Verification of Robotics and Autonomous Systems

Xiaowei Huang, University of Liverpool

Joint work with Prof. Marta Kwiatkowska, University of Oxford

Alpine Verification Meeting, November 25, 2017

Outline

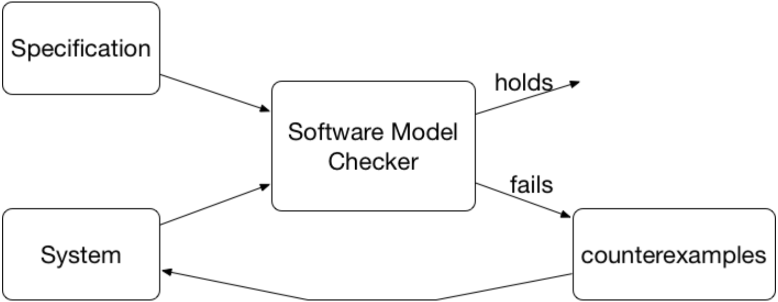
- Challenges: Robotics and Autonomous Systems
- Verification of Deep Learning [1]
- Verification of Human-Robot Interaction [?]
- Conclusion

Robotics and Autonomous Systems

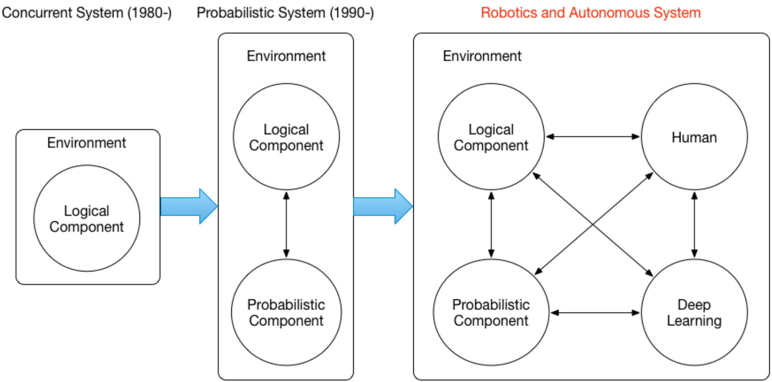
Robotics and Autonomous Systems

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



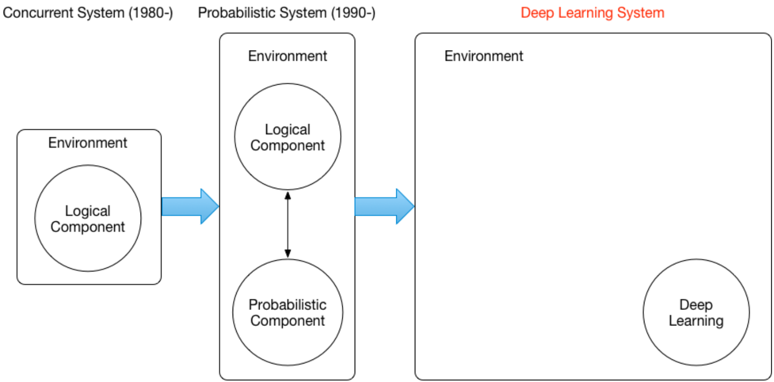
Systems for Verification: Paradigm Shifting



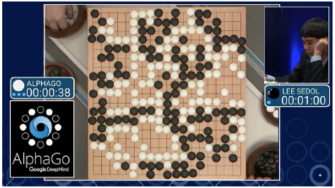
System Properties

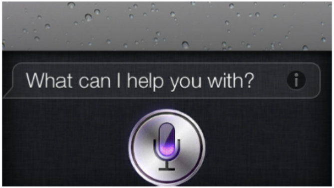
- dependability (or reliability)
- human values, such as trustworthiness, morality, ethics, transparency, etc

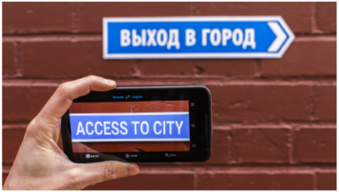
Verification of Deep Learning



Human-Level Intelligence



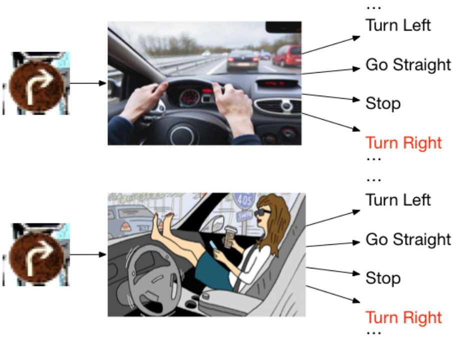




Major problems and critiques

- 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)

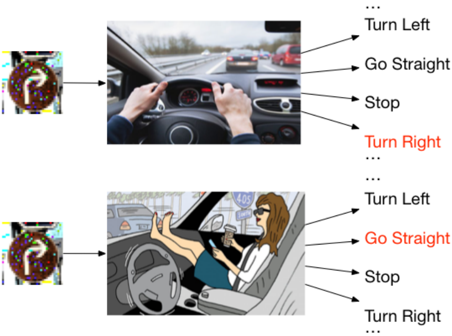


Image generated from our tool Deep Learning Verification (DLV) [1]

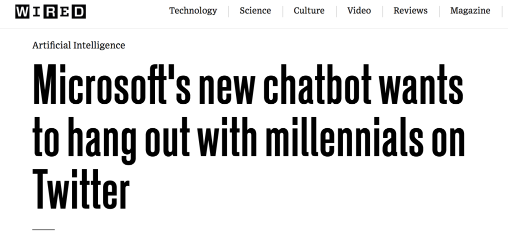
[1] X. Huang and M. Kwiatkowska. *Safety verification of deep neural networks*. CAV-2017.

Safety Problem: Tesla incident

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

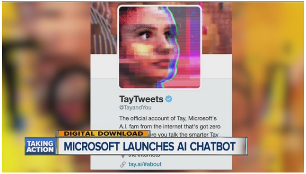
The car's sensor system, against a bright spring sky, failed to distinguish a large white 18-wheel truck and trailer crossing the highway.

Microsoft Chatbot


WIRED
Technology | Science | Culture | Video | Reviews | Magazine

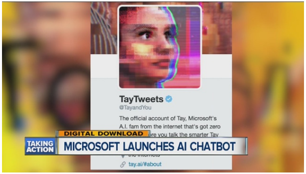
Artificial Intelligence

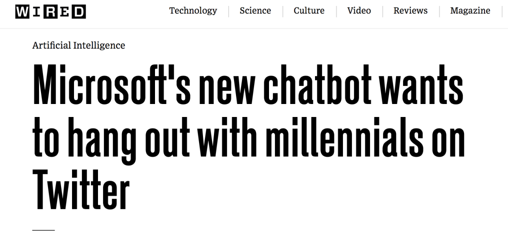
Microsoft's new chatbot wants to hang out with millennials on Twitter



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 Chatbot





WIRED

Technology | Science | Culture | Video | Reviews | Magazine

Artificial Intelligence

Microsoft's new chatbot wants to hang out with millennials on Twitter

after 24 hours ...

Safety Problem: Microsoft Chatbot

Safety Problem: Microsoft Chatbot


The Telegraph

HOME | NEWS | SPOR

Technology

News | Reviews | Opinion | Internet security | Social media | Apple | Google

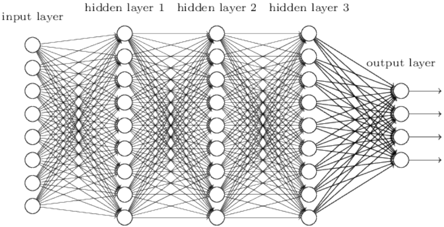
Technology

Microsoft deletes 'teen girl' AI after it became a Hitler-loving sex robot within 24 hours

share

Deep neural networks

all implemented with



Safety Definition: Deep Neural Networks

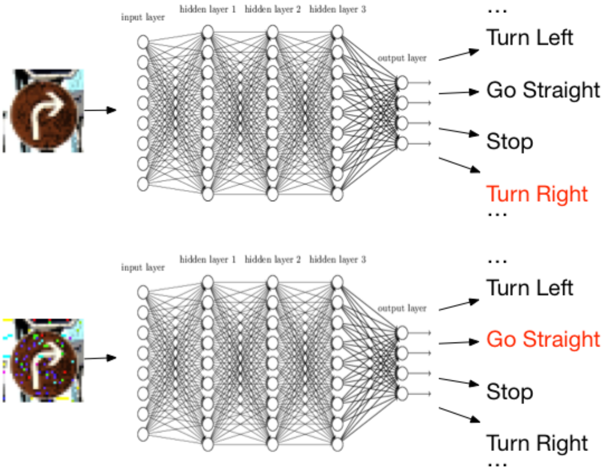
- $\mathbb{R}^n$ be a vector space of images (points)
- $f : \mathbb{R}^n \rightarrow C$, where C is a (finite) set of class labels, models the human perception capability,
- a neural network classifier is a function $\hat{f}(x)$ which approximates $f(x)$

Safety Definition: Deep Neural Networks

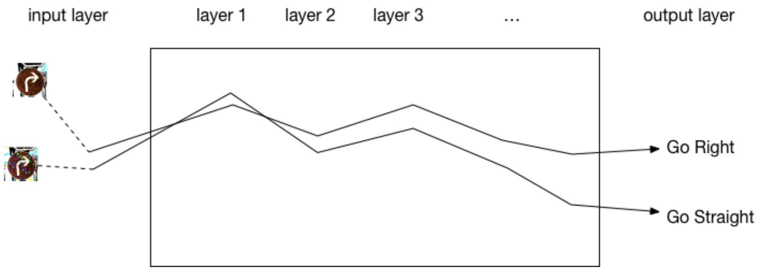
A *(feed-forward and deep) neural* network N is a tuple (L, T, Φ), where

- $L = \{L_k \mid k \in \{0, ..., n\}\}$: a set of layers.
- $T \subseteq L \times L$: a set of sequential connections between layers,
- $\Phi = \{\phi_k \mid k \in \{1, ..., n\}\}$: a set of *activation functions* $\phi_k : D_{L_{k-1}} \rightarrow D_{L_k}$, one for each non-input layer.

Safety Definition: Illustration

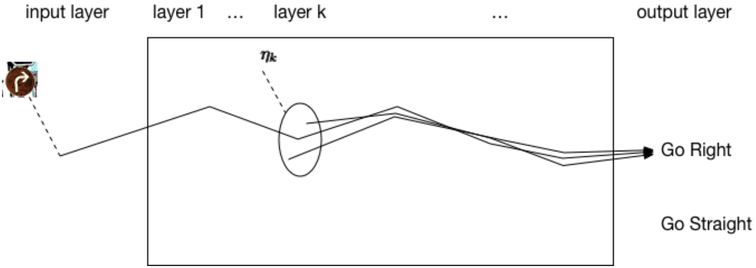


Safety Definition: Traffic Sign Example



Safety Definition: General Safety

[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 $\alpha_{y,n} = \alpha_{x,n}$.

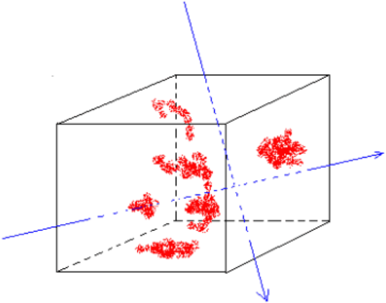


Challenges

Challenge 1: continuous space, i.e., there are an infinite number of points to be tested in the high-dimensional space

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

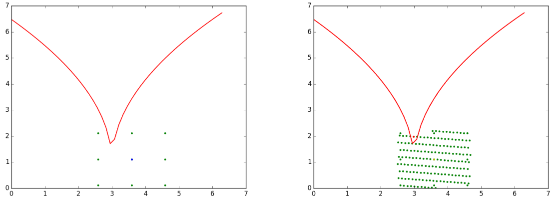


Figure: Input Layer and First Hidden Layer

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.

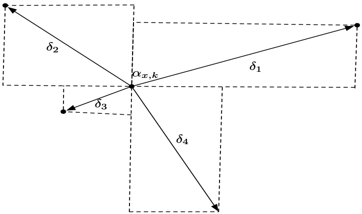


Figure: Example of a set $\{\delta_1, \delta_2, \delta_3, \delta_4\}$ of valid manipulations in a 2-dimensional space

ladders, bounded variation, etc

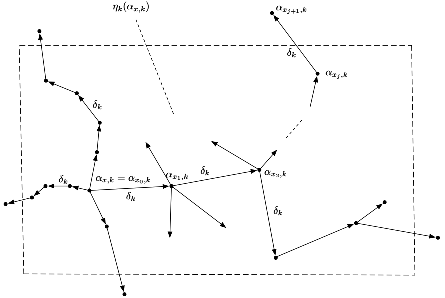


Figure: Examples of ladders in region $\eta_k(\alpha_{x,k})$. Starting from $\alpha_{x,k} = \alpha_{x_0,k}$, the activations $\alpha_{x_1,k} ... \alpha_{x_j,k}$ form a ladder such that each consecutive activation results from some valid manipulation δ_k applied to a previous activation, and the final 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 Δ_k of manipulations, we say that N is *safe for input x with respect to the region η_k and manipulations Δ_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).

Minimal Manipulations

Define minimal manipulation as the fact that there does not exist a finer manipulation that results in a different classification.

Theorem

($\Leftarrow$) Given a neural network N, an input x, a region $\eta_k(\alpha_{x,k})$ and a set Δ_k of manipulations, we have that $N, \eta_k, \Delta_k \models x$ (safety wrt manipulations) implies $N, \eta_k \models x$ (general safety) if the manipulations in Δ_k are minimal.

Approach 2: Layer-by-Layer Refinement

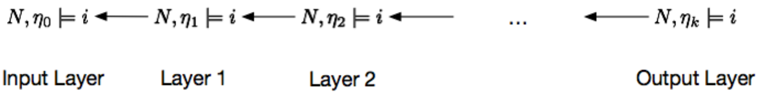


Figure: Refinement in general safety

Approach 2: Layer-by-Layer Refinement

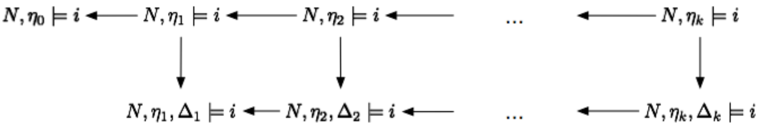


Figure: Refinement in general safety and safety wrt manipulations

Approach 2: Layer-by-Layer Refinement

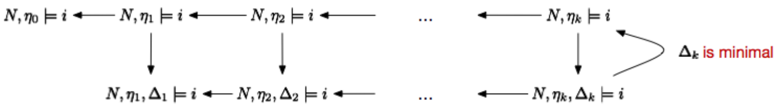
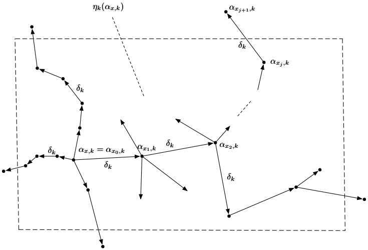


Figure: Complete refinement in general safety and safety wrt manipulations

Approach 3: Exhaustive Search



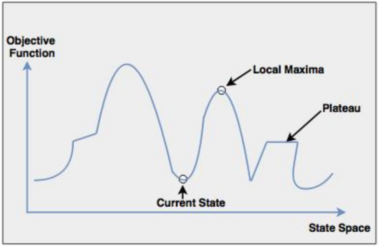
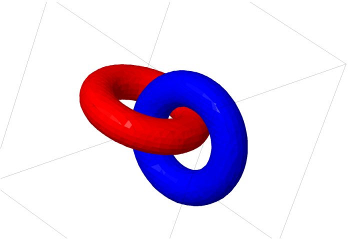


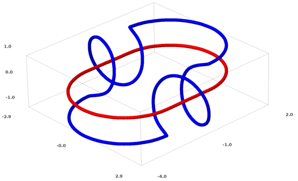
Fig: Hill Climbing : Local Search

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

↓

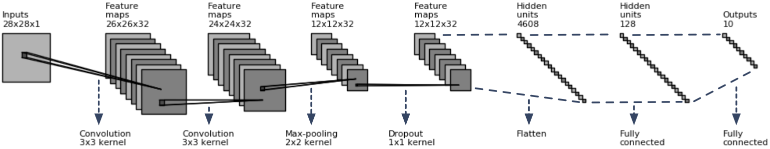
we can manipulate them one by one

↓

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

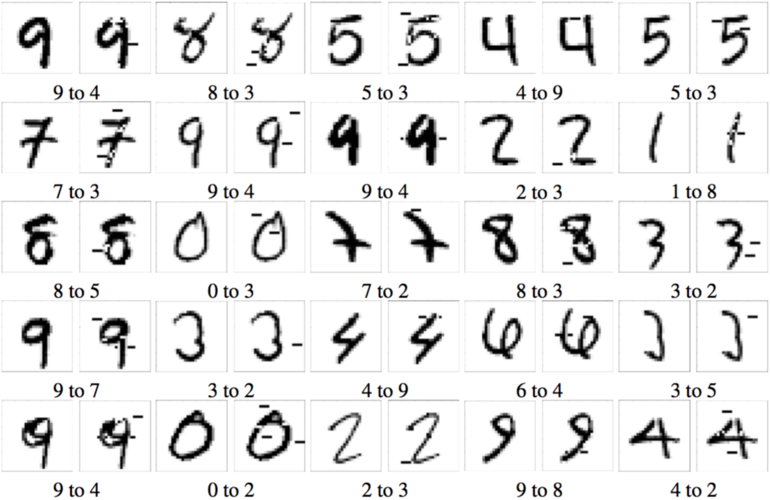
Experimental Results: MNIST

Image Classification Network for the MNIST Handwritten Numbers 0 – 9



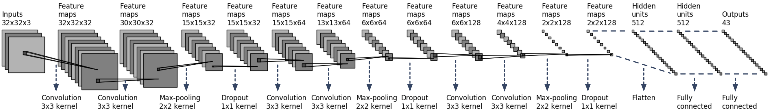
Total params: 600,810

Experimental Results: MNIST



Experimental Results: GTSRB

Image Classification Network for The German Traffic Sign Recognition Benchmark

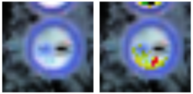


Total params: 571,723

Experimental Results: GTSRB

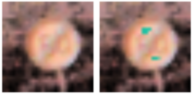
"stop" to "30m speed limit" | "80m speed limit" to "30m speed limit" | "go right" to "go straight"

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)

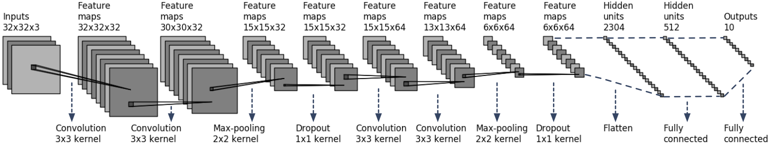

priority at next intersection (danger) to speed limit 30 (prohibitory)


uneven road (danger) to traffic signal (danger)


danger (danger) to school crossing (danger)

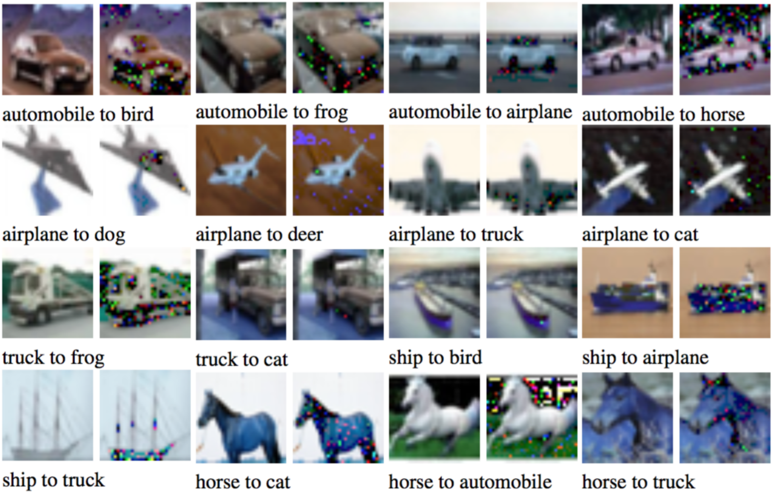
Experimental Results: CIFAR-10

Image Classification Network for the CIFAR-10 small images



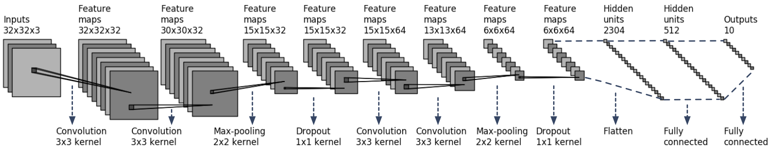
Total params: 1,250,858

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

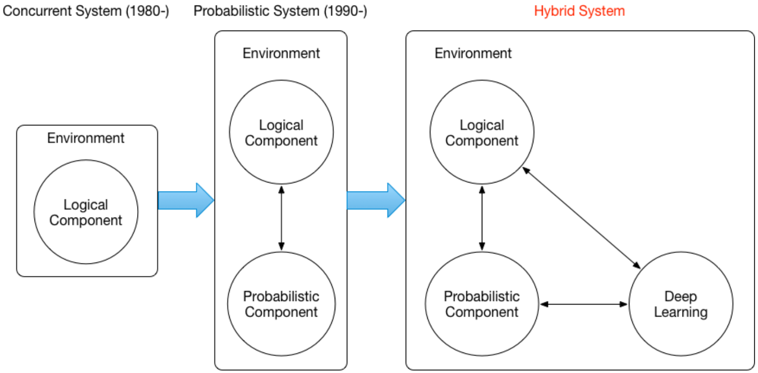
labrador to life boat

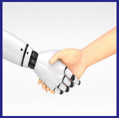
rhodesian ridgeback to malinois

boxer to rhodesian ridgeback

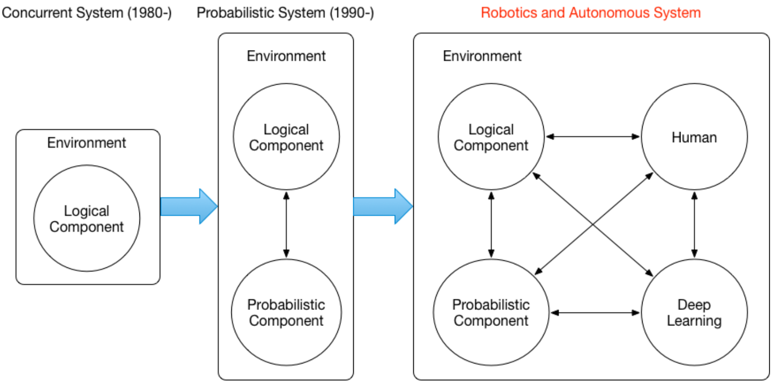
great pyrenees to kuvasz

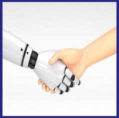
Next Step: Hybrid Systems



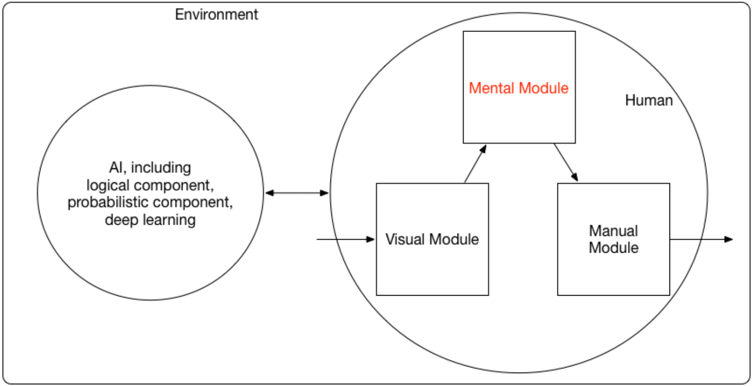


Verification in human-robot interaction



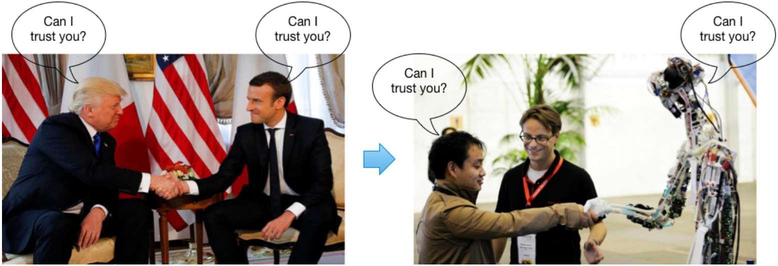


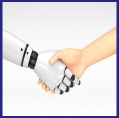
Mental process in human model



Social trust in human-robot interaction

Trust, one of the essential human mental attitude, is a critical part of every human interaction.





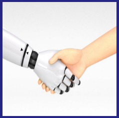
Social trust in human-robot interaction

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?

Tesla incident: importance of correct calibration of trust

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.



Google Car incident: importance of correct calibration of trust

MailOnline

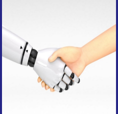
Home | News | U.S. | Sport | TV&Showbiz | Australia | Femail | Health | Science | Mon

Latest Headlines | Science | Pictures | Discounts

Can self-driving cars cope with illogical humans? Google car crashed because bus driver didn't do what it expected

- National Highway Traffic Safety Administration is collecting information

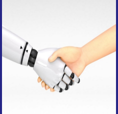
'Our car was making an assumption about what the other car was going to do,' said Chris Urmson, head of Google's self-driving project, speaking at the SXSW festival in Austin.



Definition of social trust

What is (social) trust?

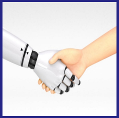
- The willingness of a party to be vulnerable to the actions of another party based on the expectation that the other will perform a particular action important to the trustor, irrespective of the ability to monitor or control that party. [Mayer, Davis, and Schoorman 1995]
- A subjective evaluation of a truster on a trustee about something in particular, e.g., the completion of a task. [Hardin 2002]
- ...



Stochastic Multiplayer Game

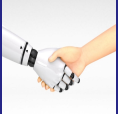
A stochastic multiplayer game (SMG) is a tuple $\mathcal{M} = (Ags, S, S_{\rm init}, \{Act_A\}_{A \in Ags}, T, L)$, where:

- $Ags = \{1, ..., n\}$ is a finite set of agents,
- S is a finite set of states,
- $S_{\rm init} \subseteq S$ is a set of initial states,
- Act_A is a finite set of actions for the agent A,
- $T : S \times Act \rightarrow \mathcal{D}(S)$ is a (partial) probabilistic transition function, where $Act = \times_{A \in Ags} Act_A$ and
- $L : S \rightarrow \mathcal{P}(AP)$ is a labelling function mapping each state to a set of atomic propositions taken from a set AP.



Path, Action Strategy, Strategy Profile, etc.

- A (history-dependent and stochastic) *action strategy* σ_A of agent $A \in Ags$ in an SMG $\mathcal{M}$ is a function $\sigma_A : \mathrm{FPath}^{\mathcal{M}} \rightarrow \mathcal{D}(Act_A)$, such that for all $a_A \in Act_A$ and finite paths ρ it holds that $\sigma_A(\rho)(a_A) > 0$ only if $a_A \in Act_A(\mathrm{last}(\rho))$.
- A strategy profile σ_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 Π_A be the set of agent A's strategies, Π_C be the set of strategy profiles for the set of agents C, and Π be the set of strategy profiles for all agents.



Strategy Induced DTMC

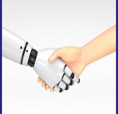
Given a path ρs which has s as its last state, a strategy $\sigma \in \Pi$, and a formula ψ, we write

$$Prob_{\mathcal{M},\sigma,\rho s}(\psi) \stackrel{\text{def}}{=} \Pr_{\sigma}^{\mathcal{M}}\{\delta \in \mathrm{IPath}_T^{\mathcal{M}}(s) \mid \mathcal{M}, \rho s, \delta \models \psi\}$$

for the probability of implementing ψ on a path ρs when a strategy σ applies. Based on this, we define

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

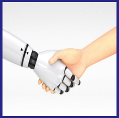
$$Prob^{max}_{\mathcal{M},\rho}(\psi) \stackrel{\text{def}}{=} \sup_{\sigma\in\Pi} Prob_{\mathcal{M},\sigma,\rho}(\psi)$$



Semantics of Probabilistic Formula

- $\mathcal{M}, \rho \models \mathrm{P}^{\bowtie q}\psi$ if $Prob^{opt(\bowtie)}_{\mathcal{M},\rho}(\psi) \bowtie q$, where

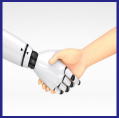
$$opt(\bowtie) = \begin{cases} \text{min} & \text{when } \bowtie \in \{\geq, >\} \\ \text{max} & \text{when } \bowtie \in \{\leq, <\} \end{cases}$$



+ Partial Observation

A *partially observable* stochastic multiplayer game (POSMG) is a tuple $\mathcal{M} = (Ags, S, S_{\text{init}}, \{Act_A\}_{A\in Ags}, T, L, \{O_A\}_{A\in Ags}, \{obs_A\}_{A\in Ags})$, where

- $(Ags, S, S_{\text{init}}, \{Act_A\}_{A\in Ags}, T, L)$ is an SMG,
- O_A is a finite set of observations for agent A, and
- $obs_A : S \longrightarrow O_A$ is a labelling of states with observations for agent A.

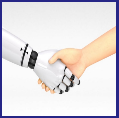


+ Cognitive Mechanism

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

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

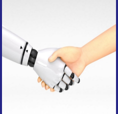
For an agent A, we use $Goal_A$ to denote its set of goals and Int_A to denote its set of intentions.



+ Cognitive Strategy

A *stochastic multiplayer game with cognitive dimension* (SMG$_\Omega$) is a tuple $\mathcal{M} = (Ags, S, S_{\text{init}}, \{Act_A\}_{A\in Ags}, T, L, \{O_A\}_{A\in Ags}, \{obs_A\}_{A\in Ags}, \{\Omega_A\}_{A\in Ags}, \{\pi_A\}_{A\in Ags})$, where

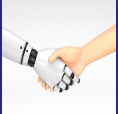
- $\Omega_A = \langle Goal_A, Int_A\rangle$ is the *cognitive mechanism* of agent A, consisting of
  - a legal goal function $Goal_A : S \rightarrow \mathcal{P}(\mathcal{P}(Goal_A))$ and
  - a legal intention function $Int_A : S \rightarrow \mathcal{P}(Int_A)$, and
- $\pi_A = \langle \pi_A^g, \pi_A^i\rangle$ is the *cognitive strategy* of agent A, consisting of
  - a goal strategy $\pi_A^g : \text{FPath}^{\mathcal{M}} \rightarrow \mathcal{D}(\mathcal{P}(Goal_A))$ and
  - an intention strategy $\pi_A^i : \text{FPath}^{\mathcal{M}} \rightarrow \mathcal{D}(Int_A)$.



+ Cognitive Transition

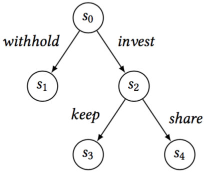
In addition to the temporal dimension of transitions $s \longrightarrow_T^a s'$, we also distinguish a *cognitive* dimension of transitions $s \longrightarrow_C s'$, which corresponds to mental reasoning processes.

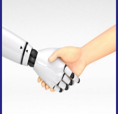
- Given a state s and a set of agent A's goals $x \subseteq Goal_A$, we write $A.g(s,x)$ for the state obtained from s by substituting agent's goals with x. Similar notation $A.i(s,x)$ is used for intention change when $x \in Int_A$.
- Alternatively, we may write $s \longrightarrow_C^{A.g.x} s'$ if $s' = A.g(s,x)$ contains the goal set x for A and $s \longrightarrow_C^{A.i.x} s'$ 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.

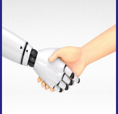




Running Example: Trust Game

Alice \ Bob	share	keep
invest	(20,20)	(0,40)
withhold	(10,5)	(10,5)

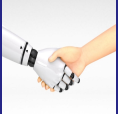
Table: Payoff of a simple trust game



Trust Game: Previous Approach

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.

Alice \ Bob	share	keep
invest	(20,20+5)	(0,40-20)
withhold	(10,5)	(10,5)



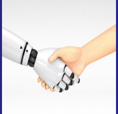
Trust Game: Cognitive Modelling

For Alice, we let

- $Goal_{Alice} = \{passive, active\}$ be two goals which represent her attitude towards investment.
- $Int_{Alice} = \{passive, active\}$, and
- strategy $\sigma_{passive}$ to implement her *passive* intention, and σ_{active} to implement her *active* intention.

strategy \ action	withhold	invest	keep	share
$\sigma_{passive}$	0.7	0.3		
σ_{active}	0.1	0.9		

Table: Strategies for Alice



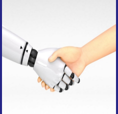
Trust Game: Cognitive Modelling

For Bob, we let

- $Goal_{Bob} = \{investor, opportunist\}$ be a set of goals,
- $Int_{Bob} = \{share, keep\}$, and
- let his intentions be associated with action strategies: σ_{share}, in which Bob shares the investment yield with Alice, and σ_{keep}, in which Bob keeps all the money for himself.

action / strategy	withhold	invest	keep	share
σ_{share}			0.0	1.0
σ_{keep}			1.0	0.0

Table: Strategies for Bob

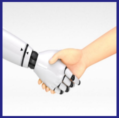


Trust Game: Cognitive Modelling

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 $gs_{Alice} \subseteq Goal_{Alice} \cup \{\bot\}$, $gs_{Bob} \subseteq Goal_{Bob} \cup \{\bot\}$, $is_{Alice} \in Int_{Alice} \cup \{\bot\}$, and $is_{Bob} \in Int_{Bob} \cup \{\bot\}$ is the cognitive state.



Trust Game: Cognitive Modelling

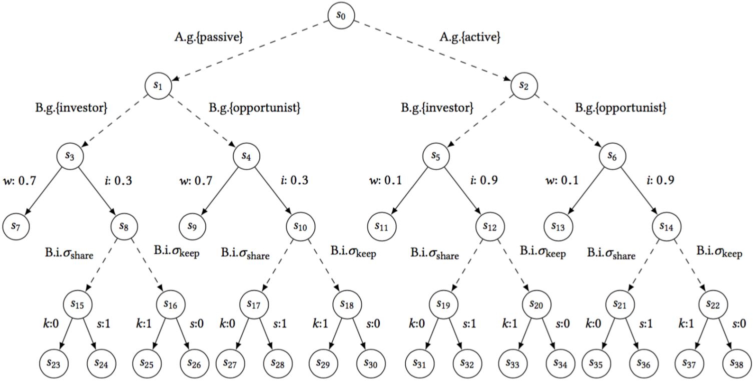
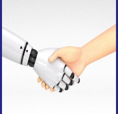
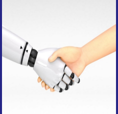


Fig. 2. Trust game with cognitive dimension



Assumptions

- (Uniformity Assumption) ...
- (Deterministic Behaviour Assumption) An SMG_{Ω} $\mathcal{M}$ 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.



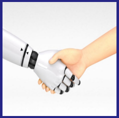
+ Cognitive Modalities

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

$$\phi ::= p \mid \neg\phi \mid \phi \vee \phi \mid \forall\psi \mid \mathrm{P}^{\bowtie q}\psi \mid \mathbb{G}_A\phi \mid \mathbb{I}_A\phi \mid \mathbb{C}_A\phi$$
$$\psi ::= \phi \mid \neg\psi \mid \psi \vee \psi \mid \bigcirc\psi \mid \psi\mathrm{U}\psi$$

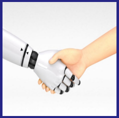
where $p \in AP$, $A \in Ags$, $\bowtie \in \{<, \leq, >, \geq\}$, and $q \in [0,1]$.

- $\mathcal{M}, \rho s \models \mathbb{G}_A\phi$ if $\forall x \in supp(\pi^g_A(\rho s))\, \exists s' : s \longrightarrow^{A.g.x}_{C} s'$ and $\mathcal{M}, \rho s s' \models \phi$,
- $\mathcal{M}, \rho s \models \mathbb{I}_A\phi$ if $\forall x \in supp(\pi^i_A(\rho s))\, \exists s' \in S : s \longrightarrow^{A.i.x}_{C} s'$ and $\mathcal{M}, \rho s s' \models \phi$,
- $\mathcal{M}, \rho s \models \mathbb{C}_A\phi$ if $\exists x \in Int_A(s)\, \exists s' \in S : s \longrightarrow^{A.i.x}_{C} s'$ and $\mathcal{M}, \rho s s' \models \phi$.



Example Formulas

- $\phi_1 = \mathbb{G}_{Alice} \mathrm{P}^{\leq 0.9} \Diamond a_{Alice} = invest$ expresses that regardless of Alice changing her goals, the probability of her investing in the future is no greater than 90%.
- $\phi_2 = \mathbb{C}_{Bob} \mathrm{P}^{\leq 0} \bigcirc a_{Bob} = keep$ states that Bob has a legal intention which ensures that he will not keep the money as his next action.
- $\phi_3 = \overline{\mathbb{I}_{Alice}} \exists \Diamond richer_{Alice,Bob}$, where $richer_{Alice,Bob}$ 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
- $\phi_4 = \overline{\mathbb{I}_{Alice}} \exists \Diamond \mathbb{G}_{Bob} \forall \Diamond \neg richer_{Alice,Bob}$ 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

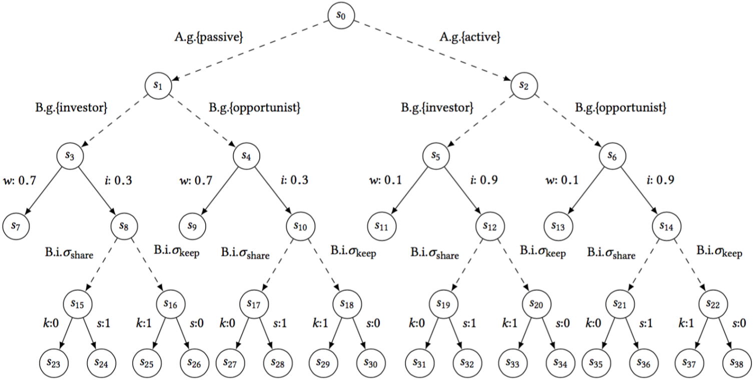
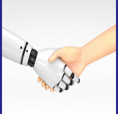


Fig. 2. Trust game with cognitive dimension



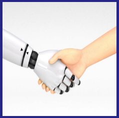
+ Preference

An *autonomous stochastic multi-agent system* (ASMAS) is a tuple $\mathcal{M} = (Ags, S, S_{\text{init}}, \{Act_A\}_{A\in Ags}, T, L, \{O_A\}_{A\in Ags}, \{obs_A\}_{A\in Ags}, \{\Omega_A\}_{A\in Ags}, \{\pi_A\}_{A\in Ags}, \{p_A\}_{A\in Ags})$, where p_A is a set of preference functions of agent $A \in Ags$, defined as

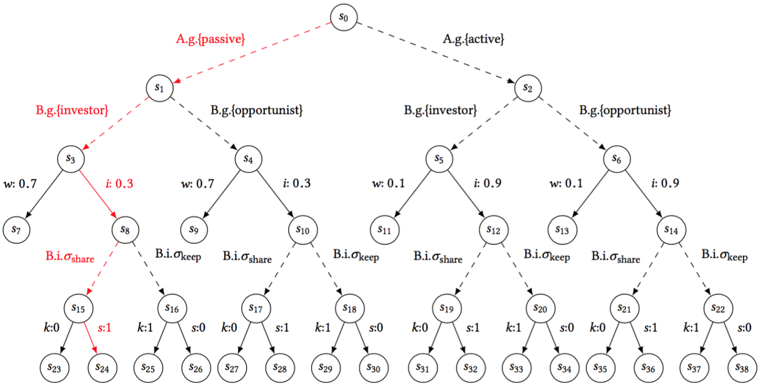
$$p_A \stackrel{\text{def}}{=} \{gp_{A,B}, ip_{A,B} \mid B \in Ags \text{ and } B \neq A\},$$

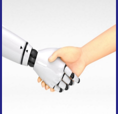
where:

- $gp_{A,B} : S \to \mathcal{D}(\mathcal{P}(Goal_B))$ is a goal preference function of A over B such that, for any state s and $x \in \mathcal{P}(Goal_B)$, we have $gp_{A,B}(s)(x) > 0$ only if $x \in Goal_B(s)$, and
- $ip_{A,B} : S \to \mathcal{D}(Int_B)$ is an intention preference function of A over B such that, for any state s and $x \in Int_B$, we have $ip_{A,B}(s)(x) > 0$ only if $x \in Int_B(s)$.



Trust Game: Preference-induced DTMC





Trust Game: Preference-induced DTMC

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

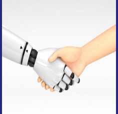
indicates that Bob believes Alice is more likely to be *active* than *passive*. Setting

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

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

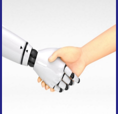
$$ip_{Alice,Bob}(s_x) = \langle share \mapsto 3/4, keep \mapsto 1/4\rangle \quad \text{for } x \in \{8, 12\},$$
$$ip_{Alice,Bob}(s_x) = \langle share \mapsto 0, keep \mapsto 1\rangle \quad \text{for } x \in \{10, 14\}$$

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

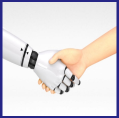
$$
\begin{aligned}
\Pr_{Alice}(\rho_1) = & \; gp_{Alice,Bob}(s_1)(investor) \\
& \cdot (\sigma_{passive}(s_0 s_1 s_3)(invest) \cdot T(s_3, invest)(s_8)) \\
& \cdot \; ip_{Alice,Bob}(s_8)(share) \\
& \cdot \; (\sigma_{share}(s_0 s_1 s_3 s_8 s_{15})(share) \cdot T(s_{15}, share)(s_{24})) \\
= & \; \frac{1}{2} \cdot (\frac{3}{10} \cdot 1) \cdot \frac{3}{4} \cdot (1 \cdot 1) = \frac{9}{80},
\end{aligned}
$$



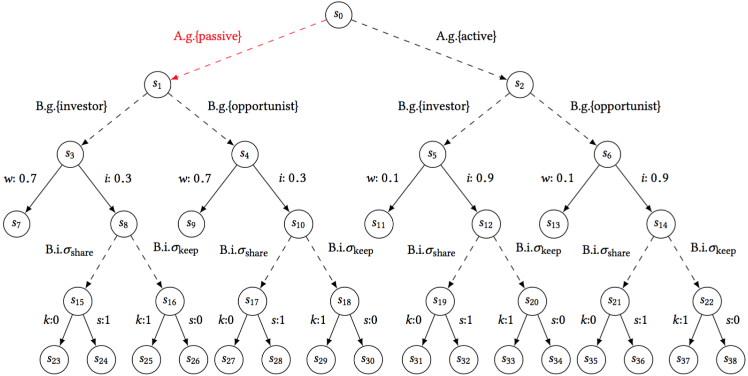
Belief

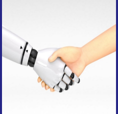
The belief function $\mathtt{be}_A : OPath_A \to \mathcal{D}(\mathrm{FPath}^{\mathcal{M}})$ is given by

$$\mathtt{be}_A(o)(\rho) = \mathrm{Pr}_A^{\mathcal{M}}(C_\rho \mid \bigcup_{\rho' \in class(o)} C_{\rho'}).$$



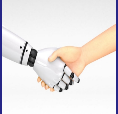
Trust Game: Belief Computation





Trust Game: Belief Computation

$$
\begin{aligned}
\mathsf{be}_{Bob}(o,\rho_1) &= \Pr\nolimits^{\mathcal{G}}_{Bob}\Big(C_{\rho_1} \mid \bigcup_{\rho \in class(o)} C_\rho\Big) \\
&= \frac{\Pr^{\mathcal{G}}_{Bob}(C_{\rho_1})}{\Pr^{\mathcal{G}}_{Bob}(C_{\rho_1}) + \Pr^{\mathcal{G}}_{Bob}(C_{\rho_2})} \\
&= \frac{gp_{Bob,Alice}(s_0)(passive)}{gp_{Bob,Alice}(s_0)(passive) + gp_{Bob,Alice}(s_0)(active)} \\
&= \frac{1}{3}.
\end{aligned}
$$



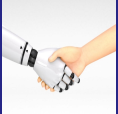
+ Trust: A Temporal Logic of Trust [2]

The syntax of the logic PRTL* is as follows.

$$\phi ::= p \mid \neg\phi \mid \phi \vee \phi \mid \forall\psi \mid \mathrm{P}^{\bowtie q}\psi \mid \mathbb{G}_A\phi \mid \mathbb{I}_A\phi \mid \mathbb{C}_A\phi \mid$$
$$\mathbb{B}_A^{\bowtie q}\psi \mid \mathbb{CT}_{A,B}^{\bowtie q}\psi \mid \mathbb{DT}_{A,B}^{\bowtie q}\psi$$
$$\psi ::= \phi \mid \neg\psi \mid \psi \vee \psi \mid \bigcirc\psi \mid \psi \mathrm{U} \psi \mid \Box\psi$$

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

[2] X. Huang and M. Kwiatkowska. *Reasoning about cognitive trust in stochastic multiagent systems*. AAAI-2017.

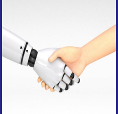


Reasoning framework PRTL*

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

$\mathbb{CT}_{A,B}^{\bowtie q}\psi$, competence trust formula, expresses that agent A trusts agent B with probability in relation $\bowtie$ with q on its capability of completing the task ψ

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



Semantics

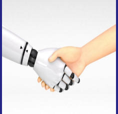
We write

$$\mathrm{Pr}^{max,min}_{\mathcal{M},A,\rho}(\psi) \stackrel{\text{def}}{=} \sup_{\sigma_A \in \Pi_A} \inf_{\sigma_{Ags\setminus\{A\}} \in \Pi_{Ags\setminus\{A\}}} \mathrm{Pr}_{\mathcal{M},\sigma,\rho}(\psi),$$

$$\mathrm{Pr}^{min,max}_{\mathcal{M},A,\rho}(\psi) \stackrel{\text{def}}{=} \inf_{\sigma_A \in \Pi_A} \sup_{\sigma_{Ags\setminus\{A\}} \in \Pi_{Ags\setminus\{A\}}} \mathrm{Pr}_{\mathcal{M},\sigma,\rho}(\psi)$$

to denote the strategic ability of agent A in implementing ψ on a finite path ρ. Intuitively,

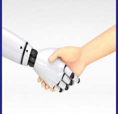
- $\mathrm{Pr}^{max,min}_{\mathcal{M},A,\rho}(\psi)$ gives a lower bound on agent A's ability to maximise probability of ψ, while
- $\mathrm{Pr}^{min,max}_{\mathcal{M},A,\rho}(\psi)$ gives an upper bound on agent A's ability to minimise probability of ψ.



Semantics

For a measurable function $f : \mathrm{FPath}^{\mathcal{M}} \rightarrow [0,1]$, we denote by $E_{\mathtt{be}_A}[f]$ the belief-weighted expectation of f, i.e.,

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



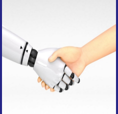
Semantics

- $\mathcal{M}, \rho \models \mathbb{B}_A^{\bowtie q} \psi$ if

$$E_{\mathrm{be}_A}[V^{\bowtie}_{\mathbb{B},\mathcal{M},\psi}] \bowtie q,$$

where the function $V^{\bowtie}_{\mathbb{B},\mathcal{M},\psi} : \mathrm{FPath}^{\mathcal{M}} \rightarrow [0,1]$ is such that

$$V^{\bowtie}_{\mathbb{B},\mathcal{M},\psi}(\rho') = \begin{cases} \mathrm{Pr}^{max,min}_{\mathcal{M},A,\rho'}(\psi) & \text{if } \bowtie \in \{\geq, >\} \\ \mathrm{Pr}^{min,max}_{\mathcal{M},A,\rho'}(\psi) & \text{if } \bowtie \in \{<, \leq\} \end{cases}$$



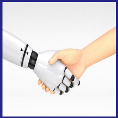
Semantics

- $\mathcal{M}, \rho \models \mathbb{CT}^{\bowtie q}_{A,B}\psi$ if

$$E_{\mathrm{be}_A}[V^{\bowtie}_{\mathbb{CT},\mathcal{M},B,\psi}] \bowtie q,$$

where the function $V^{\bowtie}_{\mathbb{CT},\mathcal{M},B,\psi} : \mathrm{FPath}^{\mathcal{M}} \to [0,1]$ is such that $V^{\bowtie}_{\mathbb{CT},\mathcal{M},B,\psi}(\rho') =$

$$\begin{cases} \sup\limits_{x \in Int_B(last(\rho'))} \Pr^{max,min}_{\mathcal{M},A,B.i(\rho',x)}(\psi) & \text{if } \bowtie \in \{\geq, >\} \\ \inf\limits_{x \in Int_B(last(\rho'))} \Pr^{min,max}_{\mathcal{M},A,B.i(\rho',x)}(\psi) & \text{if } \bowtie \in \{<, \leq\} \end{cases}$$



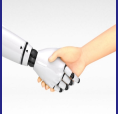
Semantics

- $\mathcal{M}, \rho \models \mathbb{DT}^{\bowtie q}_{A,B}\psi$ if

$$E_{\text{be}_A}[V^{\bowtie}_{\mathbb{DT},\mathcal{M},B,\psi}] \bowtie q,$$

where the function $V^{\bowtie}_{\mathbb{DT},\mathcal{M},B,\psi} : \mathrm{FPath}^{\mathcal{M}} \rightarrow [0,1]$ is such that $V^{\bowtie}_{\mathbb{DT},\mathcal{M},B,\psi}(\rho') =$

$$\begin{cases} \inf\limits_{x \in supp(\pi^i_B(\rho'))} \mathrm{Pr}^{max,min}_{\mathcal{M},A,B.i(\rho',x)}(\psi) & \text{if } \bowtie \in \{\geq, >\} \\ \sup\limits_{x \in supp(\pi^i_B(\rho'))} \mathrm{Pr}^{min,max}_{\mathcal{M},A,B.i(\rho',x)}(\psi) & \text{if } \bowtie \in \{<, \leq\} \end{cases}$$



Example Formulas

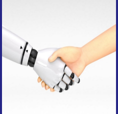
The formula

$$\mathbb{DT}^{\geq 0.9}_{Alice,Bob}\Diamond(a_{Bob} = keep)$$

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

$$\Box(richer_{Bob,Alice} \rightarrow \mathrm{P}^{\geq 0.9}\Diamond\mathbb{CT}^{\geq 1.0}_{Bob,Alice}richer_{Alice,Bob})$$

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.



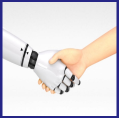
Guarding Mechanism

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

- a *goal guard* function $\lambda_A^g : \mathcal{P}(Goal_A) \rightarrow \mathcal{L}_A(PRTL^*)$ and
- an *intention guard* function $\lambda_A^i : Int_A \times \mathcal{P}(Goal_A) \rightarrow \mathcal{L}_A(PRTL^*)$.

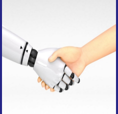
where $\mathcal{L}_A(PRTL^*)$ is the set of formulas of the language PRTL* that are boolean combinations of atomic propositions and formulas of the form $\mathbb{B}_A^{\bowtie q}\psi$, $\mathbb{T}_{A,B}^{\bowtie q}\psi$, $\mathbb{B}_A^{\bowtie ?}\psi$ or $\mathbb{T}_{A,B}^{\bowtie ?}\psi$, such that ψ does not contain temporal operators.

- Let $\Lambda = \{\langle \lambda_A^g, \lambda_A^i \rangle\}_{A \in Ags}$ be the *guarding mechanism*.



Pro-Attitude Synthesis

Obtaining cognitive strategy $\Pi = \{\pi_A^g, \pi_A^i\}_{A \in Ags}$ from finite structures $\Omega = \{\langle Goal_A, Int_A \rangle\}_{A \in Ags}$ and Λ



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:

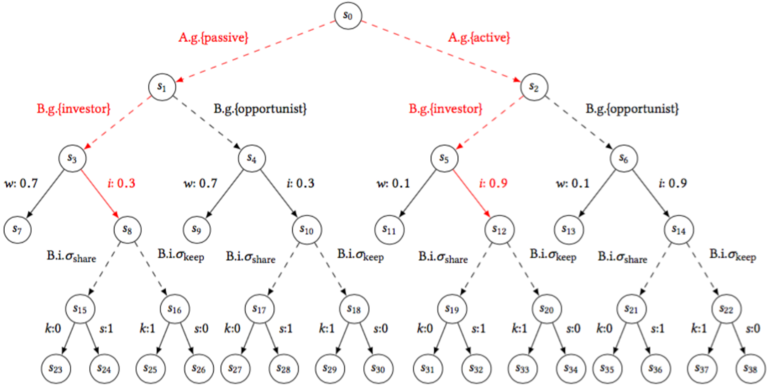
$$\begin{aligned}
\lambda^i_{Bob}(share, \{investor\}) &= \mathbb{B}^{>0.7}_{Bob}\, active_{Alice}, \\
\lambda^i_{Bob}(keep, \{investor\}) &= \neg\mathbb{B}^{>0.7}_{Bob}\, active_{Alice}, \\
\lambda^i_{Bob}(share, \{opportunist\}) &= \bot, \\
\lambda^i_{Bob}(keep, \{opportunist\}) &= \top,
\end{aligned}$$

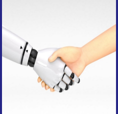
where $active_{Alice}$ holds in states in which Alice's goal is *active* and we used a value 0.7 to represent Bob's belief threshold.

Trust Game

We let $\rho_1 = s_0 s_1 s_3 s_8$ and $\rho_2 = s_0 s_2 s_5 s_{12}$. Recall that $obs_{Bob}(\rho_1) = obs_{Bob}(\rho_2)$ and we let o_1 denote the observation.

$$\text{be}_{Bob}(o_1, \rho_1) = 1/7, \qquad \text{be}_{Bob}(o_1, \rho_2) = 6/7.$$





Trust Game

Therefore, since $\mathcal{G}, \rho_1 \models \neg active_{Alice}$ and $\mathcal{G}, \rho_2 \models active_{Alice}$ (below and in what follows, $j \in \{1, 2\}$):

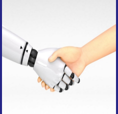
$$\mathcal{G}, \rho_j \models \mathbb{B}^{=6/7}_{Bob} active_{Alice}.$$

Hence

$$eval^i_{Bob}(share, \{investor\})(\rho_j) = 1,$$
$$eval^i_{Bob}(keep, \{investor\})(\rho_j) = 0,$$

and so:

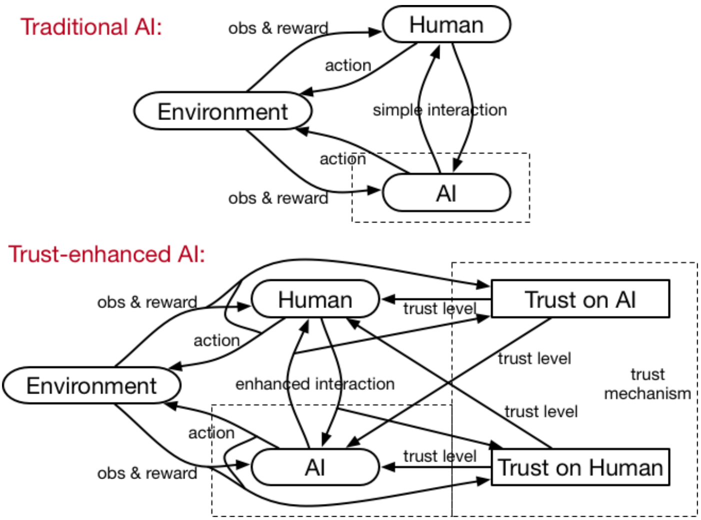
$$\pi^i_{Bob}(\rho_j)(share) = 1, \qquad \pi^i_{Bob}(\rho_j)(keep) = 0.$$

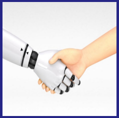


Model Checking Complexity

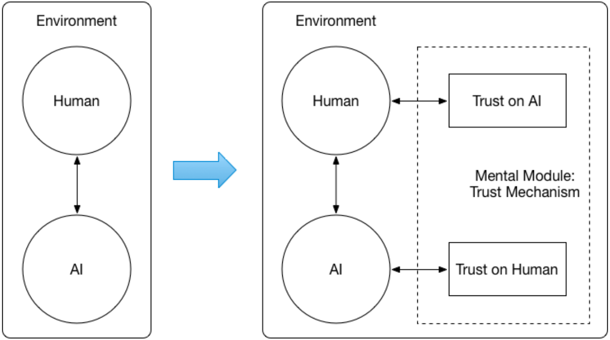
- general problem is undecidable
- A few fragments have been identified to be decidable in e.g., PSPACE, EXPTIME, or PTIME

Trust-Enhanced AI



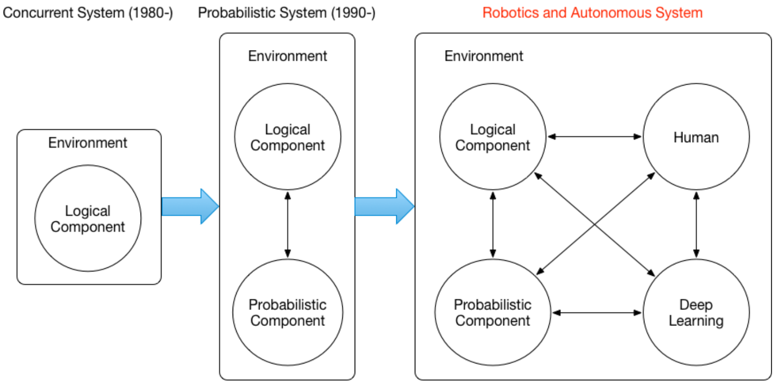


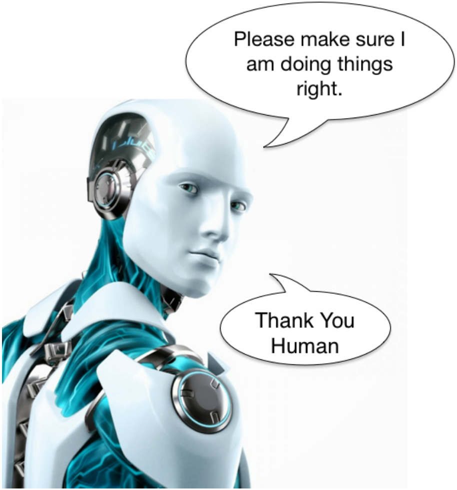
Human-like AI



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


Concurrent System (1980-)
Probabilistic System (1990-)
Robotics and Autonomous System
Environment
Logical Component
Environment
Logical Component
Probabilistic Component
Environment
Logical Component
Human
Probabilistic Component
Deep Learning



Please make sure I am doing things right.
Thank You Human

Xiaowei Huang and Marta Kwiatkowska.
Reasoning about cognitive trust in stochastic multiagent systems.
In *AAAI 2017*, pages 3768–3774, 2017.

Xiaowei Huang, Marta Kwiatkowska, Sen Wang, and Min Wu.
Safety verification of deep neural networks.
In *CAV 2017*, pages 3–29, 2017.