case splits needed). Specifically, observe a ReLU pair $x^f = \text{ReLU}(x^b)$ for which we have $l(x^b) \geq -\epsilon$ for a very small positive ϵ. We can under-approximate this range and instead set $l(x^b) = 0$; and, as previously discussed, we can then fix the ReLU pair to the active state. Symmetrical measures can be employed when learning a very small upper bound for x^f, in this case leading to the ReLU pair being fixed in the inactive state.

Any feasible solution that is found using this kind of under-approximation will be a feasible solution for the original problem. However, if we determine that the under-approximated problem is infeasible, the original may yet be feasible.

V Encoding ReLUs for SMT and LP Solvers

We demonstrate the encoding of ReLU nodes that we used for the evaluation conducted using SMT and LP solvers. Let $y = \text{ReLU}(x)$. In the SMTLIB format, used by all SMT solvers that we tested, ReLUs were encoded using an if-then-else construct:

```
(assert (= y (ite (>= x 0) x 0)))
```

In LP format this was encoded using mixed integer programming. Using Gurobi's built-in Boolean type, we defined for every ReLU connection a pair of Boolean variables, b_{on} and b_{off}, and used them to encode the two possible states of the connection. Taking M to be a very large positive constant, we used the following assertions:

$$
\begin{aligned}
&\mathtt{b_{on}} + \mathtt{b_{off}} = 1\\
&\mathtt{y} \geq 0\\
&\mathtt{x} - \mathtt{y} - \mathtt{M}*\mathtt{b_{off}} \leq 0\\
&\mathtt{x} - \mathtt{y} + \mathtt{M}*\mathtt{b_{off}} \geq 0\\
&\mathtt{y} - \mathtt{M}*\mathtt{b_{on}} \leq 0\\
&\mathtt{x} - \mathtt{M}*\mathtt{b_{on}} \leq 0
\end{aligned}
$$

When $\text{b}_{\text{on}}= 1$ and $\text{b}_{\text{off}}= 0$, the ReLU connection is in the active state; and otherwise, when $\text{b}_{\text{on}}= 0$ and $\text{b}_{\text{off}}= 1$, it is in the inactive state.

In the active case, because $\text{b}_{\text{off}} = 0$ the third and fourth equations imply that $x = y$ (observe that y is always non-negative). M is very large, and can be regarded as ∞; hence, because $\text{b}_{\text{on}}= 1$, the last two equations merely imply that $x, y \leq \infty$, and so pose no restriction on the solution.

In the inactive case, $\text{b}_{\text{on}} = 0$, and so the last two equations force $y = 0$ and $x \leq 0$. In this case $\text{b}_{\text{off}}= 1$ and so the third and fourth equations pose no restriction on the solution.

VI Formal Definitions for Properties $\phi_1,\ldots,\phi_{10}$

The units for the ACAS Xu DNNs' inputs are:

- ρ: feet.
- θ, ψ: radians.
- $v_{\text{own}}, v_{\text{int}}$: feet per second.
- τ: seconds.

θ and ψ are measured counter clockwise, and are always in the range $[-\pi, \pi]$.

In line with the discussion in Section 5, the family of 45 ACAS Xu DNNs are indexed according to the previous action a_{prev} and time until loss of vertical separation τ. The possible values are for these two indices are:

1. a_{prev}: [Clear-of-Conflict, weak left, weak right, strong left, strong right].
2. τ: $[0, 1, 5, 10, 20, 40, 60, 80, 100]$.

We use $N_{x,y}$ to denote the network trained for the x-th value of a_{prev} and y-th value of τ. For example, $N_{2,3}$ is the network trained for the case where $a_{\text{prev}} =$ weak left and $\tau = 5$. Using this notation, we now give the formal definition of each of the properties $\phi_1, \ldots, \phi_{10}$ that we tested.

Property ϕ_1.

- Description: If the intruder is distant and is significantly slower than the ownship, the score of a COC advisory will always be below a certain fixed threshold.
- Tested on: all 45 networks.
- Input constraints: $\rho \geq 55947.691$, $v_{\text{own}} \geq 1145$, $v_{\text{int}} \leq 60$.
- Desired output property: the score for COC is at most 1500.

Property ϕ_2.

- Description: If the intruder is distant and is significantly slower than the ownship, the score of a COC advisory will never be maximal.
- Tested on: $N_{x,y}$ for all $x \geq 2$ and for all y.
- Input constraints: $\rho \geq 55947.691$, $v_{\text{own}} \geq 1145$, $v_{\text{int}} \leq 60$.
- Desired output property: the score for COC is not the maximal score.

Property ϕ_3.

- Description: If the intruder is directly ahead and is moving towards the ownship, the score for COC will not be minimal.
- Tested on: all networks except $N_{1,7}$, $N_{1,8}$, and $N_{1,9}$.
- Input constraints: $1500 \leq \rho \leq 1800$, $-0.06 \leq \theta \leq 0.06$, $\psi \geq 3.10$, $v_{\text{own}} \geq 980$, $v_{\text{int}} \geq 960$.
- Desired output property: the score for COC is not the minimal score.

Property ϕ_4.

- Description: If the intruder is directly ahead and is moving away from the ownship but at a lower speed than that of the ownship, the score for COC will not be minimal.
- Tested on: all networks except $N_{1,7}$, $N_{1,8}$, and $N_{1,9}$.
- Input constraints: $1500 \leq \rho \leq 1800$, $-0.06 \leq \theta \leq 0.06$, $\psi = 0$, $v_{\text{own}} \geq 1000$, $700 \leq v_{\text{int}} \leq 800$.
- Desired output property: the score for COC is not the minimal score.

Property ϕ_5.

- Description: If the intruder is near and approaching from the left, the network advises "strong right".
- Tested on: $N_{1,1}$.
- Input constraints: $250 \leq \rho \leq 400$, $0.2 \leq \theta \leq 0.4$, $-3.141592 \leq \psi \leq -3.141592 + 0.005$, $100 \leq v_{\text{own}} \leq 400$, $0 \leq v_{\text{int}} \leq 400$.
- Desired output property: the score for "strong right" is the minimal score.

Property ϕ_6.

- Description: If the intruder is sufficiently far away, the network advises COC.
- Tested on: $N_{1,1}$.
- Input constraints: $12000 \leq \rho \leq 62000$, $(0.7 \leq \theta \leq 3.141592) \vee (-3.141592 \leq \theta \leq -0.7)$, $-3.141592 \leq \psi \leq -3.141592 + 0.005$, $100 \leq v_{\text{own}} \leq 1200$, $0 \leq v_{\text{int}} \leq 1200$.
- Desired output property: the score for COC is the minimal score.

Property ϕ_7.

- Description: If vertical separation is large, the network will never advise a strong turn.
- Tested on: $N_{1,9}$.
- Input constraints: $0 \leq \rho \leq 60760$, $-3.141592 \leq \theta \leq 3.141592$, $-3.141592 \leq \psi \leq 3.141592$, $100 \leq v_{\text{own}} \leq 1200$, $0 \leq v_{\text{int}} \leq 1200$.
- Desired output property: the scores for "strong right" and "strong left" are never the minimal scores.

Property ϕ_8.

- Description: For a large vertical separation and a previous "weak left" advisory, the network will either output COC or continue advising "weak left".
- Tested on: $N_{2,9}$.
- Input constraints: $0 \leq \rho \leq 60760$, $-3.141592 \leq \theta \leq -0.75 \cdot 3.141592$, $-0.1 \leq \psi \leq 0.1$, $600 \leq v_{\text{own}} \leq 1200$, $600 \leq v_{\text{int}} \leq 1200$.
- Desired output property: the score for "weak left" is minimal or the score for COC is minimal.

Property ϕ_9.

- Description: Even if the previous advisory was "weak right", the presence of a nearby intruder will cause the network to output a "strong left" advisory instead.
- Tested on: $N_{3,3}$.
- Input constraints: $2000 \leq \rho \leq 7000$, $-0.4 \leq \theta \leq -0.14$, $-3.141592 \leq \psi \leq -3.141592 + 0.01$, $100 \leq v_{\text{own}} \leq 150$, $0 \leq v_{\text{int}} \leq 150$.
- Desired output property: the score for "strong left" is minimal.

Property ϕ_{10}.

- Description: For a far away intruder, the network advises COC.
- Tested on: $N_{4,5}$.
- Input constraints: $36000 \leq \rho \leq 60760$, $0.7 \leq \theta \leq 3.141592$, $-3.141592 \leq \psi \leq -3.141592 + 0.01$, $900 \leq v_{\text{own}} \leq 1200$, $600 \leq v_{\text{int}} \leq 1200$.
- Desired output property: the score for COC is minimal.