Homework 22

Joe Baker, Brett Schreiber, Brian Knotten

March 15, 2018

36

In an MAM protocol, Arthur makes the wrong decision on the first round with probablility $\frac{1}{3}$. But this probability can be as low as $\frac{1}{4^m}$, where m is the number of bits that Merlin sends over. For proving some strings, Merlin may only need to send over 1 bit of information (like when proving an instance of GRAPH-NON-ISO). But Merlin can send m bits instead as a proof against m different tests. (In GRAPH-NON-ISO, Arthur sends over m relabeled graphs, each randomly corresponding to either G_0 or G_1, to which Merlin responds with m bits where the ith bit of m corresponds to the ith graph that Arthur sent over). So Arthur has up to m proofs to check against, giving him m opportunities to reject, whereas before he may have only had just one. This lowers the probability of a wrong decision in GRAPH-NON-ISO from $\frac{1}{2}$ to $\frac{1}{2^m}$.

37

Asking whether

$$\exists_x \forall_y \exists_z (x \vee y \vee \neg z) \wedge (\neg x \vee \neg y \vee z)$$

is satisfiable is equivalent to asking if:

$$\Sigma_{x=0}^{1} \Pi_{y=0}^{1} \Sigma_{z=0}^{1} (1-(1-x)(1-y)z)(1-xy(1-z) \neq 0$$

a

i

Merlin will construct a function $s(x)$ as follows:

$$\begin{aligned}
s(x) &= \Pi_{y=0}^{1} \Sigma_{z=0}^{1} (1-(1-x)(1-y)z)(1-xy(1-z)) \\
&= \Pi_{y=0}^{1} \Sigma_{z=0}^{1} 1 - z(1-x)(1-y) - xy(1-z) + xyz(1-x)(1-y)(1-z) \\
&= \Pi_{y=0}^{1} \Sigma_{z=0}^{1} 1 - (z-xz)(1-y) - xy + xyz + (xyz - x^2yz)(1-y)(1-z) \\
&= \Pi_{y=0}^{1} \Sigma_{z=0}^{1} 1 - (z-xz)(1-y) - xy + xyz + (xyz - x^2yz)(1-y)(1-z) \\
&= \Pi_{y=0}^{1} \Sigma_{z=0}^{1} 1 - (z-xz) - y(z-zx) - xy + xyz + (xyz - x^2yz)(1-y)(1-z) \\
&= \Pi_{y=0}^{1} \Sigma_{z=0}^{1} 1 - z - xz - yz + xyz - xy + xyz + xyz - x^2yz - y(xyz - x^2yz))(1-z) \\
&= \Pi_{y=0}^{1} \Sigma_{z=0}^{1} 1 - z - xz - yz + 2xyz - xy - x^2yz - (xy^2z + x^2y^2z)(1-z) \\
&= \Pi_{y=0}^{1} \Sigma_{z=0}^{1} 1 - z - xz - yz + 2xyz - xy - x^2yz - (xy^2z + x^2y^2z - z(xy^2z + x^2y^2z)) \\
&= \Pi_{y=0}^{1} \Sigma_{z=0}^{1} 1 - z - xz - yz + 2xyz - xy - x^2yz - (xy^2z + x^2y^2z - xy^2z^2 - x^2y^2z^2) \\
&= \Pi_{y=0}^{1} \Sigma_{z=0}^{1} 1 - z - xz - yz + 2xyz - xy - x^2yz - xy^2z - x^2y^2z + xy^2z^2 + x^2y^2z^2 \\
&= \Pi_{y=0}^{1} \Sigma_{z=0}^{1} 1 - z - xy - xz - yz + 2xyz - x^2yz - xy^2z - x^2y^2z + xy^2z^2 + x^2y^2z^2
\end{aligned}$$

b

i

Merlin will construct a linearized function $s(x)$ as follows:

$$\begin{aligned}
s(x) &= \Pi_{y=0}^{1}\Sigma_{z=0}^{1}(1-(1-x)(1-y)z)(1-xy(1-z)) \\
&= \Pi_{y=0}^{1}\Sigma_{z=0}^{1}1-z(1-x)(1-y)-xy(1-z)+xyz(1-x)(1-y)(1-z) \\
&= \Pi_{y=0}^{1}\Sigma_{z=0}^{1}1-(z-xz)(1-y)-xy+xyz+(xyz-xyz)(1-y)(1-z) \\
&= \Pi_{y=0}^{1}\Sigma_{z=0}^{1}1-(z-xz)(1-y)-xy+xyz+(0)(1-y)(1-z) \\
&= \Pi_{y=0}^{1}\Sigma_{z=0}^{1}1-(z-xz)(1-y)-xy+xyz \\
&= \Pi_{y=0}^{1}\Sigma_{z=0}^{1}1-(z-xz)-y(z-zx)-xy+xyz \\
&= \Pi_{y=0}^{1}\Sigma_{z=0}^{1}1-z+xz-yz+xyz-xy+xyz \\
&= \Pi_{y=0}^{1}\Sigma_{z=0}^{1}1-z+xz-yz-xy+2xyz
\end{aligned}$$