A PROOFS OF THEOREMS 1 AND 2

Theorem 1. The reduction from SSAT to POMDP guarantees that there exists a POMDP policy $\pi$ for time steps 0 to $|X|/2 - 1$ and optimal action at time step $|X|/2$ with value function $V^\pi = \Pr(\phi)$ iff there exists a policy tree $\phi$ with satisfiability probability $\Pr(\phi)$.

Proof. Consider a POMDP policy $\pi$ (for time steps 0 to $|X|/2 - 1$), which defines a policy tree $\phi$. Each branch yields a final (unnormalized) belief with mass

$$\hat{b}_{\phi_1(\pi)}(s) = b_0(\pi) \Pr(\pi_{1:|X|/2}|\text{prob}, \pi)$$

Based on the properties of the reward function, the optimal expected reward of each branch at the last time step $|X|/2$ is

$$R(\hat{b}_{\phi_1(\pi)}(s)) = \max_a \sum_s \hat{b}_{\phi_1(\pi)}(s) R(s, a)$$

Hence the value of a policy is

$$V^\pi = \sum_{\phi_{1:|X|/2}} R(\hat{b}_{\phi_1(\pi)}(s))$$

The above equation shows that the value of a policy is equal to the probability of satisfying the Boolean formula with the corresponding policy tree $\phi$.

Theorem 2. In the reduction of POMDP to SSAT, there exists a satisfiable policy tree, $\phi$, with probability $\Pr(\phi)$ iff there exists a POMDP policy, $\pi$, with value function $V^\pi = \Pr(\phi)$.

Proof. Consider a base case policy tree of size 1. Let the policy tree be $\phi = \{x_a \equiv k\}$ with clauses:

$$\bigwedge_{i \in S} x_s \equiv i \lor x_r \equiv k | S | + i$$

The probability of satisfiability of (7) is equivalent to

$$\Pr(\phi) = \sum_i \Pr(x_s \equiv i) \Pr(x_r \equiv k | S | + i)$$

by using the distributions for the randomized variables: $\Pr(x_s \equiv i) = b(i)$ and $\Pr(x_r \equiv k | S | + i) = r(i, k), \forall i, k$.

For the general case, we give a proof by induction. Assume we have a policy tree $\phi_i$, policy $\pi_{i+1}$, and we know $\Pr(\phi_{i+1}) = V^\pi_{i+1}$. Given $\phi_{i+1}$ and $\pi_{i+1}$ show that $\Pr(\phi_{i+1} = V^\pi_{i+1})$.

Since we are given the policy tree, all the actions are known. Therefore, if we simplify first by making the assignments in $\phi_{i+1}$, then only the randomized variables will remain in the quantifier prefix. Any subset of variables can now be re-ordered freely. Based on the number of randomized variables we introduced for horizon $h$ and $h + 1$, encoding the probability of satisfiability is:

$$\Pr(\phi_{i+1}) = \prod_{x_p \in dom} \Pr(x_p | x_p, x_o, x_s)$$

To achieve Eq. 10, the distribution for $x_p$ is just a uniform distribution that can be factored out as $2^{-h}$. However, each $x_p$ is controlling the length of the process, so it naturally controls how many terms contribute to the total sum if we re-arrange by horizon and then simplify. Note that given values for $x_p$, $x_o$, $x_s$ the other variables are forced by unit propagation to a specific value.

$$= 2^{-(h+1)} \sum_{h=0}^{h-1} \sum_{x_{h-1} \cdots x_2 x_1} \prod_{i=1}^{h} \Pr(x_p = i, x_o = z_i, x_r)$$

Similarly, for the distribution $x_o$ the constant, $|O|^{h-1}$, can be factored out in front and its value is used in the conditional distribution $x_O$.

$$= 2^{-(h+1)} \sum_{h=0}^{h-1} \sum_{x_{h-1} \cdots x_2 x_1} \prod_{i=1}^{h} \Pr(x_p = i, x_o = z_i, x_r)$$

the next variable $x_p$ has uniform distribution for all $l > 1$ and the initial belief when $l = 1$. Therefore, we can simplify the equation by pulling out the constant factors again.

$$= 2^{-(h+1)} \sum_{h=0}^{h-1} \sum_{x_{h-1} \cdots x_2 x_1} \prod_{i=1}^{h} \Pr(x_p = i, x_o = z_i, x_r)$$

(12)
According to the distribution \( x_{p_t} \), rewards \( x_r \) will only be given at the end of the process for each \( h \).

\[
= 2^{-(h+1)}(|O| - |S|)^{-h} \sum_{k=1}^{h+1} \sum_{z_1}^{[O]} \sum_{z_{k-1}}^{[S]} \sum_{z_h}^{[S]} \Pr(x_{s_h}^z) = i \Pr(x_{s_h}^z) 
\]

\[
\prod_{i=1}^{h} \Pr(x_{s_i}^z, x_{r_i}^z) = v_z, \quad x_{s_h}^z = i, \quad x_{s_0} = z_1 
\]

(13)

If we replace the distributions below with their definitions and replace constants with the proportional relation, we obtain

\[
\approx \sum_{h=1}^{h+1} \sum_{i=1}^{[O]} \sum_{s_{h-1}}^{[S]} b(s_1) \prod_{i=1}^{h-1} \Omega_{s_{i+1}}^{a_i} a_{s_{i+1}} r(s_{i+1}) \left( r(s_1, a_1) + \sum_{z_1}^{[O]} \sum_{z_{h-1}}^{[S]} \Omega_{z_{h-1}}^{a_{h-1}} T^{a_{h-1}}(s_{h-1}, z_{h-1}) \right) 
\]

(14)

\[
= \sum_{s_1}^{[S]} b(s_1) \left( r(s_1, a_1) + \sum_{z_1}^{[O]} \sum_{z_{h-1}}^{[S]} \Omega_{z_{h-1}}^{a_{h-1}} T^{a_{h-1}}(s_{h-1}, z_{h-1}) \right) 
\]

(15)

where \( \Pr(\phi_h) = r(s, a) + \sum_{z} \sum_{z'} \Omega_{z}^{a} r^{a_{z}, z_{h-1}} \Pr(\phi_{h-1}) \)

Now consider the reverse. Given a policy, \( \pi_{h+1} \), with value function \( V_{\pi^{h+1}} \) there exists a satisfiable policy tree, \( \phi_{h+1} \), with satisfiability probability \( \Pr(\phi_{h+1}) \) such that \( V_{\pi^{h+1}} = \Pr(\phi_{h+1}) \). First, Bellman’s equation for a \( h + 1 \) horizon policy is:

\[
V_{\pi^{h+1}} = \sum_{s} \delta_{h+1}(s) \left( r(s, a) + \sum_{\alpha} \sum_{z_{h+1}} \Omega_{z_{h+1}}^{a_{h+1}} T^{a_{h+1}}(s_{h+1}, z_{h+1}) V_{\pi^{h}}(z_{h+1}) \right), \quad a = \pi(h) 
\]

(16)

However, any \( h + 1 \) horizon policy can be written as a linear combination of \( h \) horizon policies. Since we know \( \Pr(\phi_h) = V_{\pi}^h \) by the inductive step, we conclude, that (15) and (16) are equal. Therefore, the probability of satisfying a \( h + 1 \) depth policy tree corresponds to the value function of a \( h + 1 \) step policy.

\[\square\]

## B PROBLEM STATISTICS

We test the improvements to the watch literal rule on a variety of problems from 3 different benchmark types as shown in Table 1. The POMDP problems are from Cassandra’s repository \([?]\) and consist of two easy and two hard problems that have quite a large number of literals per clause and variable cardinality. The inference problems are from a prior probabilistic inference competition \([?]\) and tend to be highly structured and contain a large number of variables and clauses.

Finally, the random benchmarks consist of a series of variables with alternating quantifiers in 3-SAT and 10-SAT forms that were generated by a procedure. Assume we are given \( V \) the number of variables, \( C \) the number of clauses, \( k \) the number of literals in a clause, \( t \) the number

\[
\begin{array}{|c|c|c|c|c|}
\hline
\text{Benchmark} & \text{Problem} & \text{Var} & \text{Clause} & \text{avg #value} & \text{avg #literal} \\
\hline
\text{RANDOM} & \text{fail-learn1} & 50 & 120 & 2.00 & 3.00 \\
& \text{big1} & 30 & 450 & 2.00 & 10.00 \\
& \text{big2} & 15 & 60 & 4.00 & 10.00 \\
\hline
\text{POMDP} & \text{tiger.95} & 157 & 304 & 2.31 & 3.60 \\
& \text{H10} & 121 & 212 & 2.16 & 4.38 \\
& \text{query.b4} & 657 & 27,868 & 42.68 & 160.40 \\
& \text{aloha.10.H3} & 1,094 & 18,637 & 17.14 & 64.39 \\
\hline
\text{INFEINCE} & \text{ej7} & 6.359 & 14,678 & 2.00 & 2.90 \\
& \text{tiger.95} & 327,787 & 803,000 & 2.00 & 2.74 \\
\hline
\end{array}
\]

Table 1: Basic information for each benchmark problem.