PSPACE Complete Problems

QSAT

$QSAT$ = Set of *Quantified CNF Expressions* which evaluate to *true*

$\exists x_1 \forall x_2 \exists \ldots\ldots Q_n x_n \varphi(x_1, x_2, \ldots\ldots, x_n) \Leftrightarrow$ true?

Is it true, that we can chose a truth-value for x_1 such that for all truth-values for x_2, we can choose a......
such that $\varphi(x_1, x_2, \ldots. x_n)$ evaluates to true?

QSAT is PSPACE-complete
Membership (I)

$\exists x_1 \forall x_2 \exists \ldots\ldots Q_n x_n \varphi(x_1, x_2, \ldots\ldots, x_n) \Leftrightarrow$ true?

The idea is to implicitly traverse the tree of all possible truth assignments to $x_1, x_2 \ldots. x_n$.

However, this tree has linear depth and exponential size. Therefore, we can only store our position in the tree -- similiar to the idea of Savich's Theorem.

QSAT is PSPACE-complete
Membership (II)

$\exists x_1 \forall x_2 \exists \ldots\ldots Q_n x_n \varphi(x_1, x_2, \ldots\ldots, x_n) \Leftrightarrow$ true?

eval($x_1, x_2, \ldots x_m, Q_{m+1} x_{m+1} \ldots. Q_n x_n \varphi$)
1. if $Q_{m+1} = \exists$ then
2. return eval($x_1, x_2, \ldots x_m$, true, $Q_{m+2} x_{m+2}, \ldots. Q_n x_n \varphi$)
3. or eval($x_1, x_2, \ldots x_m$, false, $Q_{m+2} x_{m+2}, \ldots. Q_n x_n \varphi$);
4. else
5. return eval($x_1, x_2, \ldots x_m$, true, $Q_{m+2} x_{m+2}, \ldots. Q_n x_n \varphi$);
6. and eval($x_1, x_2, \ldots x_m$, false, $Q_{m+2} x_{m+2}, \ldots. Q_n x_n \varphi$);
7. endif

QSAT is PSPACE-complete
Hardness (I)

We use the Reachability Method:

$< G_x^M, s, t >$ is a *REACH* instance with $C_M^{\log n + f(n)}$ nodes.
$< G_x^M, s, t > \in REACH$ iff $M(x) = 1$

$f(n) = n^k$ give us:

$< G_x^M, s, t >$ is a *REACH* instance with $C_M^{n^k}$ nodes.
$< G_x^M, s, t > \in REACH$ iff $M(x) = 1$

QSAT is PSPACE-complete
Hardness (II)

$< G_x^M, s, t >$ is a *REACH* instance with $C_M^{n^k}$ nodes.
$< G_x^M, s, t > \in REACH$ iff $M(x) = 1$

$PATH[i](U,V) \Leftrightarrow$ there is a path in G_x^M from u to v of length $\leq 2^i$

$PATH[0](U,V) \Leftrightarrow (U = V) \vee (< U, V > \in G_x^M)$

$PATH[i+1](U,V) \Leftrightarrow \exists Z(PATH[i](U,Z) \wedge PATH[i](Z,V))$

$PATH[n^k](S,T) \Leftrightarrow G_x^M \in REACH$

QSAT is PSPACE-complete
Hardness (III)

$PATH[0](U,V) \Leftrightarrow (U = V) \vee (<U,V> \in G_x^M)$

We can express $PATH[0]$ as circuit of polynomial size
(we will get rid of the circuit later)

$PATH[i+1](U,V) \Leftrightarrow \exists Z(PATH[i](U,Z) \wedge PATH[i](Z,V))$

$PATH[i+1](U,V)$ would become exponentially large --
we have to reuse $PATH[i]$...

QSAT is PSPACE-complete
Hardness (III)

$PATH[0](U,V) \Leftrightarrow (U = V) \vee (<U,V> \in G_x^M)$

We can express $PATH[0]$ as circuit of polynomial size
(we will get rid of the circuit later)

$PATH[i+1](U,V) \Leftrightarrow \exists Z(PATH[i](U,Z) \wedge PATH[i](Z,V))$

$$PATH[i+1](U,V) \Leftrightarrow \\ \exists Z \forall X \forall Y[((X = U \wedge Y = Z) \\ \vee (X = Z \wedge Y = V)) \Rightarrow PATH[i](X,Y)]$$

QSAT is PSPACE-complete
Hardness (IV)

$$PATH[i+1](U,V) \Leftrightarrow \\ \exists Z \forall X \forall Y[((X = U \wedge Y = Z) \\ \vee (X = Z \wedge Y = V)) \Rightarrow PATH[i](X,Y)]$$

We can bring the quantifiers to the front (used only locally)

$PATH[n^k](U,V) \Leftrightarrow QX_1QX_2...QX_m\ C(U,V,X_1,X_2,...X_m)$

We have to represent the above expression as
a polynomially sized quantified CNF-expression.

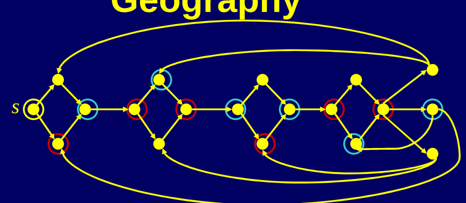
$PATH[n^k](U,V) \Leftrightarrow QX_1QX_2...QX_m \exists H\ \varphi(U,V,X_1,X_2,...X_m,H)$

QSAT is PSPACE-complete
Hardness (V)

$PATH[n^k](U,V)$ is expressible as polynomially sized
quantified CNF-expression in prenex form.

Thus, $QSAT$ is $PSPACE-hard$, using $PATH[n^k](S,T)$
as result of the reduction. •

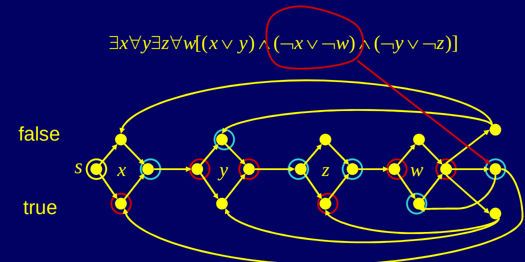
Geography



$Geography$ = Set of $<G,s>$ with $G=<V,E>$ directed and $s \in V$
where player A has a winning strategy,
i.e., can force player B into a dead end.

Geography

$\exists x \forall y \exists z \forall w[(x \vee y) \wedge (\neg x \vee \neg w) \wedge (\neg y \vee \neg z)]$



Proving Hardness & Completeness

Summary

REACH	NL-complete	Reachability Method
CIREVAL	P-complete “Inherently Sequential”	Time-Table Method
CIRSAT	NP-complete “Guess and Check -- Optimization”	Time-Table Method
QSAT	PSPACE-complete “Games – Optimal Strategies”	Reachability Method