$_{2n-1,n}$



Summary

- Exponential lower bound on the OBDD complexity of $\mathrm{MUL}_{2n-1,n}$

  $\longrightarrow$

  exponential lower bounds on the space complexity of some symbolic algorithms for the reachability problem
- $\mathrm{OBDD}(\mathrm{MUL}_{2n-1,n})$?

Here: $\mathrm{OBDD}(\mathrm{MUL}_{2n-1,n}) = \Omega(2^{n/288})$
$\mathrm{OBDD}(\mathrm{MUL}_{2n-1,n}) = O(2^{(4/5)n})$

Summary

- Exponential lower bound on the OBDD complexity of $\mathrm{MUL}_{2n-1,n}$

  $\longrightarrow$

  exponential lower bounds on the space complexity of some symbolic algorithms for the reachability problem
- OBDD($\mathrm{MUL}_{2n-1,n}$)?

Here: OBDD($\mathrm{MUL}_{2n-1,n}$)$= \Omega(2^{n/288})$
OBDD($\mathrm{MUL}_{2n-1,n}$)$= O(2^{(4/5)n})$

Long live the theory of BDDs

Don Knuth