2Exp - lower bound for GF.

Challenging

To encode a binary tree of height 2^{2^n} with a formula φ_n of size $poly(n)$.

Boring

Encoding of AExpSpace Turing machines
1734 - 1735 p. Grädel
On the Restraining Power of Guards

Any <u>pair</u> (a, b) of vertices will represent a node in a tree.

There are $\underbrace{2^n \text{ different sequences of length } n \text{ on } a, b}_{\text{will serve as bits}}$.

On 2^n bits we can encode in binary numbers from 0 to $2^{2^n} - 1$.

Let $\vec{x}, \vec{x}'$ be n-tuples composed of $\{a, b\}$.

$$\{\underset{0}{\underset{\uparrow}{a}}, \underset{1}{\underset{\uparrow}{b}}\}$$

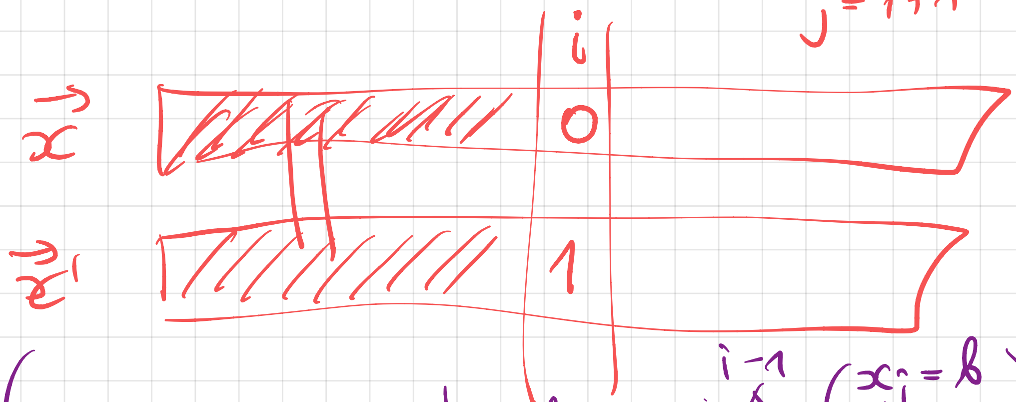
$r(\vec{x}) =$ the value encoded by $\vec{x}$

e.g. $r(abab) = 5$

$$less(\vec{x}, \vec{x}') = \bigvee_{i=1}^{n} \left(x_i = a \wedge x'_i = b \wedge \bigwedge_{j=i+1}^{n} x_i = x'_i \right)$$

"

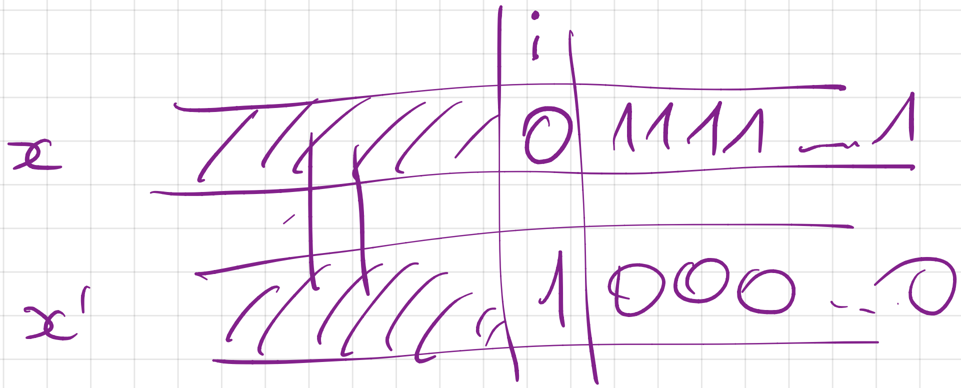
$$r(\vec{x}) < r(x')$$



$$succ(\vec{x}, \vec{x}') = \bigvee_{i=1}^{n} \left(x_i = a \wedge x'_i = b \wedge \bigwedge_{j=1}^{i-1} \left(\begin{matrix} x_j = b \\ \wedge \\ x'_j = a \end{matrix} \right) \wedge \bigwedge_{j=i+1}^{n} x_i = x'_i \right)$$

"

$$r(\vec{x}') = r(\vec{x}) + 1$$



* We will use an n-ary predicate D to "store" bits of huge numbers
" says whether a bit is on or off"

Ex $n=3$

$$\underset{0}{\overset{D}{aaa}} \le \underset{1}{\overset{\neg D}{aab}} \le \underset{2}{\overset{D}{aba}} \le \underset{3}{\overset{D}{abb}} \le \underset{4}{\overset{D}{baa}} \le \underset{5}{\overset{\neg D}{bab}} \le \underset{6}{\overset{\neg D}{bba}} \le \underset{7}{\overset{D}{bbb}}$$

$$\begin{array}{cccccccc} 128 & 64 & 32 & 16 & 8 & 4 & 2 & 1 \\ (1 & 0 & 1 & 1 & 1 & 0 & 0 & 1) \end{array} = 128+32+16+8+1$$

We will employ a $(4n+4)$-ary relation E with the following meaning:

$\vec{x}, \vec{x}'$ - n-tuples of $\{a, b\}$ $\quad \vec{y}, \vec{y}'$ - n-tuples of $\{c, d\}$

$$E(\vec{x}, \vec{x}', a, b, \vec{y}, \vec{y}', c, d) \text{ holds}$$

iff

encoded with a, b

encoded with c, d

$$r(\vec{x}) = r(\vec{y}) \quad \text{and} \quad r(\vec{x}') = r(\vec{y}')$$

$D(\vec{x})$ holds iff $\vec{x}$-th bit is true

$E(\vec{x}, \vec{x}', a, b, \vec{y}, \vec{y}', c, d)$ holds iff $r(\vec{x}) = r(\vec{y})$ and $r(\vec{x}') = r(\vec{y}')$.

Five more relations :

* T (binary) $T(a,b)$ means that a node represented by (a,b) is inside a tree.
* R (binary) $R(a,b)$ means that the node represented by (a,b) is the root of our tree
* U ($n+2$-ary) $U(\vec{x}, a, b)$ is true (at least) for all $\vec{x} \in \{a,b\}^n$ where (a,b) represents a node in our tree
* D (n-ary) encodes depth
* S_1 (4-ary) $S_1(a,b,c,d)$ means that (c,d) is a left children of (a,b)
* S_2 (4-ary) $S_2(a,b,c,d)$ means that (c,d) is a right children of (a,b).

How to define φ_n ?

$$\forall \vec{x}\ a\ b \quad U(\vec{x}, a, b) \to \bigwedge_{i=1}^{n} x_i = a \lor x_i = b$$

$$(a) \quad \underbrace{\exists a \exists b}_{\text{there is a position}} \quad \underbrace{R(a,b)}_{\text{is a root}} \land \underbrace{T(a,b)}_{\text{is inside our tree}} \land \underbrace{\left(\forall \vec{x} \quad U(\vec{x}, a, b) \to \neg D(x)\right)}_{\text{the root is on level 0, thus all D bits should be switched off.}}$$

[$D(a,b) = 0$]

$$(b) \quad \underbrace{\forall a \forall b}_{\text{for all positions}} \quad \underbrace{T(a,b)}_{\text{is inside our tree}} \to \underbrace{a \neq b}_{\text{is encoded by two diffrent elem}} \land \underbrace{U(aaa\dots a, \ a, b)}_{\text{start our generator of n-tuples of } \{a,b\}}$$

$$\forall \vec{x}\, \forall a\, \forall b \quad U(\vec{x}, a, b) \to \underbrace{\bigwedge_{i=1}^{n} U(x_1, x_2, \dots, x_{i-1}, b, x_{i+1}, \dots, x_n, a, b)}_{\text{all possible replacements are possible [generate all]}}$$

$U(\vec{x}, a, b)$ holds for any n-tuple $\vec{x}$ of a, b

(c) $\forall a \forall b \; T(a,b) \rightarrow \exists c \exists d \; S_1(a,b,c,d) \wedge \exists c \exists d \; S_2(a,b,c,d)$

probably I need to say that (a, b) is not the leaf of our tree

$$\left(\exists \vec{x} \; U(\vec{x}, a, b) \wedge \left(\bigwedge_{i=1}^{n} \begin{matrix} x_i = a \\ \vee \\ x_i = b \end{matrix} \right) \wedge \neg D(\vec{x}) \right) \rightarrow$$

(a, b) is not a leaf

(c') $\forall abcd \quad S_1(a,b,c,d) \rightarrow \neg S_2(a,b,c,d)$

$S_2(\; -\text{''}- \;) \rightarrow \neg S_1(\; -\text{''}- \;)$

keep the successors deffrent

(d) $$\bigwedge_{i=1}^{2} \forall a \forall b \forall c \forall d \quad S_i(a,b,c,d) \to$$

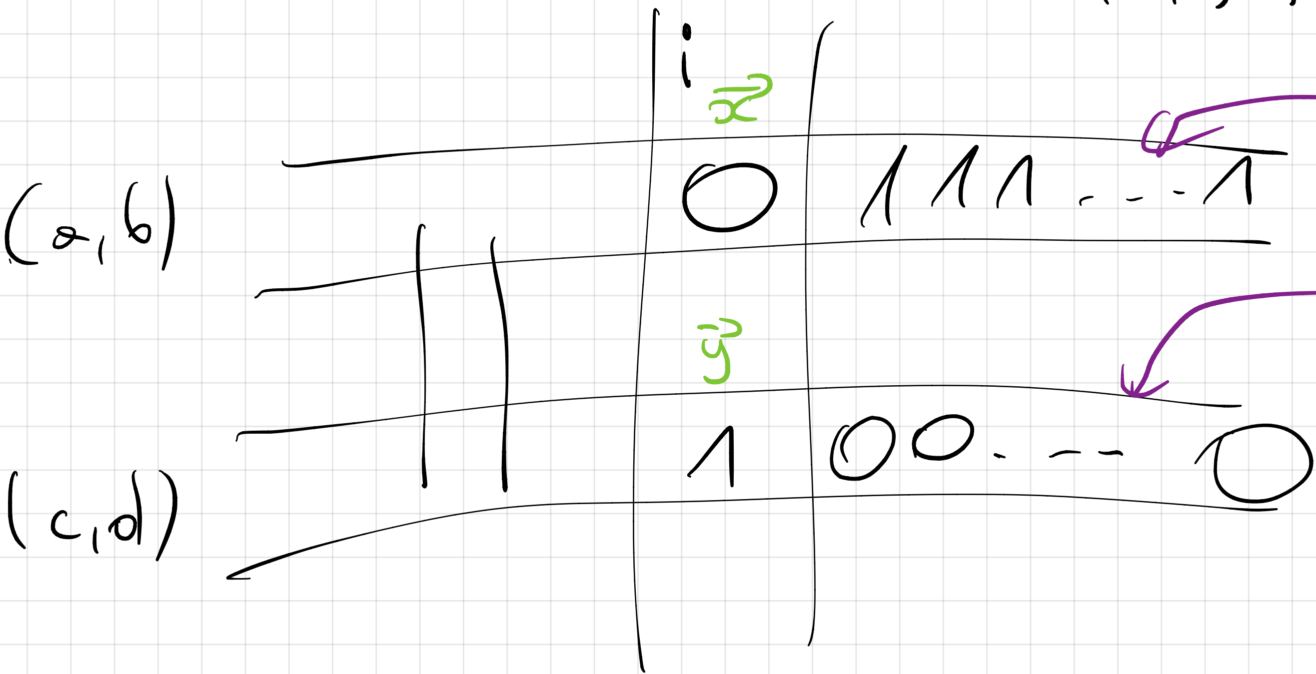
$$\underbrace{T(c,d)}_{(c,d) \text{ is inside our tree}} \wedge \underbrace{Succ(a,b,c,d)}_{depth(c,d) = depth(a,b) + 1} \wedge \underbrace{B}_{\text{a formula axiomatising } E}$$

$r(\vec{x}) = r(\vec{y})$

$$Succ(a,b,c,d) := \exists \vec{x} \exists \vec{y} \; E(\vec{x}, \vec{x}, a, b, \vec{y}, \vec{y}, c, d) \wedge \neg D(\vec{x}) \wedge D(\vec{y})$$

$$\wedge \; \forall \vec{x}' \forall \vec{y}' \; E(\vec{x}, \vec{x}', a, b, \vec{y}, \vec{y}', c, d) \to less(\vec{x}', \vec{x}) \to \begin{pmatrix} D(\vec{x}') \\ \wedge \\ \neg D(\vec{y}') \end{pmatrix}$$

$$\wedge \; \forall \vec{x}' \forall \vec{y}' \; E(\vec{x}, \vec{x}', a, b, \vec{y}, \vec{y}', c, d) \to less(\vec{x}, \vec{x}') \to (D(\vec{x}') \Leftrightarrow D(\vec{y}'))$$



(e) Axiomatise E

It remains to axiomatize E. To simplify notation, we abbreviate the concatenation $\vec{x}\vec{x}'$ by $\vec{u}$ and $\vec{y}\vec{y}'$ by $\vec{v}$, we let $\vec{a}$ and $\vec{c}$ be the tuples $(a, a, \ldots, a)$ and $(c, c, \ldots, c)$ of length $2n$ and R be the set of all substitutions $[u_i/a, v_i/c]$ or $[u_i/b, v_i/d]$ for $i = 0, \ldots, 2n-1$. Now let

$$\beta(a, b, c, d) := E\vec{a}ab\vec{c}cd \wedge (\forall \vec{u}\vec{v}\,.\, E\vec{u}ab\vec{v}cd) \bigwedge_{\rho \in R} E\vec{u}ab\vec{v}cd[\rho]$$
$$\wedge (\forall \vec{u}\vec{v}\,.\, E\vec{u}ab\vec{v}cd) \bigwedge_{i<2n} ((u_i = a \wedge v_i = c) \vee (u_i = b \wedge v_i = d)).$$

There are $4n$ substitutions in R, so the length of β is $O(n \log n)$. It should be obvious that β enforces the required properties for E.