Safe Learning: A Challenge Talk

CPS V&V I&F Workshop 2019

Kristin Yvonne Rozier
Iowa State University
December 11, 2019
The purpose of this workshop is to make academic solutions meet industrial challenges with the goal of identifying the most important present and future foundational challenges in CPS V&V (verification & validation) to ensure safe optimization and learning in autonomous CPS. Methods of interest span design and
What is Safe Learning?

Learning in an environment that is physically, emotionally, and socially safe

\(^1\)study.com
What is learning?

- adding a behavior to an automated system in response to some observed pattern of operation
  - can be performed by a person or a machine
  - can take many forms (automated, semi-automated)
- “safe learning:” learned behavior is a safe action
What is learning?

- adding a behavior to an automated system in response to some observed pattern of operation
  - can be performed by a person or a machine
  - can take many forms (automated, semi-automated)
- “safe learning:” learned behavior is a safe action

What is safe acting?

- performing an action that:
  - does not harm humans
  - may prevent harm resulting from no action
How do we know an action is safe?
How do we know an action is safe?

- Need a **proof**!
How do we know an action is safe?

- Need a **proof**!
  - Proof that the action is within a safety region?
How do we know an action is safe?

- Need a **proof**!
  - Proof that the action is within a safety region?
  - Proof that harmful actions aren’t within the behavior space?
How do we know an action is safe?

- Need a **proof**!
  - Proof that the action is within a safety region?
  - Proof that harmful actions aren’t within the behavior space?
  - ...

**We need a specification of what is safe!**
How do we know an action is safe?

- Need a **proof**!
  - Proof that the action is within a safety region?
  - Proof that harmful actions aren’t within the behavior space?
  - ... 

- Need a **specification**!
  - What are the safety requirements?
  - What are the assumed safety bounds?
  - How do we identify a violation?
How do we know an action is safe?

- Need a **proof**!
  - Proof that the action is within a safety region?
  - Proof that harmful actions aren’t within the behavior space?
  - ...

- Need a **specification**!
  - What are the safety requirements?
  - What are the assumed safety bounds?
  - How do we identify a violation?

- Need a way of checking the **implementation** follows the **proof**, generated from the **specification**
Down a Level: What is Safe Learning?

What are the inputs and outputs?
<table>
<thead>
<tr>
<th></th>
<th>Safe Learning In Six Definitions</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Learning within <strong>safety bounds</strong></td>
</tr>
</tbody>
</table>
Safe Learning In Six Definitions

1. Learning within safety bounds

2. Learning safe behaviors
Safe Learning In Six Definitions

1. Learning within **safety bounds**

2. Learning safe behaviors $\rightarrow$ learning safety requirements
Safe Learning In Six Definitions

1. Learning within safety bounds

2. Learning safe behaviors → learning safety requirements → safe behavior genesis
Safe Learning In Six Definitions

1. Learning within **safety bounds**

2. Learning safe behaviors $\rightarrow$ learning safety requirements $\rightarrow$ **safe behavior genesis**

3. **Refining behaviors** to be more safe/conservative
Safe Learning In Six Definitions

1. Learning within **safety bounds**

2. Learning safe behaviors → learning safety requirements → **safe behavior genesis**

3. **Refining behaviors** to be more safe/conservative

4. Learning that **generates verification artifacts**
Safe Learning In Six Definitions

1. Learning within **safety bounds**

2. Learning safe behaviors $\rightarrow$ learning safety requirements $\rightarrow$ **safe behavior genesis**

3. **Refining behaviors** to be more safe/conservative

4. Learning that **generates verification artifacts**
   - Learning that **passes verification tests**
Safe Learning In Six Definitions

1. Learning within \textit{safety bounds}

2. Learning safe behaviors $\rightarrow$ learning safety requirements $\rightarrow$ \textit{safe behavior genesis}

3. \textbf{Refining behaviors} to be more safe/conservative

4. Learning that \textit{generates verification artifacts}
   - Learning that \textit{passes verification tests}

5. Learning of \textit{proofs}
https://medium.com/@deepmindsafetyresearch/building-safe-artificial-intelligence-52f5f75058f1
Specification: The Biggest Bottleneck in Formal Methods and Autonomy

- Where are we now?
  - Continuously re-assess . . .
- Where will we get specifications from?
- How should we measure specification quality?
- How do we best use specifications?
- How should we organize specifications?

Motivation
Safety Bounds
Safe Behaviors
Future

Specification: The Biggest Bottleneck
in Formal Methods and Autonomy

You are here
Specifications
Completeness
Correctness
Coverage
Quality

Where are we now?
- Continuously re-assess . . .

Where will we get specifications from?

How should we measure specification quality?

How do we best use specifications?

How should we organize specifications?

... in the context of learning, autonomously acting systems?

Learning in a Safety Region

Design-time requirement

Logically follows: □ + 3 sides
Safety Bounds

- Can use logical deduction (e.g., bound by SAT/SMT)
- Can use a priori known bounds (e.g., bounded learning)

- Can we use design-time requirements?
- Can we use technological limits?
  - what we can measure
  - computational complexity
  - what we can verify

Bottleneck: Where do we get these bounds from?
Safety Bound Extraction from Learning

Post Learning: What Safety Bounds Were Learned?

- Rule extraction for Deep Neural Networks\textsuperscript{4}
- ML feature selection
- ML feature extraction\textsuperscript{5}

An Observation...

These bounds look a lot like sanity checks...
Dynamic Sanity Checking: Some Challenges

- change with different mission modes
- accommodate re-planning
- respond to unexpected environmental conditions
- allow human interaction
  - how to explain the purpose behind findings to humans
  - how to create and monitor additional sanity checks per human request
  - how to allow humans to refine definition of safety
Challenge: What Do The Bounds Look Like?

To be useful, bounds must obey patterns...

What are the patterns?

- Measurable
- Precise
- Domain-specific (in the system domain, level of abstraction, units of the action being bounded)
- Translatable: English $\iff$ System-level
- (Semi-) Automatable
- What else?
Challenge: What Do The Bounds Look Like?

To be useful, bounds must obey patterns...

What are the patterns?

- Measurable
- Precise
- Domain-specific (in the system domain, level of abstraction, units of the action being bounded)
- Translatable: English $\iff$ System-level
- (Semi-) Automatable
- What else?
Learning a Behavior

Motivation
Safety Bounds
Safe Behaviors
Future

Laboratory for Temporal Logic
Kristin Yvonne Rozier
Safe Learning Challenges

Learning a Behavior

Code

int main() {
    short int a = 1024;
    int i;
    for (i = 0; i < 10; i++)
        a *= 2;
    return a;
}

Event Trace

Build
Model

Execute

Analyse

Bug 1
Bug 2
...

Highly customized for property of interest
Learning a Behavior or Runtime Verification?

```
int main() {
    short int a = 1024;
    int i;
    for (i = 0; i < 10; i++)
    
    a *= 2;
}
return a;
```
Learning a Behavior or Runtime Verification? Simulation?


The purpose of simulation is insight \(^8\) whereas the purpose of RV is fault detection \(^9\).

---


The purpose of simulation **and learning?** is insight 10 whereas the purpose of RV is fault detection 11.

---


The Specification Bottleneck

Specification is the biggest bottleneck to RV.\textsuperscript{12}

Can learning provide RV requirements?

Learning → RV

Figure: Possible workflow for connecting the outputs of learning simulation runs to the inputs for runtime verification: if we can formalize and automate the translation of simulation output statistics to supply the requirements from which we create runtime monitors, we can mitigate the biggest bottleneck in RV.¹⁴


¹⁴
Figure: Possible workflow for filtering individual runs using RV. But how do we know what checks to run?
Safe Learning: Verification Artifacts and Proofs

How can learning algorithms generate verification inputs?
Safe Learning: Verification Artifacts and Proofs

How can learning algorithms generate verification inputs?

Can any learning algorithms generate verification artifacts?
Safe Learning: Verification Artifacts and Proofs

How can learning algorithms generate verification inputs?

Can any learning algorithms generate verification artifacts?

Can they generate explainability artifacts?
How can learning algorithms generate verification inputs?

Can any learning algorithms generate verification artifacts?

Can they generate explainability artifacts?

Can we even start to generate proofs?
Safe Learning Defined?

Doing one of more of the following, with or without automation or help from humans, driven by a specification that is checkable, with the provable result of minimizing harm to humans (through action or inaction):

1. **Learning within safety bounds**

2. **Learning safe behaviors** → learning safety requirements → **safe behavior genesis**

3. **Refining behaviors** to be more safe/conservative

4. **Learning that generates verification artifacts**
   - Learning that passes verification tests

5. **Learning of proofs**