The RAS Safety Framework

Nikita Johnson Needle[1], James Douthwaite[1], Yunus Emre Cogurcu[1], Nicholas Hall[2], and James Law[1]

1 Sheffield Robotics, The University of Sheffield, UK
2 Health & Safety Executive, UK

KEYWORDS: RAS safety, collaborative robot safety, structured argumentation, assurance framework

ABSTRACT

There exist several emerging approaches for systematic safety assurance of robotics and autonomous systems. These range from implementation of processes and safety cases, to low-level analysis of particular autonomous functions. Even with these advancements, there remains a paucity of guidance of how to identify, capture and communicate hazards that emerge as a result of autonomous system operation. This paper presents the RAS Safety Framework – an end-to-end assurance process that makes use of digital twins and constraint rules to represent safety claims and provide evidence for them. The RAS Safety Framework represents a fundamental shift in paradigm of assuring 'behavioural hazards'. The approach is applied to a case study to demonstrate the utility and benefit of having a structured, traceable safety case and digital twin for complex autonomous systems.

1 INTRODUCTION

Unlike many existing approaches to safety assurance of Robotic and Autonomous Systems (RAS), that rely on detailed prior knowledge of the operational environment and full control over its agents, the approach in this paper allows for dynamic reasoning about unsafe behaviours in more open environments. This shift from static to dynamic assurance is crucial for maximising the potential of mobile robotics and cobots. Rather than relying solely on physical barriers and separation, this method identifies system actors, actions, and behaviours within a given context, then imposes rules to prevent unsafe actions. Behavioural hazards, emerging from specific actions, are managed using safety and security assurance rules. By clearly defining these rules, operational behaviour can be monitored, for instance, via a Digital Twin (DT), to detect rule violations and mitigate hazards. These rules form the basis of safety and security claims, and during runtime the claims are supported by data and verification evidence from the system or DT, thereby providing a structured run-time RAS Assurance Case.

2 RECENT ADVANCEMENTS IN RAS SAFETY

The increasing deployment of RAS) necessitates robust safety assurance methodologies for dynamic and open environments. This review explores key advancements in structured argumentation, digital twins, and safety hazard management, essential for comprehensive safety frameworks.

Safety assurance in RAS requires robust methodologies for system reliability in dynamic environments. Kelly and Weaver [1] introduced goal-structured notation (GSN) for structured safety cases, while Hawkins and Kelly [2] integrated GSN with assurance case patterns to describe common safety argumentation strategies. Bloomfield and Bishop [3] describe structured argumentation with evidence collection, enabling real-time data integration in Safety Cases. Another crucial element to modern RAS safety assurance is the use of Digital Twins (DTs) for predicting behaviour, generating synthetic data and understanding real-time system behaviour. Grieves and Vickers [4] highlighted DTs potential for predictive maintenance, and Rosen et al. [5] used digital twins to effectively monitor safety-critical systems. Tao et al. [6] developed a digital twin-driven design framework for enhanced safety, and Fuller et al. [7] highlighted their benefits for dynamic risk assessment and real-time safety assurance.

Hazards that emerge as a result of system and process interactions are not unique to autonomous systems, however due to the automation these types of hazards pose new challenges for safety. Koopman and Wagner [8] discuss some of these challenges, especially around safety testing and validation. To address some of these challenges for RAS, Buysse et al [9] apply the System-Theoretic Process Analysis (STPA) for hazard mitigation, influencing RAS dynamic safety claims. Furthermore, Douthwaite et al. [10] and Gleirscher et al. [11] present advanced methodologies for enhancing safety in collaborative robotic environments. Douthwaite et al. propose a modular digital twinning framework that integrates Digital Models and Real-World Data for real-time monitoring and safety assessment, standardizing communication across hardware platforms and validating safety

claims virtually. Gleirscher et al. introduce a tool-supported approach for synthesizing, verifying, and testing safety control software, emphasizing safety controller correctness under explicit assumptions, informed by risk analysis and safety regulations.

State-of-the-art for RAS Safety underscores the importance of digital tools and rigorous verification methods in improving human-robot interaction safety but what is still missing is a systematic approach to linking these elements together. The RAS Safety Framework introduces a dynamic approach to safety assurance, utilising digital twins and structured argumentation for real-time monitoring and management of behavioural hazards. This methodology allows continuous assessment and updating of safety cases, incorporating safety and security into a unified process. The framework systematically defines and enforces safety rules, enhancing risk mitigation and ensuring safety claims are supported by current evidence.

3 RAS SAFETY FRAMEWORK OVERVIEW

The RAS Safety Framework (SF) is proposed in this paper, consists of a structured assurance case and the assurance process. The RAS SF process implements a Dynamic Safety Case for industrial applications using a Digital Twin, a one-to-one model of the real-world application that receives sensor data. Sensors within the Digital Twin can generate additional data during operation, which is processed and monitored in real-time, updating the evidence for claims in the Safety Case. The main benefits of this Framework include the ability to update the safety case at run-time in response to changes in the operational environment, such as behavioural hazards or application reconfiguration. This allows robotic systems to be repurposed with reduced cost and effort after the initial Digital Twin setup. The next sections provide further detail about the assurance case and process.

3.1 RAS SF Assurance Case

The Assurance Case component of the RAS Safety Framework is represented by the GSN model in Figure 1. The assurance case is a risk-based structured argument comprising claims supported by evidence, focusing on safety and security. Each risk is identified and managed using safety controls, which map to safety requirements and trace to system requirements, implemented in software or hardware.

The top-level claim G1 asserts that RAS is acceptably safe and secure throughout its lifetime based on requirements from standards, regulations, or customers. The concept of acceptable safety is derived from UK legislation and the ALARP principle, emphasizing proportional risk reduction. Claim G2 states that RAS is acceptably safe by addressing hazards listed in C1, including environmental, energetic, mechanical, and behavioural hazards. Behavioural hazards, unique to the RAS Safety Framework, result from interactions during operation that could cause harm. Traditional systems with 'caged robots' do not consider behavioural hazards, relying on physical barriers to prevent human-robot contact. Behavioural hazards include a robot failing to detect and avoid a moving human, misjudging its speed when approaching an obstacle, or incorrectly prioritizing tasks that lead to simultaneous actions causing a safety incident. An integral part of the RAS SF Assurance Case is the Safety Rules (G6 and C3), bridging high-level safety requirements (G5) and functional system requirements (G7). This approach can model behavioural hazards in the Digital Twin (DT) and integrate results with other cobot safety methods using safety controllers or policies.

3.2 RAS SF Assurance Process

The RAS Safety Framework Process shown in Figure 2 ensures the safety and security of robotic and autonomous systems through a structured, iterative approach. It begins with defining entities in the operational environment, including actors and their characteristics. Actions during operation and their associated dynamic tags are identified in parallel with the operational behaviours that emerge from these actions. This mapping forms the basis for risk analysis, where potential behavioural hazards are formalized. Assurance rules are established to constrain hazardous behaviours, followed by a risk assessment to evaluate the acceptability of these risks. The overall process involves developing assurance claims, forming a dynamic safety case that updates with changes in the operational environment. Real-time data from sensors and digital twins enable continuous monitoring and adaptation to new hazards. The final steps include risk evaluation and applying necessary treatments to maintain safety margins. This iterative process, guided by the Plan-Do-Check-Act cycle, ensures continuous improvement and robust safety assurance in dynamic environments. Responsibilities for each step primarily lie with a safety engineer, with input from system, software, and hardware engineers. Note that each of these steps can be performed without the Digital Twin, however to gain the dynamic safety assurance benefits, Steps 1-3 and elements of Step 6 must be instantiated in the Twin.

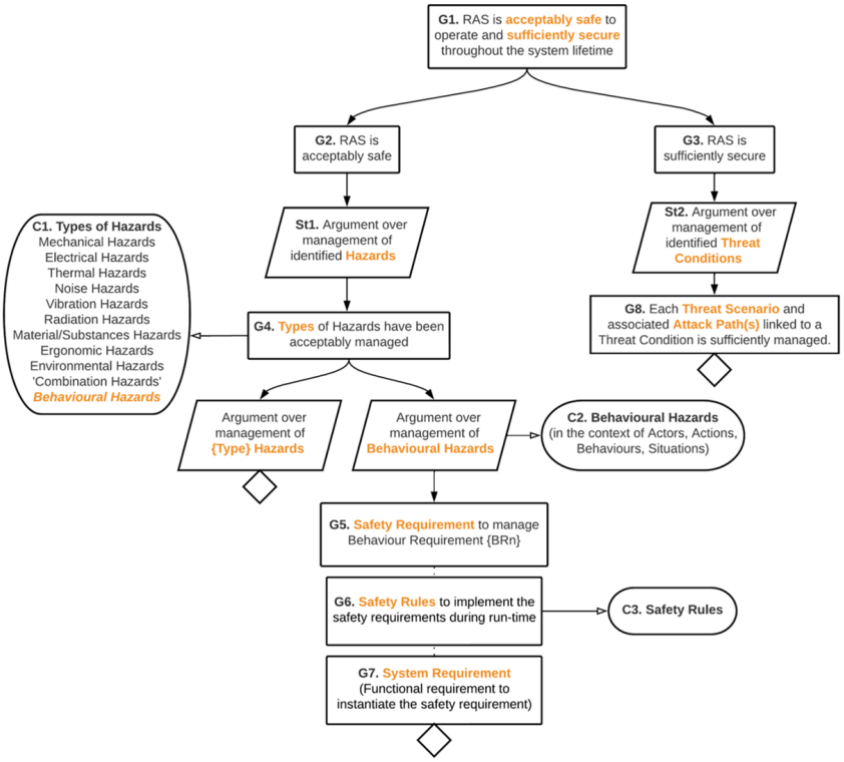

Figure 1. *RAS Safety Framework Generic Assurance Case Structure*.

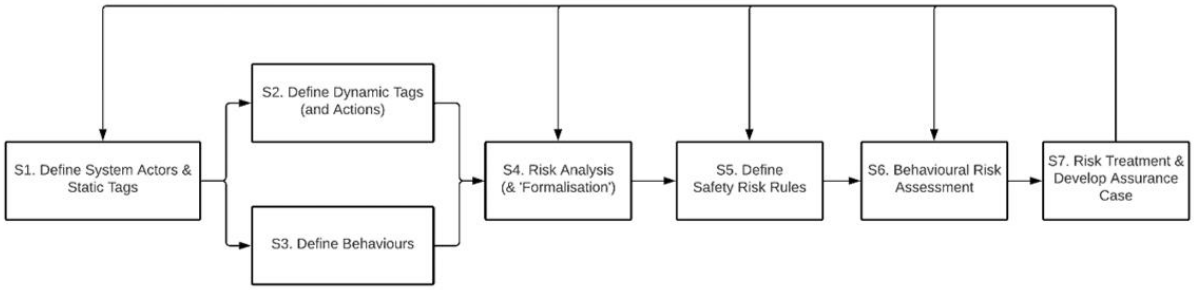

Figure 2. *RAS Safety Framework Process Steps*.

4 CASE STUDY EVALUATION

This section provides a case study applying the RAS Safety Framework process and elements of the assurance case. The scenario is based on a human-robot welding process which utilises a Universal Robot (UR10) to transfer a component part between a shared collaborative workspace the welding machine. In this process the robot is separated from the operator by a workbench and a small light barrier. The human assembles a component and passes it through the light barrier, placing it on a shelf within reach of the robot. Once the operator has withdrawn, and the light barrier is no longer broken, the robot picks up the assembly, performs a weld, returns the assembly to the shelf and withdraws. The operator then reaches through the light barrier to retrieve the welded assembly, and replaces it with another. In addition to the light barrier there is a rangefinder at shin-height at the base of the welding machine that detects anyone who has entered the cell and a camera sensor mounted above the cell. The RAS SF Process steps applied to this scenario involve:

- **Step 1**: The process begins with defining entities within the operational environment, including actors such as the human, welder, table, robot arm, and robot base, each tagged with properties like canHarm, emitsHeat, isSurface, isMobile, and isStationary.
- **Step 2**: Actions for each entity are then defined, such as the human moving, the welder increasing temperature, the table puncturing, and the robot arm moving.
- **Step 3**: These actions are used to identify operational behaviours like welding parts, human transferring parts from bin to table, robot transferring parts from table to welder, and human assembling parts.

— **Step 4**: Risk analysis is conducted to identify potential hazards, including robot colliding with a human, human tripping over a parts bin, and human being burned by the welding machine.
— **Step 5**: Safety rules are then established to prevent these hazards. For instance, to avoid collisions, a rule is set that the human and robot must not have contact. Similarly, to prevent tripping, it is ruled that a moving human and stationary objects must not be in contact, and to avoid burns, a human must not contact objects with increased temperature.
— **Step 6**: The next step involves risk assessment, where the likelihood and impact of each behaviour are evaluated. The overall risk is calculated as the product of the likelihood and impact of these behaviours.
— **Step 7**: Assurance claims are then developed based on the safety rules and evidence from sensors. These claims ensure that safety rules are continuously monitored and enforced using real-time data. For example, the claim G6 asserts that the system ensures no human-robot contact based on sensor data.

In the welding process, light barriers and sensors detect human presence and prevent robot movement if a human is detected in the vicinity. This approach ensures that behavioural hazards, such as collisions or burns, are continuously monitored and mitigated in real-time, maintaining an acceptable level of safety.

Table 1. *Behavioural Hazards, Safety Rules and Risk Calculation Example.*

Behavioural Hazard	Safety Rule	SR Formal Logic Representation	Likelihood	Impact	Risk Calculation	Category
Robot collides with human	Human and robot must not have contact	¬(Human ∧ Robot ∧ Contact)	Low	High	Low x High = Medium	Collision
Human trips over parts bin	Human moving and stationary actor must not be in contact	¬(HumanMoving ∧ StationaryActor ∧ Contact)	Medium	Medium	Medium x Medium = Medium	Tripping
Human is burned by weld machine	Human must not contact hot objects	¬(Human ∧ Contact ∧ HotObject)	Low	High	Low x High = Medium	Burn

Table 1 captures key information for managing behavioural hazards in the human-robot welding process. Each row details a specific behavioural hazard examples and the associated safety rule designed to prevent it. The formal logic representation column provides a logical expression of the safety rule, these must be defined by the process engineer. Likelihood indicates the probability of the hazard occurring, while impact assesses the severity of the hazard if it does occur. Risk calculation is derived from the product of likelihood and impact, giving an overall measure of risk. Finally, the category column classifies the type of hazard, such as collision, tripping, or burn. This structured approach ensures comprehensive risk management and safety assurance for dynamic cases.

Table 2. *Safe Rules Linked to Assurance Case Claims Example.*

Safety Claim	Safety Rule	Link between Safety Claim and Rule
G4.1: RAS is acceptably safe and secure	Human and robot must not have contact	Ensures physical separation to prevent collisions.
G4.2: Robot operation is safe during welding	Human moving and stationary actor must not be in contact	Prevents tripping accidents during robot and human interaction.
G4.3: Welding process does not harm the operator	Human must not contact hot objects	Prevents burns by ensuring no contact between the human and the welding machine.
G4.4: Environmental safety is maintained	Sensors must detect human presence	Ensures the robot stops when a human is detected, maintaining a safe environment.
G4.5: Operational safety rules are enforced	Light barriers must be functional	Verifies that the light barriers are operational to prevent hazardous situations.
G4.6: Continuous monitoring and adaptation	Real-time data must be collected and analysed	Supports dynamic adjustment of safety protocols based on real-time sensor data.

For the cobot weld scenario, Table 2 outlines high-level safety claims for the system, representing claims at G4 and lower from the generic assurance case (Figure 1). Table 2 provides specific rules designed to enforce the

safety claims. Each link between a claim and a rule explains how the safety rule supports the corresponding safety claim, thereby creating a clear bridge between system data and the safety requirements.

4.1 Industrial Systems Development Context

The risk and assurance processes are integral to the overall system or industrial platform design. An end-to-end development model is shown in Figure 3. The RAS Safety Framework is seamlessly integrated into the industrial development process, particularly within the V-lifecycle model, which is widely used for developing complex systems. This integration ensures that safety and security considerations are embedded from the initial stages of system development.

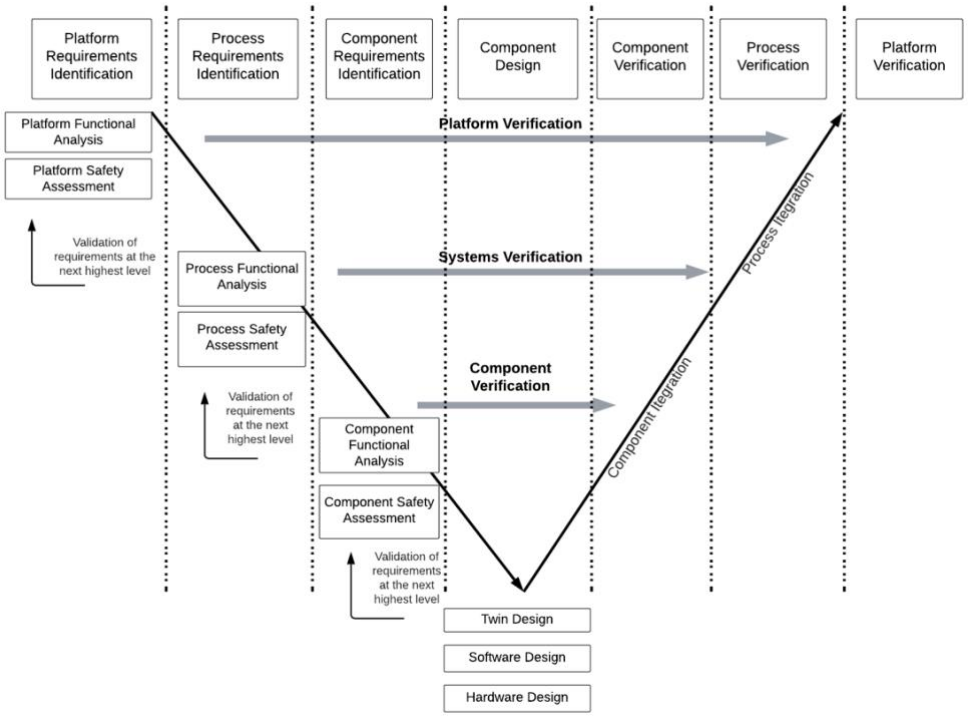

***Figure 3**. RAS Safety Framework and Digital Twin through the System V Life-cycle.*

1. Requirements Definition Phase: During this phase, RAS assurance requirements are established alongside system functional requirements, creating a comprehensive basis for subsequent design activities. For example, in developing an autonomous welding robot, safety requirements might include preventing collisions between the robot arm and human operators, while security requirements could include safeguarding the system from unauthorized access.
2. Analysis & Design Phase: The RAS Safety Framework uses digital twins to model the system's behaviour and interactions in a virtual environment. For instance, the design of the autonomous welding robot includes creating a digital twin to simulate the robot's movements and test various safety scenarios, allowing engineers to identify potential hazards and design safety controls to mitigate them.
3. Implementation Phase: During this phase, the digital twin validates and verifies the system's functionality and safety measures in real-time. Sensors and control systems continuously monitor operations, and the digital twin collects data to validate against safety and security requirements. For example, if a potential collision is detected, the system can trigger an emergency stop.
4. Verification and Validation Phase: The system undergoes rigorous testing using the digital twin to verify that it performs safely under various conditions. This includes stress testing for potential failures and collecting evidence to support the safety claims made in the assurance case, ensuring compliance with industry standards.
5. Deployment and Maintenance Phase: Continuous monitoring and updates are facilitated through the digital twin, which operates alongside the physical system to detect and respond to emerging hazards. For instance, if new machinery is introduced, the digital twin can simulate these changes and update the safety case accordingly.

The primary benefit of this approach is the ability to perform verification and validation activities during both development and run-time, offering flexibility and robust assurance for RAS systems. While establishing a

safety case and linked digital twin may have a high initial overhead, it enhances system flexibility and reconfigurability, supporting ongoing safety and security assurance throughout the system's lifecycle.

5 CONCLUSION

The RAS Safety Framework offers a novel approach to safety assurance for robotic and autonomous systems by shifting from traditional static methods to a dynamic model using digital twins and constraint (safety) rules. This framework enables real-time monitoring and mitigation of 'behavioural hazards,' ensuring safety claims are continuously supported by evidence collected during system operation. It allows for ongoing assessment and updates to the safety case, adapting to changes in the operational environment and providing a clear methodology for addressing unsafe behaviours in dynamic settings.

Key benefits of the approach include real-time verification and validation through the DT, which provides accurate models of real-world applications. This approach supports flexibility and reconfiguration of robotic systems at reduced cost and effort. By defining and monitoring safety rules, the framework can dynamically respond to operational changes, effectively managing complex interactions and behaviours. Additionally, it incorporates both safety and security considerations into the assurance case, offering a comprehensive understanding of risks.

However, there remain some open challenges for the Framework. Setting up the digital twin and developing the assurance case can be resource-intensive and complex, especially in highly dynamic environments. There is also a heavy reliance on real-time data collection and analysis which requires advanced infrastructure and robust sensor integration. Despite these challenges, the RAS Safety Framework represents a significant advancement in safety assurance, offering a dynamic and evidence-based approach to managing and assuraing behavioural hazards in autonomous systems.

6 REFERENCES

1. Kelly, T. and Weaver, R., 2004, July. The goal structuring notation–a safety argument notation. In *Proceedings of the dependable systems and networks 2004 workshop on assurance cases* (Vol. 6). Princeton, NJ
2. Hawkins R. D., Kelly T., A Systematic Approach for Developing Software Safety Arguments, Proc. of the 27th Int. System Safety Conf., Huntsville, United States, 2010, pp. 25-33.
3. Bloomfield, R. and Bishop, P., 2009, December. Safety and assurance cases: Past, present and possible future–an Adelard perspective. In *Making Systems Safer: Proceedings of the Eighteenth Safety-Critical Systems Symposium, Bristol, UK, 9-11th February 2010* (pp. 51-67). London: Springer London.
4. Grieves, M. and Vickers, J., 2017. Digital twin: Mitigating unpredictable, undesirable emergent behavior in complex systems. *Transdisciplinary perspectives on complex systems: New findings and approaches*, pp.85-113.
5. Rosen, R., Von Wichert, G., Lo, G. and Bettenhausen, K.D., 2015. About the importance of autonomy and digital twins for the future of manufacturing. *Ifac-papersonline*, *48*(3), pp.567-572.
6. Tao, F., Cheng, J., Qi, Q., Zhang, M., Zhang, H. and Sui, F., 2018. Digital twin-driven product design, manufacturing and service with big data. *The International Journal of Advanced Manufacturing Technology*, *94*, pp.3563-3576.
7. Fuller, A., Fan, Z., Day, C. and Barlow, C., 2020. Digital twin: Enabling technologies, challenges and open research. *IEEE access*, *8*, pp.108952-108971.
8. Koopman, P. and Wagner, M., 2016. Challenges in autonomous vehicle testing and validation. *SAE International Journal of Transportation Safety*, *4*(1), pp.15-24.
9. Buysse, L., Vanoost, D., Vankeirsbilck, J., Boydens, J. and Pissoort, D., 2022, September. Case study analysis of STPA as basis for dynamic safety assurance of autonomous systems. In *European Dependable Computing Conference* (pp. 37-45). Cham: Springer International Publishing.
10. Douthwaite J. A., Lesage B., Gleirscher M., Calinescu R., Aitken J. M., Alexander R., Law J., A modular digital twinning framework for safety assurance of collaborative robotics, Frontiers in Robotics and AI, Vol. 8, 2021, pp. 758099.
11. Gleirscher M., Calinescu R., Douthwaite J., Lesage B., Paterson C., Aitken J., Alexander R., Law J., Verified synthesis of optimal safety controllers for human-robot collaboration, Science of Computer Programming, Vol. 218, 2022, pp. 102809.