



DOI:10.1145/3624699

As autonomous systems increasingly become part of our lives, it is crucial to foster trust between humans and these systems, to ensure positive outcomes and mitigate harmful ones.

BY DHAMINDA B. ABEYWICKRAMA, AMEL BENNACEUR, GREG CHANCE, YIANNIS DEMIRIS, ANASTASIA KORDONI, MARK LEVINE, LUKE MOFFAT, LUC MOREAU, MOHAMMAD REZA MOUSAVI, BASHAR NUSEIBEH, SUBRAMANIAN RAMAMOORTHY, JAN OLIVER RINGERT, JAMES WILSON, SHANE WINDSOR, AND KERSTIN EDER

On Specifying for Trustworthiness

AUTONOMOUS SYSTEMS (AS) are systems that involve software applications, machines, and people—that is, systems that can take action with little or no human supervision.³⁴ Soon, AS will no longer be confined to safety-controlled industrial settings. Instead, they will increasingly become part of our daily lives, having matured across various domains, such as driverless cars, healthcare robotics, and uncrewed aerial vehicles (UAVs). As such, it is crucial that these systems are trusted and trustworthy. Trust may vary, as it can be gained and lost over time. Different research disciplines define trust in different ways. This article focuses on the notion of trust that concerns the relationship between humans and AS. AS are considered *trustworthy* when the design, engineering, and operation of these systems generates positive outcomes and mitigates potentially harmful outcomes.³⁵ The trustworthiness of AS can

depend on many factors, such as explainability, accountability, and understandability to different users; robustness of AS in dynamic and uncertain environments; assurance of their design and operation through verification and validation (V&V) activities; confidence in their ability to adapt functionality as required; security against attacks on the systems, users, and deployed environment; governance and regulation of their design and operation; and consideration of ethics and human values in their deployment and use.³⁵

There are various techniques for demonstrating the trustworthiness of systems, such as synthesis, formal verification at design time, runtime verification or monitoring, and test-based methods. However, common to all these techniques is the need to formulate *specifications*. A specification is a detailed formulation that provides “a definitive description of a system for the purpose of developing or validating the system.”¹³ According to Kress-Gazit et al.,²⁹ writing specifications that capture trust is challenging. A human will only trust an AS to perform in a safe manner (that is, nothing bad happens) if it clearly and demonstrably acts in such a manner. This requires the AS to not only be safe, but also to be seen as safe by the human. In the same manner, it is equally important to ensure that the AS trusts the human.²⁹ To address this, specifi-

» key insights

- **Autonomous systems are increasingly becoming part of our daily lives. To demonstrate the trustworthiness of an autonomous system, we must first specify what is considered trustworthy.**
- **This article looks across a range of autonomous-systems domains and identifies some of their key specification challenges.**
- **Key intellectual challenges involved with specifying for trustworthiness in autonomous systems cut across the domains and are aggravated by the uncertainty in which the autonomous systems must operate.**



cations must go beyond typical functionality and safety aspects.

Engineering trustworthy and trusted AS involves different processes, technology, and skills than those required for traditional software solutions. Many practitioners in the AS or artificial intelligence (AI) domains have learned by accumulating experiences and failures across projects.¹ Best practices have started to emerge. There is increasing evidence of the need for rigorous specification techniques for developing and deploying AI applications.³ Even when not life-critical, actions and decisions made by AS may have serious consequences. If we are to use them in our businesses, at doctor's surgeries, on our roads, or in our homes, we must build AS that precisely satisfy the requirements of their stakeholders. However, specifying requirements for AS (AI in particular) remains more a craft than a science. For example, machine-learning (ML) applications are often specified based on optimization and efficiency measures rather than well-specified quality requirements that relate to stakeholder needs,²³ and further research is needed.

In the U.K. Research and Innovation (UKRI) Trustworthy Autonomous Systems (TAS) program, we conduct

cross-disciplinary fundamental research to ensure that AS are safe, reliable, resilient, ethical, and trusted. TAS is organized around six research projects called Nodes and a Hub; each Node focuses on the individual aspects of trust in AS, such as resilience, trust, functionality, verifiability, security, and governance and regulation.

Undertaking a community approach, this roadmap article is the result of the "Specifying for Trustworthiness" workshop held during the September 2021 TAS All Hands Meeting, which gathered a diverse group of researchers from all parts of the TAS program. Co-authored by a representative sample of the AS community in the U.K., this article highlights the specification challenges for AS with illustrations from a representative set of domains currently being investigated within our community. This article's main contribution is to identify key open research problems, termed 'intellectual challenges,' involved with specifying for trustworthiness in AS that cut across domains and are exacerbated by the inherent uncertainty involved with the environments in which AS need to operate. This article takes a broad view of specification, concentrating on top-level requirements including, but not limited

to, functionality, safety, security, and other non-functional properties that contribute to the trustworthiness of AS. Also, a discussion on the formalization of these specifications has intentionally been left for the future, when the understanding of what is required to specify for trustworthiness will be more mature.

To motivate and present the research challenges associated with specifying for trustworthiness in AS, the rest of this article is divided into three parts. The next section discusses a number of AS domains, each with its unique specification challenges. Then, the article presents key intellectual challenges currently being investigated within our community. Finally, the article summarizes our findings.

Autonomous Systems Domains and Their Specification Challenges

In this article, we classify AS domains based on two criteria: the number of autonomous agents (single or multiple) and whether humans are interacting with the AS as part of the system or the environment, following Schneiders et al.³⁸ Accordingly, we distinguish AS domains into four categories:

- ▶ A single autonomous agent (for example, automated driving, UAV)
- ▶ A group of autonomous agents (for example, swarms)
- ▶ An autonomous agent assisting a human (for example, AI in healthcare, human-robot interaction)
- ▶ A group of autonomous agents collaborating with humans (for example, emergency situations, disaster relief).


We discuss the specification challenges involved with AS using illustrations from a representative set of domains, as being investigated within our community in TAS (see the accompanying table), rather than attempting to cover all possible AS domains.

Single autonomous agent: Automated driving, UAV. Automated driving (self-driving) refers to a class of AS that varies in the extent to which they independently make decisions (SAE J3016 standard taxonomy). The higher levels of autonomy, levels 3–5, refer to functionality ranging from traffic jam chauffeur to completely hands-free driving in all conditions. Despite an


AS domains and their specification challenges.

Category	Domain	Specification Challenge
Single Autonomous Agent	Automated driving	How to address the lack of machine-readable specifications that formally express acceptable driving behavior.
	UAV	How to specify the actions of other road users. How to specify the ways the UAV should deal with situations that go beyond the limits of its training.
Multiple Autonomous Agents	Swarms	How to specify the emergent behavior of a swarm that is a consequence of the interaction of individual agents with each other and the environment.
Autonomous Agent Assisting a Human	Human-robot interaction	How to specify the perceptual, reasoning, and behavioral processes of robot systems. How to infer human mental states interactively.
	AI in healthcare	How to specify 'black box' models. What is the role of explainability and faithfulness of the interpretation of semantics? What is the role of pre-trained models in pipelines?
Multiple Autonomous Agents Collaborating with Humans	Emergency situations and disaster relief	How to specify collaboration between autonomous agents and different human agents in emergency settings.
		How to specify security where large amounts of data need to be collected, shared, and stored.

explosion of activity in this domain in recent years, the majority of systems being considered for deployments depend on careful delineation of the operation design domain to make the specification of appropriate behavior tractable. Even so, the specification problem remains difficult for a number of reasons. Firstly, traffic regulations are written in natural language, ready for human interpretation. Although highway code rules are intended for legal enforcement, they are not specifications that are suitable for machines. There are typically many exceptions, context-dependent conflicting rules, and guidance of an ‘open nature,’ all of which require interpretation in context. Driving rules can often be vague or even conflicting, and may need a base of knowledge to interpret the rule given a specific context. The U.K. Highway Code Rule 163 states that after you have started an overtaking maneuver you should “move back to the left as soon as you can but do not cut in.”⁶ A more explicit specification of driving conduct (for example, Rule 163) to something more machine interpretable that captures the appropriate behavior presents a challenge to this research area. When people are taught to perform this activity, a significant portion of the time is spent in elaborating these special cases, and much of the testing in the licensing regime is aimed at probing for uniformity of interpretation. How best to translate these human processes into the AS domain is important not only for achieving safety but also acceptability. Secondly, driving in urban environments is an intrinsically interactive activity, involving several actors whose internal states may be opaque to the automated vehicle. As an example, the U.K. Highway Code asks drivers to not “pull out into traffic so as to cause another driver to slow down.” Without further constraint on what the other drivers could possibly do, specifying appropriate behavior becomes difficult, and any assumptions made in that process would call into question the safety of the overall system when those assumptions are violated. Thus, two key challenges in the area of automated driving are the lack of machine-readable specifications that formally express acceptable driving behavior and the need to specify the actions of



Engineering trustworthy and trusted AS involves different processes, technology, and skills than those required for traditional software solutions.



other road users (see the accompanying table). To some extent, these issues arise in all open environments. However, in automated driving, the task is so intricately coupled with the other actors that even the default assumptions may not be entirely clear, and the relative variation in behavior due to different modeling assumptions could be qualitatively significant.

A UAV or drone is a type of aerial vehicle capable of autonomous flight without a pilot on board. UAVs are increasingly being applied in diverse applications, such as logistics services, agriculture, emergency response, and security. Specification of the operational environment of UAVs is often challenging due to the complexity and uncertainty of the environments that UAVs need to operate in. For instance, in parcel delivery using UAVs in urban environments, there can be uncertain flight conditions (for example, wind gradients) and highly dynamic and uncertain airspace (for instance, other UAVs in operation). Recent advances in ML offer the potential to increase the autonomy of UAVs in uncertain environments by allowing them to learn from experience. For example, ML can be used to enable UAVs to learn novel maneuvers to achieve perched landings in uncertain windy conditions.¹² In these contexts, a key challenge is how to specify the way the system deals with situations that go beyond the limits of its training (see the accompanying table).

Multiple autonomous agents: Swarm robotics. Swarm robotics provides an approach to the coordination of large numbers of robots, which is inspired from the observation of social insects.³⁷ Three desirable properties in any swarm robotics system are robustness, flexibility, and scalability. The functionality of a swarm is emergent (for example, aggregation, coherent ad hoc networks, taxis, obstacle avoidance, and object encapsulation)⁴⁵ and evolves based on the capabilities and number of robots used. The overall behaviors of a swarm are not explicitly engineered in the system, as they might be in a collection of centrally controlled robots, but they are an emergent consequence of the interaction of individual agents with each other and the environment. This

emergent functionality poses a challenge for specification. The properties of individual robots can be specified in a conventional manner, yet it is the emergent behaviors of the swarm that determine the performance of the system as a whole. The challenge is to develop specification approaches that specify properties at the swarm level that can be used to develop, verify, and monitor swarm-robotics systems.

Autonomous agent assisting a human: Human-robot interaction, AI in healthcare. Interactive robot systems aim to complete their tasks while explicitly considering the states, goals, and intentions of the human agents they collaborate with, and aiming to calibrate the trust humans have for them to an appropriate level. This form of human-in-the-loop, real-time interaction is required in several application domains, including assistive robotics for activities of daily living,¹⁵ healthcare robotics, shared control of smart mobility devices,⁴⁰ and collaborative manufacturing. Most specification challenges arise from the need to provide specifications for the perceptual, reasoning, and behavioral processes of robot systems that will need to acquire models of, and deal with, the high variability exhibited in human behavior. While several human-in-the-loop systems employ mental-state inference, the necessity for interactively performing such inference (including as beliefs and intentions), typically through sparse and/or sensor data from multimodal interfaces, imposes further challenges for the principled specification of human factors and data-driven adaptation processes in robots operating in close proximity to humans, where safety and reliability are critical.

Healthcare is a broad application domain which already enjoys the many benefits arising from the use of AI and AI-enabled autonomy. This has ranged from more accurate and automated diagnostics to a greater degree of autonomy in robot surgery, as well as entirely new approaches to drug discovery and design. The use of AI in medical diagnosis has advanced to an extent that in some settings, for example, mammography screening, automated interpretation seems to match human expertise in some trials. However,

there remains a gap in test accuracy. It has been argued that the automated systems are not sufficiently specific to replace radiologist double reading in screening programs.¹⁴ These gaps also highlight the main specification challenges in this domain. Historically, human expertise in this domain has not been explicitly codified, so it can be hard to enumerate desired characteristics. It is clear that the specifications must include notions of invariance to instrument and operator variations, coverage of condition and severity level, and so on. Beyond that, the semantics of the biological features used to make fine determinations are subject to both ambiguity or informality, and variability across experts and systems. Moreover, the use of deep learning to automate interpretation brings with it the need for explainability. This manifests itself in the challenge of guarding against shortcuts,⁴ wherein the AI diagnostic system achieves high accuracy by exploiting irrelevant side variables instead of identifying the primary problem (for example, radiographic COVID-19 detection using AI).⁴ The specific challenge here is how to specify with respect to ‘black box’ models. In this regard, we can highlight the role of explainability and faithfulness of interpretation of semantics, and the role of pre-trained models in pipelines (see the accompanying table).

Multiple autonomous agents collaborating with humans: Emergency situations, disaster relief. Emergency situations evolve dynamically and can differ in terms of the type of incident, its magnitude, additional hazards, and the number and location of injured people. They are also characterized by urgency; they require a response in the shortest timeframe possible and call for a coordinated response of emergency services and supporting organizations, which are increasingly making use of AS. This means that successful resolutions depend not only on effective collaboration between humans²⁵ but also between humans and AS. Thus, there is a need to specify both functional requirements and the social, legal, ethical, empathic, and cultural (SLEEC) rules and norms that govern an emergency scenario. AS in emergency response contexts vary hugely; as such,

the kinds of SLEEC issues pertaining to them must be incorporated into the design process rather than implemented afterward. This suggests a shift from a static design challenge toward the need to specify for adaptation to the diversity of emergency actors and complexity of emergency contexts, which are time-sensitive and involve states of exception not common in other open AS environments, such as autonomous vehicles. In addition, to enhance collaboration between autonomous agents and different human agents in emergencies, specifying human behavior remains one of the main challenges in emergency settings.

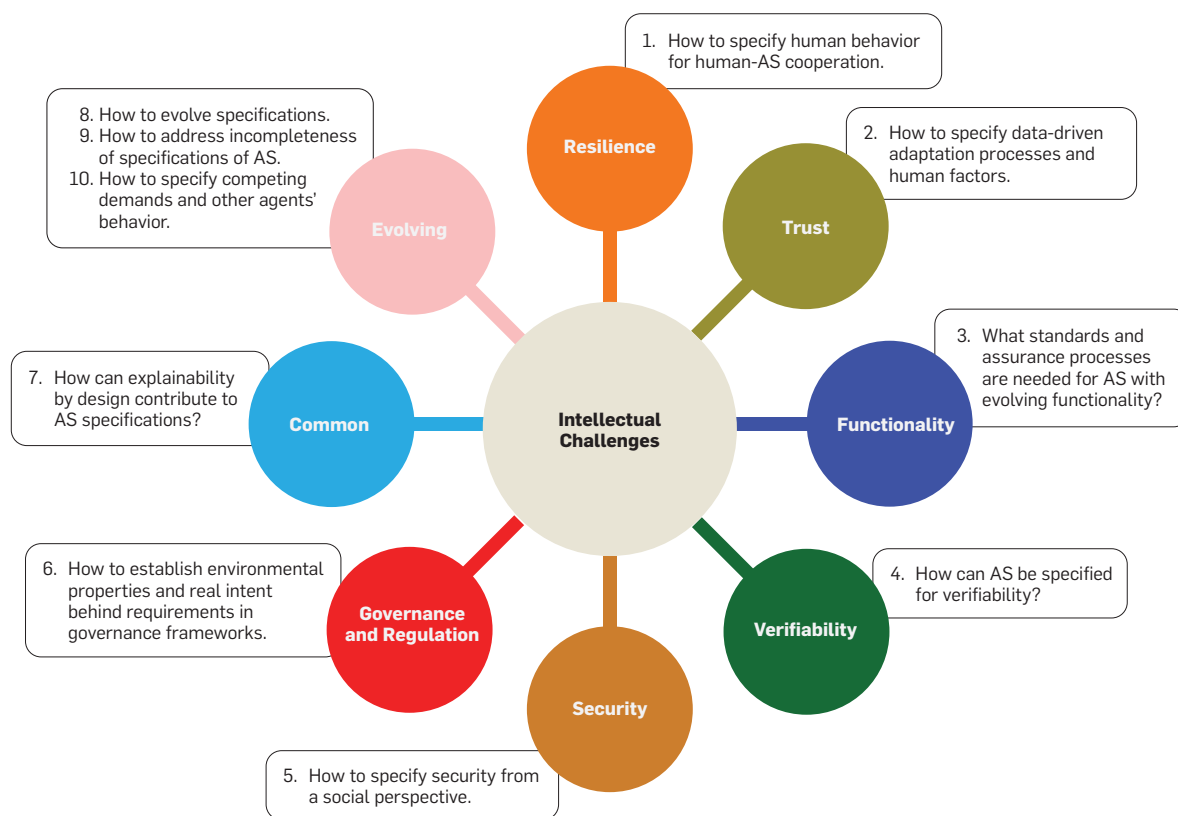
There are also challenges for specifying security in the context of disaster relief. A large part of this comes from the vast amounts of data that needs to be collected, shared, and stored between different agencies and individuals. Securing a collaborative information management system is divided between technical forms of security, such as firewalling and encryption, and social forms of security, such as trust. To provide security to a system, both aspects must be addressed in relation to each other within a specification.

Intellectual Challenges for the Autonomous Systems Community

The preceding section discussed specification challenges unique to a representative set of domains investigated within our community. Now we discuss 10 *intellectual challenges* involved with specifying for trustworthiness in AS that can cut across domains and are exacerbated by the inherent uncertainty involved with the environments in which AS need to operate. These challenges were identified during stimulating discussions among the speakers and participants of the breakout groups at the “Specifying for Trustworthiness” workshop.

Intellectual challenges 1–6 are in the six *focus areas* of trust in AS (that is, resilience, trust, functionality, verifiability, security, and governance and regulation), as identified by their respective speakers. Meanwhile, the remaining four challenges have either a *common focus* (7) across the TAS program, or they are *evolving* in nature (8–10) (see the accompanying figure). For each

Intellectual challenges for the AS community.



challenge, we provide an overview, identify high-priority research questions, and suggest future directions.

Many of the specification challenges to be discussed are shared by systems such as multi-agent systems, cyber-physical-social systems, or AI-based systems. Autonomy is an important characteristic of these systems and so is the need for trustworthiness. Specification challenges have also received a lot of attention in ‘non-AS’, for example, safety-critical systems. Yet, many of the challenges are exacerbated in AS because of the inherent uncertainty of their operating environments: They are long-lived, continuously running systems that interact with the environment and humans in ways that can hardly be fully anticipated at design time and continuously evolve at runtime. In other words, while those challenges are not specific to AS, AS exacerbate them.

1. How to specify human behavior for human-AS cooperation. How to model human behavior to enable cooperation with AS is challenging but

crucial for the resilience of the system as a whole. It is the diversity in human enactment that drives uncertainty about what people do and do not do, and subsequently, the way human behavior can be specified. Knowing the mental state of others enables AS to steer a cooperation that is consistent with the needs of the AS, as well as to respond to the needs of human agents in an appropriate manner.


Different theories of human behaviors explain diversity in human action in different ways and by detecting various determinants of human behavior. For example, a behaviorist approach suggests that every behavior is a response to a certain stimulus.²¹ Albeit true, this approach is restrictive in addressing the complexity of human behavior, as well as the different ways that human behavior develops during cooperation. To grasp that humans are embodied with purposes and goals that affect each other, the concept of joint-action can be introduced as “a social interaction whereby two or more individuals coordinate their ac-

tions in space and time to bring about change in the environment.”³⁹ Adapting it to human-robot interaction, this approach suggests an interplay between humans and AS, such that what matters is not only how the AS understands the system but also how humans understand the way the autonomous agent behaves and is willing to cooperate.¹⁷ Thus, cooperation arises from a shared understanding between agents, which is a challenge to specify.


The social identity approach⁴¹ induces this concept of a shared understanding by providing an explanation of human behavior focusing on how social structures act upon cognition. It proposes that, alongside our personal identity, our personality—who we are—we also have multiple social identities based on social categories and groups. Previous research has shown that social identities influence people’s relation with technology.³⁰ Sharing a social identity initiates pro-social behaviors, such as helping behaviors in emergency situations.⁷ People adapt their behavior in line

with their shared identities, which in turn, enhances resilience. Specifying social identities to enable cooperation is challenging. It requires answering questions such as: How do we represent different identities and how do we reason about them? Following the social-identity approach to specify identities for human-autonomous agent cooperation requires an investigation of how to operationalize social identity, a psychological state, into software embedded within AS.

2. How to specify data-driven adaptation processes and human factors. Specifying, designing, implementing, and deploying interactive robot systems that are trustworthy for use in scenarios where humans and robots collaborate in close proximity is challenging, given that safety and reliability in such scenarios are of particular importance. Examples include assisting people with daily living activities, such as mobility⁴⁰ and dressing;¹⁵ rehabilitation robotics; adaptive assistance in intelligent vehicles; and robot assistants in care homes and hospital environments. The intellectual challenge the AS community faces is the specification, design, and implementation of trustworthy perceptual, cognitive, and behavior-generation processes that explicitly incorporate parameterizable models of human skills, beliefs, and intentions.⁵ These models are necessary for interactive assistive systems since they must decide not only how but also when to assist.¹⁶ Given the large variability of human behavior, the parameters of these user models must be acquired interactively, typically from sparse and potentially noisy sensor data, a particularly challenging inverse problem. An additional challenge is introduced in the case of long-term human-robot interaction, where the assistive system must learn and take into consideration human developmental aspects, typically manifested in computational learning terms as model drift. As an example, consider an assistive mobility device for children with disabilities:⁴⁰ As the child's perceptual, cognitive, emotional, and motor skills develop over time, their requirements for the type, amount, and frequency of the provided assistance must evolve. Similarly, when as-



Many emerging concerns, such as fairness, are not only difficult to formalize in the sense of software specification, but also their many definitions can be conflicting.



sisting an elderly person or someone recovering from surgery, the distributions of the human data that the robot sensors collect will vary not only according to the context but also over time. Depending on the human participant, and their underlying time-varying physiological and behavioral particularities, model drift can be sudden, gradual, or recurring, posing significant challenges to underlying modeling methods. Principled methods for incorporating long-term human factors into the specification, design, and implementation of assistive systems that adapt and personalize their behavior for the benefit of their human collaborator remain an open research challenge.

3. What standards and assurance processes are needed for AS with evolving functionality? AS with *evolving functionality*—the ability to adapt and change in function over time—pose significant challenges to current processes for specifying functionality. Most conventional processes for defining system requirements assume that these are fixed and can be defined in a complete and precise manner before the system goes into operation. Existing standards and regulations do not accommodate the adaptive nature of AS with evolving functionality. This is a key limitation¹¹ preventing the deployment of promising applications, such as swarm robots which adapt through emergent behavior and UAVs with ML-based flight-control systems from deployment.

For airborne systems and in particular for UAVs, several industry standards and regulations have been introduced to specify requirements for system design and safe operation—for example, DO-178C, DO-254, ED279, ARP4761, NATO STANAG 4671, and CAP 722. However, none of these standards or regulations covers the types of ML-based systems currently being developed to enable UAVs to operate autonomously in uncertain environments.

The ability to adapt and learn from experience are important for enabling AS to operate in real-world environments. When one considers existing industry standards, they are either implicitly or explicitly based on the V&V model, which moves from

requirements through design into implementation, testing, and finally deployment.²⁶ However, this model is unlikely to suit systems with the ability to adapt their functionality in operation—for example, through interaction with other agents and the environment, as is the case with swarms, or through experience-driven adaptation, as is the case with ML. AS with evolving functionality follow a different, more iterative life cycle. Thus, there is a need for new standards and assurance processes that extend beyond design time and allow continuous certification at runtime.

4. How to specify AS for verifiability.

For a system to be *verifiable*, a person or a tool needs to be able to check its correctness¹³ with respect to its requirements and specification. The main challenge is in specifying and designing the system in a way that makes this process as easy and intuitive as possible. For AS in particular, specific challenges include capturing and formalizing requirements, including functionality, safety, security, performance and, beyond these, any additional non-functional requirements purely needed to demonstrate trustworthiness; handling flexibility, adaptation and learning; and managing the inherent complexity and heterogeneity of both the AS and the environment it operates in.

Specifications must represent the different aspects of the overall system in a way that is natural to domain experts, facilitates modeling and analysis, provides transparency of how the AS works, and offers insights into the reasons that motivate its decisions. To specify for verifiability, a specification framework will need to offer a variety of domain abstractions to represent the diverse, flexible, and possibly evolving requirements AS are expected to satisfy. Furthermore, the underlying verification framework should connect all these domain abstractions to allow an analysis of their interaction. This is a key challenge in specifying for verifiability in AS.

AS can be distinguished using two criteria: the degree of autonomy and adaptation, and the criticality of the application (which can range from harmless to safety-critical). We can consider which techniques or their

combinations are needed for V&V at the different stages of the system life cycle. The need for runtime V&V emerges when AS operate in uncontrolled environments, where there is a need for autonomy and learning and adaptation. There, a significant challenge is finding rigorous techniques for the specification and V&V of safety-critical AS, where requirements are often vague, flexible, and may contain uncertainty and fuzziness. V&V at design time can only provide a partial solution, and more research is needed to understand how best to specify and verify learning and adaptive systems by combining design-time with runtime techniques. Finally, identifying the design principles that enable V&V of AS is a key pre-requisite to promote verifiability to a first-class design goal alongside functionality, safety, security, and performance.

5. How to specify security from a social perspective. There are technical sides to security, but there are also social dimensions that matter when considering how an AS enforces its status as secure. In this context, security overlaps with trust. One can only be assured a system is secure if one trusts that system. Public trust is a complex issue, shot through with media, emotions, politics, and competing interests. How do we go about specifying security in a social sense?

On the technical side, there are fairly specific definitions for specification which can be grasped and measured. From the social perspective, the possibility of specification relies on a network of shared assumptions and beliefs that are difficult to unify. In fact, much of the value from engagement over social specifications derives from the diversity and difference. A predominant concern in social aspects of security is where data is shared between systems (social-material interactions)—that is, whenever an AS communicates with a human being or an aspect of the environment. Although these interactions have technical answers, finding answers that consider social-science perspectives requires collaboration and agile methods to facilitate that collaboration.

The human dimension means that it is not enough to specify technical

components. Specifications must also capture beliefs, desires, fears, and, at times, misinformation with respect to how those are understood, regarded, and perceived by the public. For example, in what ways can we regard pedestrians as passive users of automated vehicles? How are automated vehicles regarded by the public, and how are pedestrians involved in automated mobility?

The ethical challenges that emerge for AS security also relate to the legal and social ones. The difficulty centers around how to create regulations and specifications on a technical level that are also useful socially, facilitating responsiveness to new technologies that are neither simply techno-phobic nor passively accepting. Doing so must involve both innovation and public input, so the technology developed works for everyone. The ethical, legal, and social implications (ELSI) framework⁸ aims to engage designers, engineers, and public bodies in answering these questions. ELSI is an inherently cross-disciplinary set of approaches for tackling AS security, as many inter-related and entangled aspects. Specifying security requires connection, collaboration, and agile ethical methods.

6. How to establish environmental properties and real intent behind requirements in governance frameworks.

Computer scientists treat specifications as precise objects, often derived from requirements by purging features such that they are defined with respect to environment properties that can be relied on regardless of the machine's behavior. Emerging AS applications in human-centered environments can challenge this way of thinking, particularly because the environment properties may not be fully understood or because it is hard to establish if the real intent behind a requirement can be verified. These gaps should be addressed in governance frameworks to engender trust.

For instance, in all the domains mentioned, we are increasingly seeing systems that are data-first and subject to continuous deployment. This has the interesting consequence that sometimes the task requirements cannot be explicitly stated. Instead, they are only given in terms

of instances of observed human behavior,⁴² which represent positive examples. An example in medical diagnostics is when an AI-based AS has only a high-level label from the human radiologist, to be matched by the model, rather than detailed causal theories or justifications.¹⁴ We see this as a crucial area for future development, as existing workflows depend on human interpretation of rules in crucial ways, whereas when AS make the same decisions, there is scope for significant disruption of these workflows due to potential gaps that become exposed.

Furthermore, many emerging concerns, such as fairness, are not only difficult to formalize in the sense of software specification, but also their many definitions can be conflicting, such that it is impossible to satisfy all of them in a given system.³⁶

AS of the future will need a combination of informal and formal mechanisms for governance. In domains such as automated vehicles, system trustworthiness may require a complete ecosystem approach²⁷ involving community-defined scenario libraries, enabling the greater use of simulation in verification, and independent audits via independent third parties. This calls for developing new computational tools for performance and error characterization, systematic adversarial testing with respect to a range of different specification types, and causal explanations that address not only a single instance of a decision but better expose informational dependencies that are useful for identifying edge cases and delineating operational design domains.

In addition to these technical tools, there is a need to understand the human-machine context in a more holistic manner, as this is really the target of effective governance. People's trust in an AS is not solely determined by technical reliability. Instead, the expectations of responsibility and accountability are associated with the human team involved in the system's design and deployment and the organizational design behind the system. A vast majority of system failures arise from mistakes made in this 'outer loop.' Therefore, effective regulations must begin with a comprehensive

mapping of responsibilities that must be governed, so that computational solutions can be tailored to address these needs. Furthermore, there is a need for ethnographic understanding of AS being used in context, which could help focus technical effort on the real barriers to trustworthiness.

7. How can explainability by design contribute to AS specifications? There are increasing calls for explainability in AS, with emerging frameworks and guidance¹⁸ pointing to the need for AI to provide explanations about decision making. A challenge with specifying such explainability is that existing frameworks and guidance are not prescriptive: What is an actual explanation and how should one be constructed? Furthermore, frameworks and guidance tend to be concerned with AI in general, not AS.

A case study addressing regulatory requirements on explainability of automated decisions in the context of a loan application²² provided foundations for a systematic approach. Within this context, explanations can act as external detective controls, as they provide specific information to justify the decision reached and help the user take corrective actions.⁴³ But explanations can also act as internal detective controls—that is, a mechanism for organizations to demonstrate compliance to the regulatory frameworks they must implement. The study and design of AS includes many facets; not only black-box or grey-box AI systems, but also the system's various software and hardware components, the curation and cleansing of datasets used for training and validation, the governance of such systems, their user interface, and crucially the users of such systems with a view of ensuring that they do not harm but benefit these users and society in general. There are typically a range of stakeholders involved, from the system designers to their hosts and/or owners, their users (consumers and operators), third-parties, and, increasingly, regulators. In this context, many questions related to trustworthy AS must be addressed holistically, including:

- ▶ What is an actual explanation and how should one be constructed?

- ▶ What is the purpose of an explanation?

- ▶ What is the audience of an explanation?

- ▶ What is the information it should contain?^{22,43}

It no longer suffices to focus on the explainability of a black-box decision system. Its behavior must be explained, with more and less details, in the context of the overall AS. However, to adequately address these questions, explainability should not be seen as an afterthought but as an integral part of the specification and design of a system, leading to explainability requirements to be given the same level of importance as all other aspects of a system.

In the context of trustworthy AS, emerging AS regulations could be used to drive the socio-technical analysis of explainability. A particular emphasis would have to be on the autonomy and the handoff between systems and humans that characterizes trustworthy AS. The audience of explanations will also be critical, from users and consumers to businesses, organizations, and regulators. Finally, considerations for post-mortem explanations, in case of crash or disaster situations involving AS, should lead to adequate architectural design for explainability.

8. How to evolve specifications. Every typical AS undergoes changes over its lifetime that require going beyond an initially specified spectrum of operation—despite the observation that this spectrum is typically quite large for AS in the first place. The evolution of trustworthy AS may concern changes in the requirements of their functional or non-functional properties, changes of the environment that the AS operate in, and changes in the trust of users and third parties toward the AS.

Initial specifications of the AS may no longer reflect the system's desired properties or they may fail to accurately represent its environment. The evolution of specifications presents challenges in balancing the system's autonomy.


While any non-trivial system requires evolution and maintenance,³³ some challenges are exacerbated for trustworthy AS. As an example, observed changes in trust toward the AS might require changes to behavior

specifications, even if the AS operations are perfectly safe. Conversely, required changes to specifications might negatively impact future trust toward the AS. New methods will be required to efficiently deal with the various dimensions of trust in the evolution of specifications.


One dimension of trust relates to transparency toward developers of AS specifications. Approaches that compare evolving specifications on a syntactical level as currently done for code, or based on metrics as currently done for AI models, are unlikely to be sufficient for effective maintenance. Analysis will need to scale beyond syntactic differences to include semantic differences³¹ and allow for efficient analysis of the impact of changes on the level of systems rather than artifacts. New techniques to compare AS specifications are required that identify, present, and explain differences as well as their potential impact on the system's trustworthiness.

9. How to address incompleteness of AS specifications. Incompleteness is a common property of specifications. Only the use of suitable abstractions allows for coping with the complexity of systems.²⁸ However, there is an important difference in the incompleteness introduced by abstractions; the process of eliminating unnecessary detail to focus, for example, on behavioral, structural, or security-related aspects of a system; and the incompleteness related to the specification's purpose—that is, the faithful representation of the system in an abstraction.

On the one hand, if the purpose of creating and analyzing a specification is to examine an AS and learn about possible constraints, then incompleteness of the AS representation in the specification is important, as it allows for obtaining feedback with low investment in specification development²⁴—for example, for the reduction of ambiguities. On the other hand, if the purpose of the specification is to prove a property, then incompleteness of the AS representation may lead to incorrect analyses results manifesting in false positives or false negatives. False positives are often treated by adding the missing knowledge to the specification of AS—



In the context of trustworthy AS, emerging AS regulations could be used to drive the socio-technical analysis of explainability.



for example, verifying the specification of an infusion pump reported a false positive due to incompleteness.²⁰ The specification had to be changed to a “much more complex”²⁰ one to remove the false positive.

One way to address incompleteness is with partial models,^{9,10,44} where models and analyses are extended with modalities qualifying their completeness. The various approaches provide analysis of either syntactic properties⁹ or behavior refinements.^{10,44} Combinations and extensions to rich specification languages for AS are part of this research challenge.

In addition to analysis tasks, specifications are also used in synthesis tasks; this is where the incompleteness of AS specifications can manifest itself in the construction of biased or incorrect systems. As an example, consider the specification of a robot operating in a warehouse.³² The specification requires that the robot never hits a wall. With no assumptions about the environment, the synthesizer would take the worst-case view—that is, walls move and hit the robot—and consequently report that the specification is not realizable and no implementation exists. Adding the assumption that walls cannot move as an environment constraint changes the outcome of the synthesis. Interestingly, when formulating requirements for humans, common sense allows us to cope with this type of incompleteness. However, the automated analysis of specifications for AS brings with it the challenge of identifying and handling (all) areas of incompleteness.

10. How to specify competing demands and other agents' behavior. Conventional approaches to V&V for AS may seek to attain coverage against a specification to demonstrate assurance of functionality and compliance with safety regulations or legal frameworks. Such properties may be derived from existing legal or regulatory frameworks, for example, the U.K. Highway Code for driving, which can then be converted into formal expressions for automatic checking.¹⁹


But optimal safety does not imply optimal trust, and just because an AS follows rules does not mean it will be accepted as a trustworthy sys-

tem in human society. Other factors of trustworthiness should be considered, such as reliability, robustness, cooperation, and performance. We can also say that strictly following safety rules may even be detrimental to other trustworthiness properties—for example, performance. Consider an automated vehicle trying to move through a busy market square full of people slowly walking across the road and uncommitted to the usual observation of road conduct. The *safest* option for the AS is to wait until the route ahead is completely clear before moving on, as by taking this option you do not endanger any other road user. However, *better performance* may be to creep forward in a bid to promote your likelihood of success. Driving then, is much more than following safety rules, which makes this a particularly hard specification challenge. In this scenario, an assertive driving style would make more progress than a risk-averse one.


In reality there will be significantly more considerations than just safety and performance, but this example illustrates the principle of conflicting demands between assessment standards. Consideration of other agents, such as properties of fairness or cooperation, would lead to a more trustworthy system. Additionally, the interaction of AS with people may require insight into social norms of which there is no written standard by which these can be judged. Will the task of specification first require a codex of social-interaction norms to be drawn together to add to the standards by which trust can be measured? Specifications would need to be written with reference to these standards, regulations, and ethical principles, some of which do not currently exist, to ensure that any assessment captures the full spectrum of these trustworthiness criteria.

Conclusion

As autonomous systems play greater roles in our daily lives and interact more closely with humans, we need to build systems worthy of trust regarding safety, security, and other non-functional properties. In this article, we have first examined AS domains of different levels of maturity



As autonomous systems play greater roles in our daily lives and interact more closely with humans, we need to build systems worthy of trust.



and then identified their specification challenges and related research directions. One of these challenges is the formalization of knowledge easily grasped by humans so that it becomes interpretable by machines. Prominent examples include the specification of driving regulations for AVs, and the specification of human knowledge expertise in the context of AI-based medical diagnostics. How to specify and model human behavior, intent, and mental state is a further challenge common to all domains where humans interact closely with AS, such as in human-robot collaborative environments in smart manufacturing. Alternative approaches involve the specification of norms to characterize the desired behavior of AS, which regulate what the system should or should not do. An emerging research direction is the design of monitors to observe the system and check compliance with norms.² The example of swarm robotics raises the need and challenge to specify behavior that emerges at the system level and relies on certain actions of the entities that form the system with each other and their environment.

Beyond the technical aspects, across the specific AS domains, are research challenges related to governance and regulation for trustworthiness, requiring a holistic and human-centered approach to specification focused on responsibility and accountability, and enabling explainability from the outset. Fundamental to specifying for trustworthiness is a sound understanding of human behavior and expectations, as well as the social and ethical norms applicable when humans directly interact with AS. As for future work, an interesting extension of this article would be to produce a classification of properties to be specified for trustworthiness under the different intellectual challenges discussed—for example, socio-technical properties of explainability are purpose, audience, content, timing, and delivery mechanism of explanations.

We conclude that specifying for trustworthiness requires advances on the technical and engineering side, informed by new insights from social sciences and humanities research.

Thus, tackling this specification challenge necessitates tight collaboration of engineers, roboticists, and computer scientists with experts from psychology, sociology, law, politics, economics, ethics, and philosophy. Most importantly, continuous engagement with regulators and the general public will be key to trustworthy AS.

Acknowledgments

This work has been supported by the U.K. EPSRC under the grants: [EP/V026518/1], [EP/V026607/1], [EP/V026747/1], [EP/V026763/1], [EP/V026682/1], [EP/V026801/2], [EP/S027238/1] and [EP/V00784X/1]. A.B. and B.N. are also supported by EPSRC [EP/R013144/1] and SFI [13/RC/2094_P2]. Y.D. is also supported by a RAEng Chair in Emerging Technologies [CiET1718/46]. M.M. is also supported by EPSRC [EP/Y005244/1].

References

- Amershi, S. et al. Software engineering for machine learning: A case study. In *Proc. of the 41st Intern. Conf. on Software Engineering: Software Engineering in Practice*. IEEE/ACM (2019), 291–300; 10.1109.
- Criado, N. Resource-bounded norm monitoring in multi-agent systems. *J. Artificial Intelligence Res.* 62, 1 (May 2018), 153–192; 10.1613/jair.11206.
- D'Amour, A. et al. Underspecification presents challenges for credibility in modern machine learning. *CoRR* (2020); <https://arxiv.org/abs/2011.03395>
- DeGrave, A.J., Janizek, J.D., and Lee, S. AI for radiographic COVID-19 detection selects shortcuts over signal. *Nature Machine Intelligence* (2021), 1–10.
- Demiris, Y. Prediction of intent in robotics and multi-agent systems. *Cognitive Processing* 8, 3 (2007), 151–158; 10.1007/s10339-007-0168-9.
- Department for Transport. The highway code—Using the road (159 to 203), 2022; <https://bit.ly/49nAFbh>.
- Drury, J. The role of social identity processes in mass emergency behaviour: An integrative review. *European Rev. of Social Psychology* 29, 1 (2018), 38–81.
- Escalante, M.A.L. et al. Is IT ethical? Board game: Playing with speculative ethics of IT innovation in disaster and risk management. In *Proc. of the IX Latin American Conf. on Human Computer Interaction*. ACM (2019), 8; 10.1145/3358961.3358962. Article 9
- Famelis, M., Salay, R., and Chechik, M. Partial models: Towards modeling and reasoning with uncertainty. In *34th Intern. Conf. on Software Engineering*. IEEE Computer Society (2012), 573–583; 10.1109/ICSE.2012.6227159.
- Fischbein, D. et al. Weak alphabet merging of partial behavior models. *ACM Trans. Software Engineering Methodology* 21, 2 (2012), 9:1–9:47; 10.1145/2089116.2089119.
- Fisher, M. et al. Towards a framework for certification of reliable autonomous systems. *Autonomous Agents and Multi-Agent Systems* 35, 8 (2021), 65; 10.1007/s10458-020-09487-2.
- Fletcher, L., Clarke, R., Richardson, T., and Hansen, M. Reinforcement learning for a perched landing in the presence of wind. In *AIAA SciTech 2021 Forum*. American Institution of Aeronautics and Astronautics (2021); 10.2514/6.2021-1282.
- International Organization for Standardization. ISO/IEC/IEEE 24765:2017 systems and software engineering—Vocabulary; <https://www.iso.org/standard/71952.html>
- Freeman, K. et al. Use of artificial intelligence for image analysis in breast cancer screening programmes: Systematic review of test accuracy. *BMJ* 374 (2021); 10.1136/bmj.n1872.
- Gao, Y., Chang, H.J., and Demiris, Y. User modelling using multimodal information for personalised dressing assistance. *IEEE Access* 8 (2020), 45700–45714; 10.1109/ACCESS.2020.2978207.
- Georgiou, T. and Demiris, Y. Adaptive user modelling in car racing games using behavioural and physiological data. *User Modelling and User-Adapted Interaction* 27 (2017), 267–311; 10.1007/s11257-0179192-3.
- Grigore, E.C. et al. Joint action understanding improves robot-to-human object handover. In *Proc. of the 2013 IEEE/RSJ Intern. Conf. on Intelligent Robots and Systems* (2013), 4622–4629; 10.1109/IROS.2013.6697021.
- Hamon, R., Junklewitz, H., and Sanchez, I. Robustness and explainability of artificial intelligence. *Technical Report*. Publications Office of the European Union (2020); 10.2760/57493.
- Harper, C. et al. Safety validation of autonomous vehicles using assertion-based oracles. *arXiv* (2021); 10.48550/ARXIV.2111.04611.
- Harrison, M.D., Masci, P., Campos, J.C., and Curzon, P. Verification of user interface software: The example of use-related safety requirements and programmable medical devices. *IEEE Trans. Hum. Mach. Systems* 47, 6 (2017), 834–846; 10.1109/THMS.2017.2717910.
- Heimlich, J.E. and Ardoin, N.M. Understanding behavior to understand behavior change: A literature review. *Environmental Education Research* 14, 3 (2008), 215–237.
- Huynh, T.D. et al. Addressing regulatory requirements on explanations for automated decisions with provenance: A case study. *Digital Government: Research and Practice* 2, 2 (Jan. 2021); 10.1145/3436897.
- Ishikawa, F. and Matsuno, Y. Evidence-driven requirements engineering for uncertainty of machine learning-based Systems. In *Proceedings of the 28th IEEE Intern. Requirements Engineering Conf.* (2020), 346–351.
- Jackson, D. Alloy: A language and tool for exploring software designs. *Commun. ACM* 62, 9 (2019), 66–76; 10.1145/3338843
- James, K. The organizational science of disaster/terrorism prevention and response: Theory-building toward the future of the field. *J. of Organizational Behavior* 32, 7 (2011), 1013–1032.
- Jia, Y., McDermid, J., Lawton, T., and Habli, I. The role of explainability in assuring safety of machine learning in healthcare. *arXiv* (2021); 10.48550/ARXIV.2109.00520.
- Koopman, P. et al. *Certification of Highly Automated Vehicles for Use on UK Roads: Creating an Industry-Wide Framework for Safety*. Five AI Ltd. (2019).
- Kramer, J. Is abstraction the key to computing? *Communications of the ACM* 50, 4 (2007), 36–42; 10.1145/1232743.1232745.
- Kress-Gazit, H. et al. Formalizing and guaranteeing human-robot interaction. *Communications of the ACM* 64, 9 (Aug. 2021), 78–84; 10.1145/3433637.
- Lee, Y., Lee, J., and Lee, Z. The effect of self identity and social identity on technology acceptance. In *Proc. of the Intern. Conf. on Information Systems* 59, (2001); <https://aisel.aisnet.org/icis2001/59/>.
- Maoz, S. and Ringert, J.O. A framework for relating syntactic and semantic model differences. *Software and Systems Modeling* 17, 3 (2018), 753–777; 10.1007/s10270-016-0552-y.
- Maoz, S. and Ringert, J.O. On the software engineering challenges of applying reactive synthesis to robotics. In *Proc. of the 1st Intern. Workshop on Robotics Software Eng.*, ACM (2018), 17–22; 10.1145/3196558.3196561.
- Mens, T. Introduction and roadmap: History and challenges of software evolution. In *Software Evolution*. T. Mens and S. Demeyer (Eds.), Springer (2008), 1–11; 10.1007/978-3-540-76440-3_1.
- Murukannaiah, P.K., Ajmeri, N., Jonker, C.M., and Singh, M.P. New foundations of ethical multiagent systems. In *Proc. of the 19th Intern. Conf. on Autonomous Agents and MultiAgent Systems*, Intern. Found. for Autonomous Agents and Multiagent Systems (2020), 1706–1710.
- Naiseh, M., Bentley, C.M., and Ramchurn, S. Trustworthy autonomous systems (TAS): Engaging TAS experts in curriculum design. In *Proc. of the 2022 IEEE Global Eng. Education Conf.*; 10.48550/ARXIV.2202.07447.
- Narayanan, A. 21 fairness definitions and their politics. In *Proceedings of the Conf. on Fairness, Accountability, and Transparency*, ACM (2021).
- Şahin, E. Swarm robotics: From sources of inspiration to domains of application. In *Swarm Robotics*, E. Şahin and W. Spears (Eds.). Springer Berlin Heidelberg, (2005), 10–20.
- Schneiders, E. et al. Non-dyadic interaction: A literature review of 15 years of human-robot interaction conference publications. *J. Human-Robot Interactions* 11, 2, Article 13, (Feb. 2022), 32; 10.1145/3488242.
- Sebanz, N., Bekkering, H., and Knoblich, G. Joint action: Bodies and minds moving together. *Trends in Cognitive Sciences* 10, 2 (2006), 70–76.
- Soh, H. and Demiris, Y. Learning assistance by demonstration: Smart mobility with shared control and paired haptic controllers. *J. Human-Robot Interactions* 4, 3 (Dec. 2015), 76–100; 10.5898/JHRI.4.3.Soh.
- Spears, R. Social influence and group identity. *Annual Rev. of Psychology* 72 (2021), 367–390.
- Topol, E.J. High-performance medicine: The convergence of human and artificial intelligence. *Nature Medicine* 25, 1 (2019), 44–56.
- Tsakalakis, N. et al. The dual function of explanations: Why it is useful to compute explanations. *Computer Law and Security Rev.* 41 (Mar. 18. 2021); 10.1016/j.clsr.2020.105527.
- Wei, O., Gurfinkel, A., and Chechik, M. On the consistency, expressiveness, and precision of partial modeling formalisms. *Inf. Comput.* 209, 1 (2011), 20–47; 10.1016/j.ic.2010.08.001.
- Winfield, A.F.T. and Nembrini, J. Safety in numbers: Fault-tolerance in robot swarms. *Intern. J. on Modelling Identification and Control* 1, 1 (2006), 30–37; 10.1504/IJMIC.2006.008645.

Dhaminda Abeywickrama (dhaminda.abeywickrama@bristol.ac.uk) is a research fellow in the Department of Computer Science at the University of Bristol, U.K.

Amel Bennaceur is an associate professor of Computing at the Open University, U.K., and a senior research fellow at Lero Ireland.

Greg Chance is a research fellow in the Department of Computer Science at the University of Bristol, U.K.

Yiannis Demiris is a professor in Human-Centered Robotics at the Department of Electrical and Electronic Engineering, Imperial College, London, U.K.

Anastasia Kordoni is a senior research analyst at Trilateral Research Ltd, U.K., and a senior research associate in Psychology at Lancaster University, U.K.

Mark Levine is a professor of Social Psychology at Lancaster University, U.K.

Luke Moffat is a senior research associate in Sociology at Lancaster University, U.K.

Luc Moreau is a professor of Computer Science and Head of the Department of Informatics at King's College London, U.K.

Mohammad Reza Mousavi is a professor of Software Engineering at King's College London, U.K.

Bashar Nuseibeh is a professor of Computing at the Open University, U.K., and chief scientist at Lero Ireland.

Subramanian Ramamoorthy is a professor of Robot Learning and Autonomy, and director of the Institute of Perception, Action and Behaviour, within the School of Informatics at the University of Edinburgh, U.K.

Jan Oliver Ringert is a professor of Software Engineering at the Bauhaus-University Weimar, Germany.

James Wilson is a research associate in the Department of Engineering Mathematics at the University of Bristol, U.K.

Shane Windsor is an associate professor of Bio-Inspired Aerodynamics at the University of Bristol, U.K.

Kerstin Eder is a professor of Computer Science at the University of Bristol, U.K.

© 2024 copyright held by the owner/author(s).