Wireless Protocol Validation Under Uncertainty

Jinghao Shi, Shuvendu K. Lahiri*, Ranveer Chandra*, Geoffrey Challen

University at Buffalo, NY, USA

*Microsoft Research Redmond, WA, USA

University at Buffalo
The State University of New York

Microsoft

Customized Wireless Protocols Are Everywhere

AirPlay

Proprietary Protocol

New Functionality

By extending existing Protocol

Special Requirements

Latency

Power consumption

…

Industry Wireless Design/Implementation Flow

Protocol Designers

Microsoft, Apple, Google…

Wireless Chip Vendors

Qualcomm, MTK…

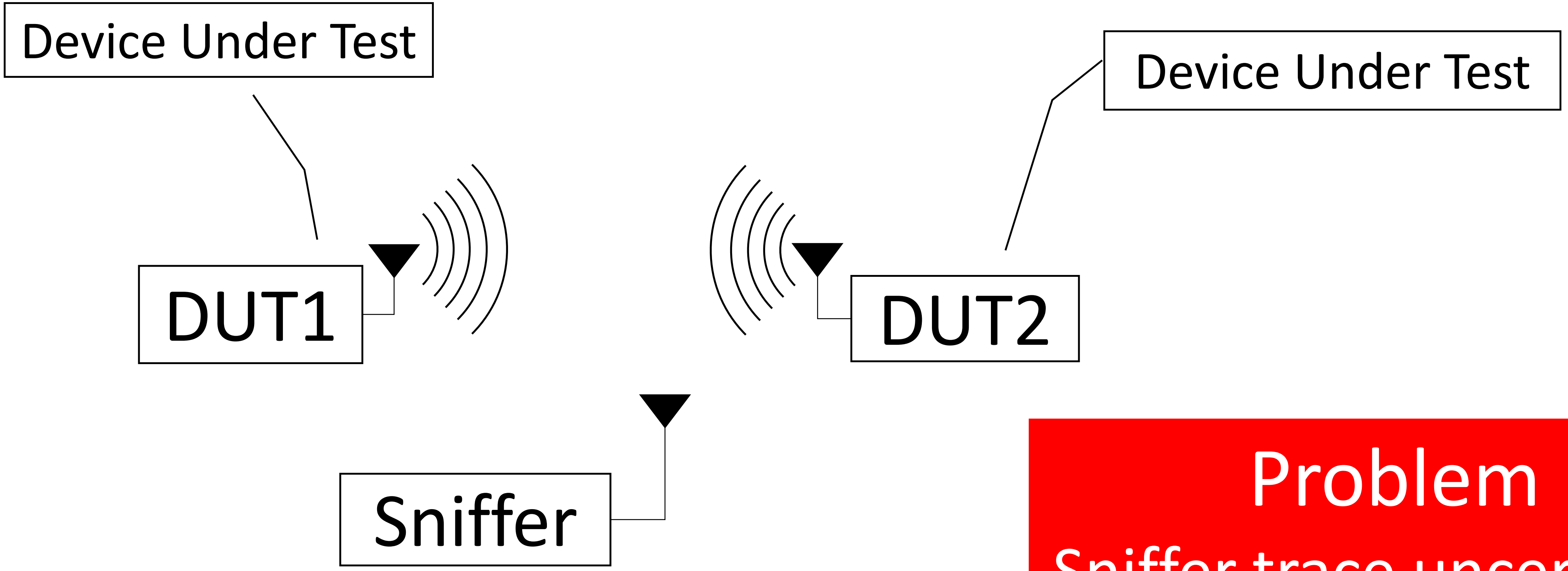
1. Design protocol using simulation
   - Qualnet, NS-3,…

2. Low level **proprietary** implementation

3. How to validate the **implementation** meets the **spec**?

- Proprietary implementation
- Resource limitation
- “Heisenberg” effect

Wireless Sniffer as Observer



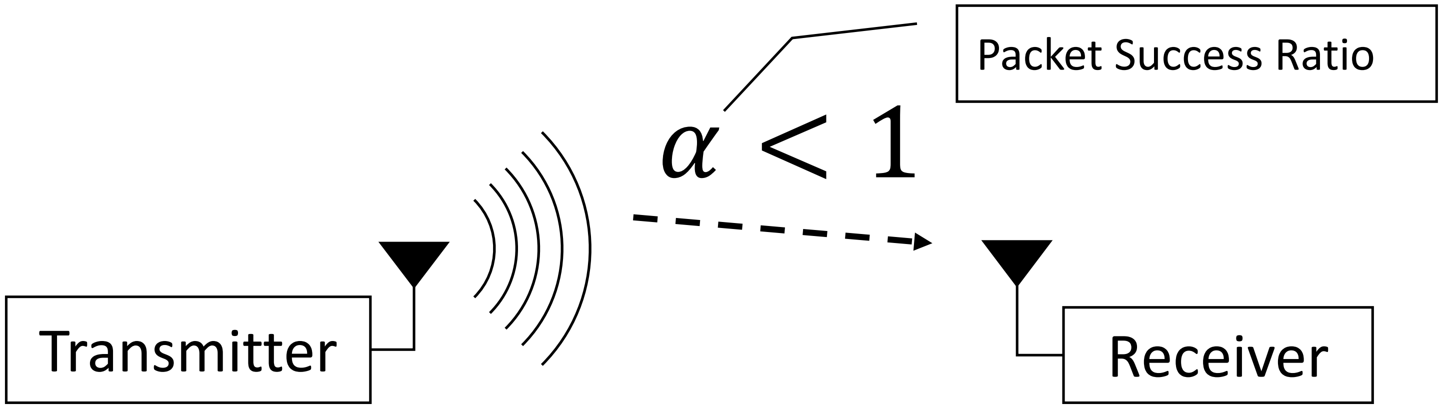
Problem

Sniffer trace uncertainty

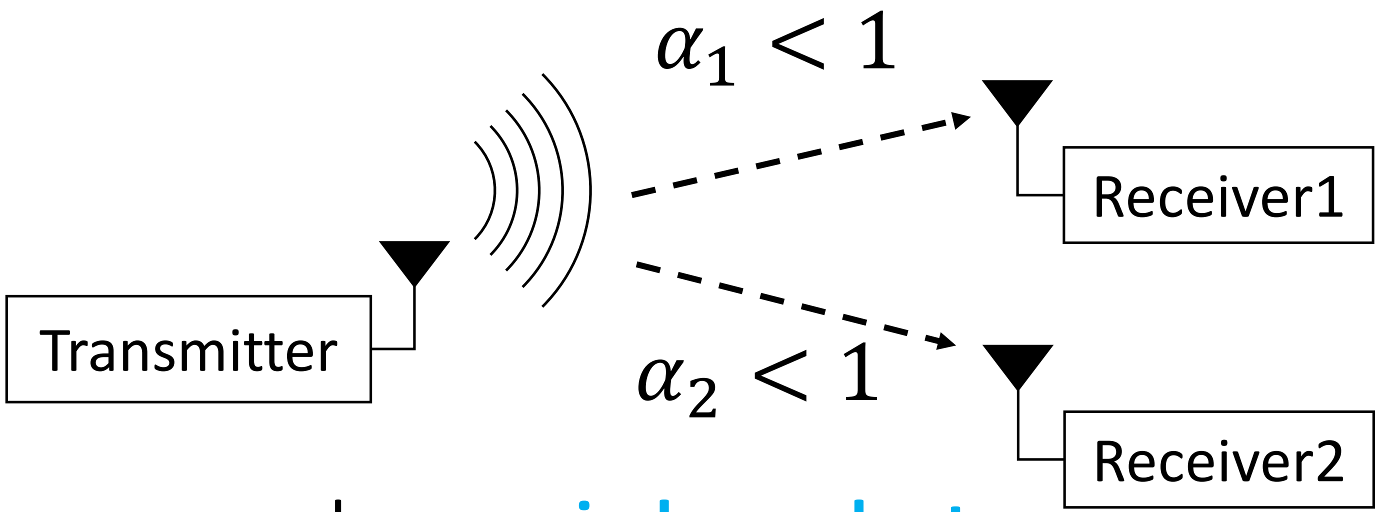
Trace: $Pkt_1 \ Pkt_2 \ Pkt_3 \ \cdots$

Wireless Communication Properties

Packet loss

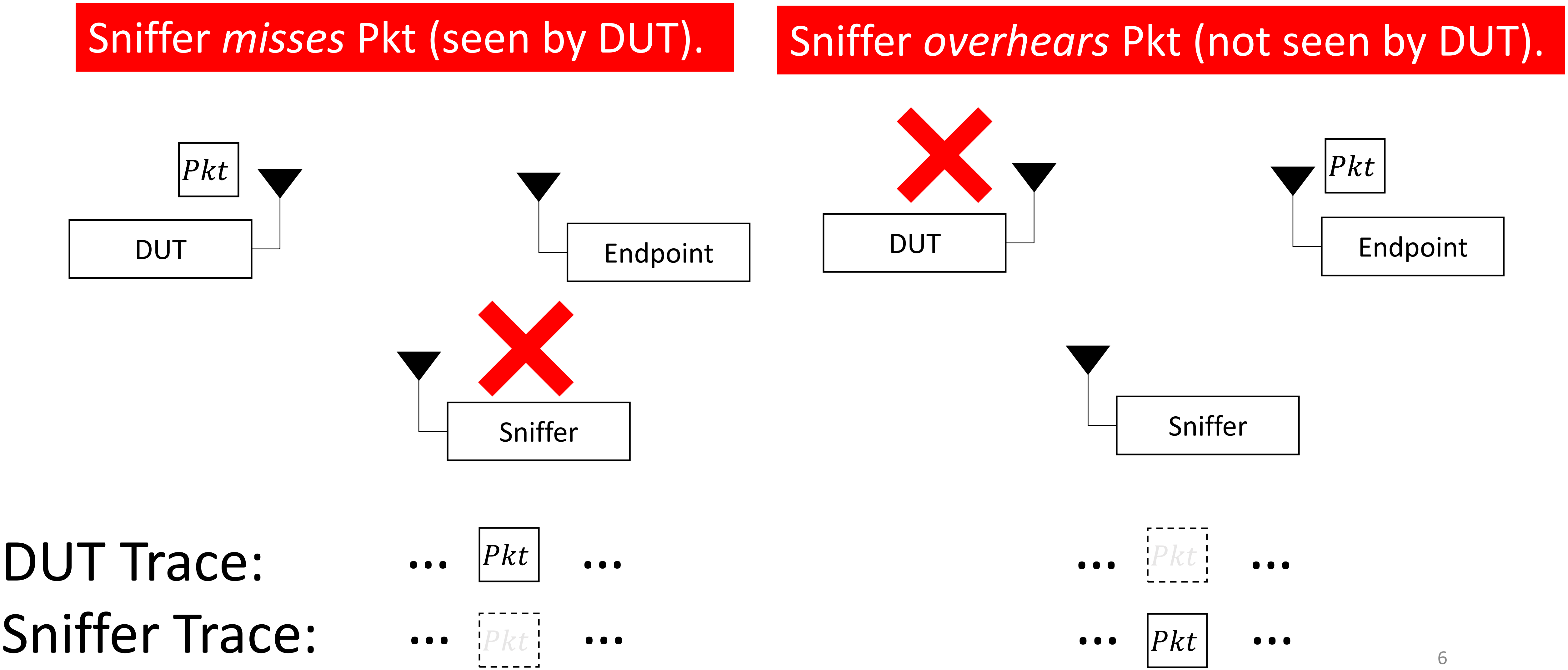


Physical Diversity

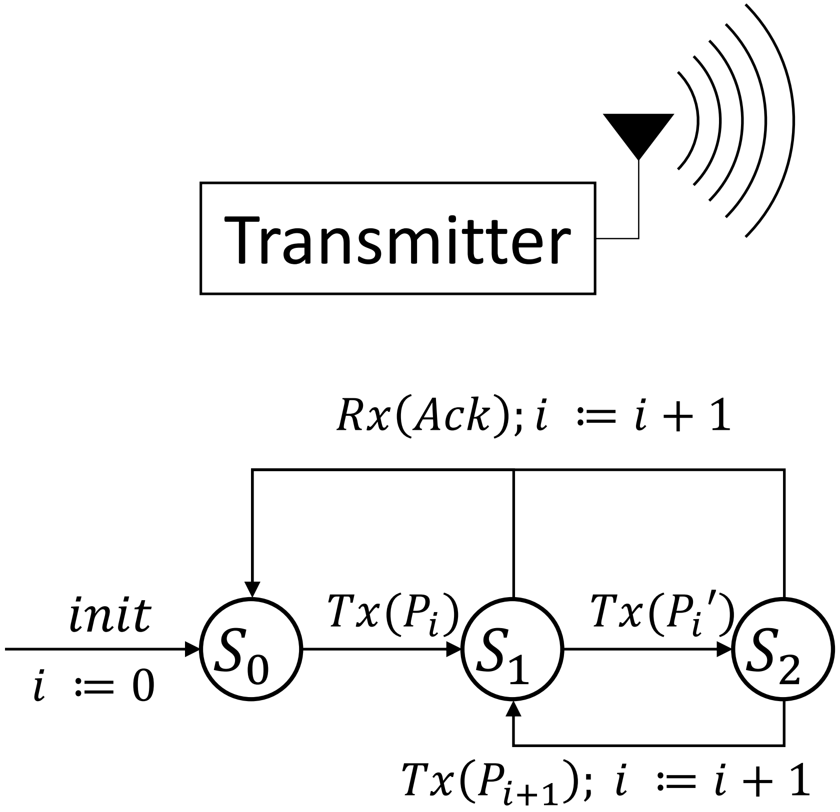


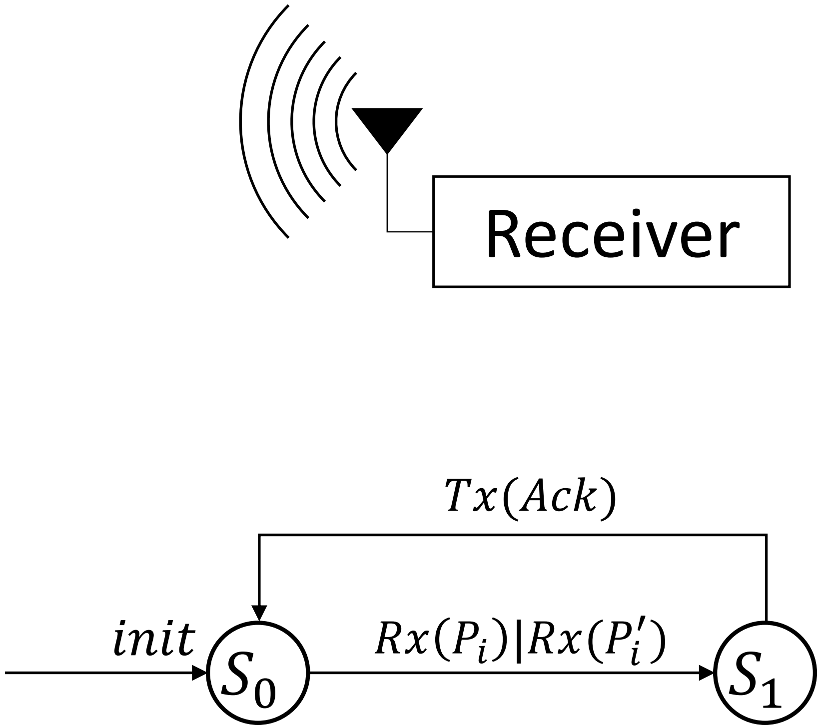
α_1 and α_2 are **independent**

Two Sources of Sniffer Trace Uncertainty



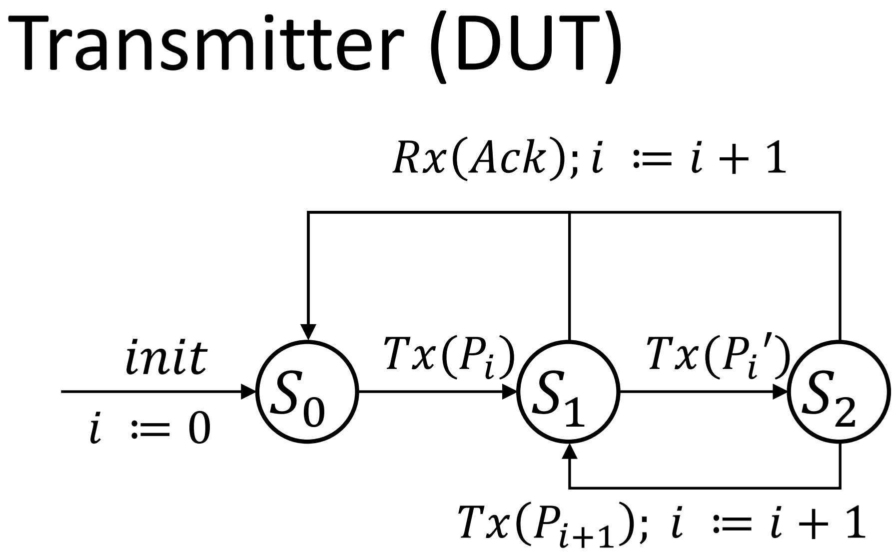
An Example Protocol: Packet Transmission

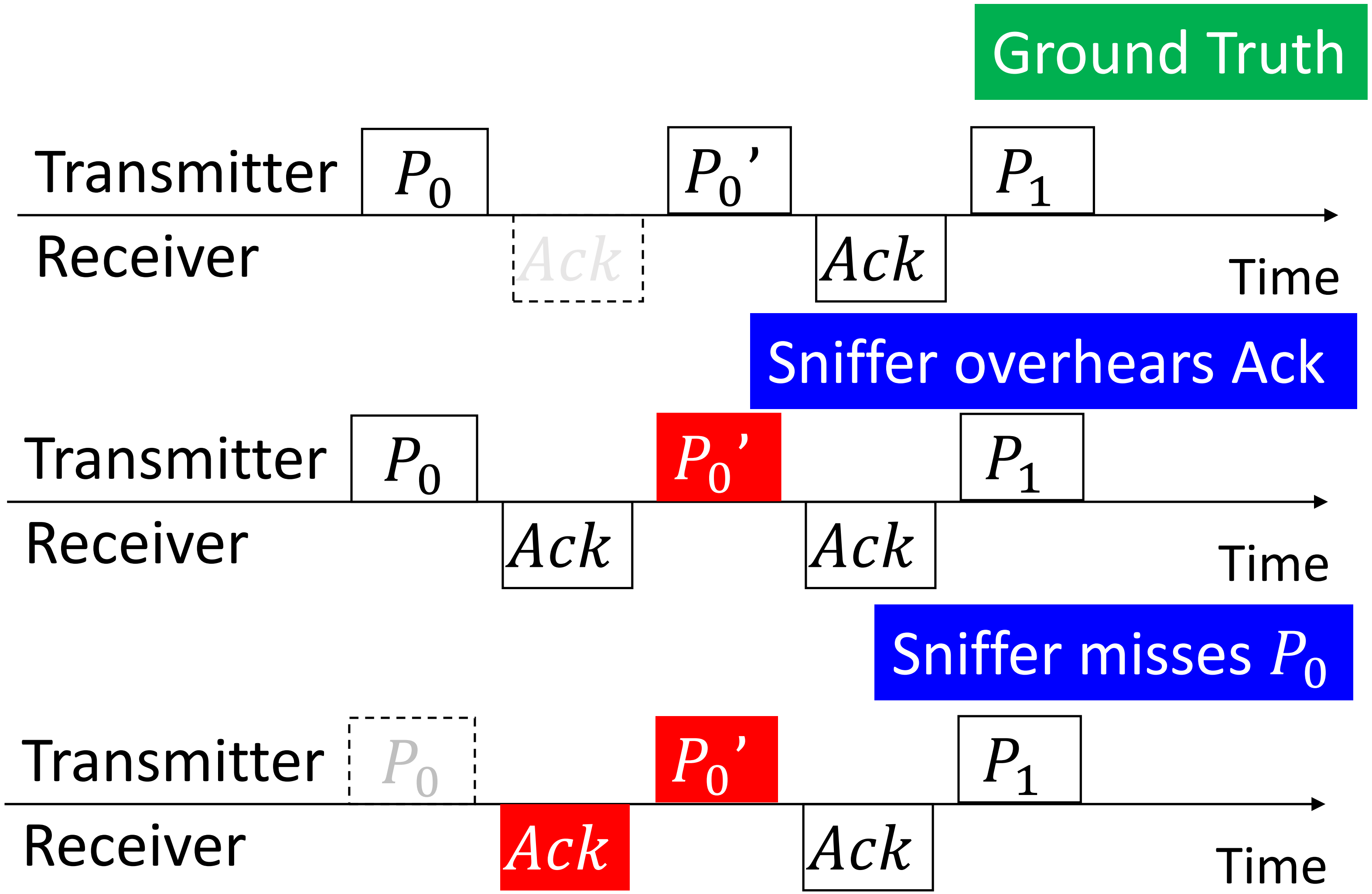




P_i: *packet with seq num i*
P_i': *retransmission of* P_i

False Alarms





Root Cause

Sniffer and DUT may see different traces

Sniffer may either:

- Miss packets that are present in DUT’s trace
- Overhear extra packet that not in DUT’s trace

Can not directly use sniffer trace for validation

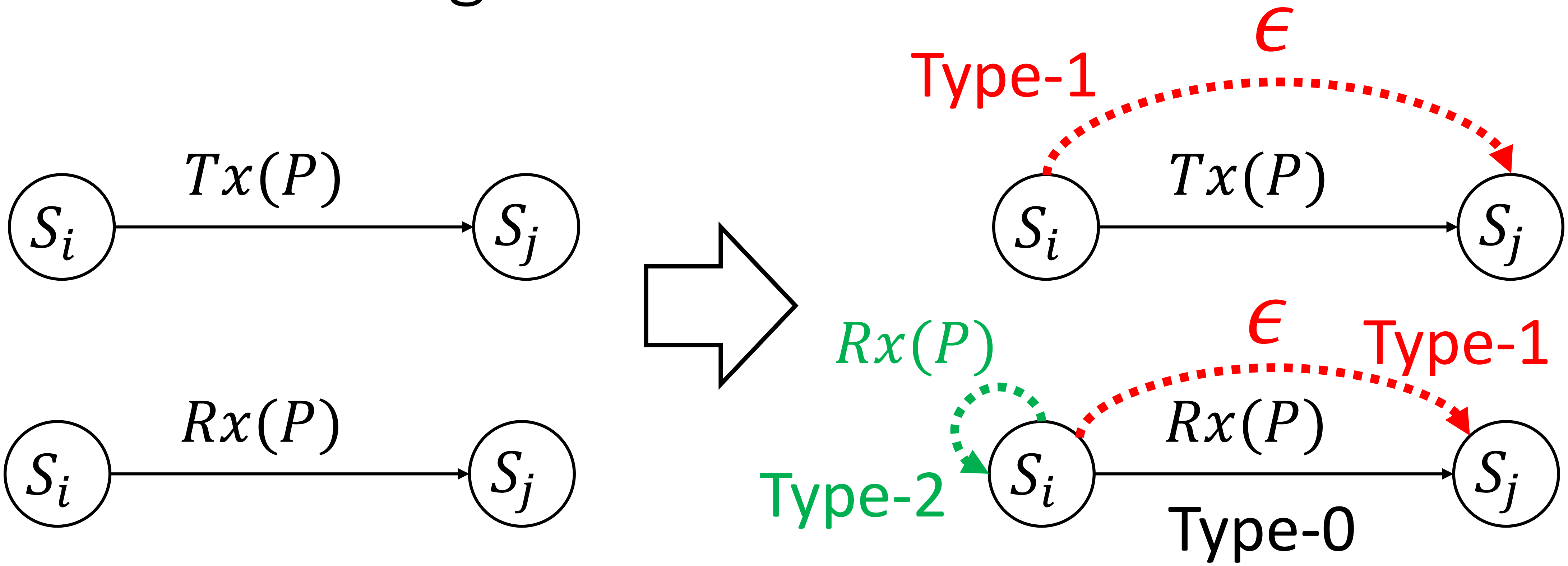
False alarms may occur

Key Idea

Relax the original state machine with <u>non-deterministic transitions</u>

- Avoid raising false alarms, while…
- Still capture true violations

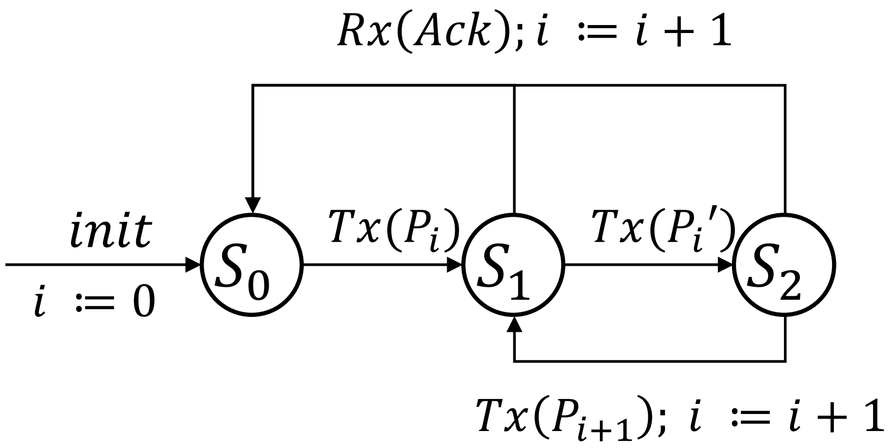
Augmented Transitions

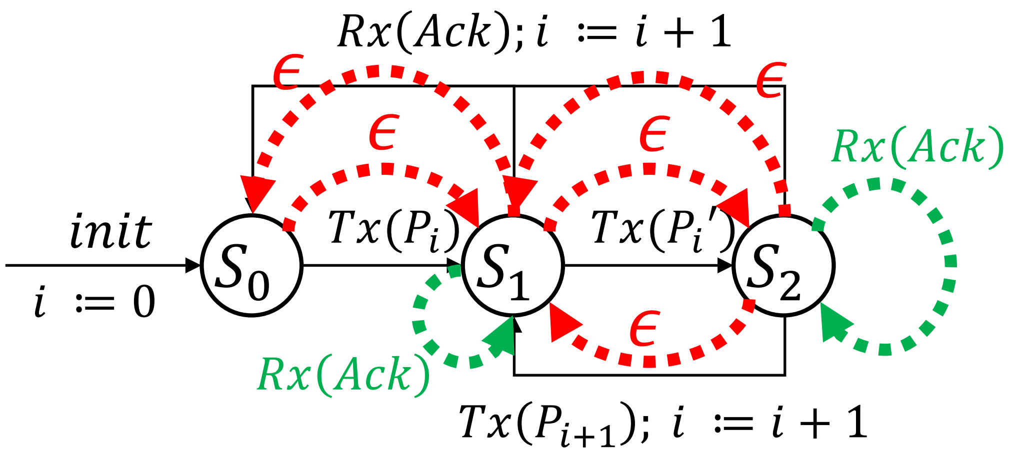


Original State Machine S
Deterministic

Augmented State Machine S^+
Non-Deterministic

Augmented State Machine

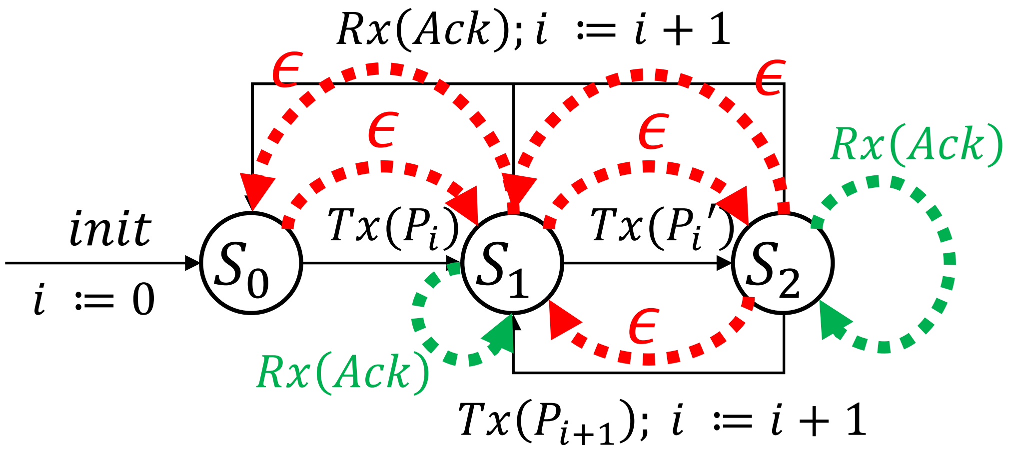




Original State Machine S
Deterministic

Augmented State Machine S^+
Non-Deterministic

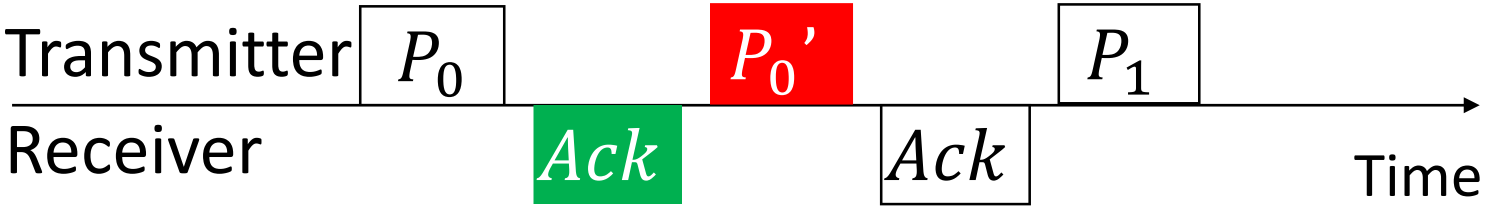
Eliminating False Alarms



Augmented State Machine S^+

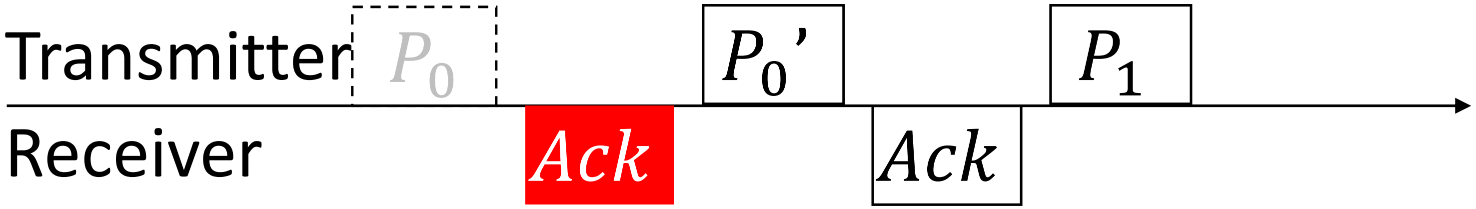
Non-Deterministic

$S_0 \rightarrow S_1 \rightarrow S_1 \rightarrow S_2 \rightarrow \cdots$



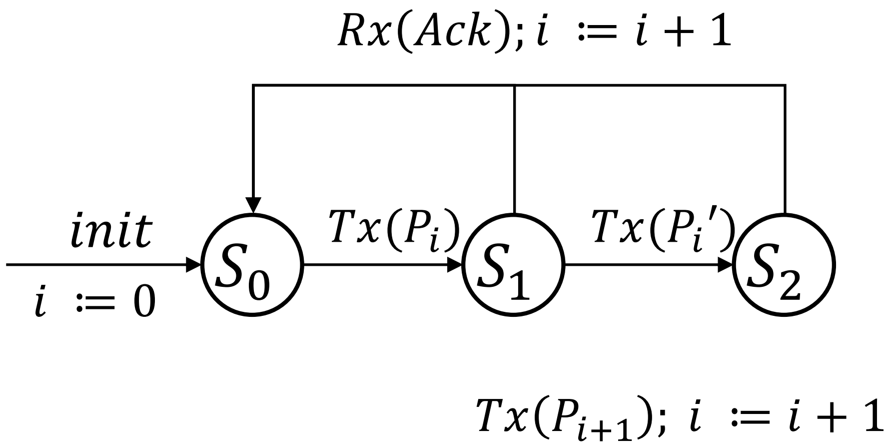
Sniffer overhears Ack

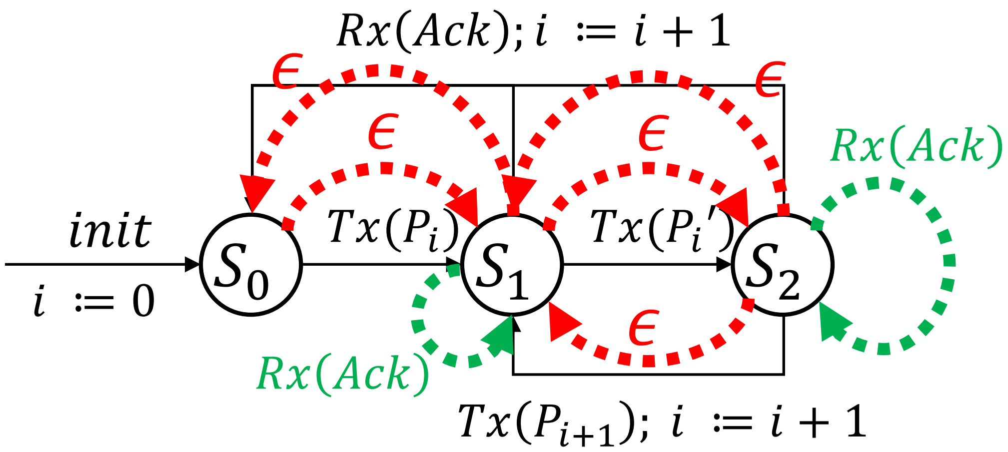
$S_0 \rightarrow S_1 \rightarrow S_2 \rightarrow \cdots$



Sniffer misses P_0

The Problem: Does S accept Tr_{DUT}?





Original State Machine S

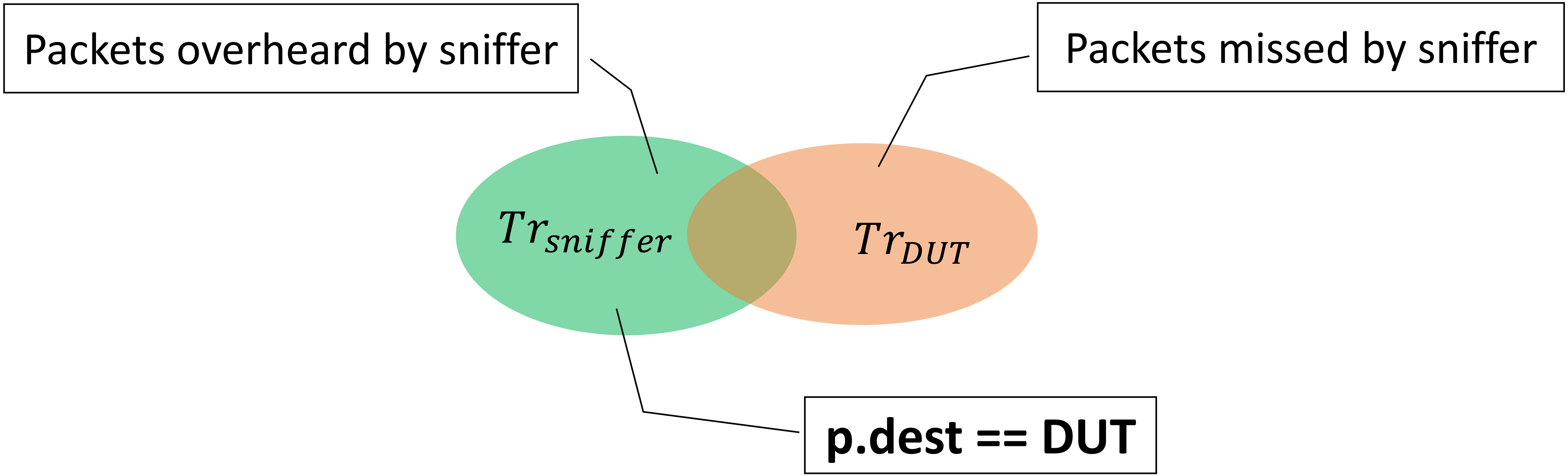
DUT Trace: Tr_{DUT}

?

Augmented State Machine S^+

Sniffer Trace: $Tr_{sniffer}$

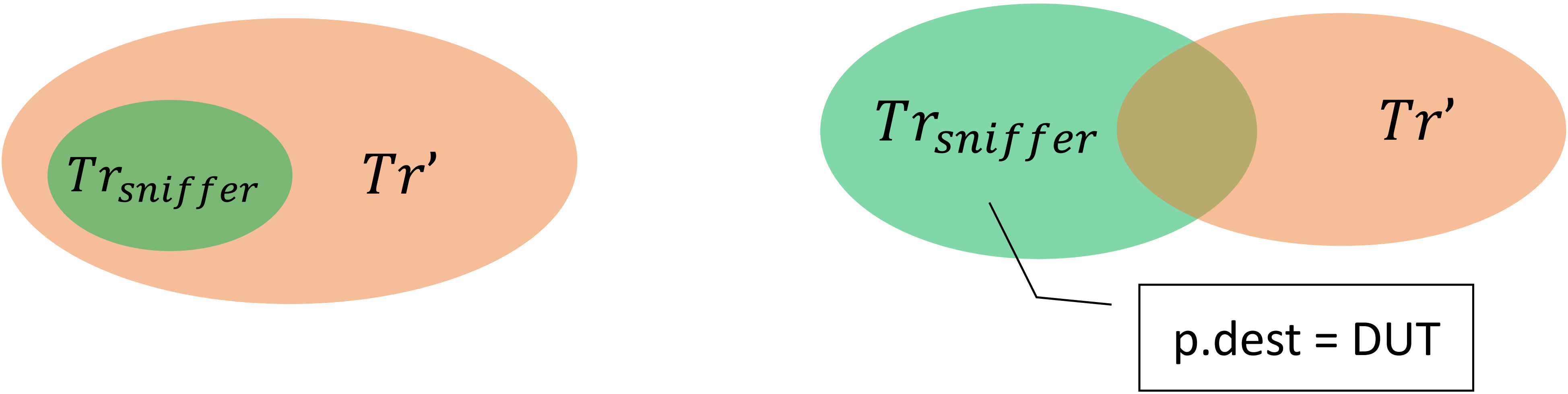
Relationship of Tr_{DUT} and $Tr_{sniffer}$



Sniffer can not overhear packets that are not sent by DUT

Mutation Trace

- Definition: Mutation Trace
  - A packet trace Tr’ is a **mutation** of sniffer trace $Tr_{sniffer}$ w.r.t a DUT if for all $(t, p) \in Tr_{sniffer} / Tr$’, p.dest = DUT.
- Lemma: $Tr_{DUT} \in M(Tr_{sniffer})$ (Set of mutation traces of $Tr_{sniffer}$)



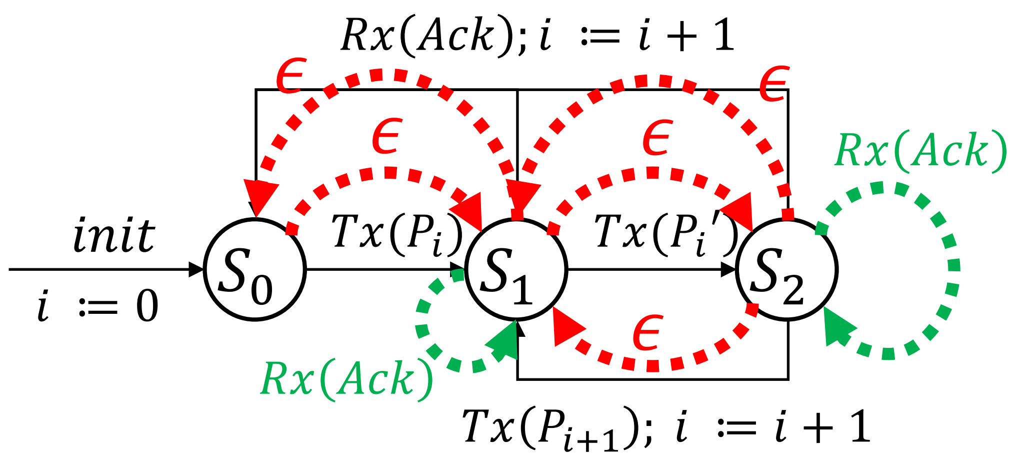
Satisfiability Theorem

S^+ accepts $Tr_{sniffer}$ iff. $\exists$ $Tr' \in M(Tr_{sniffer})$ that S accepts Tr'

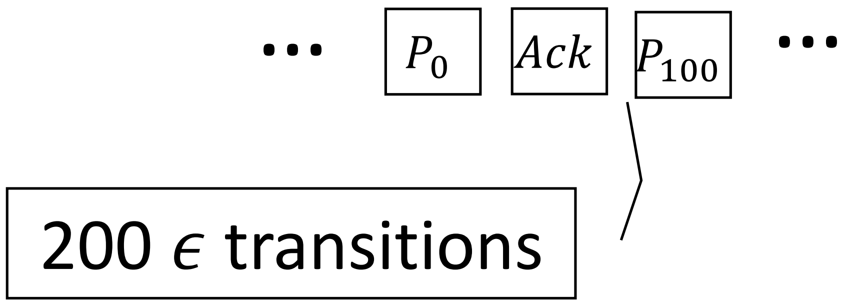
- Lemma
  If S^+ rejects $Tr_{sniffer}$, then S rejects Tr_{DUT}
- S^+ accepts $Tr_{sniffer} \not\Rightarrow S$ accepts Tr_{DUT} .
  - Fundamental limitation of sniffer trace

Proof: https://www.microsoft.com/en-us/research/publication/wireless-protocol-validation-under-uncertainty/

Instance of (Likely) Violation



Sniffer Trace



Relaxed too much…

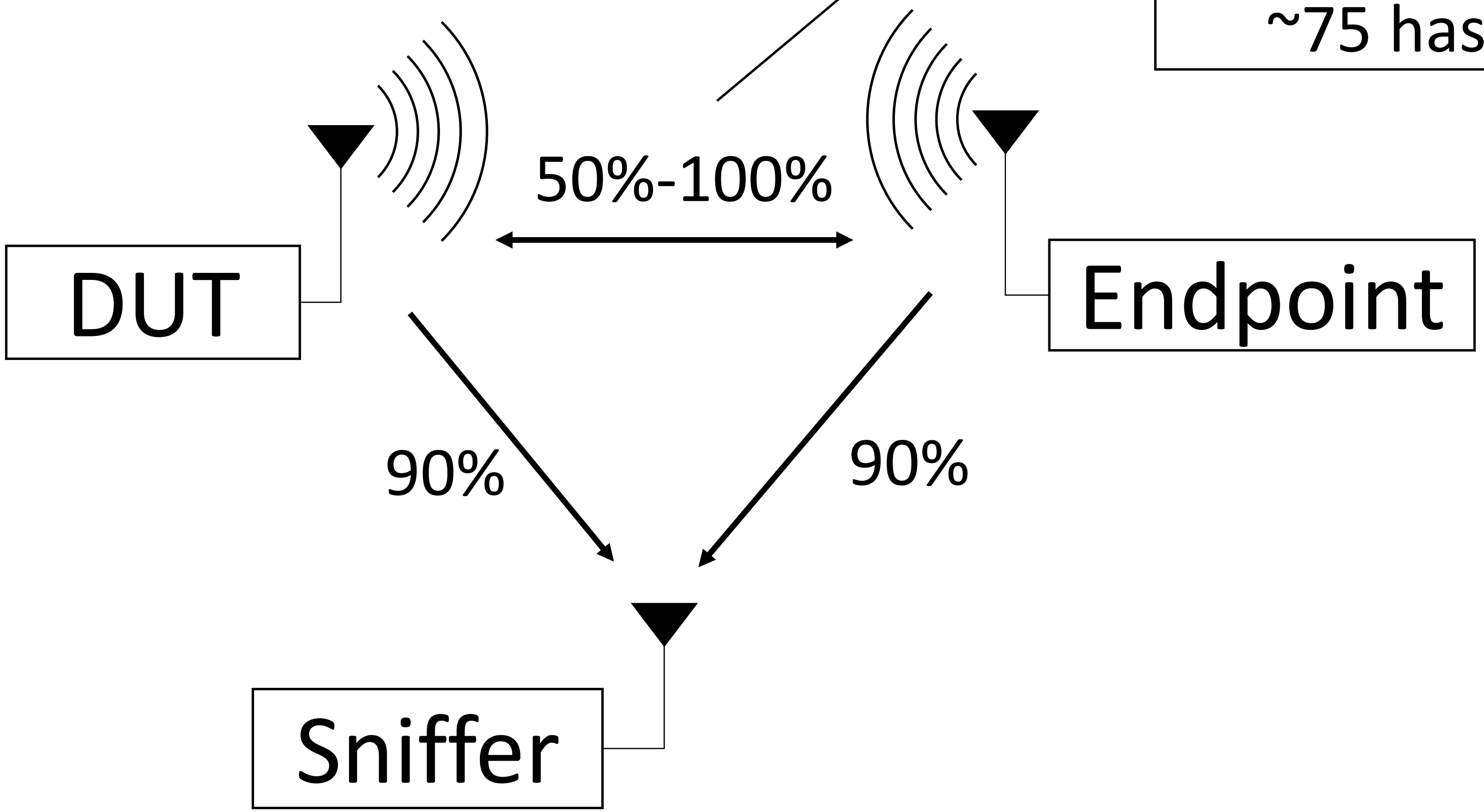
Pruning Heuristics

- Goal:
  - Constraint augmented transitions to report true violations
  - Make runtime practical
- $NumMissing(d, k, l)$
  - For device d, number of missing packets (Type-1) of and subtrace of length l must not exceed k
- $GoBack(k)$
  - Backtrace up to k packets

Evaluation on NS-3

Experiment Setup

1% step
100 runs for each ratio
~75 has synthesized bugs



Evaluation Metrics

$$\text{Precision} = \frac{\{\text{Reported Bugs}\} \cap \{\text{True Bugs}\}}{\{\text{Reported Bugs}\}}$$

Accuracy

Higher precision,
Less false positive.

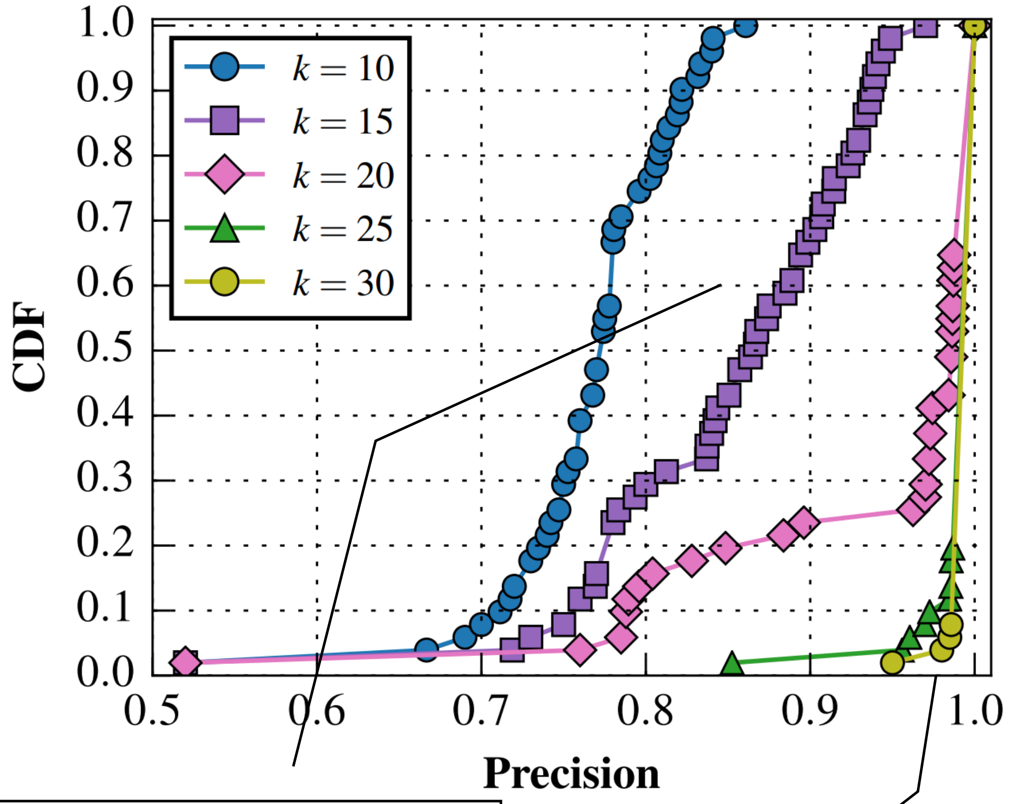
$$\text{Recall} = \frac{\{\text{Reported Bugs}\} \cap \{\text{True Bugs}\}}{\{\text{True Bugs}\}}$$

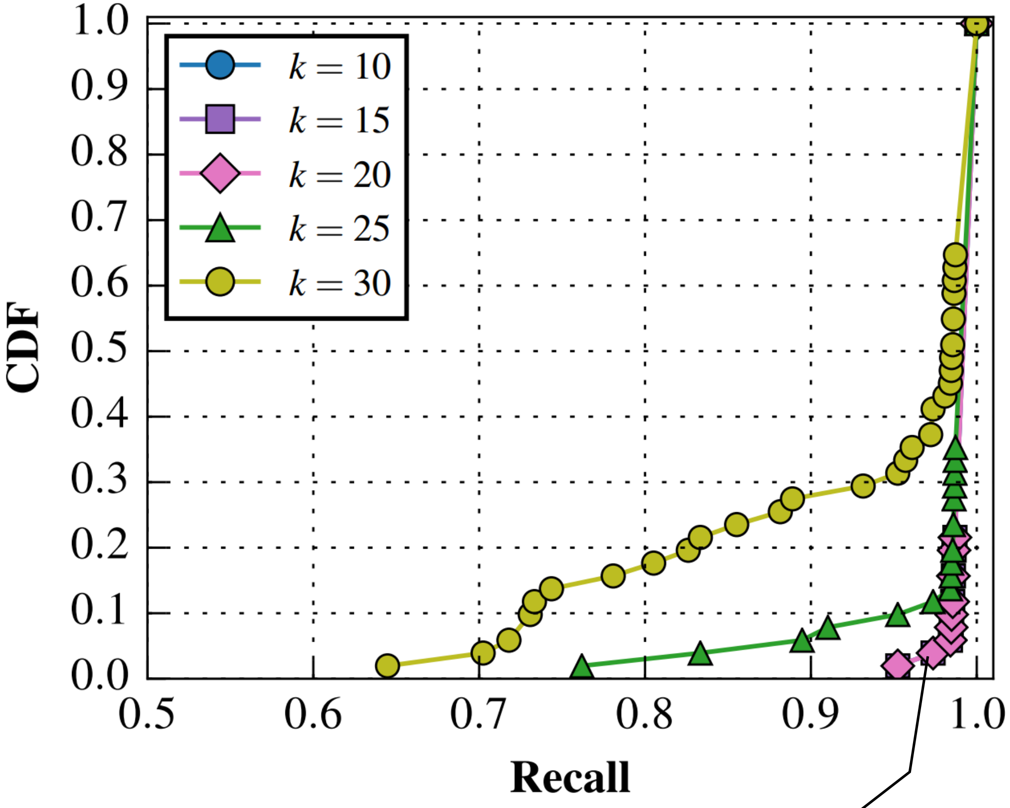
Completeness

Higher recall,
Less false negative.

Results

Heuristics: $NumMissing(d, k, 100)$(fixed l = 100), GoBack(7)





More tolerant to sniffer loss,
Less false positive.

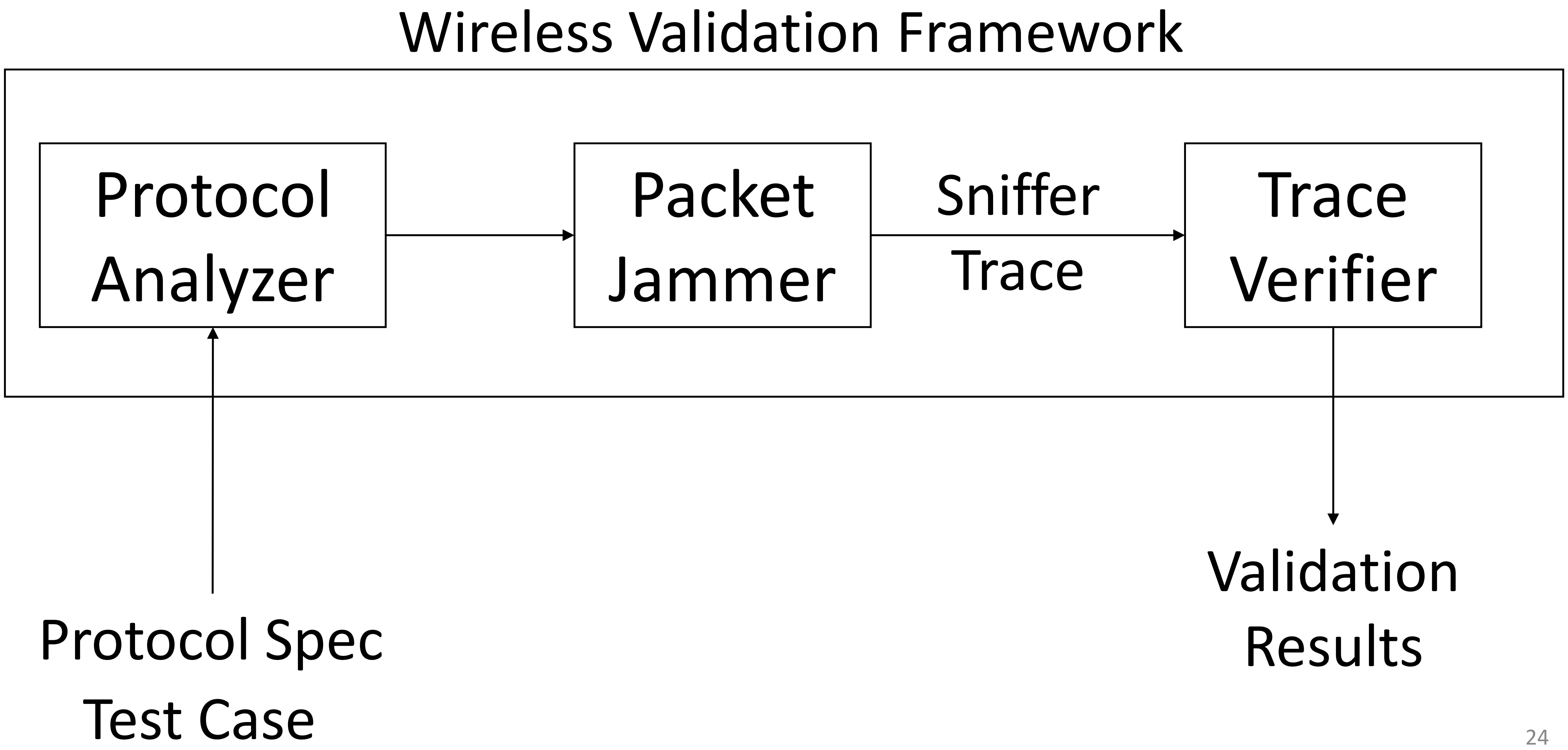
No False Positive

No False Negative

Real-World Application

- Found 3 latent bugs in the development phase of Xbox One wireless controller
- Being actively used by Xbox accessory testing team (since 08/2015)

Ongoing/Future Works



Summary

- Sniffer trace uncertainty
  - Miss or overhear packets
- Augmented transition to tolerate sniffer trace uncertainty
  - Type-1 and Type-2 edges
- Satisfiability theorem and NP-hardness
  - Lemma: S^+ rejects $Tr_{sniffer} \Rightarrow S$ rejects Tr_{DUT}
- Pruning heuristics
  - $NumMissing(d, k, l)$
  - $GoBack(k)$