Hard Cases for SAT Solvers

William Gasarch-U of MD
Example

The AND of the following:

1. $x_{11} \lor x_{12}$
2. $x_{21} \lor x_{22}$
3. $x_{31} \lor x_{32}$
4. $\neg x_{11} \lor \neg x_{21}$
5. $\neg x_{11} \lor \neg x_{31}$
6. $\neg x_{21} \lor \neg x_{31}$
7. $\neg x_{12} \lor \neg x_{22}$
8. $\neg x_{12} \lor \neg x_{32}$
9. $\neg x_{22} \lor \neg x_{32}$

This is Pigeonhole Principle: $x_{ij}$ is putting $i$th pigeon in $j$th hole! Can't put 3 pigeons into 2 holes! So Fml is NOT satisfiable.
Example

The AND of the following:

1. \( x_{11} \lor x_{12} \)
2. \( x_{21} \lor x_{22} \)
3. \( x_{31} \lor x_{32} \)
4. \( \neg x_{11} \lor \neg x_{21} \)
5. \( \neg x_{11} \lor \neg x_{31} \)
6. \( \neg x_{21} \lor \neg x_{31} \)
7. \( \neg x_{12} \lor \neg x_{22} \)
8. \( \neg x_{12} \lor \neg x_{32} \)
9. \( \neg x_{22} \lor \neg x_{32} \)

This is Pigeonhole Principle: \( x_{ij} \) is putting \( i \)th pigeon in \( j \) hole!
Example

The AND of the following:

1. \( x_{11} \lor x_{12} \)
2. \( x_{21} \lor x_{22} \)
3. \( x_{31} \lor x_{32} \)
4. \( \neg x_{11} \lor \neg x_{21} \)
5. \( \neg x_{11} \lor \neg x_{31} \)
6. \( \neg x_{21} \lor \neg x_{31} \)
7. \( \neg x_{12} \lor \neg x_{22} \)
8. \( \neg x_{12} \lor \neg x_{32} \)
9. \( \neg x_{22} \lor \neg x_{32} \)

This is Pigeonhole Principle: \( x_{ij} \) is putting \( i \)th pigeon in \( j \) hole!
Can’t put 3 pigeons into 2 holes! So Fml is NOT satisfiable.
PHP: Pigeon Hole Principle

Let $n < m$. $n$ is NUMBER OF HOLES, $m$ is NUMBER OF PIGEONS. $x_{ij}$ will be thought of as Pigeon $i$ IS in Hole $j$.

Definition

$PHP^m_n$ is the AND of the following:

1. For $1 \leq i \leq m$

   $x_{i1} \lor x_{i2} \lor \cdots \lor x_{in}$

   (Pigeon $i$ is in SOME Hole.)

2. For $1 \leq i_1 < i_2 \leq m$ and $1 \leq j \leq n$

   $\neg x_{i_1 j} \lor \neg x_{i_2 j}$

   (Hole $j$ does not have BOTH Pigeon $i_1$ and Pigeon $i_2$.)

NOTE: $PHP^m_n$ has $nm$ VARs and $O(mn^2)$ CLAUSES and is NOT satisfiable.
What is Known

1. If $n < m$ then $\text{PHP}_n^m$ is not satisfiable.
2. The proof of this is by the Pigeon hole principle and not by Truth Table, it was by mathematical reasoning.
3. There is a proof technique called Resolution that is used to show formulas are not satisfiable. It is known that resolution proofs that $\text{PHP}_n^m$ is not satisfiable are large.
4. Our speculation is that the SAT Solvers we have been studying will take a long time on $\text{PHP}_n^m$.
5. Try our out SAT solvers on $\text{PHP}_n^{n+1}, \text{PHP}_n^{n+2}, \ldots$ and see if it takes a long time. See what happens as the $m$ gets bigger.