Understanding and Verifying Autonomy and Teamwork

Michael Fisher

Department of Computer Science, University of Manchester, UK

https://web.cs.manchester.ac.uk/~michael

21st March, 2022

Overview

- *Autonomous Systems*
  – *what is autonomy?*
- *Architectures* can help!
  – *designing for transparency, verifiability, explainability...*
- *Teams*
  – *captured as collections of agents*
- *What are the Benefits?*
- *What are the Drawbacks?*

Overview: How to model a ‘team’?

Traditional (Engineering) route to modelling collectives is to:

1. assume each element is a simple control/learning component

   (often: modelled using stochastic differential equations)

2. model the interactions between the elements

   (*often: how states of one element affect states of another)*

Then, if some members of the collective are to represent humans,

3. try to add cognitive aspects to individual representations

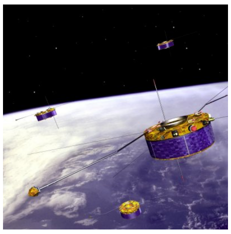
Rather than this *bottom-up* approach, we instead start with sets of fully autonomous decision-making agents and then add interactions to capture collaboration and teamwork; think *top-down*

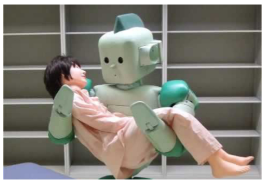
Let’s start then with a recap of ‘autonomy’.

Autonomous Systems

Autonomy:

the ability of a system to make its own decisions and to act on its own, and to do both without direct human intervention.





rtc.nagoya.riken.jp/RI-MAN

www.volvo.com

Who makes the Decisions?

Even within ‘autonomy’, there are important variations concerning decision-making:

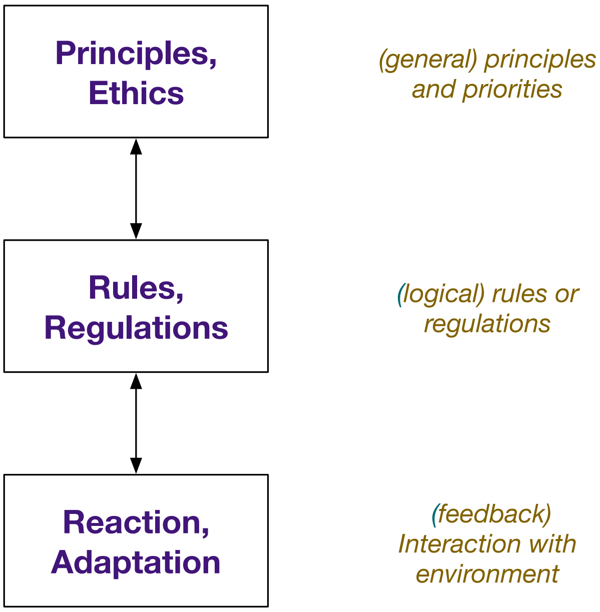
Automatic: involves a number of fixed, and prescribed, activities; there may be options, but these are generally fixed in advance.

Adaptive: improves its performance/activity based on feedback from environment — typically developed using tight continuous control and optimisation, e.g. feedback control system.

Autonomous: decisions made based on system’s (belief about its) current situation at the time of the decision — environment still taken into account, but internal motivations/beliefs are important.

Verification is very different across these variations.

Another Dimension: Classes of Decision



Principles, Ethics

(general) principles and priorities

Rules, Regulations

(logical) rules or regulations

Reaction, Adaptation

(feedback) Interaction with environment

Critical Decision-Making in Uncertain Environments

In a predictable and known environment

.... *we can enumerate all decisions that might be needed and can pre-code the best/optimal outcomes*

In non-critical scenarios

.... *we probably don't care too much how decisions are made*

But: what about

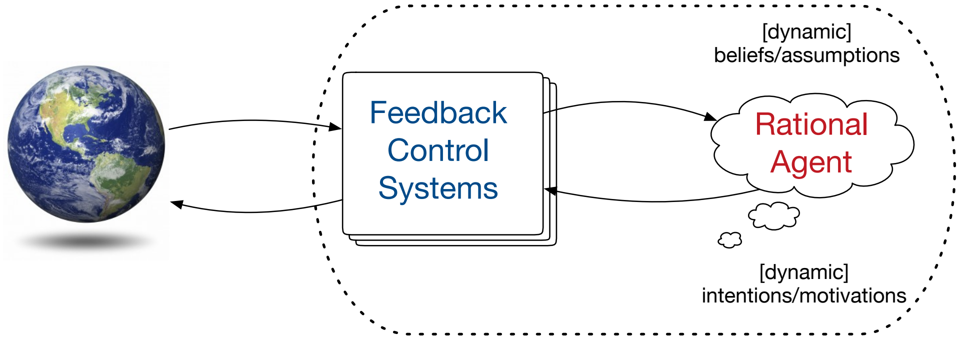
critical decisions in uncertain environments?

We need: *trustworthiness* (transparency, verifiability).

Our Approach

Our approach is that

we should be certain what the autonomous system **intends** *to do and how it* **chooses** *to go about this*

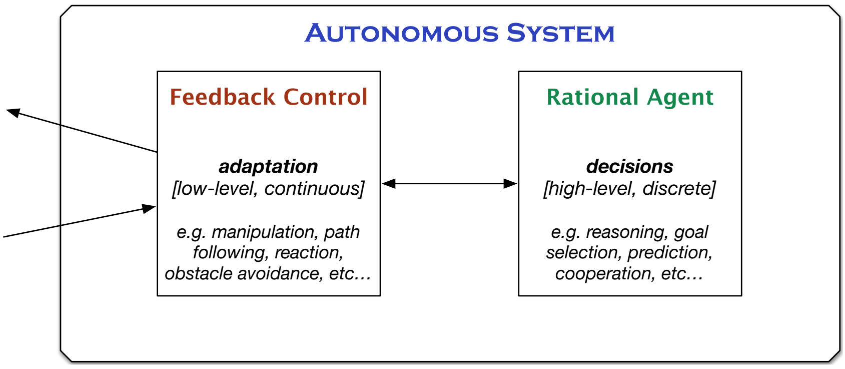


A *rational agent* (typically, a BDI Agent):

must have explicit **reasons** *for making the choices it does, and should expose/explain these when needed*

Hybrid Agent Architectures

1. *rational agent(s)* for high-level autonomous decisions, and
2. traditional *feedback control systems* for low-level activities,


Autonomous System
Feedback Control
adaptation
[low-level, continuous]
e.g. manipulation, path following, reaction, obstacle avoidance, etc…
Rational Agent
decisions
[high-level, discrete]
e.g. reasoning, goal selection, prediction, cooperation, etc…

Decision-making process (and reasons for decisions) is transparent.

Example: from Pilot to Rational Agent

Autopilot can essentially fly an aircraft

- keeping on a particular path,
- keeping flight level/steady under environmental conditions,
- planning routes around obstacles, etc.

Human pilot makes high-level decisions, such as

- where to go to,
- when to change route,
- what to do in an emergency, etc.

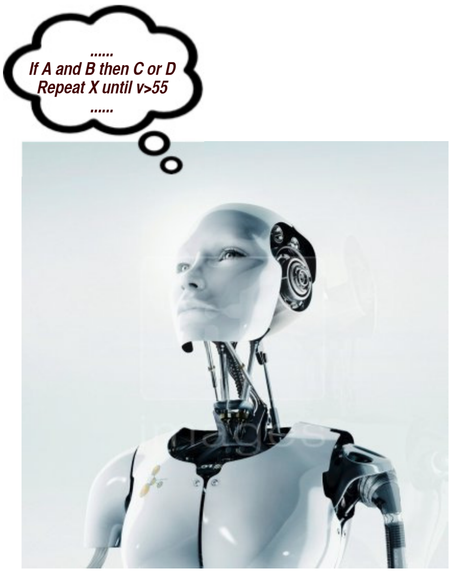
Rational Agent now makes the decisions the pilot used to make.

Benefits: Transparency [No Psychiatrists for Robots?]

As we move towards increased autonomy, we need to expose not just **what** the robot will do, but **why** it chooses to do it.

With an *agent architecture* we can (in principle) examine its internal programming and find out exactly

1. *what* it is “thinking”,
2. what *choices* it has, and
3. why it *decides* to take particular ones.



Dennis, Fisher — Verifiable Self-aware Agent-based Autonomous Systems. Proceedings of the IEEE, 2020.

Benefits: Explainability for Free

We have a rational agent that

1. has symbolic representations of its motivations (goals, intentions) and beliefs
2. reasons about these in order to decide what to do, and
3. records all the other options/reasons explored.

So, we have a trace of reasoned decisions.

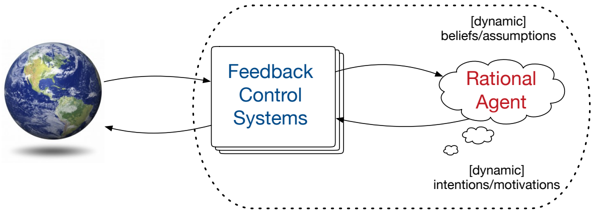
We can provide:

- recording facilities — *ethical black box*;
- interactive capabilities — “why did you do that?”
- .... ‘‘what will you do next, and why?”

Koeman, Dennis, Webster, Fisher, Hindriks — The “Why did you do that” Button: Answering Why-questions for end users of Robotic Systems. In Proc. EMAS, 2019.

Benefits: Verification

With a hybrid agent-based architecture we can employ different verification techniques to different parts.



- We can *formally verify* the agent's decision-making
- We can *simulate/test/monitor* the feedback control
- We can *practically test* whole system ⟶ more confidence?

Note: we can only be certain of what the agent *intends* to do!

Dennis, Fisher, Lincoln, Lisitsa, Veres — Practical Verification of Decision-Making in Agent-Based Autonomous Systems. Journal of Automated Software Engineering, 2016.

Teams as Multi-Agent Systems

In our case, teams comprise:

- agents associated with our embodied autonomous systems;
- agents modelling (simplified) aspects of human behaviour; and
- software agents

To these we add communication/coordination constraints.

But, we have a range of choices in modelling humans:

- humans as *random* agents
  → simple, covers wide behaviour, but big state-space
- human behaviour encoded as *simple* agents
  → small number of goals and plans in agents
- model humans as more *complex* agents
  → requires sophisticated agent modelling

Example: Domestic Robotic Assistants

Robotic Assistants are now being designed to help the elderly or incapacitated.

One robot and one human.

But more likely also: doctor; nurse; smart-home; cleaning robot; social robot; etc

This team must be able to work together and we must be able to verify that the team (in principle) functions correctly.

Webster, Dixon, Fisher, Salem, Saunders, Koay, Dautenhahn, Saez-Pons — Toward Reliable Autonomous Robotic Assistants Through Formal Verification: A Case Study. IEEE Trans. Human-Machine Systems, 2016.

Example: Road Trains

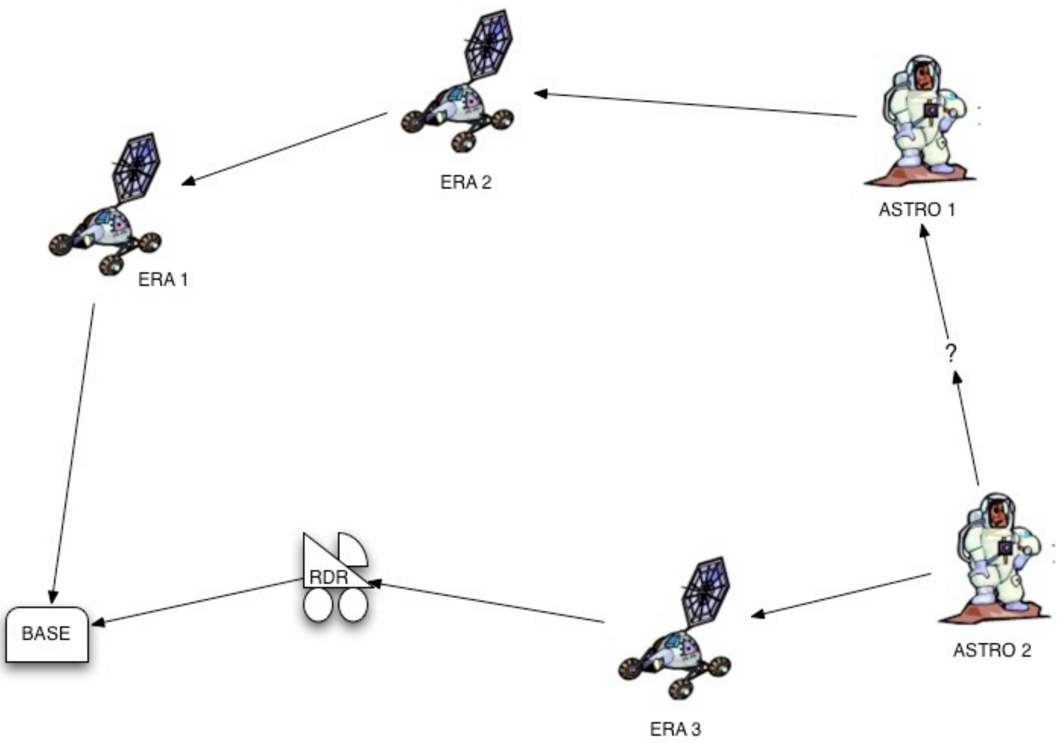
Scania Platoon

[http://www.sartre-project.eu]

We can formally verify joining, leaving, emergency, etc, protocols.

Kamali, Dennis, McAree, Fisher, Veres — Formal Verification of Autonomous Vehicle Platooning. Science of Computer Programming, 2017.

Example: NASA Mars Exploration



(Logically) Describing Team Behaviour

Individual interactions:

- *ASTRO1 believes that ERA2 aims to ensure it is never more than 30m from ASTRO1*

  $\mathsf{Bel}_{ASTRO1}\,\Box\,\mathsf{Int}_{ERA2}\; distance(ASTRO1, ERA2) < 30m$

Distributed capabilities:

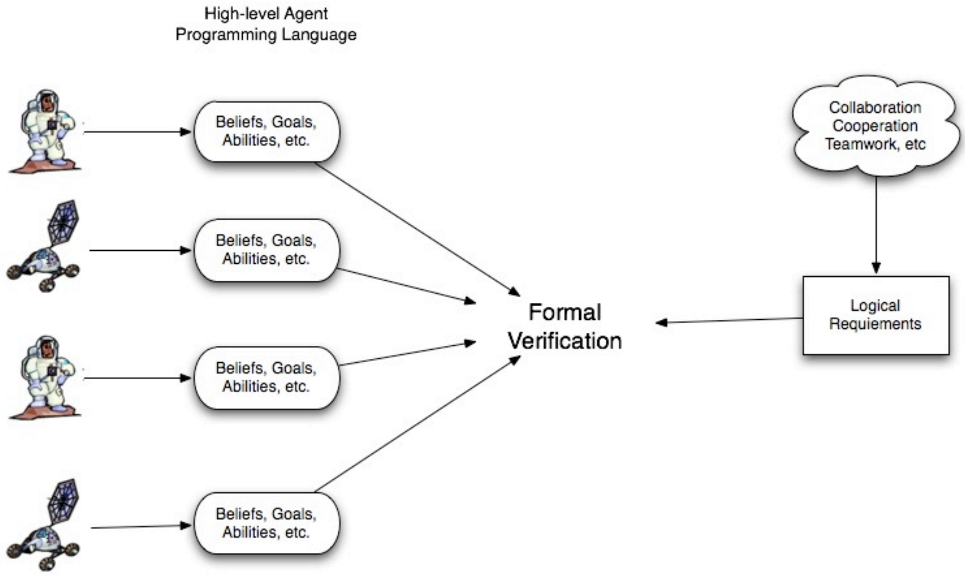
- There is at least one team member who will be within 30m

  $\exists X.\ \Box\ distance(ASTRO1, X) < 30m$

Team achievements:

- ASTRO1 will never be isolated $\Box\neg isolated(ASTRO1)$
- eventually the area will have been searched $\Diamond area\_searched$

Benefits: Again, some Formal Verification



High-level Agent Programming Language

Beliefs, Goals, Abilities, etc.

Beliefs, Goals, Abilities, etc.

Beliefs, Goals, Abilities, etc.

Beliefs, Goals, Abilities, etc.

Formal Verification

Collaboration Cooperation Teamwork, etc

Logical Requiements

Webster, Dennis, Dixon, Fisher, Stocker, Sierhuis — Formal Verification of Astronaut-Rover Teams for Planetary Surface Operations. In Proc. IEEE Aerospace Conference, 2020.

Problems: Failure is Inevitable

Verifying resilience (and self-awareness) and building in fault-tolerance is simple in simple cases but **very** difficult in general.

We regularly use **Runtime Verification** as a mechanism to catch unexpected behaviours.

Nevertheless, getting the agents to understand what caused a failure, and to self-modify to fix it, is **very** complex.

Ferrando, Cardoso, Fisher, Ancona, Franceschini, Mascardi — ROSMonitoring: A Runtime Verification Framework for ROS. In Proc. Towards Autonomous Robotic Systems, 2020.

Problems: Opaque Components Cause Issues

Formally verifying team behaviour is essential.

But: Lack of transparency in some components, e.g. deep ML components, cause difficulties.

We just have a (very) weak guarantee of its behaviour, typically a probabilistic envelope.

This doesn't help us much verifying team behaviours.

Note: probabilistic models/estimates are always wrong, sometimes **very** wrong

Problems: Transparency and Explainability not Enough

Transparency is necessary.

So is explainability (about both past actions and future aims).

But we specifically need understandability (cf dialogue)

Dennis, Oren — Explaining BDI agent behaviour through dialogue. In Proc. AAMAS, 2021.

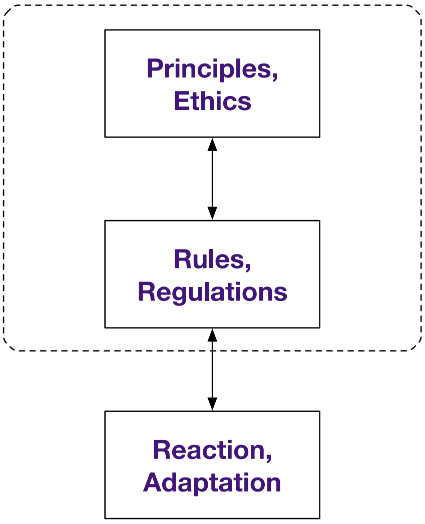
Problems: Complexity

Agents can get too complex, especially if we model very detailed human behaviours.

Correspondingly, formal verification becomes difficult.

However, we benefit from teams usually being small.

Benefits: Clarity over Decision-Making



Principles, Ethics

Rules, Regulations

Reaction, Adaptation

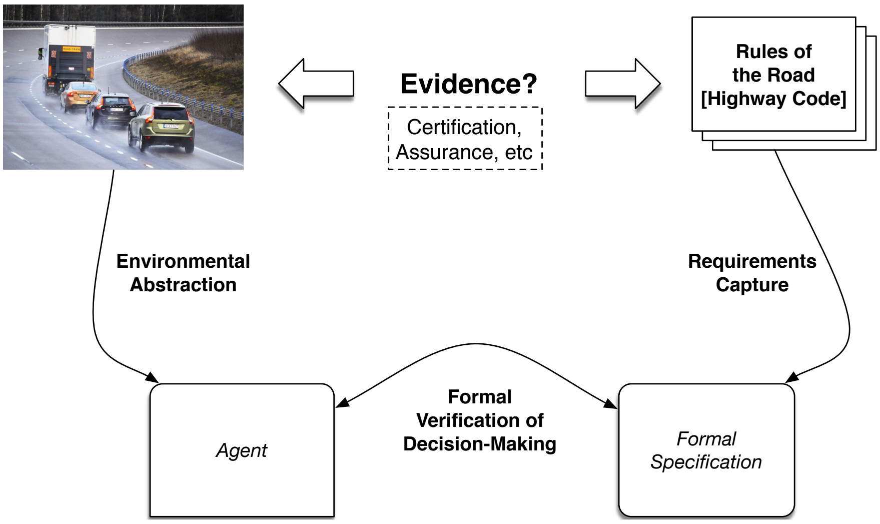
Symbolic Component

Rational Agent

Explicit Intentions, Beliefs, etc.

Fisher, Mascardi, Rozier, Schlingloff, Winikoff, Yorke-Smith — Towards a Framework for Certification of Reliable Autonomous Systems. Autonomous Agents and Multi-Agent Systems, 2021.

Benefits: Can Verify Rule Conformance



Evidence?

Certification, Assurance, etc

Rules of the Road [Highway Code]

Environmental Abstraction

Requirements Capture

Formal Verification of Decision-Making

Agent

Formal Specification

Alves, Dennis, Fisher — Formalisation of the Rules of the Road for embedding into an Autonomous Vehicle Agent. In Validation and Verification of Automated Systems Book, 2020, and Journal of Sensor and Actuator Networks, 2021.

Benefits: Can Verify Ethical Reasoning

Once the agent decisions take ethical concerns into account then we can extend formal verification to also assess these.

For example, we can formally verify that

if *a chosen course of action violates some substantive ethical concern, A*

then *the other available choices all violated some concern that was equal to, or more severe than, A.*

Dennis, Fisher, Slavkovik, Webster. Formal Verification of Ethical Choices in Autonomous Systems. Robotics & Autonomous Systems, 2016.

Bremner, Dennis, Fisher, Winfield — On Proactive, Transparent, and Verifiable Ethical Reasoning for Robots. Proceedings of the IEEE, 2019.

Summary: Teams and Trust

Teams are essentially social constructs and so using high-level agent approaches, typically used in social modelling/issues, makes sense rather than appealling to Engineering mechanisms.

Trust is subjective, but systems can be designed to be *trustworthy*.

Trustworthiness in standard CPS essentially equates to reliability.

Trustworthiness in Autonomous Systems adds *beneficiality*.

This requires transparency/truthfulness in intentions and (correspondingly) strong verification around behaviour

Dennis, Fisher. Verifiable Autonomous Systems. Cambridge University Press (in press)

Chatila, Dignum, Fisher, Giannotti, Morik, Russell, Yeung. Trustworthy AI. In Reflections on Artificial Intelligence for Humanity, 2021.

Sample Relevant Publications

- Amirabdollahian, et al. Can you Trust your Robotic Assistant? In *Proc. Int. Conf. Social Robotics*, 2013.
- Bremner, Dennis, Fisher, Winfield. On Proactive, Transparent, and Verifiable Ethical Reasoning for Robots. *Proceedings of the IEEE*, 2019.
- Cardoso, Farrell, Luckcuck, Ferrando, Fisher. Heterogeneous Verification of an Autonomous Curiosity Rover. In *Proc. NASA Formal Methods*, 2020.
- Chatila, Dignum, Fisher, Giannotti, Morik, Russell, Yeung. Trustworthy AI. In *Reflections on Artificial Intelligence for Humanity*, 2021.
- Dennis, Fisher. Verifiable Self-aware Agent-based Autonomous Systems. *Proceedings of the IEEE*, 2020.
- Dennis, Fisher, Lincoln, Lisitsa, Veres. Practical Verification of Decision-Making in Agent-Based Autonomous Systems. *Journal of Automated Software Engineering*, 2016.
- Ferrando, Cardoso, Fisher, Ancona, Franceschini, Mascardi. ROSMonitoring: A Runtime Verification Framework for ROS. In *Proc. Towards Autonomous Robotic Systems*, 2020.
- Ferrando, Dennis, Cardoso, Fisher, Ancona, Mascardi. Toward a Holistic Approach to Verification and Validation of Autonomous Cognitive Systems. *ACM Trans. Software Engineering Methodology*, 2021.
- Fisher, Dennis, Webster. Verifying Autonomous Systems. *CACM*, 2013.
- Fisher, Mascardi, Rozier, Schlingloff, Winikoff, Yorke-Smith. Towards a Framework for Certification of Reliable Autonomous Systems. *Autonomous Agents and Multi-Agent Systems*, 2021.
- Kamali, Dennis, McAree, Fisher, Veres. Formal Verification of Autonomous Vehicle Platooning. *Science of Computer Programming*, 2017.
- Luckcuck, Farrell, Dennis, Dixon, Fisher. Formal Specification and Verification of Autonomous Robotic Systems: A Survey. *ACM Computer Surveys*, 2019.
- Stocker, Dennis, Dixon, Fisher. Verifying Brahms Human-Robot Teamwork Models. In *Proc. JELIA*, 2012.
- Webster, Cameron, Fisher, Jump. Generating Certification Evidence for Autonomous Unmanned Aircraft Using Model Checking and Simulation. *Journal of Aerospace Information Systems*, 2014.
- Webster, Dennis, Dixon, Fisher, Stocker, Sierhuis. Formal Verification of Astronaut-Rover Teams for Planetary Surface Operations. In *Proc. IEEE Aerospace Conference*, 2020.
- Webster, Western, Araiza-Illan, Dixon, Eder, Fisher, Pipe. A Corroborative Approach to Verification and Validation of Human–Robot Teams. *International Journal of Robotics Research*, 2019.