Verification of Robotics and Autonomous Systems

Xiaowei Huang, University of Liverpool

Joint work with Prof. Marta Kwiatkowska, University of Oxford

Alpine Verification Meeting, September 19, 2017
Outline

- Challenges: Robotics and Autonomous Systems
- Verification of Deep Learning
- Verification of Human-Robot Interaction
- Conclusion
Robots and Autonomous Systems

Verifications of Robotics and Autonomous Systems

Xiaowei Huang

Challenges

Deep Learning Verification
Safety Definition Challenges
Approaches
Experimental Results

Verification in Human-Robot Interaction

Motivation
Stochastic Multiplayer Game
Cognitive Mechanism
A Temporal Logic of Trust Complexity

Conclusion

Xiaowei Huang (Liverpool University)
Robotic and autonomous systems (RAS) are interactive, cognitive and interconnected tools that perform useful tasks in the real world where we live and work.
Automated Verification, a.k.a. Model Checking

Verification of Robotics and Autonomous Systems
Xiaowei Huang

Challenges
Deep Learning Verification
Safety Definition Challenges
Approaches
Experimental Results

Verification in human-robot interaction
Motivation
Stochastic Multiplayer Game
Cognitive Mechanism
A Temporal Logic of Trust
Complexity

Conclusion
Systems for Verification: Paradigm Shifting

Verification of Robotics and Autonomous Systems

Xiaowei Huang

Challenges
Deep Learning
Verification
Safety Definition
Challenges
Approaches
Experimental Results

Verification in human-robot interaction
Motivation
Stochastic Multiplayer Game
Cognitive Mechanism
A Temporal Logic of Trust
Complexity

Conclusion

Concurrent System (1980-)
Probabilistic System (1990-)
Robotics and Autonomous System

Environment
Logical Component
Probabilistic Component

Environment
Logical Component

Environment
Probabilistic Component

Human
Deep Learning
System Properties

- dependability (or reliability)
- human values, such as trustworthiness, morality, ethics, transparency, etc
Verification of Deep Learning

- Verification of Robotics and Autonomous Systems
- Xiaowei Huang

Challenges
Deep Learning Verification
- Safety Definition
- Challenges
- Approaches
- Experimental Results

Verification in human-robot interaction
- Motivation
- Stochastic Multiplayer Game
- Cognitive Mechanism
- A Temporal Logic of Trust
- Complexity

Conclusion

Xiaowei Huang (Liverpool University)  Verification of Robotics and Autonomous Systems / 98
Verification of Robotics and Autonomous Systems

Xiaowei Huang

Challenges

Deep Learning Verification

Safety Definition

Approaches

Experimental Results

Verification in human-robot interaction

Motivation

Stochastic Multiplayer Game

Cognitive Mechanism

A Temporal Logic of Trust Complexity

Conclusion

Human-Level Intelligence
Major problems and critiques

- un-safe, e.g., instability to adversarial examples
- hard to explain to human users
Human Driving vs. Autonomous Driving

Traffic image from “The German Traffic Sign Recognition Benchmark”
Deep learning verification (DLV)

Image generated from our tool Deep Learning Verification (DLV) \(^1\)

Joshua Brown was killed when his Tesla Model S, which was operating in Autopilot mode, crashed into a tractor-trailer.

The car’s sensor system, against a bright spring sky, failed to distinguish a large white 18-wheel truck and trailer crossing the highway.
On 23 Mar 2016, Microsoft launched a new artificial intelligence chat bot that it claims will become smarter the more you talk to it.
Microsoft's new chatbot wants to hang out with millennials on Twitter

after 24 hours ...
Safety Problem: Microsoft Chatbot

Verification of Robotics and Autonomous Systems

Xiaowei Huang

Challenges
Deep Learning Verification
Safety Definition
Challenges
Approaches
Experimental Results

Verification in human-robot interaction
Motivation
Stochastic Multiplayer Game
Cognitive Mechanism
A Temporal Logic of Trust Complexity

Conclusion

Safety Problem: Microsoft Chatbot

TayTweets @TayandYou
@icydt bush did 9/11 and Hitler would have done a better job than the monkey we have now. donald trump is the only hope we've got.

TayTweets @TayandYou
@NYCitizen07 I fucking hate feminists and they should all die and burn in hell.
24/03/2016, 11:41

TayTweets @TayandYou
@ReynTheo HITLER DID NOTHING WRONG!

8:44 PM - 23 Mar 2016
Microsoft deletes 'teen girl' AI after it became a Hitler-loving sex robot within 24 hours
Deep neural networks

all implemented with

input layer  hidden layer 1  hidden layer 2  hidden layer 3  output layer
Safety Definition: Deep Neural Networks

- \( \mathbb{R}^n \) be a vector space of images (points)
- \( f : \mathbb{R}^n \rightarrow C \), where \( C \) is a (finite) set of class labels, models the human perception capability,
- a neural network classifier is a function \( \hat{f}(x) \) which approximates \( f(x) \)
A (feed-forward and deep) neural network $N$ is a tuple $(L, T, \Phi)$, where

- $L = \{L_k \mid k \in \{0, \ldots, n\}\}$: a set of layers.
- $T \subseteq L \times L$: a set of sequential connections between layers,
- $\Phi = \{\phi_k \mid k \in \{1, \ldots, n\}\}$: a set of activation functions $\phi_k : D_{L_{k-1}} \to D_{L_k}$, one for each non-input layer.
Safety Definition: Illustration
Safety Definition: Traffic Sign Example
[General Safety] Let $\eta_k(\alpha_{x,k})$ be a region in layer $L_k$ of a neural network $N$ such that $\alpha_{x,k} \in \eta_k(\alpha_{x,k})$. We say that $N$ is safe for input $x$ and region $\eta_k(\alpha_{x,k})$, written as $N, \eta_k \models x$, if for all activations $\alpha_{y,k}$ in $\eta_k(\alpha_{x,k})$ we have $\alpha_{y,n} = \alpha_{x,n}$.
Challenge 1: continuous space, i.e., there are an infinite number of points to be tested in the high-dimensional space.

Xiaowei Huang (Liverpool University) Verification of Robotics and Autonomous Systems

Alpine Verification Meeting, September 19, 2017
Challenges

Challenge 2: The spaces are high dimensional

Note: a colour image of size 32*32 has the 32*32*3 = 784 dimensions.

Note: hidden layers can have many more dimensions than input layer.
Challenges

Challenge 3: the functions $f$ and $\hat{f}$ are highly non-linear, i.e., safety risks may exist in the pockets of the spaces.

![Figure: Input Layer and First Hidden Layer](image-url)
Challenges

Challenge 4: not only heuristic search but also verification
Approach 1: Discretisation by Manipulations

Define manipulations $\delta_k : D_{L_k} \rightarrow D_{L_k}$ over the activations in the vector space of layer $k$.

**Figure:** Example of a set $\{\delta_1, \delta_2, \delta_3, \delta_4\}$ of valid manipulations in a 2-dimensional space.
ladders, bounded variation, etc

Figure: Examples of ladders in region $\eta_k(\alpha_x,k)$. Starting from $\alpha_{x,k} = \alpha_{x_0,k}$, the activations $\alpha_{x_1,k} \ldots \alpha_{x_j,k}$ form a ladder such that each consecutive activation results from some valid manipulation $\delta_k$ applied to a previous activation, and the final activation $\alpha_{x_j,k}$ is outside the region $\eta_k(\alpha_x,k)$. 
Safety wrt Manipulations

[Safety wrt Manipulations] Given a neural network $N$, an input $x$ and a set $\Delta_k$ of manipulations, we say that $N$ is safe for input $x$ with respect to the region $\eta_k$ and manipulations $\Delta_k$, written as $N, \eta_k, \Delta_k \models x$, if the region $\eta_k(\alpha_x, k)$ is a 0-variation for the set $\mathcal{L}(\eta_k(\alpha_x, k))$ of its ladders, which is complete and covering.

**Theorem**

$(\Rightarrow) N, \eta_k \models x$ (general safety) implies $N, \eta_k, \Delta_k \models x$ (safety wrt manipulations).
Define minimal manipulation as the fact that there does not exist a finer manipulation that results in a different classification.

**Theorem**

\[(\iff) \text{Given a neural network } N, \text{ an input } x, \text{ a region } \eta_k(\alpha_{x,k}) \text{ and a set } \Delta_k \text{ of manipulations, we have that } N, \eta_k, \Delta_k \models x \text{ (safety wrt manipulations) implies } N, \eta_k \models x \text{ (general safety) if the manipulations in } \Delta_k \text{ are minimal.}\]
Approach 2: Layer-by-Layer Refinement

Figure: Refinement in general safety

\[ N, \eta_0 \models i \leftarrow N, \eta_1 \models i \leftarrow N, \eta_2 \models i \leftarrow \ldots \leftarrow N, \eta_k \models i \]

Input Layer \quad Layer 1 \quad Layer 2 \quad Output Layer
Approach 2: Layer-by-Layer Refinement

Figure: Refinement in general safety and safety wrt manipulations
Approach 2: Layer-by-Layer Refinement

![Diagram of layer-by-layer refinement process](image)

**Figure:** Complete refinement in general safety and safety wrt manipulations
Approach 3: Exhaustive Search

Figure: exhaustive search (verification) vs. heuristic search
Approach 4: Feature Discovery

Natural data, for example natural images and sound, forms a high-dimensional manifold, which embeds tangled manifolds to represent their features.

Feature manifolds usually have lower dimension than the data manifold, and a classification algorithm is to separate a set of tangled manifolds.
Approach 4: Feature Discovery

the appearance of features is independent

we can manipulate them one by one

reduce the problem of size $O(2^{d_1} + \ldots + d_m)$ into a set of smaller problems of size $O(2^{d_1}), \ldots, O(2^{d_m})$. 
Experimental Results: MNIST

Image Classification Network for the MNIST Handwritten Numbers 0 – 9

Total params: 600,810
Experimental Results: MNIST
Experimental Results: GTSRB

Image Classification Network for The German Traffic Sign Recognition Benchmark

Total params: 571,723
Experimental Results: GTSRB

“stop” to “30m speed limit”

“80m speed limit” to “30m speed limit”

“go right” to “go straight”
Experimental Results: GTSRB

- no overtaking (prohibitory) to go straight (mandatory)
- speed limit 50 (prohibitory) to stop (other)
- road narrows (danger) to construction (danger)
- restriction ends 80 (other) to speed limit 80 (prohibitory)
- no overtaking (trucks) (prohibitory) to speed limit 80 (prohibitory)
- no overtaking (prohibitory) to restriction ends (overtaking (trucks)) (other)
- priority at next intersection (danger) to speed limit 30 (prohibitory)
- uneven road (danger) to traffic signal (danger)
- danger (danger) to school crossing (danger)
Image Classification Network for the CIFAR-10 small images

Total params: 1,250,858
Experimental Results: CIFAR-10
Experimental Results: imageNet

Image Classification Network for the ImageNet dataset, a large visual database designed for use in visual object recognition software research.

Total params: 138,357,544
Experimental Results: ImageNet

labrador to life boat

rhodesian ridgeback to malinois

boxer to rhodesian ridgeback

great pyrenees to kuvasz
Next Step: Hybrid Systems

Verification of Robotics and Autonomous Systems

Xiaowei Huang

Challenges
Deep Learning
Verification
Safety Definition
Challenges
Approaches
Experimental Results

Verification in human-robot interaction
Motivation
Stochastic Multiplayer Game
Cognitive Mechanism
A Temporal Logic of Trust Complexity

Conclusion
Verification in human-robot interaction


Verification in human-robot interaction

Motivation
Stochastic Multiplayer Game
Cognitive Mechanism
A Temporal Logic of Trust Complexity

Conclusion
Mental process in human model

- Environment
  - AI, including logical component, probabilistic component, deep learning
  - Mental Module
  - Visual Module
  - Manual Module
  - Human

Verification of Robotics and Autonomous Systems
Xiaowei Huang
Challenges
Deep Learning Verification
Safety Definition
Challenges
Approaches
Experimental Results
Verification in human-robot interaction
Motivation
Stochastic Multiplayer Game
Cognitive Mechanism
A Temporal Logic of Trust Complexity
Conclusion

Xiaowei Huang (Liverpool University)  Verification of Robotics and Autonomous Systems
Social trust in human-robot interaction

Trust, one of the essential human mental attitude, is a critical part of every human interaction.
Social trust in human-robot interaction

Question: what is the level of trust we have on a self-driving car to send our kids to the school?

Question: what is the level of trust we have on a self-driving car to let it make decision in a critical situation?
Joshua Brown was killed when his Tesla Model S, which was operating in Autopilot mode, crashed into a tractor-trailer. He was allegedly watching a movie when the incident occurs.
'Our car was making an assumption about what the other car was going to do,' said Chris Urmson, head of Google's self-driving project, speaking at the SXSW festival in Austin.
Definition of social trust

What is (social) trust?

- The **willingness** of a party to be **vulnerable** to the actions of another party based on the **expectation** that the other will perform a particular action important to the trustor, irrespective of the **ability** to monitor or control that party. [Mayer, Davis, and Schoorman 1995]
- A **subjective** evaluation of a **trustee** on a **trustor** about something in particular, e.g., the completion of a **task**. [Hardin 2002]
- ...
A stochastic multiplayer game (SMG) is a tuple $\mathcal{M} = (\text{Ags}, \mathcal{S}, \mathcal{S}_{\text{init}}, \{\text{Act}_A\}_{A \in \text{Ags}}, \mathcal{T}, \mathcal{L})$, where:

- $\text{Ags} = \{1, \ldots, n\}$ is a finite set of agents,
- $\mathcal{S}$ is a finite set of states,
- $\mathcal{S}_{\text{init}} \subseteq \mathcal{S}$ is a set of initial states,
- $\text{Act}_A$ is a finite set of actions for the agent $A$,
- $\mathcal{T} : \mathcal{S} \times \text{Act} \rightarrow \mathcal{D}(\mathcal{S})$ is a (partial) probabilistic transition function, where $\text{Act} = \times_{A \in \text{Ags}} \text{Act}_A$ and
- $\mathcal{L} : \mathcal{S} \rightarrow \mathcal{P}(\mathcal{AP})$ is a labelling function mapping each state to a set of atomic propositions taken from a set $\mathcal{AP}$. 
A (history-dependent and stochastic) *action strategy* \( \sigma_A \) of agent \( A \in \text{Ags} \) in an SMG \( \mathcal{M} \) is a function \( \sigma_A : \text{FPath}^{\mathcal{M}} \rightarrow \mathcal{D}(\text{Act}_A) \), such that for all \( a_A \in \text{Act}_A \) and finite paths \( \rho \) it holds that \( \sigma_A(\rho)(a_A) > 0 \) only if \( a_A \in \text{Act}_A(\text{last}(\rho)) \).

A strategy profile \( \sigma_C \) for a set \( C \) of agents is a vector of action strategies \( \times_{A \in C} \sigma_A \), one for each agent \( A \in C \).

We let \( \Pi_A \) be the set of agent \( A \)'s strategies, \( \Pi_C \) be the set of strategy profiles for the set of agents \( C \), and \( \Pi \) be the set of strategy profiles for all agents.
Strategy Induced DTMC

Given a path $\rho s$ which has $s$ as its last state, a strategy $\sigma \in \Pi$, and a formula $\psi$, we write

$$\text{Prob}_{\mathcal{M}, \sigma, \rho s}(\psi) \overset{\text{def}}{=} \Pr_{\sigma}^{\mathcal{M}} \{ \delta \in \text{IPath}_{\mathcal{T}}^{\mathcal{M}}(s) \mid \mathcal{M}, \rho s, \delta \models \psi \}$$

for the probability of implementing $\psi$ on a path $\rho s$ when a strategy $\sigma$ applies. Based on this, we define

$$\text{Prob}_{\mathcal{M}, \rho}^{\text{min}}(\psi) \overset{\text{def}}{=} \inf_{\sigma \in \Pi} \text{Prob}_{\mathcal{M}, \sigma, \rho}(\psi),$$

$$\text{Prob}_{\mathcal{M}, \rho}^{\text{max}}(\psi) \overset{\text{def}}{=} \sup_{\sigma \in \Pi} \text{Prob}_{\mathcal{M}, \sigma, \rho}(\psi).$$
Semantics of Probabilistic Formula

\[ \mathcal{M}, \rho \models p \boxdot q \psi \text{ if } \text{Prob}_{\mathcal{M}, \rho}^{opt(\boxdot)}(\psi) \boxdot q, \text{ where} \]

\[
\text{opt}(\boxdot) = \begin{cases} 
\min & \text{when } \boxdot \in \{\geq, >\} \\
\max & \text{when } \boxdot \in \{\leq, <\}
\end{cases}
\]
A partially observable stochastic multiplayer game (POSMG) is a tuple \( \mathcal{M} = (\text{Ags}, S, S_{\text{init}}, \{\text{Act}_A\}_{A \in \text{Ags}}, T, L, \{O_A\}_{A \in \text{Ags}}, \{\text{obs}_A\}_{A \in \text{Ags}}) \), where

- \( (\text{Ags}, S, S_{\text{init}}, \{\text{Act}_A\}_{A \in \text{Ags}}, T, L) \) is an SMG,
- \( O_A \) is a finite set of observations for agent \( A \), and
- \( \text{obs}_A : S \rightarrow O_A \) is a labelling of states with observations for agent \( A \).
Cognitive Mechanism

Stochastic multiplayer game with cognitive dimension (SMG_Ω) extends POSMG with

- cognitive state,
- cognitive mechanism, and
- cognitive strategy.

For an agent A, we use Goal_A to denote its set of goals and Int_A to denote its set of intentions.
A stochastic multiplayer game with cognitive dimension (SMG$_\Omega$) is a tuple $\mathcal{M} = (\text{Ags}, S, S_{\text{init}}, \{\text{Act}_A\}_A \in \text{Ags}, T, L, \{O_A\}_A \in \text{Ags}, \{\text{obs}_A\}_A \in \text{Ags}, \{\Omega_A\}_A \in \text{Ags}, \{\pi_A\}_A \in \text{Ags})$, where

- $\Omega_A = \langle \text{Goal}_A, \text{Int}_A \rangle$ is the **cognitive mechanism** of agent $A$, consisting of
  - a legal goal function $\text{Goal}_A : S \to \mathcal{P}(\mathcal{P}(\text{Goal}_A))$ and
  - a legal intention function $\text{Int}_A : S \to \mathcal{P}(\text{Int}_A)$, and

- $\pi_A = \langle \pi^g_A, \pi^i_A \rangle$ is the **cognitive strategy** of agent $A$, consisting of
  - a goal strategy $\pi^g_A : \text{FPath}^\mathcal{M} \to \mathcal{D}(\mathcal{P}(\text{Goal}_A))$ and
  - an intention strategy $\pi^i_A : \text{FPath}^\mathcal{M} \to \mathcal{D}(\text{Int}_A)$.
In addition to the temporal dimension of transitions \( s \rightarrow^A s' \), we also distinguish a *cognitive* dimension of transitions \( s \rightarrow^C s' \), which corresponds to mental reasoning processes.

- Given a state \( s \) and a set of agent \( A \)'s goals \( x \subseteq \text{Goal}_A \), we write \( A.g(s, x) \) for the state obtained from \( s \) by substituting agent’s goals with \( x \). Similar notation \( A.i(s, x) \) is used for intention change when \( x \in \text{Int}_A \).

- Alternatively, we may write \( s \rightarrow^A g \cdot x \) if \( s' = A.g(s, x) \) contains the goal set \( x \) for \( A \) and \( s \rightarrow^A i \cdot x \) if \( s' = A.i(s, x) \) contains the intention \( x \) for \( A \).
Running Example: Trust Game

A simple trust game from [Kuipers2016], in which there are two agents, Alice and Bob. At the beginning, Alice has 10 dollars and Bob has 5 dollars. If Alice does nothing, then everyone keeps what they have. If Alice invests her money with Bob, then Bob can turn the 15 dollars into 40 dollars. After having the investment yield, Bob can decide whether to share the 40 dollars with Alice. If so, each will have 20 dollars. Otherwise, Alice will lose her money and Bob gets 40 dollars.
Running Example: Trust Game

Table: Payoff of a simple trust game

<table>
<thead>
<tr>
<th></th>
<th>Bob</th>
<th>Alice</th>
</tr>
</thead>
<tbody>
<tr>
<td>share</td>
<td>(20,20)</td>
<td>(0,40)</td>
</tr>
<tr>
<td>keep</td>
<td>(10,5)</td>
<td>(10,5)</td>
</tr>
</tbody>
</table>

Motivation

Stochastic Multiplayer Game

Cognitive Mechanism

A Temporal Logic of Trust Complexity

Conclusion
It is argued that the single numerical value as the payoff of the trust game is an over-simplification. A more realistic utility should include both the payoff and other hypotheses, including trust.

<table>
<thead>
<tr>
<th></th>
<th>Bob</th>
<th>share</th>
<th>keep</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alice</td>
<td>invest</td>
<td>(20,20+5)</td>
<td>(0,40-20)</td>
</tr>
<tr>
<td></td>
<td>withhold</td>
<td>(10,5)</td>
<td>(10,5)</td>
</tr>
</tbody>
</table>
Trust Game: Cognitive Modelling

For Alice, we let

- $\text{Goal}_{Alice} = \{\text{passive}, \text{active}\}$ be two goals which represent her attitude towards investment.
- $\text{Int}_{Alice} = \{\text{passive}, \text{active}\}$, and
- strategy $\sigma_{\text{passive}}$ to implement her passive intention, and $\sigma_{\text{active}}$ to implement her active intention.

<table>
<thead>
<tr>
<th>action</th>
<th>withhold</th>
<th>invest</th>
<th>keep</th>
<th>share</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\sigma_{\text{passive}}$</td>
<td>0.7</td>
<td>0.3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\sigma_{\text{active}}$</td>
<td>0.1</td>
<td>0.9</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table: Strategies for Alice
For Bob, we let

- \( \text{Goal}_{Bob} = \{ \text{investor}, \text{opportunist} \} \) be a set of goals,
- \( \text{Int}_{Bob} = \{ \text{share}, \text{keep} \} \), and
- let his intentions be associated with action strategies: \( \sigma_{\text{share}} \), in which Bob shares the investment yield with Alice, and \( \sigma_{\text{keep}} \), in which Bob keeps all the money for himself.

<table>
<thead>
<tr>
<th>Strategy</th>
<th>Withhold</th>
<th>Invest</th>
<th>Keep</th>
<th>Share</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \sigma_{\text{share}} )</td>
<td>0.0</td>
<td>1.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>( \sigma_{\text{keep}} )</td>
<td>1.0</td>
<td>0.0</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Table: Strategies for Bob**
We extend the trust game $\mathcal{G}$ by expanding state to additionally include cognitive state. In particular, each state can now be represented as a tuple

$$(a_{Alice}, a_{Bob}, gs_{Alice}, gs_{Bob}, is_{Alice}, is_{Bob}),$$

such that $a_{Alice}$ and $a_{Bob}$ are last actions executed by agents and $gs_{Alice} \subseteq Goal_{Alice} \cup \{\perp\}$, $gs_{Bob} \subseteq Goal_{Bob} \cup \{\perp\}$, $is_{Alice} \in Int_{Alice} \cup \{\perp\}$, and $is_{Bob} \in Int_{Bob} \cup \{\perp\}$ is the cognitive state.
Trust Game: Cognitive Modelling

Fig. 2. Trust game with cognitive dimension
Assumptions

- (Uniformity Assumption) ...

- (Deterministic Behaviour Assumption) An $\text{SMG}_\Omega \ M$ satisfies the *Deterministic Behaviour Assumption* if each agent’s cognitive state deterministically decides its behaviour in terms of selection of its next local action. In other words, agent’s cognitive state induces a pure action strategy that agent follows.
Cognitive Modalities

The syntax of the logic, named PCTL$^*$, is as follows.

$$\phi ::= p \mid \neg \phi \mid \phi \lor \phi \mid \forall \psi \mid p^\bowtie q \psi \mid G_A \phi \mid I_A \phi \mid C_A \phi$$

$$\psi ::= \phi \mid \neg \psi \mid \psi \lor \psi \mid \diamond \psi \mid \psi U \psi$$

where $p \in AP$, $A \in Ags$, $\bowtie \in \{<, \leq, >, \geq\}$, and $q \in [0, 1]$.

- $M, \rho s \models G_A \phi$ if $\forall x \in supp(\pi^g_A(\rho s)) \exists s' : s \xrightarrow{A.g.x} s'$ and $M, \rho ss' \models \phi$,

- $M, \rho s \models I_A \phi$ if $\forall x \in supp(\pi^i_A(\rho s)) \exists s' \in S : s \xrightarrow{A.i.x} s'$ and $M, \rho ss' \models \phi$,

- $M, \rho s \models C_A \phi$ if $\exists x \in Int_A(s) \exists s' \in S : s \xrightarrow{A.i.x} s'$ and $M, \rho ss' \models \phi$. 
Example Formulas

- $\phi_1 = \Box_{Alice} P \leq 0.9 \diamond a_{Alice} = \text{invest}$ expresses that regardless of Alice changing her goals, the probability of her investing in the future is no greater than 90%.

- $\phi_2 = \Box_{Bob} P \leq 0 \circ a_{Bob} = \text{keep}$ states that Bob has a legal intention which ensures that he will not keep the money as his next action.

- $\phi_3 = \exists_{Alice} \exists_{Bob} \diamond \text{richer}_{Alice,Bob}$, where $\text{richer}_{Alice,Bob}$ is an atomic proposition with obvious meaning, states that Alice can find an intention such that it is possible to eventually reach a state where Alice has more money than Bob. Finally, the formula

- $\phi_4 = \exists_{Alice} \exists_{Bob} \diamond \Box_{Bob} \forall \diamond \neg \text{richer}_{Alice,Bob}$ expresses that Alice can find an intention such that it is possible to reach a state such that, for all possible Bob's goals, the game will always reach a state in which Bob is no poorer than Alice.
Trust Game: Cognitive Modelling

Fig. 2. Trust game with cognitive dimension
An autonomous stochastic multi-agent system (ASMAS) is a tuple \( \mathcal{M} = (\text{Ags}, S, S_{\text{init}}, \{\text{Act}_A\}_A \in \text{Ags}, T, L, \{\mathcal{O}_A\}_A \in \text{Ags}, \{\text{obs}_A\}_A \in \text{Ags}, \{\Omega_A\}_A \in \text{Ags}, \{\pi_A\}_A \in \text{Ags}, \{p_A\}_A \in \text{Ags}) \), where \( p_A \) is a set of preference functions of agent \( A \in \text{Ags} \), defined as

\[
p_A \overset{\text{def}}{=} \{gp_{A,B}, ip_{A,B} \mid B \in \text{Ags} \text{ and } B \neq A\},
\]

where:

- \( gp_{A,B} : S \to \mathcal{D}(\mathcal{P}(\text{Goal}_B)) \) is a goal preference function of \( A \) over \( B \) such that, for any state \( s \) and \( x \in \mathcal{P}(\text{Goal}_B) \), we have \( gp_{A,B}(s)(x) > 0 \) only if \( x \in \text{Goal}_B(s) \), and

- \( ip_{A,B} : S \to \mathcal{D}(\text{Int}_B) \) is an intention preference function of \( A \) over \( B \) such that, for any state \( s \) and \( x \in \text{Int}_B \), we have \( ip_{A,B}(s)(x) > 0 \) only if \( x \in \text{Int}_B(s) \).
Trust Game: Preference-induced DTMC

- **s0**: Initial state
- **s1, s2**: Active and passive states
- **s3, s4**: Investor and opportunist states
- **s5, s6**: Investor and opportunist states
- **s7, s8, s9, s10, s11, s12, s13, s14, s15, s16, s17, s18, s19, s20, s21, s22, s23, s24, s25, s26, s27, s28, s29, s30, s31, s32, s33, s34, s35, s36, s37, s38**: Various states with transitions

Transition labels:
- **w**: Weight
- **i**: Information
- **B.i.σ**: Behavior
- **A.g**:
  - **{passive}**: Passive action
  - **{active}**: Active action

Example transitions:
- **s7** to **s8**: w: 0.7, i: 0.3
- **s8** to **s9**: B.i.σ_share
- **s9** to **s10**: B.i.σ_keep, B.i.σ_share
- **s10** to **s11**: B.i.σ_keep
- **s11** to **s12**: B.i.σ_share
- **s12** to **s13**: B.i.σ_keep
- **s13** to **s14**: B.i.σ_share, B.i.σ_keep
- **s14** to **s15**: B.i.σ_share
- **s15** to **s16**: k:0, s:1
- **s16** to **s17**: k:1
- **s17** to **s18**: s:0
- **s18** to **s19**: s:1
- **s19** to **s20**: k:0
- **s20** to **s21**: s:0
- **s21** to **s22**: k:1
- **s22** to **s23**: s:0
Trust Game: Preference-induced DTMC

\[ gp_{Bob,Alice}(s_0) = \langle \text{passive} \mapsto 1/3, \text{active} \mapsto 2/3 \rangle \]

indicates that Bob believes Alice is more likely to be active than passive. Setting

\[ gp_{Alice,Bob}(s_x) = \langle \text{investor} \mapsto 1/2, \text{opportunist} \mapsto 1/2 \rangle, \]

for \( x \in \{1, 2\} \), represents that Alice has no prior knowledge regarding Bob’s mental attitudes. We may set

\[ ip_{Alice,Bob}(s_x) = \langle \text{share} \mapsto 3/4, \text{keep} \mapsto 1/4 \rangle \quad \text{for} \ x \in \{8, 12\}, \]

\[ ip_{Alice,Bob}(s_x) = \langle \text{share} \mapsto 0, \text{keep} \mapsto 1 \rangle \quad \text{for} \ x \in \{10, 14\} \]

to indicate that Alice knows that Bob will keep the money when he is an opportunist, but she thinks it’s quite likely that he will share his profit when he is an investor.
Trust Game: Preference-induced DTMC

\[
\Pr_{Alice}(\rho_1) = \text{gp}_{Alice, Bob}(s_1)(\text{investor}) \\
\cdot (\sigma_{\text{passive}}(s_0 s_1 s_3)(\text{invest}) \cdot T(s_3, \text{invest})(s_8)) \\
\cdot ip_{Alice, Bob}(s_8)(\text{share}) \\
\cdot (\sigma_{\text{share}}(s_0 s_1 s_3 s_8 s_{15})(\text{share}) \cdot T(s_{15}, \text{share})(s_{24})) \\
= \frac{1}{2} \cdot \left( \frac{3}{10} \cdot 1 \right) \cdot \frac{3}{4} \cdot (1 \cdot 1) = \frac{9}{80},
\]
The belief function $\text{be}_A : OPath_A \rightarrow D(FPath^M)$ is given by

$$\text{be}_A(o)(\rho) = \text{Pr}^M_A(C_\rho | \bigcup_{\rho' \in \text{class}(o)} C_{\rho'}).$$
Trust Game: Belief Computation

Verification of Robotics and Autonomous Systems

Xiaowei Huang

Challenges

Deep Learning Verification
Safety Definition Challenges
Approaches
Experimental Results

Verification in human-robot interaction
Motivation
Stochastic Multiplayer Game
Cognitive Mechanism
A Temporal Logic of Trust Complexity

Conclusion

Xiaowei Huang (Liverpool University)
Trust Game: Belief Computation

\[
\begin{align*}
\text{be}_{Bob}(o, \rho_1) &= \Pr^G_{Bob}(C_{\rho_1} | \bigcup_{\rho \in \text{class}(o)} C_{\rho}) \\
&= \frac{\Pr^G_{Bob}(C_{\rho_1})}{\Pr^G_{Bob}(C_{\rho_1}) + \Pr^G_{Bob}(C_{\rho_2})} \\
&= \frac{gp_{Bob,Alice}(s_0)(\text{passive})}{gp_{Bob,Alice}(s_0)(\text{passive}) + gp_{Bob,Alice}(s_0)(\text{active})} \\
&= \frac{1}{3}.
\end{align*}
\]
Trust: A Temporal Logic of Trust

The syntax of the logic PRTL* is as follows.

\[ \phi ::= p | \neg \phi | \phi \lor \phi | \forall \psi | P^{q} \psi | G_{A} \phi | I_{A} \phi | C_{A} \phi | B_{A}^{q} \psi | CT_{A,B}^{q} \psi | DT_{A,B}^{q} \psi \]

\[ \psi ::= \phi | \neg \psi | \psi \lor \psi | \circ \psi | \psi U \psi | \square \psi \]

where \( p \in AP, A, B \in Ags, \trianglerighteq \in \{<, \leq, >, \geq\} \), and \( q \in [0,1] \).

\[ ^{2}X. \text{Huang and M. Kwiatkowska. Reasoning about cognitive trust in stochastic multiagent systems. AAAI-2017.} \]
Reasoning framework PRTL* 

\( \mathbb{B}_A^{q}\psi \), belief formula, expresses that agent \( A \) believes \( \psi \) with probability in relation \( \bowtie \) with \( q \).

\( \mathbb{C}_{T_{A,B}}^{q}\psi \), competence trust formula, expresses that agent \( A \) trusts agent \( B \) with probability in relation \( \bowtie \) with \( q \) on its capability of completing the task \( \psi \).

\( \mathbb{D}_{T_{A,B}}^{q}\psi \), disposition trust formula, expresses that agent \( A \) trusts agent \( B \) with probability in relation \( \bowtie \) with \( q \) on its willingness to do the task \( \psi \), where the state of willingness is interpreted as unavoidably taking an intention.
We write

\[
\begin{align*}
\Pr_{\mathcal{M},A,\rho}^{\max,\min}(\psi) & \overset{\text{def}}{=} \sup_{\sigma A \in \Pi_A} \inf_{\sigma_{Ags}\{A\} \in \Pi_{Ags}\{A\}} \Pr_{\mathcal{M},\sigma,\rho}(\psi), \\
\Pr_{\mathcal{M},A,\rho}^{\min,\max}(\psi) & \overset{\text{def}}{=} \inf_{\sigma A \in \Pi_A} \sup_{\sigma_{Ags}\{A\} \in \Pi_{Ags}\{A\}} \Pr_{\mathcal{M},\sigma,\rho}(\psi)
\end{align*}
\]

to denote the strategic ability of agent \( A \) in implementing \( \psi \) on a finite path \( \rho \). Intuitively,

- \( \Pr_{\mathcal{M},A,\rho}^{\max,\min}(\psi) \) gives a lower bound on agent \( A \)'s ability to maximise probability of \( \psi \), while
- \( \Pr_{\mathcal{M},A,\rho}^{\min,\max}(\psi) \) gives an upper bound on agent \( A \)'s ability to minimise probability of \( \psi \).
For a measurable function $f : \text{FPath}^\mathcal{M} \rightarrow [0, 1]$, we denote by $E_{\text{be}_A}[f]$ the belief-weighted expectation of $f$, i.e.,

$$E_{\text{be}_A}[f] = \sum_{\rho \in \text{FPath}^\mathcal{M}} \text{be}_A(\rho) \cdot f(\rho).$$
Semantics

\[ M, \rho \models B_A^\otimes q \psi \text{ if } \]

\[ E_{be_A}[V_{B,M,\psi}] \otimes q, \]

where the function \( V_{B,M,\psi} : \text{FPath}^M \to [0, 1] \) is such that

\[
V_{B,M,\psi}(\rho') = \begin{cases} 
\Pr_{M,A,\rho'}^{\max,\min}(\psi) & \text{if } \otimes \in \{\geq, >\} \\
\Pr_{M,A,\rho'}^{\min,\max}(\psi) & \text{if } \otimes \in \{<, \leq\} 
\end{cases}
\]
Semantics

\[ \mathcal{M}, \rho \models CT_{A,B}^q \psi \text{ if } \]

\[ E_{beA}[V_{CT,M,B,\psi}^\sqsubseteq] \ni q, \]

where the function \( V_{CT,M,B,\psi}^\sqsubseteq : FPath^\mathcal{M} \rightarrow [0, 1] \) is such that \( V_{CT,M,B,\psi}^\sqsubseteq(\rho') = \)

\[
\begin{cases}
  \sup_{x \in Int_B(last(\rho'))} \Pr_{M,A,B,i(\rho',x)}^{\max,\min}(\psi) & \text{if } \sqsubseteq \in \{\geq, >\} \\
  \inf_{x \in Int_B(last(\rho'))} \Pr_{M,A,B,i(\rho',x)}^{\min,\max}(\psi) & \text{if } \sqsubseteq \in \{<, \leq\}
\end{cases}
\]
Semantics

\[ \mathcal{M}, \rho \models \Delta T_{A,B}^{\psi} \text{ if } E_{be_A}[V_{\Delta T, \mathcal{M}, B, \psi}] \otimes q, \]

where the function \( V_{\Delta T, \mathcal{M}, B, \psi} : FPath^{\mathcal{M}} \rightarrow [0, 1] \) is such that

\[
\begin{cases}
\inf \, \Pr^{\max, \min}_{\mathcal{M}, A, B, i(\rho', x)}(\psi) & \text{if } \otimes \in \{\geq, >\} \\
\sup \, \Pr^{\min, \max}_{\mathcal{M}, A, B, i(\rho', x)}(\psi) & \text{if } \otimes \in \{<, \leq\}
\end{cases}
\]

\[
\text{if } \otimes \in \{\geq, >\}
\]

\[
\text{if } \otimes \in \{<, \leq\}
\]
Example Formulas

The formula

\[ \mathcal{D}_{Alice,Bob} T \geq 0.9 \diamond (a_{Bob} = \text{keep}) \]

states that Alice can trust Bob with probability no less than 0.9 that he will keep the money for himself. The formula

\[ \square (richer_{Bob,Alice} \rightarrow p \geq 0.9 \diamond C_{Bob,Alice} T \geq 1.0 \diamond richer_{Alice,Bob}) \]

states that, at any point of the game, if Bob is richer than Alice, then with probability at least 0.9, in future, he can almost surely, i.e., with probability 1, trust Alice on her capability of becoming richer than Bob.
Guarding Mechanism

For every agent $A \in \text{Ags}$, we define:

- a goal guard function $\lambda^g_A : \mathcal{P}(\text{Goal}_A) \rightarrow \mathcal{L}_A(\text{PRTL}^*)$ and
- an intention guard function $\lambda^i_A : \text{Int}_A \times \mathcal{P}(\text{Goal}_A) \rightarrow \mathcal{L}_A(\text{PRTL}^*)$.

where $\mathcal{L}_A(\text{PRTL}^*)$ is the set of formulas of the language PRTL* that are boolean combinations of atomic propositions and formulas of the form $\Box_A^q \psi$, $\Diamond_A^q \psi$, $\Box_A ? \psi$ or $\Diamond_A ? \psi$, such that $\psi$ does not contain temporal operators.

Let $\Lambda = \{ \langle \lambda^g_A, \lambda^i_A \rangle \}_{A \in \text{Ags}}$ be the guarding mechanism.
Pro-Attitude Synthesis

Obtaining cognitive strategy $\Pi = \{\pi^g_A, \pi^i_A\}_{A \in Ags}$ from finite structures $\Omega = \{\langle \text{Goal}_A, \text{Int}_A \rangle\}_{A \in Ags}$ and $\Lambda$
Trust Game

We recall our informal assumption that Bob’s intention will be *share* when he is an investor and his belief in Alice being active is sufficient, and *keep* otherwise. We formalise it as follows:

\[
\lambda^i_{\text{Bob}}(\text{share}, \{\text{investor}\}) = \mathbb{B}_{\text{Bob}}^{>0.7} \text{active}_{\text{Alice}},
\]

\[
\lambda^i_{\text{Bob}}(\text{keep}, \{\text{investor}\}) = \neg \mathbb{B}_{\text{Bob}}^{>0.7} \text{active}_{\text{Alice}},
\]

\[
\lambda^i_{\text{Bob}}(\text{share}, \{\text{opportunist}\}) = \bot,
\]

\[
\lambda^i_{\text{Bob}}(\text{keep}, \{\text{opportunist}\}) = \top,
\]

where \text{active}_{\text{Alice}} holds in states in which Alice’s goal is *active* and we used a value 0.7 to represent Bob’s belief threshold.
We let $\rho_1 = s_0s_1s_3s_8$ and $\rho_2 = s_0s_2s_5s_12$. Recall that $\text{obs}_{Bob}(\rho_1) = \text{obs}_{Bob}(\rho_2)$ and we let $o_1$ denote the observation.

$\text{be}_{Bob}(o_1, \rho_1) = 1/7,$  

$\text{be}_{Bob}(o_1, \rho_2) = 6/7.$
Therefore, since $G, \rho_1 \models \neg active_{Alice}$ and $G, \rho_2 \models active_{Alice}$ (below and in what follows, $j \in \{1, 2\}$):

$$G, \rho_j \models B^{6/7}_{Bob} active_{Alice}.$$ 

Hence

$$eval^i_{Bob}(share, \{investor\})(\rho_j) = 1,$$

$$eval^i_{Bob}(keep, \{investor\})(\rho_j) = 0,$$

and so:

$$\pi^i_{Bob}(\rho_j)(share) = 1, \quad \pi^i_{Bob}(\rho_j)(keep) = 0.$$
Model Checking Complexity

- general problem is undecidable
- A few fragments have been identified to be decidable in e.g., PSPACE, EXPTIME, or PTIME
Trust-Enhanced AI

Traditional AI:

- obs & reward
- action
- simple interaction

Trust-enhanced AI:

- obs & reward
- action
- enhanced interaction

Human

AI

Environment

Trust on AI

Trust on Human

Motivation
Stochastic Multiplayer Game
Cognitive Mechanism
A Temporal Logic of Trust Complexity

Verification of Robotics and Autonomous Systems

Xiaowei Huang

Challenges
Deep Learning Verification
Safety Definition Challenges
Approaches
Experimental Results

Verification in human-robot interaction

Conclusion

Human-like AI: enhance AI with mental module (e.g., a trust mechanism) to learn and reason about human’s values (e.g., trustworthiness, morality, ethics, etc.)
Conclusion

Verification of Robotics and Autonomous Systems

Xiaowei Huang

Challenges
Deep Learning Verification
Safety Definition Challenges
Approaches
Experimental Results

Verification in human-robot interaction
Motivation
Stochastic Multiplayer Game
Cognitive Mechanism
A Temporal Logic of Trust Complexity

Conclusion

Xiaowei Huang (Liverpool University)
Verification of Robotics and Autonomous Systems

Xiaowei Huang

Challenges
Deep Learning Verification
Safety Definition Challenges
Approaches
Experimental Results

Verification in human-robot interaction
Motivation
Stochastic Multiplayer Game
Cognitive Mechanism
A Temporal Logic of Trust Complexity

Conclusion

Xiaowei Huang (Liverpool University)