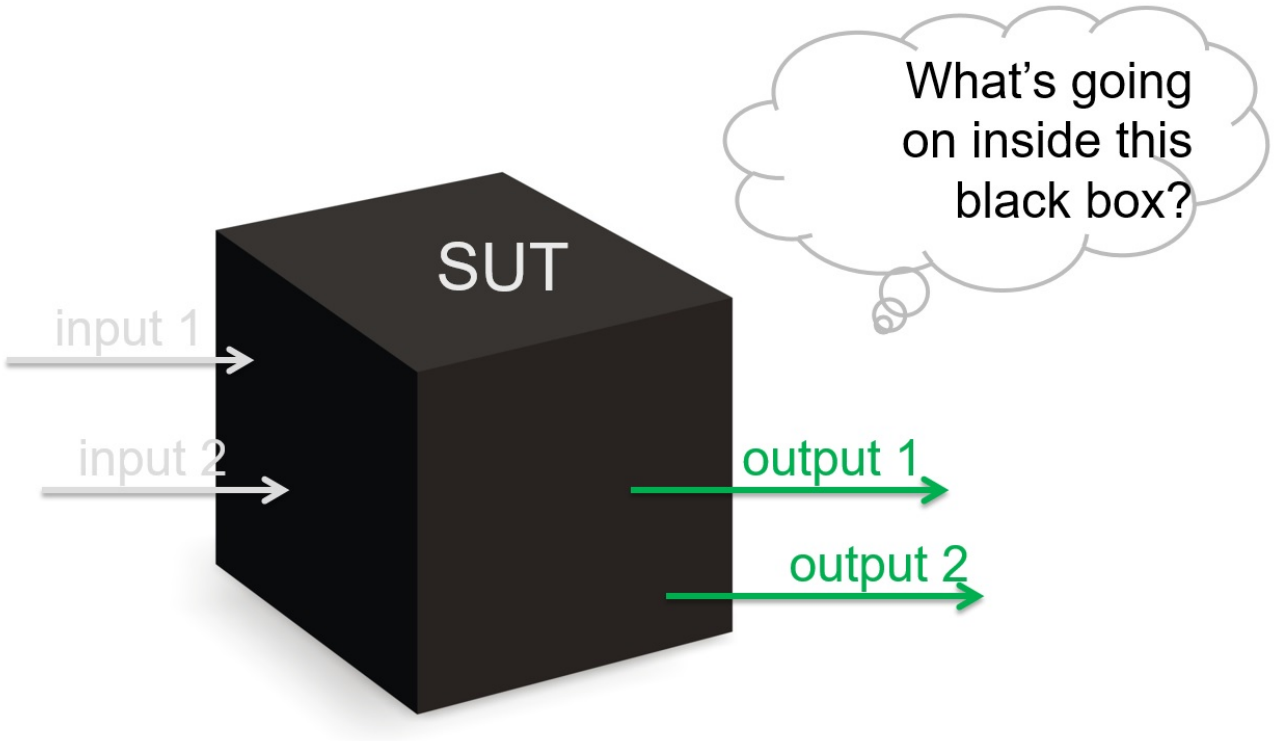
Model Learning as an SMT Problem

Rick Smetsers Paul Fiterău-Broștean Frits Vaandrager

Radboud University Nijmegen

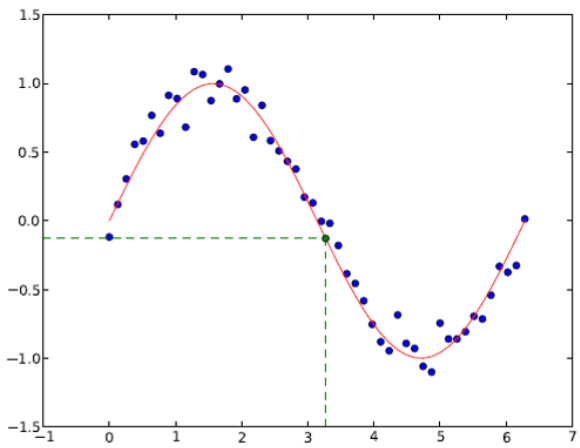
LATA 2018, Bar-Ilan, April 9-11, 2018

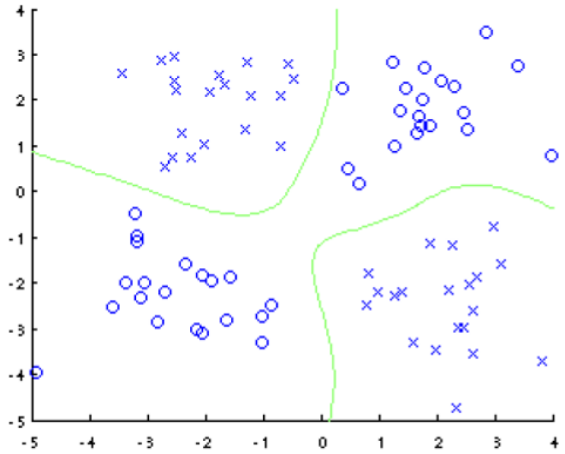
Goal active automaton learning



Machine Learning in General

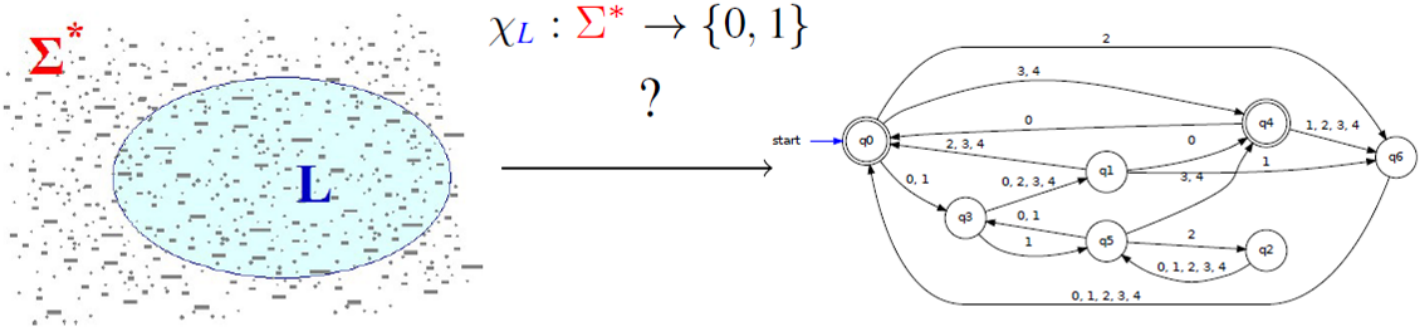
- Given a sample $M = \{(x, y) \mid x \in X, y \in Y\}$
- Find $f : X \to Y$ such that $f(x) = y, \forall (x, y) \in M$
- Predict $f(x)$ for all $x \in X$





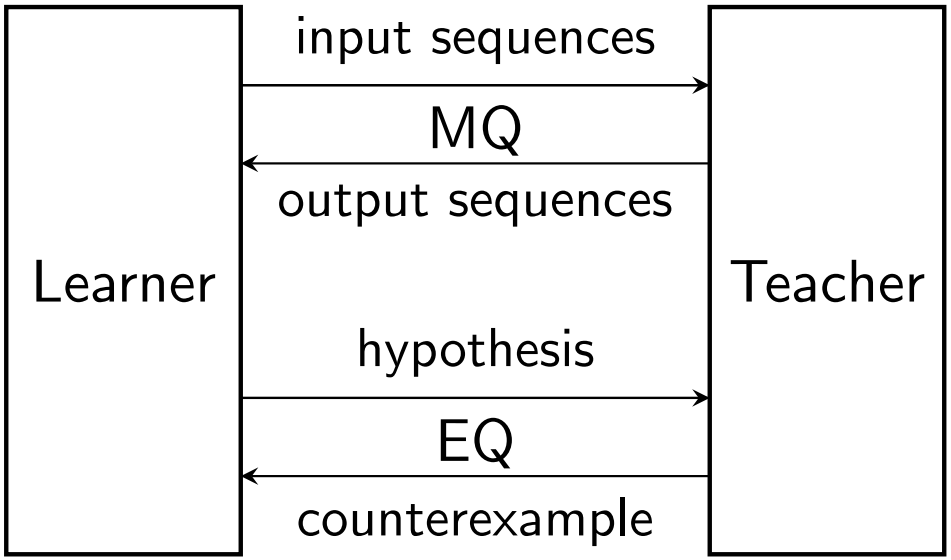
Learning Regular Languages

Let Σ be an alphabet and let $L \subseteq \Sigma^*$ be a regular language (*the target language*)

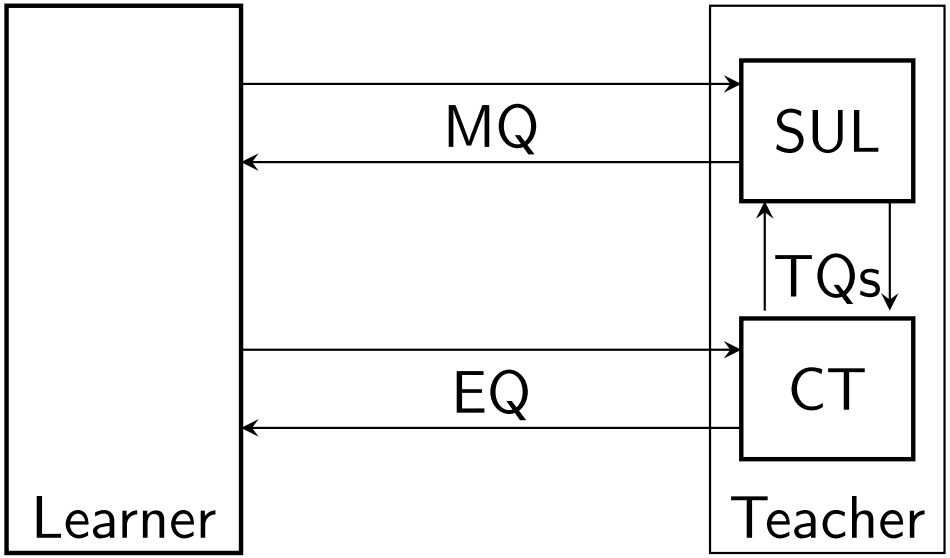


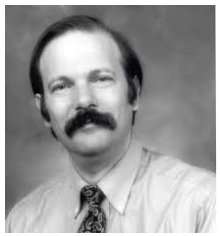
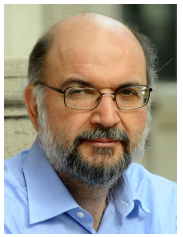
- Edward F Moore, *Gedanken-experiments on sequential machines*, 1956
- E. Mark Gold, *System Identification via State Characterization*, 1972
- Dana Angluin, *Learning regular sets from queries and counterexamples*, 1987

Minimally adequate teacher (Angluin)



Black box checking (Peled, Vardi & Yannakakis)

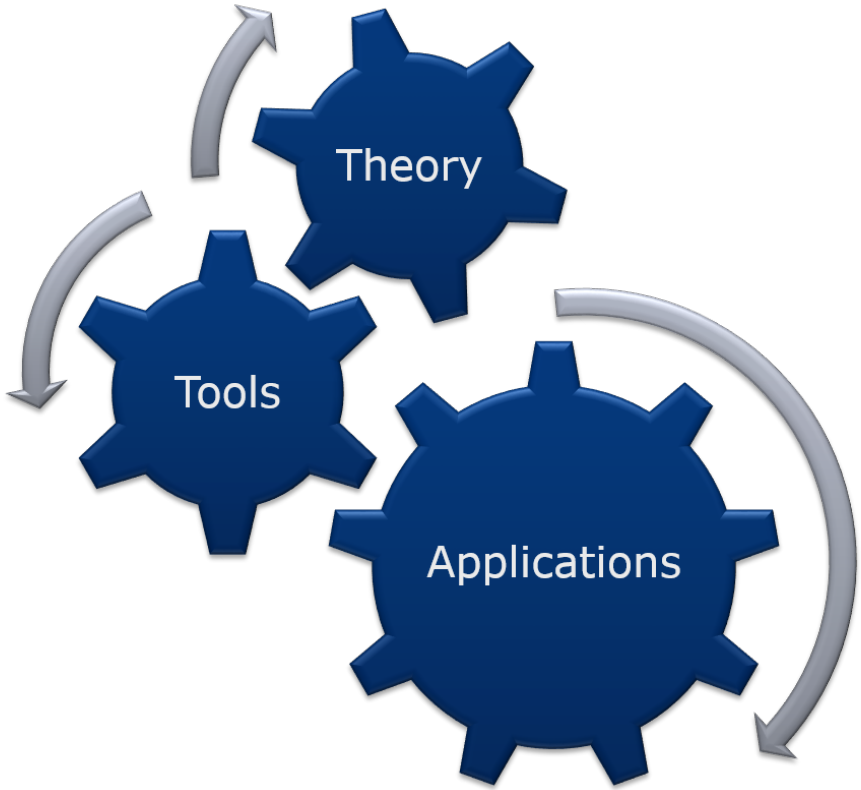


Learner: Formulate hypotheses

Conformance Tester (CT): Test correctness hypotheses

Research method



Application: E.dentifier2

NOS Zoeken binnen NOS.nl Vandaag 8° Morgen 9° Verkeer 2 km AEX 379,41 meer

NOS.nl | Nieuws | Binnenland | Buitenland | Politiek | Economie | Opmerkelijk | Sport | Televisie | Radio | Mobiel

Economie | Overzicht | Nieuwsarchief | Video & audio | Journaal 24 | Politiek 24 | Dossiers | Financieel

E-bankieren ABN Amro kwetsbaar

donderdag 16 aug 2012, 18:02 (Update: 17-08-12, 08:39)

ABN Amro NOS

Internetbankieren bij ABN Amro is gevoelig voor fraude. Internetcriminelen kunnen sommige betalingstransacties onderscheppen, aanpassen en doorsluizen naar hun eigen rekening.

Video

Internetbankieren met E.dentifier ABN Amro onveilig

Internetbankieren met de E.dentifier2 van de ABN AMRO is onveilig als ie wordt gebruikt met een USB-kabel. Dat zegt de Radboud... (meer)

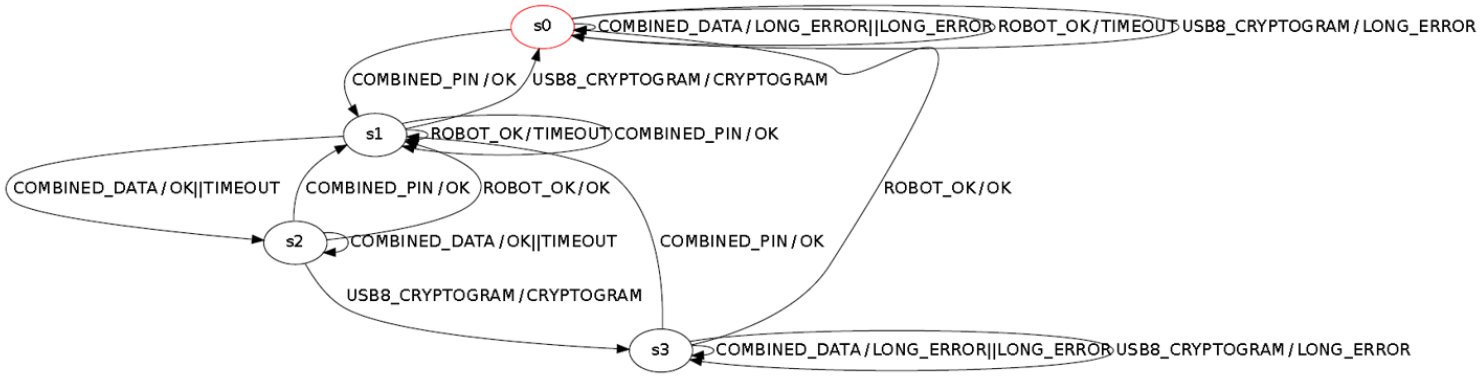
Audio

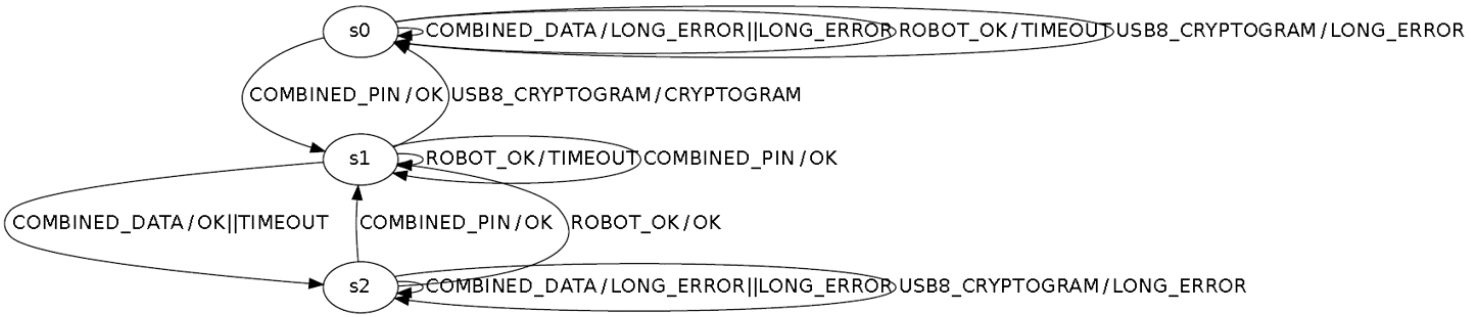
Softwarefout in identifier ABN Amro

Onderzoekers van de Radboud Universiteit hebben aangetoond dat het apparaatje waarmee je via ABN Amro kunt internet-bankieren... (meer)

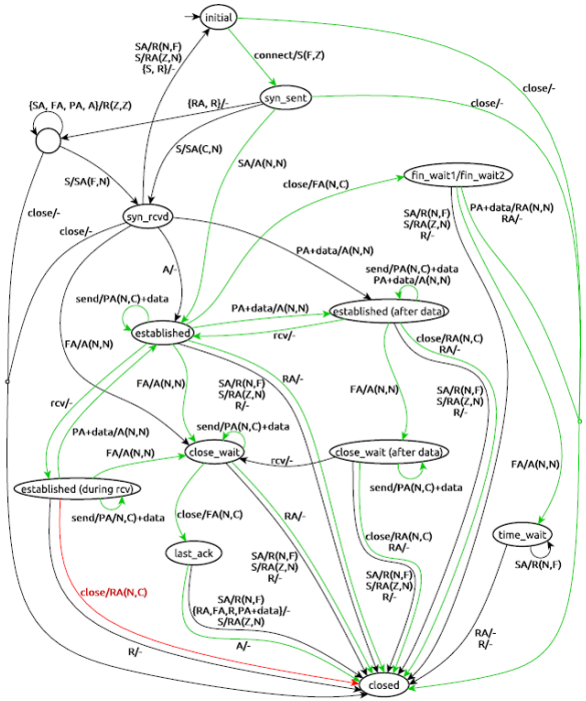
00:00 00:00

State machines for old and new E.dentifier2



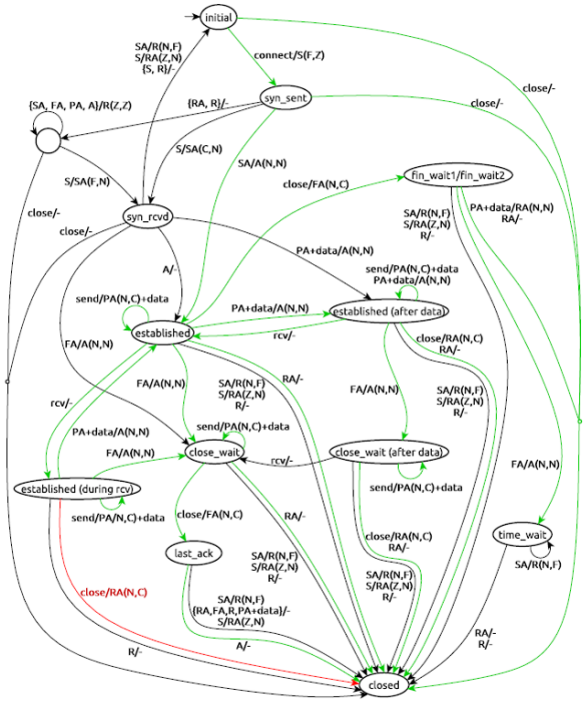


Bugs in protocol implementations



Standard violations found in implementations of major protocols, e.g., TCP (CAV'16, FMICS'17), TLS (Usenix Security'15), SSH (Spin'17).

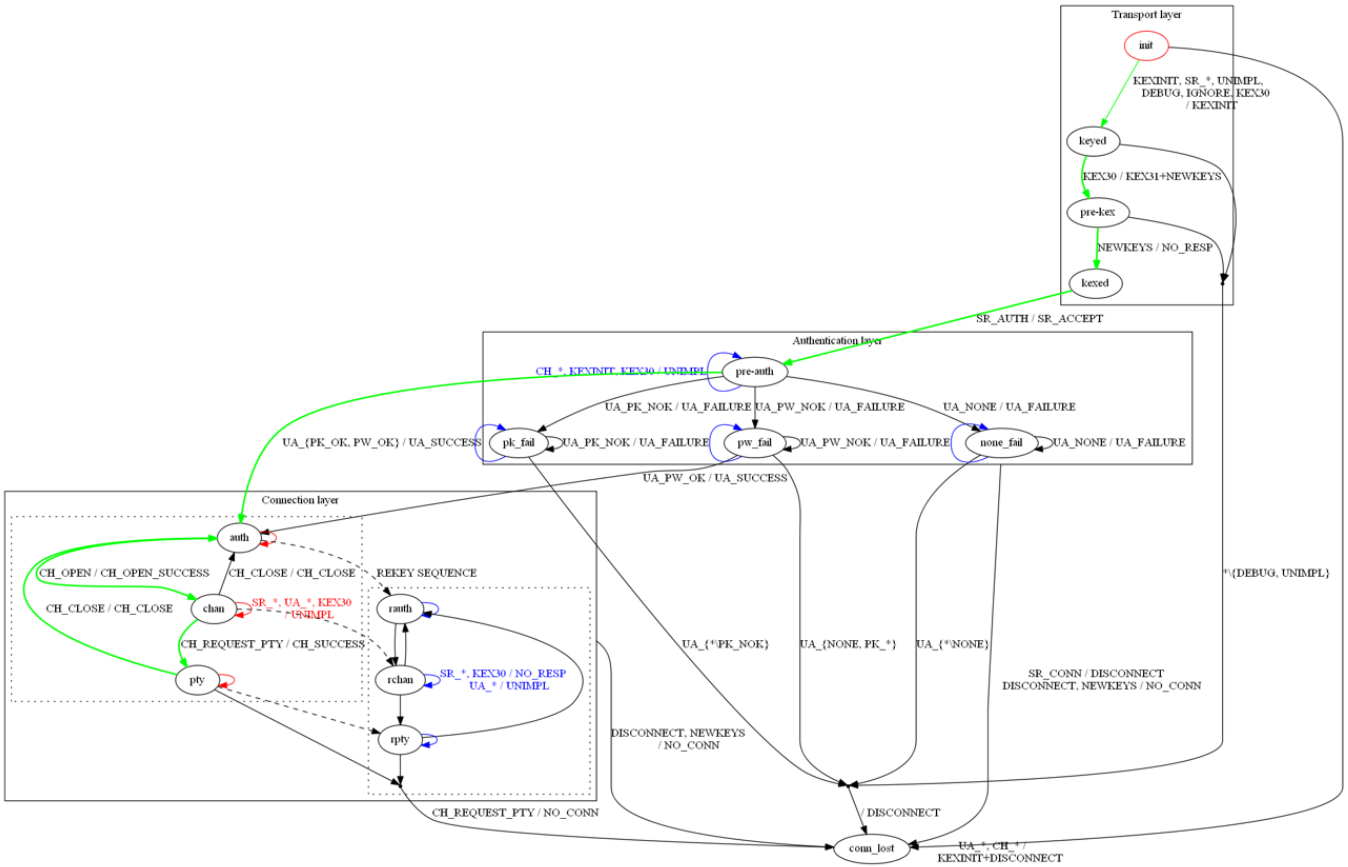
Bugs in protocol implementations



Standard violations found in implementations of major protocols, e.g., TCP (CAV'16, FMICS'17), TLS (Usenix Security'15), SSH (Spin'17).

These findings led to several bug fixes in implementations.

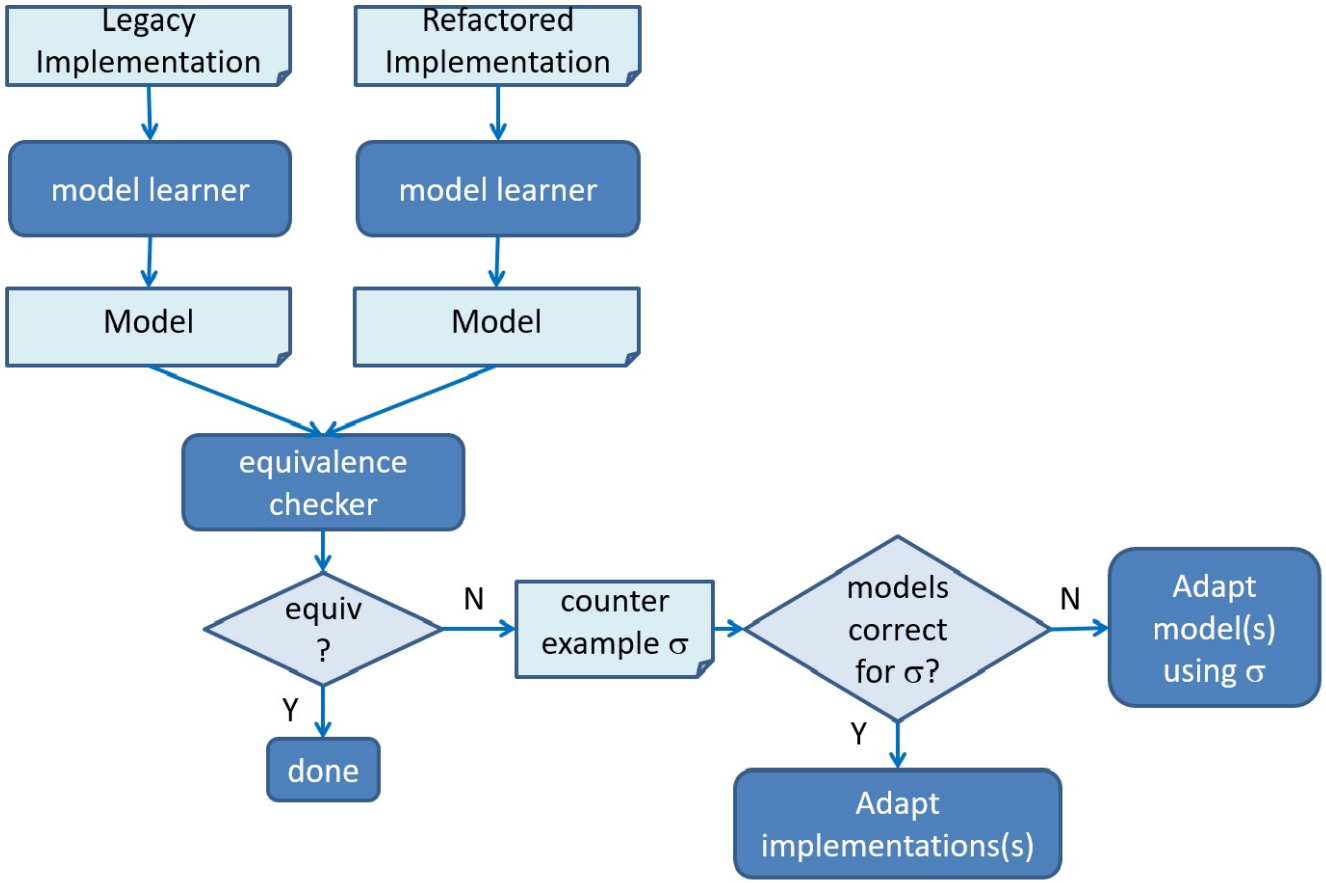
Learned model for SSH implementation



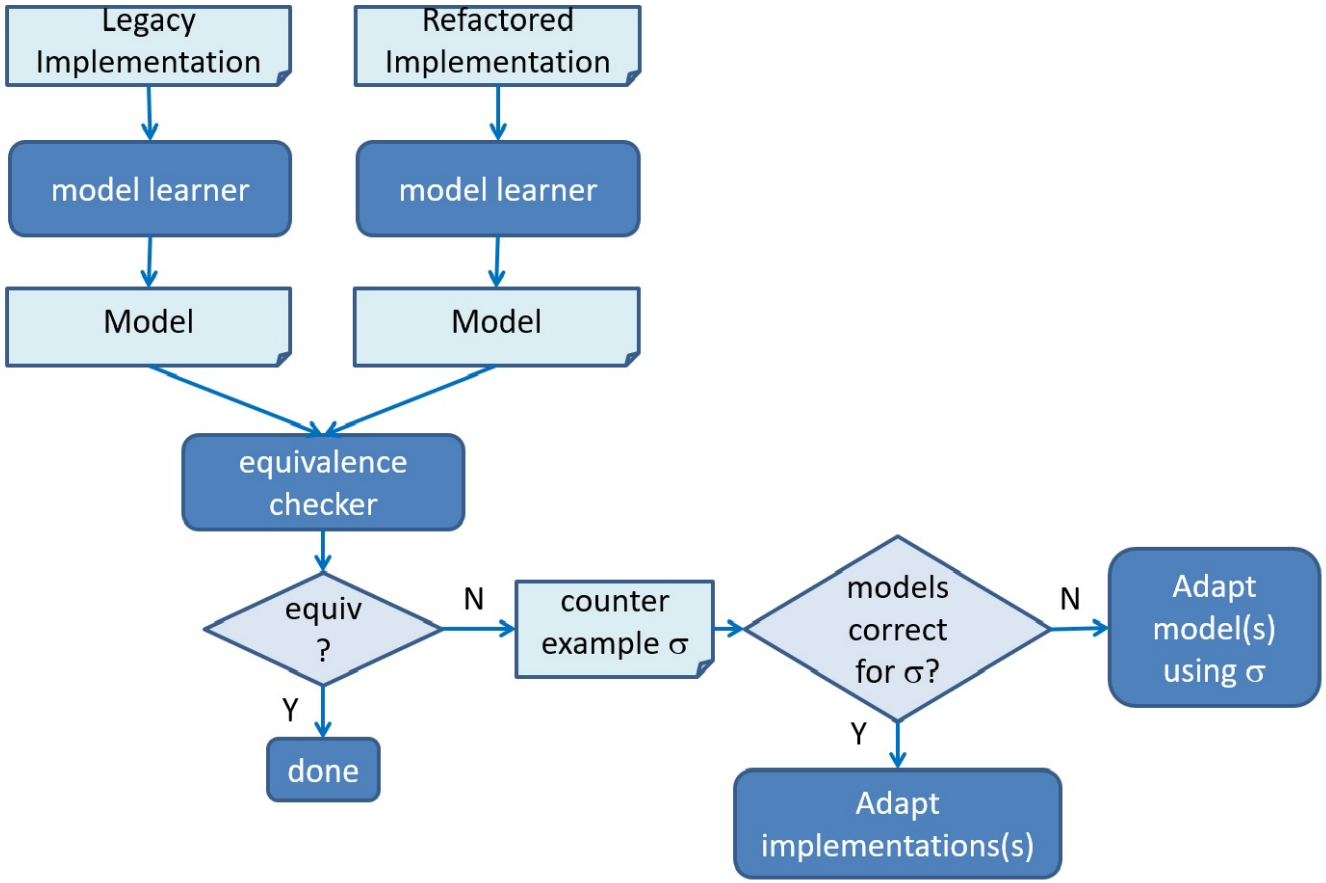
Power Control Service from Philips Healthcare

Are legacy component and refactored implementation equivalent?

Refactoring Legacy Implementations

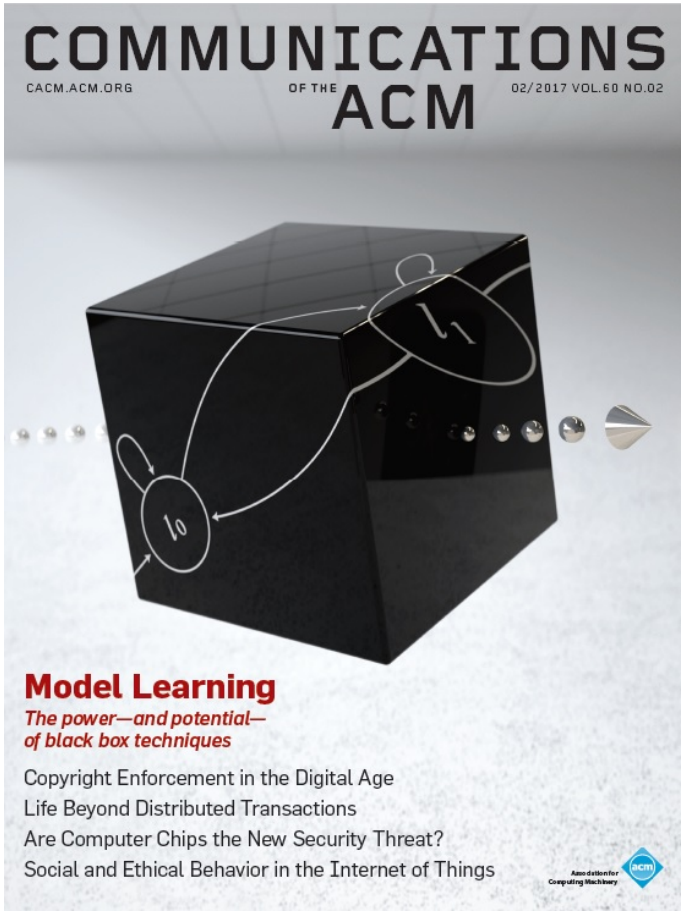


Refactoring Legacy Implementations

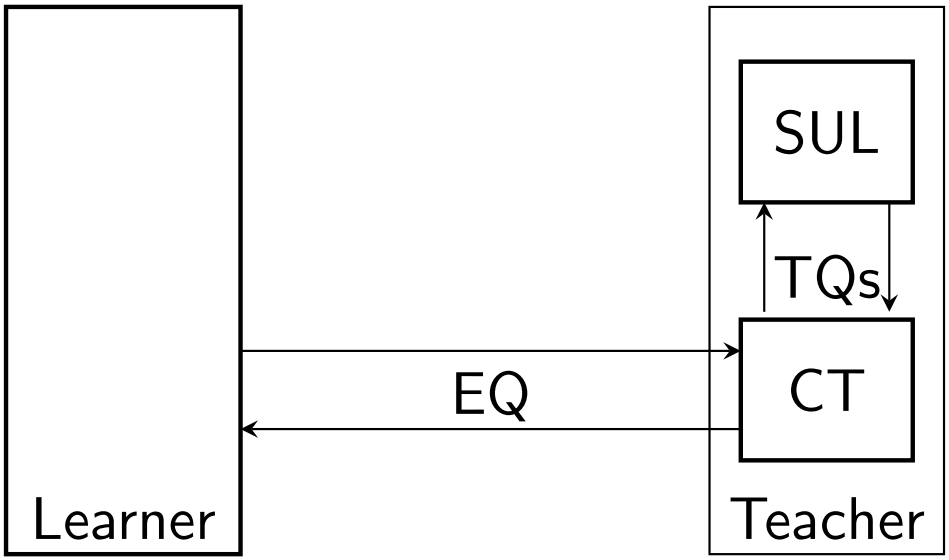


This approach allowed us to find several bugs in refactored implementations.

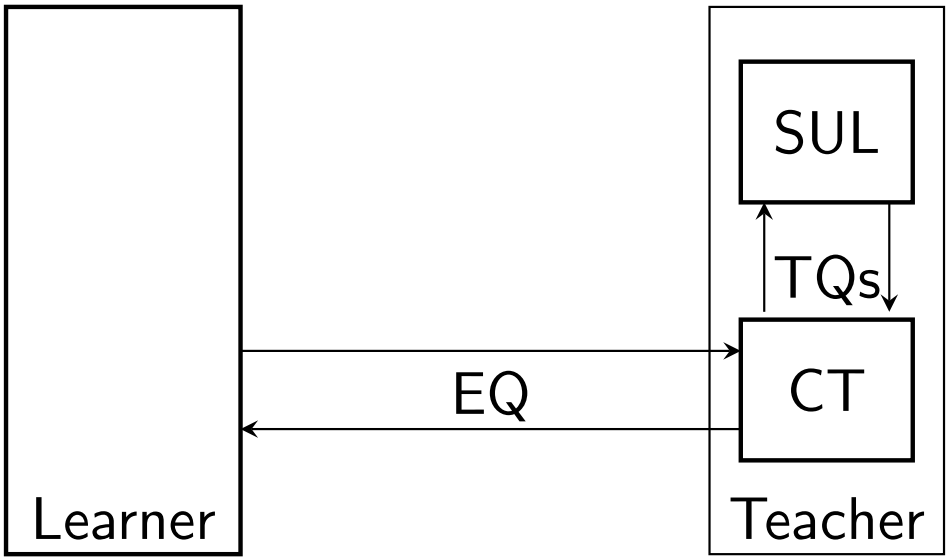
For background and applications see CACM review article



Our approach

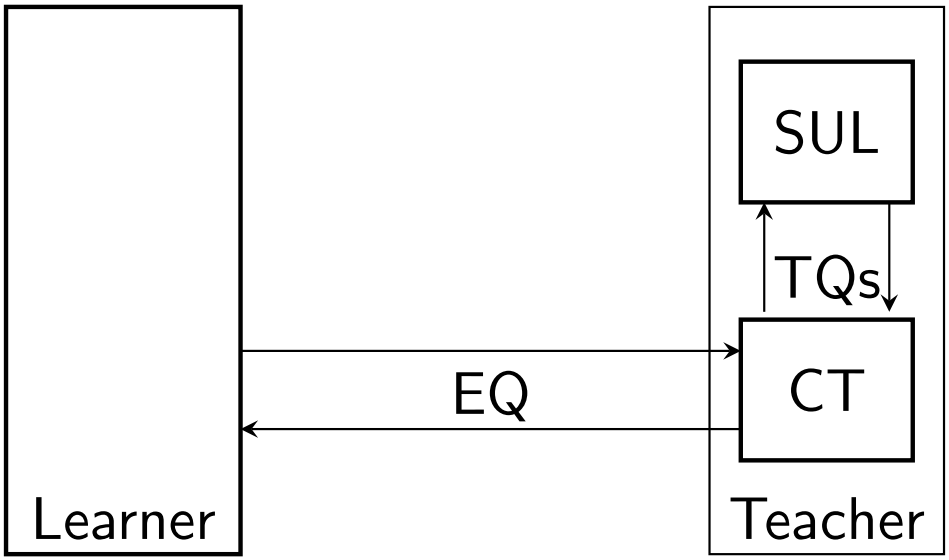


Our approach



Crazy idea?

Our approach



Crazy idea? Might work because of:

- Advances in constraint solvers
- Conformance testers not adversarial
- In applications case studies are often small

Using a Constraint Solver for Passive Learning

1. Arrange positive and negative examples in *observation tree* O,
2. Ask solver if there exists a DFA A with at most n states and a homomorphic mapping from O to A,
3. Repeat for $n = 1, 2, 3 \ldots$ until minimal DFA is found.

Question

Can our approach compete with active learning algorithms?

- Implement passive learner using Z3 SMT solver
- Compare total number of inputs needed to learn models with Angluin's L^* and state-of-the-art TTT algorithm of Isberner et al
- Use state-of-the-art conformance testing algorithm based on adaptive distinguishing sequences of Lee and Yannakakis
- Evaluate on a number of realistic benchmarks

Experimental Results

Model	States loc	Alph size	SMT			TTT		L*	
			Tests	Inputs	Time	Tests	Inputs	Tests	Inputs
Biometric passport	6	9	220	1057	26	220	941	333	1143
MAESTRO	6	14	835	4375	359	860	4437	1190	4718
MasterCard	6	14	839	4379	353	996	5260	1190	4718
PIN	6	14	757	3945	338	911	4769	1190	4718
SecureCode	4	14	313	1485	90	194	682	798	2758
VISA	9	14	796	4770	2115	750	4094	2040	9015
PCS_1	8	9	629	3530	189	417	2179	657	2682
PCS_2	3	9	71	279	9	75	196	252	657
PCS_3	7	9	508	2651	154	476	2472	576	2196
PCS_4	7	9	559	3024	154	451	2297	576	2196
PCS_5	9	9	1120	6260	778	417	1753	1308	5340
PCS_6	9	9	1158	6442	704	457	1977	1308	5340
Mealy_FIFOSet(2)	3	2	6	27	0	12	38	14	38
Mealy_FIFOSet(7)	8	2	52	481	7	71	588	235	2494
Mealy_FIFOSet(10)	11	2	179	2152	63	163	1822	486	6743
Mealy_Login(2)	6	3	37	214	7	57	242	57	219
Mealy_Login(3)	10	3	89	644	64	120	704	240	1720

Other Modelling Frameworks

We also defined SMT encodings for

- Mealy machines
- Register automata
- Input output register automata
- Learning setting without resets a la Petrenko et al

Experimental results described in paper.

Conclusions

1. Our approach to use SMT solvers for model learning is highly versatile
2. Approach does not scale well, but is able to learn small models
3. Competitive with state of the art

Future Work

1. Improve scalability via smarter encodings
2. Reduce number of queries via smarter testing
3. Explore approach for rapid prototyping new types of models, such as Mealy machines with timers
4. From query complexity to input symbol complexity: better theoretical understanding