INSYTE: A Classification Framework for Traditional to Agentic AI Systems

ZOE PORTER*, Department of Computer Science, University of York, UK
RADU CALINESCU*, Department of Computer Science and Centre for Assuring Autonomy, University of York, UK
ERNEST LIM*, Ufonia Ltd., UK and Centre for Assuring Autonomy, University of York, UK
VICTORIA HODGE, Centre for Assuring Autonomy, University of York, UK
PHILIPPA RYAN, Centre for Assuring Autonomy, University of York, UK
SIMON BURTON, Centre for Assuring Autonomy, University of York, UK
IBRAHIM HABLI, Centre for Assuring Autonomy, University of York, UK
TOM LAWTON, Improvement Academy, Bradford Teaching Hospitals NHS Foundation Trust, UK
JOHN MCDERMID, Centre for Assuring Autonomy, University of York, UK
JOHN MOLLOY, Centre for Assuring Autonomy, University of York, UK
HELEN MONKHOUSE, HORIBA MIRA Ltd., UK
PHILLIP MORGAN, York Law School, University of York, UK
PAUL NOORDHOF, Department of Philosophy, University of York, UK
COLIN PATERSON, Department of Computer Science and Centre for Assuring Autonomy, University of York, UK
ISOBEL STANDEN, Department of Philosophy, University of York, UK
JIE ZOU, Centre for Assuring Autonomy, University of York, UK

Existing classification frameworks for artificial intelligence (AI) and autonomous systems are being outpaced by recent advancements in AI technologies. This limits their applicability to modern intelligent systems, particularly *agentic AI systems* (autonomous systems that leverage foundation models to achieve wide-ranging, multi-layered goals). To address this deficiency, we introduce INSYTE, a multi-faceted framework that supports the classification of AI systems ranging from traditional rule-based systems to cutting-edge embodied AI and agentic systems. To that end, INSYTE considers the essential characteristics of an AI system across eight key dimensions grouped into four categories: system design (*underspecification* and *adaptiveness*), functionality (*breadth* and *depth*), operating environment (*diversity* and *dynamism*) and independence from human operational control (*intervention* and *oversight*). Different AI systems (or versions of systems) yield different "patterns" on an eight-axis radar chart that INSYTE uses to provide an immediate visual summary of an AI system's overall capability, and a detailed representation of its individual characteristics. The INSYTE

*Corresponding author.

Authors' addresses: Zoe Porter*, Department of Computer Science, University of York, UK, zoe.porter@york.ac.uk; Radu Calinescu*, Department of Computer Science and Centre for Assuring Autonomy, University of York, UK, radu.calinescu@york.ac.uk; Ernest Lim*, Ufonia Ltd., UK and Centre for Assuring Autonomy, University of York, UK, el@ufonia.com; Victoria Hodge, Centre for Assuring Autonomy, University of York, UK, victoria.hodge@york.ac.uk; Philippa Ryan, Centre for Assuring Autonomy, University of York, UK, philippa.ryan@york.ac.uk; Simon Burton, Centre for Assuring Autonomy, University of York, UK, simon.burton@york.ac.uk; Ibrahim Habli, Centre for Assuring Autonomy, University of York, UK, ibrahim.habli@york.ac.uk; Tom Lawton, Improvement Academy, Bradford Teaching Hospitals NHS Foundation Trust, UK, tom.lawton@bthft.nhs.uk; John McDermid, Centre for Assuring Autonomy, University of York, UK, john.mcdermid@york.ac.uk; John Molloy, Centre for Assuring Autonomy, University of York, UK, john.molloy@york.ac.uk; Helen Monkhouse, HORIBA MIRA Ltd., UK, helen.monkhouse@horiba-mira.com; Phillip Morgan, York Law School, University of York, UK, phillip.morgan@york.ac.uk; Paul Noordhof, Department of Philosophy, University of York, UK, paul.noordhof@york.ac.uk; Colin Paterson, Department of Computer Science and Centre for Assuring Autonomy, University of York, UK, colin.paterson@york.ac.uk; Isobel Standen, Department of Philosophy, University of York, UK, isobel.standen@york.ac.uk; Jie Zou, Centre for Assuring Autonomy, University of York, UK, jie.zou@york.ac.uk.

framework aligns with OECD's definition of deployed AI systems, which is becoming the standard definition used by legislators and developers worldwide.

Additional Key Words and Phrases: AI system, AI-enabled system, autonomous system, artificial intelligence, agentic AI, taxonomy, classification framework

1 INTRODUCTION

Since the first 'Levels of Automation' taxonomy was used to categorise factory floor automation in the 1950s [15] and undersea robots in the late 1970s [114], the increasing operational independence of robotic and software-enabled systems has typically been framed in terms of a single hierarchy of levels, with the highest level reached when the human operator is completely "out of the loop" [129]. This framing has remained the dominant paradigm for automated systems enabled by artificial intelligence (AI) – such as self-driving vehicles, uncrewed maritime vessels, and surgical robots – and is now often referred to as the 'Levels of Autonomy'. The Levels of Autonomy have helped diverse stakeholders compare systems and plan development roadmaps [33, 34].

However, the Levels of Autonomy and related taxonomies of autonomy also have increasingly constraining limitations, including a lack of precision and an inability to express many of the characteristics, and combinations of characteristics, embodied by advanced AI-enabled systems. This is particularly the case for *agentic AI systems*, i.e., *"AI systems that can pursue complex goals with limited direct supervision"* [1, 113]. Like traditional autonomous systems, agentic AI systems have a direct impact on the world, achieving objectives on behalf of humans [69] rather than merely aiding human decision-making [21, 22, 113]. The key additional factor is that the core functionality of agentic AI systems is enabled by *frontier AI models* [5], i.e., models at the cutting edge of AI research and development. Today's frontier models include foundation models (also known as *large language models*) based on transformer architectures [130], such as the GPT, Claude and Gemini family of models. They have led to an unprecedented scaling and diversification of intelligent capabilities [119]. New and emerging AI and autonomous systems accomplish complex, wide-ranging objectives; pursue missions based on only limited, high-level specifications; mitigate uncertainty and change using sophisticated adaptation tactics; and operate in highly diverse, open-world dynamic environments. These abilities are qualitatively different and present to varying degrees within a given system—yet they are often conflated by the rigid definitions of autonomy levels within current taxonomies, if they are considered by these taxonomies at all. The INtelligent SYsTEms (INSYTE) classification framework introduced in our paper overcomes these shortcomings, enabling the accurate and nuanced classification of a wide range of AI and AI-enabled autonomous systems. As such, INSYTE provides an analytic tool that facilitates robust, straightforward, and comprehensive evaluations of these systems.

The INSYTE framework comprises three key components. The first component is the breakdown of an AI system's essential characteristics into eight dimensions, each of which may be instantiated at a level between 0 and 5. At Level 0, a system would possess little to none of the characteristic represented; at Level 5, it would possess the characteristic to the highest physically and technically possible degree. The framework does not prioritise any of the eight dimensions, although users may choose to do so, depending on their purpose. The eight dimensions are grouped into four categories: system design (*underspecification* and *adaptiveness*); system functionality (*breadth* and *depth*); operating environment (*diversity* and *dynamism*); and independence from human operational control (*intervention* and *oversight*). Together, these eight dimensions give a contextual, 'whole system' view of the system.

The second component of the INSYTE framework is a process for determining, for each of the eight dimensions, at what level a given system should be described. This involves systematically working through the level descriptions given for each dimension, and selecting the appropriate levels for the system, guided by our online open-source worksheet and the illustrative worked examples provided in the Appendices AI -A3 to this paper.

The third component is the depiction of the system on a radar chart (also known as a cobweb model, spider diagram, or Kiviat figure [68]). The eight dimensions are represented as eight axes on the radar chart, with a system's position on each axis established by its level (between 0 and 5) of the respective characteristic. The "INSYTE pattern" that a system yields on the radar chart conveys the combination of characteristics the system instantiates and to what degree. It therefore constitutes its visual classification. Our open-source worksheet [60] culminates in a radar chart generator for creating and downloading INSYTE patterns. This is described further in the tool support section (Section 4) of the paper, with worked examples in Appendices AI - A3.

To ensure the broad relevance of our INSYTE framework, we developed its components using a systematic, multi-stage methodology. A preliminary version of the framework was assembled based on insights we gained from our Assuring Autonomy International Programme's 25 demonstrator projects [8], in which health and social care, automotive, maritime, manufacturing, mining, space, agriculture, aviation, and quarrying AI and AI-enabled autonomous systems were prototyped and/or studied by teams of researchers, practitioners, regulators, policy makers, and other stakeholders between 2019–2024. We supplemented these insights by considering what key system characteristics are highlighted in existing autonomy-based classification frameworks, such as the Levels of Autonomy. To ensure that INSYTE's dimensions are appropriate for state-of-the-art AI systems, we further refined the framework through consulting the growing body of literature on agentic AI. The preliminary version of the INSYTE framework underwent internal testing and revision by a subteam of authors who were not directly involved in defining the levels for its eight dimensions, nor in the development of its online tool support. The revised framework was then evaluated in two rounds by academic and non-academic stakeholders with different levels of experience, and drawn from a wide range of application domains and disciplines. After the first round of evaluation, the framework was fine tuned, based on feedback received, to improve the clarity of the level descriptions and the usability of the process for second-round evaluators.

Terminology. We use the term 'system' for a set of interconnected components that work as a unit to achieve a specific purpose, for instance a vehicle, vessel, chatbot or decision-support tool. The term 'AI-enabled system' (or, interchangeably, 'intelligent system') is used to denote a system comprising at least one component that employs AI techniques that play an important role in delivering the core functionality of that system. Many terms from AI and robotics are either contested or vague and, despite significant contributions to the debate [40, 70, 77, 78, 82], a shared understanding of notions such as 'autonomy' and 'agency' in systems engineering is still largely lacking. To stipulate, in this paper, the terms 'autonomy' (or 'autonomous') and 'agency' (or 'agent') refer to 'the capacity to accomplish objectives independently of human operational control' and 'the capacity to accomplish complex objectives independently of human operational control in complex environments', respectively [22, 82, 90]. No position is taken in philosophical debates on whether AI and AI-enabled systems are literally intelligent or can truly instantiate the powers of the human mind [76, 97, 111, 123, 135], nor do we make assumptions about their potential for moral autonomy and moral agency [50]. Finally, we use the term 'whole system' to refer to the wider operational context of the intelligent system, including its functionality and operating environment.

Organisation of the paper. The paper is organised as follows. Section 2 places the INSYTE framework in the context of the related literature, specifically autonomy-based classification frameworks. This section identifies the key features of these frameworks, and explains how INSYTE overcomes their limitations. In Section 3, we present the INSYTE framework. Starting with an overview of the framework's three components in Section 3.1, the paper then details its eight dimensions in Section 3.2, providing systematic descriptions of Levels 0 to 5 for each dimension. Section 4 introduces our freely available online INSYTE tool, which incorporates a worksheet for users to work through the process of selecting a given system's level on each of INSYTE's eight dimensions, and a radar chart generator to create and download their system's INSYTE pattern. In Section 5, we present the evaluation of the framework, conducted to assess its usability and usefulness. In Section 6, the insights of our evaluators are supplemented with our own reflections on the foreseeable uses of the INSYTE framework, drawing on the authors' multidisciplinary backgrounds. The paper also includes an appendix providing three worked examples of the application of the INSYTE framework to different AI systems and their variants.

2 RELATED CLASSIFICATION FRAMEWORKS

The classification of AI and autonomous systems is an important problem. The effective classification of these systems is crucial for guiding research, deployment, regulation, risk management, and standardisation. Researchers and policymakers have therefore made significant efforts to establish well-defined classification frameworks for these technologies. While some frameworks classify AI and autonomous systems according to their risk or their impact on human values [88, 93, 125], INSYTE falls into a category of classification frameworks we call 'autonomy-based'. In this section, we summarise the two main classes of autonomy-based classification frameworks, and explain how our INSYTE framework overcomes their increasingly significant limitations.

2.1 Unidimensional 'Levels of Autonomy' frameworks

The majority of autonomy-based classification frameworks are 'Levels of Autonomy' (LoA) frameworks, with most of these grouping different characteristics of autonomy into unified levels. We refer to this class as 'unidimensional LoA frameworks'. Each system classified by such a framework is assigned a single, overarching level. Unidimensional LoA frameworks are prevalent across diverse sectors, including automotive (e.g., SAE's Levels of Driving Automation [108], and also [2, 46, 103, 127, 128]), maritime [4, 17, 20, 59, 63, 75, 105, 114], agriculture [32], aerospace [3, 6, 13, 24, 94, 109, 112, 139], defence [122, 126], manufacturing [37, 45, 87], healthcare [39, 41, 51, 62, 72, 118, 121, 137], mining [47], rail [61, 99], and space [79, 134]. Additionally, human-computer interaction research has yielded several sector-neutral unidimensional LoA frameworks [10, 35, 70].

The central thread running through all unidimensional LoA frameworks is a system's progressive independence from operator intervention. How this independence is interpreted varies between frameworks. Some define higher autonomy by a system's ability to make decisions and execute actions without human intervention [4, 17, 24, 31, 33, 35, 110, 112], while others prioritise minimal human involvement, such as only for fallback control [13, 72, 103, 115, 137]. Still other frameworks consider sustained independent operation over time as a key indicator of higher autonomy [32, 105, 136].

Several unidimensional LoA frameworks are 'contextual' frameworks [53], meaning that they consider the system's task and operating environment as well as its operational independence [11, 51, 73, 89, 104, 108]. Some consider the complexity of the task performed by the system, but not that of its environment [24, 32, 40, 41, 43, 72, 126, 134, 137, 139]; others consider the complexity of the environment but disregard the complexity of the task [2, 6, 17, 27, 37, 42, 45–47, 70, 87, 122]. Few frameworks in this class explicitly consider details of system design, such as whether the system is based on encoded rules or machine learning [65, 100], whether its outputs for a given input are fixed or can improve over time [32], or whether it can deal with 'edge cases' it was not programmed or trained for [32].

Limitations and comparison to INSYTE. While unidimensional LoA frameworks have long supported multi-stakeholder discussions about the range of automation options and optimal function allocations between human and machine [33–35], they have some major limitations. These shortcomings are summarised below, with an explanation of how the INSYTE framework overcomes each one.

Rigidity. The arbitrary groupings of characteristics into single, predefined levels does not allow for a distinct classification of many possible systems. In contrast, INSYTE's breakdown into eight separate dimensions allows for the independent consideration of individual system characteristics. The synthesis of a system's instantiation of these characteristics in a radar chart then allows for a classification based on its specific combination of characteristics. As such, INSYTE offers a flexible classification framework that can differentiate between systems that occupy the same level on a unidimensional LoA. Indeed, there are 6^8 conceptual possibilities of a system's combination of variable characteristics on the INSYTE framework.

Insufficient detail. Unidimensional LoA frameworks do not go into much detail about important system characteristics. This is evident, for example, in the fact that the automotive industry uses terms like Level 2+ and Level 2++ to overcome regulatory uncertainty around the definition of Level 3 [44]. Moreover, even the few unidimensional LoAs that consider system design, like [42, 65, 100], overlook the significant variation in the degree of specification of learned models. They also fail to differentiate between different dimensions of task or environmental complexity. Thanks to its eight dimensions, INSYTE does not suffer from this limitation. In addition to considering two aspects of operational independence, it separates the complexity of a system's operating environment and its functionality into two distinct dimensions, and a dedicated underspecification dimension ranks AI and AI-enabled autonomous systems by how their requirements are defined: through encoded rules, learned from data, reward-guided trial and error, or high-level objectives.

Inadequacy for classifying agentic AI. Distinctive features of agentic AI systems are inadequately captured by existing unidimensional LoA frameworks. These features include: the minimal explicit specification underlying their performance (they are often driven by reinforcement learning and self-supervised learning) [1, 21]; the capacity for adaptiveness [1, 113]; and the ability to multi-task across different subjects [1, 113]. Unidimensional LoAs also lack the detailed dimensional breakdown needed to fully capture the operational context and dynamism relevant to agentic AI [1]. INSYTE, on the other hand, is deliberately designed to support the classification of agentic AI systems. The highest levels on our framework's underspecification dimension correspond to reduced or minimal explicit specification. INSYTE's adaptiveness dimension measures the degree to which a system can handle increasing levels of uncertainty and change. The breadth of functionality dimension captures a system's range of task types and versatility. Finally, environmental dynamism is included as a distinct INSYTE dimension.

Insufficiently precise language. Regulatory and standards adoption can be inhibited by the imprecise formulation of unidimensional LoA level definitions. For instance, during UK Parliamentary debates on the bill that became the Automated and Electric Vehicles (AEV) Act, the Government minister concerned stated that: *"The SAE levels lack the precision needed for technical standards and are not currently recognised as a technical standard in either the technical committee or the forum looking at use within the UNECE [(United Nations Economic Commission for Europe)]"* [80, 124], although this LoA has been used in some US state legislation [23]. In contrast, INSYTE offers distinct descriptions for each level of each of its eight dimensions. These level descriptions are abstract, but precise.

Proneness to misuse. The insufficient detail of unidimensional LoAs may facilitate the downplaying of risk. For instance, a system might be classified as low autonomy (e.g., ‘just’ a Level 1 or Level 2) and therefore low risk, while its actual risk and controllability factors associated with other variable characteristics, such as underspecification or environmental dynamism, are overlooked. This can increase the difficulty for the human operator to ensure safety [9, 71]. Conversely, systems are sometimes over-marketed, inflating perceived capabilities (e.g., "Autopilot" for an SAE Level 2 vehicle), which has been shown to unjustifiably increase users’ perceptions of a system’s safety [120]. By offering detail about more system characteristics, INSYTE may help spotlight when and where systems are described as exhibiting more or less sophistication than they actually embody. It also offers a framework for immediately cross-referencing a system’s sophistication against the level of intervention and monitoring expected of its human operator.

Inconsistency between sectors. Discrepancies between level descriptions make it difficult to compare systems in different domains using unidimensional LoA frameworks. For example, at the time of writing, an LoA taxonomy in the automotive sector states that a Level 3 autonomous car has a human in fallback operational control [108], whereas a taxonomy in the maritime sector states that a Level 3 autonomous ship does not [63]. Against a backdrop in which, we note, classifications are evolving in many domains, INSYTE represents a universal framework standardised across application domains.

2.2 Multidimensional classification frameworks

A few autonomy-based classification frameworks explicitly distinguish multiple dimensions of a system’s autonomy. A key exemplar is the ALFUS Framework, produced for the US National Institute for Standards in Technology (NIST) in 2005 [56, 57], which classifies systems along three separate dimensions: mission complexity; environmental complexity; and independence from a human operator.

In 2000, Parasuraman et al. introduced a Levels of Autonomy (LoA) framework differentiating four system functions: information acquisition, analysis, decision/action selection, and action implementation [95]. This mirrors the sense-understand-decide-act (SUDA) model of an autonomous system’s architecture [82]. The aim was to support designers to decide what level of automation is appropriate for each of the four functions in any given system [95, 109]. The SESAR program later refined this into the Level of Autonomy Taxonomy (LOAT) for Air Traffic Management, assigning explicit numbered levels to each function [109, 112]. Several other similar four-level LoA frameworks exist across various domains [38, 42, 43, 101, 110].

A couple of other examples in this class identify five dimensions of autonomy, depicted on radar charts [58, 131]. For instance, one classifies unmanned robotic systems according to their embodiment of five key technologies that enable system autonomy: decision-making; perception; navigation; human-robot interaction; and co-operation with other autonomous systems [131].

At the time of writing (during the INSYTE framework’s second round of evaluation), a new four-dimensional classification framework was published, focused specifically on the characteristics of AI agents [66]. Its four dimensions are: *autonomy* (an AI agent’s independence from the oversight and control); *efficacy* (an AI agent’s interaction with, and causal impact on, the real world, considering the significance of its impact and whether its operating environment is simulated, mediated, or physical); *goal complexity* (the intricacy and balancing of an AI agent’s subgoal sequences, and its capacity to generate goal structures and interpret underspecified objectives); and *generality* (an AI agent’s ability to operate across diverse tasks and domains, and the range of human cognitive tasks it can fulfil). By combining an AI agent’s different levels (between 0-5) for each of these four dimensions, this framework culminates in a visual four-dimensional “agent profile”. This profile shares similarities with the eight-dimension "INSYTE pattern" generated by our framework. However, it disregards the variable complexity of physical operating environments, only considers underspecification at the highest level of a system's goal complexity, does not not explicitly include a system's adaptiveness within its level descriptions, and conflates freedom from intervention and freedom from oversight within a single dimension.

Limitations and comparison to INSYTE. Multidimensional classification frameworks provide a basis for the simultaneous measurement of multiple variables, allowing for flexible and precise classification of AI systems. However, the extant frameworks in this class have important shortcomings. The remainder of this section summarises these shortcomings, and explains how INSYTE overcomes them.

Focus on a single type of system. Most multi-dimensional frameworks are limited to the classification of mobile, embodied robots. By contrast, the new four-dimensional for characterising AI agents from [66] is geared specifically to advanced AI systems, whether embodied or not, but, as mentioned earlier, lacks sufficient detail on these systems' characteristics. This leaves a gap for a single multidimensional classification framework that can represent a wide range of system types, and consider broader range of advanced AI system characteristics than [66]. The INSYTE framework fills this gap. It has been constructed specifically to characterise a range of intelligent system types, both embodied and non-embodied, incorporating both advanced and more traditional AI, and covering AI characteristics comprehensively.

Noncontextuality. Some four-dimensional LoA frameworks [38, 42, 43, 95, 101, 109, 110, 112] are concerned solely with the system platform. They do not consider the complexity of a system's mission and operating environment. However, contextuality is crucial for a framework to be able to support safety and risk management, which represent key activities within the lifecycle of many AI systems. Recognising this, the INSYTE framework is contextual: its eight dimensions are grouped into four categories which furnish a 'whole system' perspective.

Insufficient disambiguation for classifying agentic AI. Most multi-dimensional frameworks were devised over a decade ago, prior to the development of foundation models and the diversification of capabilities they enable. While [66] updates the space to include a framework for characterising AI agents, its four dimensions combine characteristics that could themselves be decoupled for greater precision. INSYTE offers enhanced detail for the classification of agentic AI systems. While the "goal complexity" and "generality" dimensions of the framework from [66] align with INSYTE's depth of functionality and breadth of functionality, respectively, INSYTE provides greater differentiation of other characteristics. In particular, INSYTE separates freedom from intervention and oversight into two dimensions, allows for a finer-grained analysis of the complexity of a system's operating environment, and specifically distinguishes underspecification as a dimension in its own right, along with adaptiveness.

3 THE INSYTE CLASSIFICATION FRAMEWORK

3.1 INSYTE framework overview

Having situated our proposed INSYTE framework in the context of related classification frameworks, we now start its presentation with an overview of its three components.

3.1.1 Eight dimensions of intelligent capability. The first component of the framework is the breakdown of the essential characteristics of an AI or AI-enabled system along eight key dimensions. These are grouped into four categories, giving a 'whole system' perspective, as shown in Table 1.

Table 1. The eight dimensions of the INSYTE framework

Category	Dimension	Description
System Design	1. Underspecification	Ability of the system to accomplish its objectives without an explicit specification of how to do so.
	2. Adaptiveness	Ability of the system to accomplish its objectives in the face of uncertainty and change encountered in operation.
System Functionality	3. Breadth of functionality	Ability of the system to perform a range of different task types or generate a range of output types.
	4. Depth of functionality	Ability of the system to execute computationally complex, multi-layered tasks or outputs.
Operating Environment	5. Environmental diversity	Ability of the system to accomplish its objectives in a rich, open external operating environment.
	6. Environmental dynamism	Ability of the system to accomplish its objectives in a frequently, rapidly changing external operating environment.
Operational Independence	7. Independence from intervention	Ability of the system to accomplish its objectives without human operational intervention.
	8. Independence from oversight	Ability of the system to accomplish its objectives without constant, real-time human monitoring.

INSYTE's eight dimensions align with the Organisation of Economic Cooperation and Development (OECD) definition of a deployed AI system [90], which was approved by OECD member states in May 2024, to encourage interoperability and harmonisation between jurisdictions:

> *"An AI system is a machine-based system that, for explicit or implicit objectives, infers, from the input it receives, how to generate outputs such as predictions, content, recommendations, or decisions that can influence physical or virtual environments. Different AI systems vary in their levels of autonomy and adaptiveness after deployment."*

The OECD definition has been used by the European Union [125], the Council of Europe, the United States [88] , the United Kingdom and the United Nations, amongst others. Take, for example, Article 3(1) the EU AI Act [125]:

> *"'AI system' means a machine-based system that is designed to operate with varying levels of autonomy and that may exhibit adaptiveness after deployment, and that, for explicit or implicit objectives, infers, from the input it receives, how to generate outputs such as predictions, content, recommendations, or decisions that can influence physical or virtual environments."*

In line with this definition, dimension 1 of the INSYTE framework describes how tightly defined is the specification that enables the system, *'for explicit or implicit objectives,'* to *'infer, from the input it receives, how to generate outputs'*. Dimension 2 elucidates how systems might *'vary in their level of [. . .] adaptiveness after deployment.'* Dimensions 3 and 4 illustrate the range and complexity of the *outputs* the system can *generate*. Dimensions 5 and 6 identify two dimensions of the complexity of a system's *'physical or virtual environment'*. Finally, dimensions 7 and 8 highlight two core aspects of the varying *'levels of autonomy'* a deployed system might have. The OECD's defining features of an AI-enabled system can be manifested in varying degrees and combinations. INSYTE offers a way to model this, providing the means to exemplify and enrich the OECD's definition.

3.1.2 A process for determining a system's level on each dimension. The second component of the INSYTE framework encourages users to analyse their systems closely, considering each of the eight characteristics, or dimensions, in detail. It comprises a systematic process for determining, for each of the eight dimensions, at what level a given intelligent system should be described.

While this range is probably best seen as a continual scale, we break each dimension down into discrete levels, running from 0 to 5, to support the application of the framework. Six levels allow for the progressive sophistication or technical maturity on each dimension to be comprehensively described without overloading the framework with too much detail. To help users apply INSYTE, an online worksheet has been created and made freely available; this takes the user step-by-step through identifying a system's level for each dimension. The worksheet, which we call the 'online tool', is described in the Section 4.

3.1.3 A radar chart for depicting a system's INSYTE "pattern". The third component of the INSYTE framework is a synthesis of the selection of levels for each dimension, enabling a system's combination of characteristics to be depicted on a visually informative radar chart (Figure 1). Each of the eight dimensions is represented as a distinct axis on the radar, running from level 0 to level 5. In this way, the multivariate classification of an AI system's characteristics can be represented by the "pattern" it makes on the radar chart, conveying considerable detail about the system at a single glance. To ease the adoption and use of the INSYTE, the online tool described in the Section 4 includes functionality to generate and download a system's INSYTE pattern.

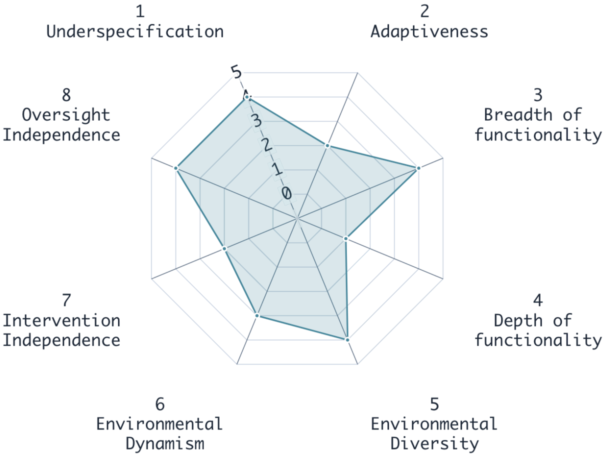


Fig. 1. Illustrative INSYTE pattern for a hypothetical AI system

3.2 The eight dimensions of INSYTE

In this section, we give a detailed description of each of the INSYTE framework's eight dimensions, grouped into four categories: system design (*underspecification* and *adaptiveness*); system functionality (*breadth* and *depth*); operating environment (*diversity* and *dynamism*); operational independence (from *intervention* and *oversight*). The descriptions of Levels 0-5 for each dimension are presented in Tables 2 to 5. The process for determining a system's level on each dimension involves working through the level descriptions given for each dimension, and choosing the most appropriate one for the system as it is intended to be deployed. Cross-referencing against examples of different system types and domains can help in this process. As such, we provide several exemplars in our online tool [60] and Appendix A of this paper.

3.2.1 System Design INSYTE Dimensions. System design is the process of defining how the system will meet the needs of its users. There are many elements of the design process that might be included in the framework, but we focus on two: the degree to which the system is (not) hard-coded with instructions and rules; and the degree to which it is designed adapt to uncertainty and change.

Dimension 1: Underspecification. This dimension measures the system's ability to accomplish its objectives without an explicit set of rules or instructions of how to do so. Specification is the blueprint of a system's required properties, which is then mapped into the system's design. In the INSYTE framework, we are specifically concerned with the specification of step-by-step

Table 2. System design INSYTE dimensions

Level	Description
	Dimension 1: Underspecification
0	The system requirements have been fully encoded as a set of instructions or rules.
1	The majority of the system's requirements (typically over two thirds) have been encoded as a set of rules, with the rest learnt from data, or through fine-grained rewards-guided trial and error.
2	A non-negligible part of the system requirements (typically under one third) have been encoded as a set of rules, with the rest learnt from data, or through fine-grained rewards-guided trial and error.
3	The majority of system requirements have been learnt from labelled data, or through trial and error within a fully specified environment and/or with frequent human feedback.
4	System requirements have been learned from (mostly unlabelled) data, or through trial and error within a partially specified environment and/or with infrequent human feedback.
5	System requirements have mostly been specified as high-level objectives, possibly supported by some learning from data and/or through trial and error within a partially specified environment and without human feedback.
	Dimension 2: Adaptiveness
0	The system works only in a controlled environment without uncertainty or change.
1	The system reacts to a specific type of uncertainty or change using a few predefined adaptation tactics.
2	The system reacts to multiple, known types of uncertainty or change with several predefined adaptation tactics.
3	The system applies both predefined and dynamic adaptation tactics, sometimes proactively, to handle a few, foreseen types of uncertainty or change.
4	The system proactively uses a suite of adaptation tactics, many of which are dynamic, to handle multiple, foreseen types of uncertainty or change.
5	The system mainly adapts proactively to a wide range of foreseen types of uncertainty and change, and reacts to multiple unforeseen types of uncertainty or change while learning and improving its resilience over time.

instructions to derive outputs from inputs (i.e., a system's input-output transfer function) [96, 116]. Systems programmed with explicit, encoded rules are more tightly specified than those trained on labelled data (supervised learning algorithms); these, in turn, are more tightly specified than those trained on both labelled and unlabelled data (semi-supervised learning algorithms) or those trained through an iterative process of trial and error (reinforcement learning algorithms); these, in turn, are more tightly specified than those trained on unlabelled data (unsupervised learning algorithms). Underspecification is a way of transferring to the model itself the task of inferring its own intended function. While one essential dimension of system autonomy is manifested downstream (its degree of freedom from the continuous input of a human operator, as represented in dimension 7), this aspect of autonomy is manifested upstream (a system's degree of freedom from the continuous input of a human engineer) [81].

Designers may choose to underspecify for several reasons: to reduce cost and time; due to multiple viable routes to achieving the same objective [25, 138]; or because what the system is intended to achieve is difficult to formalise unambiguously. This difficulty in formalisation can stem from the complexity of its environment and mission, or because of the reliance on tacit knowledge when the task is performed by humans [19].

Levels 0-5 of the underspecification dimension are presented in the top half of Table 2. In the online tool, an agricultural robot, a medical decision-support system, and an insurance chatbot are given to illustrate the different levels of this dimension. At Level 0, an embodied system would, for example, achieve its objective at a fixed rate on a predefined schedule, while a traditional AI system would achieve its objectives on the basis of IF-THEN rules crafted by experts. At Level 5, an embodied system based on reinforcement learning might have been trained in a high-fidelity simulation environment where people and objects have been specified, while an agentic AI system may have had no norms or constraints specified, and infer a general understanding of how to achieve its objectives from the data used to train its underpinning foundational model.

Dimension 2: Adaptiveness. This dimension measures the system's ability to accomplish its objectives, and, at the upper levels of adaptiveness, to optimise how it does so, in the face of uncertainty and change encountered in operation [132, 133]. This ability is central to the concept of resilience and, at the upper end, to that of *antifragility*, i.e., to the capacity of a system to become more resilient over time [18, 49]. Adaptiveness depends on the range of uncertainty and change *locations* and *sources*, and on the *level(s)* of uncertainty that the system can handle. Uncertainty taxonomies [36, 102] identify numerous combinations of such locations and sources—which, for simplicity, we will henceforth call *types of uncertainty/change*. These include the system itself (sensor noise and faults, world-model inaccuracy, effector imprecision, etc.), and its operational context (environment parameter variability, user behaviour, etc.). The more of these types an AI-enabled system can handle, the more adaptive it is. Furthermore, uncertainty taxonomies define levels of uncertainty, e.g., [98] distinguishes between deterministic knowledge, known/foreseen uncertainty, unknown/unforeseen uncertainty, or unknowns unknowns, and total ignorance, or lack of ability to identify the presence of, and to reason about, a type of uncertainty. Increased adaptiveness is normally associated with the ability to manage higher levels of uncertainty.

Adaptiveness also depends on how the types of uncertainty manage by a system impact its ability to deliver the required functionality, and on the *adaptation tactics* (i.e. uncertainty management techniques) employed. At lower adaptiveness levels, a system may incur temporary loss of functionality, and use reactive, predefined adaptation tactics subsequently to recover some or all of its functionality. At mid levels of adaptiveness, a system may use a combination of predefined/reactive and dynamically synthesised adaptation tactics, sometimes proactively, and its functionality will likely be fully restored after a period of graceful degradation. Finally, at high adaptiveness levels, a system will proactively adapt its configuration, architecture, internal models of the world and/or behaviour to prevent degradation or loss of functionality, and to optimise its operation, and may employ online learning to improve its ability to do so over time.

Levels 0–5 of the adaptiveness dimension consider all the factors mentioned above, and are presented in the bottom half of Table 2. In the online tool, an inspection drone, a traditional AI system used for quality control inspection, and an agentic AI system used in supply chain management are given as examples to illustrate the different levels of this dimension. At Level 0, an embodied system may, for example, only be able to accomplish its objectives on known subjects in tightly controlled conditions, while a traditional AI system may only be able to identify known issues in a standardised environment. At Level 5, an embodied system may be able to achieve its objectives on subjects within a wide range of variation, and, for example, maintain optimal position and energy efficiency in highly unpredictable conditions, while an agentic AI system may be able to use dynamic adjustments to maintain optimal performance in the face of a wide array of foreseen and unanticipated disruptions, and learn from unanticipated disruptions to update its predictive models over time.

3.2.2 System Functionality INSYTE Dimensions. System functionality is what the system does to meet the needs of its users. Because our aim is to construct a framework that allows for the classification of systems in any application domain, we do not describe system functionality in terms of specific tasks a system might accomplish, but couch it at a higher level of abstraction: how broad a range of tasks it can accomplish; and how difficult and involved those tasks are.

Dimension 3: Breadth. This dimension measures the system's ability to perform diverse task types or generate varied output types, reflecting its versatility and multi-tasking capacity — a growing feature of AI tools powered by frontier models. We characterise breadth of functionality by the diversity (and variations) of task types that the system can perform, output types it can generate, or decision criteria it can fulfil.

Table 3. System functionality INSYTE dimensions

Level	Description
	Dimension 3: Breadth
0	The system performs one task type, or generates one output type, or fulfils one decision criterion.
1	The system performs variants of one task type or output type, or fulfils variants of one decision criterion.
2	The system performs a few different task or output types (typically fewer than five), or fulfils a few decision criteria.
3	The system performs variants of a few different task or output types, or fulfils variants of a few decision criteria.
4	The system performs many different task or output types, or fulfils multiple decision criteria.
5	The system performs variants of many different task or output types, or performs many task types, or fulfils variants of multiple decision criteria.
	Dimension 4: Depth
0	The system's task or output types involve one routine sub-task or step.
1	The system's task or output types involve a few routine sub-tasks or steps (typically fewer than five).
2	The system's task or output types involve a few complex sub-tasks or steps, or many routine sub-tasks or steps.
3	The system's task or output types involve many sub-tasks or steps, some of which are routine and others complex.
4	The system's task or output types involve many, mostly complex sub-tasks or steps.
5	The system's task or output types involve many sub-tasks or steps that are very complex.

'Task' refers to the specific problem the system addresses (e.g., identifying anomalies in a radiography image, executing highway lane changes, transcribing a conversation). By 'output' we mean the result after the system has processed the input data (e.g., a classification, a physical manoeuvre, a transcript). 'Task types' or 'output types' are sets of variants of tasks or outputs which are not functionally distinct (e.g., for an assistive robot, dressing and feeding are different task types, while its variety of arm-raising manouevres are variants of the same output type). By 'decision criteria' we mean the multiple optimisation objectives and/or constraints that the system needs to take into account (e.g., for an automated driving system, minimising journey time and environmental impact, and optimising safety, while planning and executing a navigation route). Greater functional breadth may increase the likelihood of emergent behaviour, as the system can learn general principles across a range of tasks [30].

Levels 0-5 of the breadth of functionality dimension are presented in the top half of Table 3. In the online tool, a surgical robot, a traditional AI-based fraud detection system, and an agentic AI research assistant are given as examples to illustrate the different levels of this dimension. At Level 0, an embodied system would only perform a single task, such as performing a line of sutures in a specific type of surgery, while a non-embodied traditional AI system may simply flag a single type of threshold-exceeding input to a human operator. At Level 5, an embodied system may be able to perform the entire suite of tasks associated with a complex task pathway, such as surgery, while an agentic AI system may be able to do likewise, fulfilling a wide range of decision criteria.

Dimension 4: Depth. This dimension measures the system's capacity to execute computationally complex, multi-layered task types or generate sophisticated output types. We define depth of functionality by the number of hierarchical subtasks (and their complexity) needed to complete a system's task, or the number of processing steps/layers required to produce its outputs.

A complex subtask involves multiple, interacting steps that require re-planning or depend on contextual factors. An output step is complex if it involves multiple parts or numerous inputs requiring distinct analyses. One way a system exhibits functional depth is through multi-step planning over extended periods, a recognised characteristic of agentic AI systems [21, 22, 30, 66]. This depth often intertwines with functional breadth: balancing multiple decision criteria necessitates deeper processing, while multi-tasking requires coordination between different subsystems.

Levels 0-5 of the depth of functionality dimension are given in the bottom half of Table 3. In the online tool, a highly automated driving system, a traditional AI system for assessing glucose test results, and an agentic smart grid management system are given as examples to illustrate the different levels of this dimension. At Level 0, an embodied system will just perform a task with a single routine sub-task, such as a vehicle maintaining a fixed speed, while a traditional AI might just evaluate inputs against a predefined normal range. At Level 5, an embodied system may be able perform a highly complex, nested set of nuanced tasks, while an agentic AI system might, for example, develop dynamic, multi-stage plans and provide chain-of-thought (CoT) reasoning for its decisions.

3.2.3 Operating Environment INSYTE Dimensions. The operating environment is the external environment in which the system is intended to be deployed. This may be a physical, real-world environment or a digital environment. INSYTE distinguishes two environmental dimensions: the diversity of elements and interactions within the environment; and the propensity of the environment to frequent, rapid, and significant change. These dimensions refer to the environment the system can actually perceive or detect, not simply the broader world it exists in.

Dimension 5: Environmental Diversity. This dimension measures the richness and openness of the operating environment. We characterise environmental diversity by the number and variety of element types within that environment and the possible interactions between them.

When classifying a system using the INSYTE framework, framework users should consider the following environmental elements: objects; other agents (both human and artificial); physical conditions; norms and rules. The classification of subcategories of these element types depends on the context. For example, in healthcare, human clinicians and patients would likely be distinct element types, and different clinician roles might also count as separate types if they engage in a wider task differently. Because this dimension covers interactions between environmental elements, including the system itself and human users, human-machine interaction is partly captured by this dimension. However, the level of human operational input is covered under INSYTE's dimension 7.

Levels 0-5 of the environmental diversity dimension are presented in the top half of Table 4. In the online tool, a manufacturing environment, an air traffic control environment, and a financial market environment are given as examples to illustrate the different levels of this dimension. At Level 0, an embodied system might be operating in a sterile and highly controlled environment, while a traditional AI might only receive a single stream of data. At Level 5, and embodied system might negotiate multiple types of obstacle, and interact with many humans, while an agentic AI system might operate in an environment with widely diverse data sources and variables, and communicate with different types of customer and other systems, including other AI agents.

Dimension 6: Environmental Dynamism. This dimension measures the frequency, rapidity, and magnitude of change in the operating environment. The change is of interest if it has the potential to impact the system's ability to deliver its required functionality. As above, the operating environment comprises objects, other agents (both human and artificial), physical conditions, norms and rules.

Levels 0-5 of the environmental dynamism dimension are presented in the bottom half of Table 4. In the online tool, the ground and air environment of an unmanned aerial vehicle, a legal research environment, and a software development environment are given as examples to illustrate the different levels of this dimension. At Level 0, an embodied system might, for example, operate in an environment with stable protocols and climate control, while a traditional AI might operate

Table 4. Operating environment INSYTE dimensions

Level	Description
	Dimension 5: Environmental Diversity
0	The system is the only element in the environment.
1	There are few types of element in the environment (typically fewer than five), and negligible interactions between them.
2	There are few types of element in the environment (typically fewer than five), and few interactions between them.
3	There are many types of element in the environment, and few interactions between them; or there are few types of element in the environment, and many interactions between them.
4	There are many types of element in the environment, and many interactions between them.
5	There are unbounded types of element, and unbounded possible interactions between them.
	Dimension 6: Environmental Dynamism
0	There is no change or negligible change in the environment while the system is operating.
1	Frequency, speed and magnitude of change in the environment are all low.
2	One or two of frequency, speed and magnitude of change in the environment are medium, and the other(s) are low.
3	Frequency, speed and magnitude of change in the environment are all medium.
4	One of frequency, speed and magnitude of change in the environment is high, and the others are low or medium.
5	Two or three of frequency, speed and magnitude of change in the environment are high.

in a static environment consisting of a fixed database. At Level 5, and embodied system might accomplish its objective in an environment with frequent temperature changes, rapidly moving physical objects, and significant changes to norms and protocols, while an agentic AI system might do so in an environment with continuous, immediate changes that affect functionality.

3.2.4 Operational Independence INSYTE Dimensions. Operational independence is the system's capacity and delegated authority to achieve its objectives without human operational control. We break this down into two key dimensions: the system's ability operate without frequent human intervention, including during extended operation and unusual circumstances; and the intended level of freedom given to the system to accomplish its objectives without constant or regular human monitoring or scrutiny. Though monitoring is often a prerequisite for effective intervention, these two dimensions of human control can be disambiguated. For instance, a system may require little intervention but still need frequent monitoring due to, for example, legal requirements.

Dimension 7: Independence from intervention. This dimension measures the degree to which a system can, and is intended to, accomplish its objectives without the intervention of a human operator. We characterise this by the frequency of human intervention, whether it is required for adequate or optimal functioning, and whether it is required in normal or exceptional circumstances. In the descriptions for Levels 2 and 3, we also cover how a decision-support system might exert operational independence: its outputs may simply provide information to a human decision-maker (Level 2); or its outputs may be intended to provide direction to a human decision-maker (Level 3).

Levels 0-5 of the independence from intervention dimension are presented in the top half of Table 5. In the online tool, an autonomous maritime vessel, a traditional medical AI system which outputs a risk prediction for post-operative complications, and an agentic AI meeting scheduler are given as examples to illustrate the different levels of this dimension. At Level 0, a human operator would be in full operational control of an embodied system, while they would, for example, categorise and interpret all data inputted into and outputted by a traditional AI system. At Level 5, no human operational intervention would be required across an embodied system's entire mission, while an

Table 5. Operational independence INSYTE dimensions

Level	Description
	Dimension 7: Independence from intervention
0	The system requires continuous human intervention to function at all.
1	The system requires frequent human intervention to function adequately, or only informs human decision making.
2	The system requires occasional human intervention to function optimally, or directs human decision making.
3	The system requires minimal human intervention only for extended operation.
4	The system only requires occasional human intervention, even for extended operation.
5	The system functions without any human intervention across all intended operational contexts.
	Dimension 8: Independence from oversight
0	The system requires continuous human monitoring while operating.
1	The system requires regular human 'sanity checking' while operating.
2	The system requires 'sense checking' while operating in occasional, exceptional circumstances (typically prompted by the system itself).
3	The system requires human assessment at the completion of each mission.
4	The system requires regular, retrospective auditing.
5	System is only audited retrospectively to investigate accidents and incidents.

agentic AI system would be handle all situations arising in the course of accomplishing its objective without human intervention.

Dimension 8: Independence from oversight. This dimension measures the extent to which the system is allowed to accomplish its objectives without constant monitoring or scrutiny from internal actors, namely operators or users (as opposed to the scrutiny of external actors, such as regulatory officials [5]). We characterise this by how often human monitoring occurs, whether it occurs during operation or retrospectively, and under which circumstances.

Levels 0-5 of the independence from oversight dimensions are shown in the bottom half of Table 5. In the online tool, a forest inspection drone, an automated content moderation system, and a clinical conversational AI agent are given as examples to illustrate the levels of this dimension. At Level 0, human operators constantly remotely-monitor an embodied system during the course of its mission, while, for example, a traditional AI system might present every output in real time to a human operator. At Level 5, human operators might, for example, only review an embodied system's mission logs after damage or an accident has occurred, while they might only review the transcript of an agentic AI system's mission after a harm has occurred and been reported.

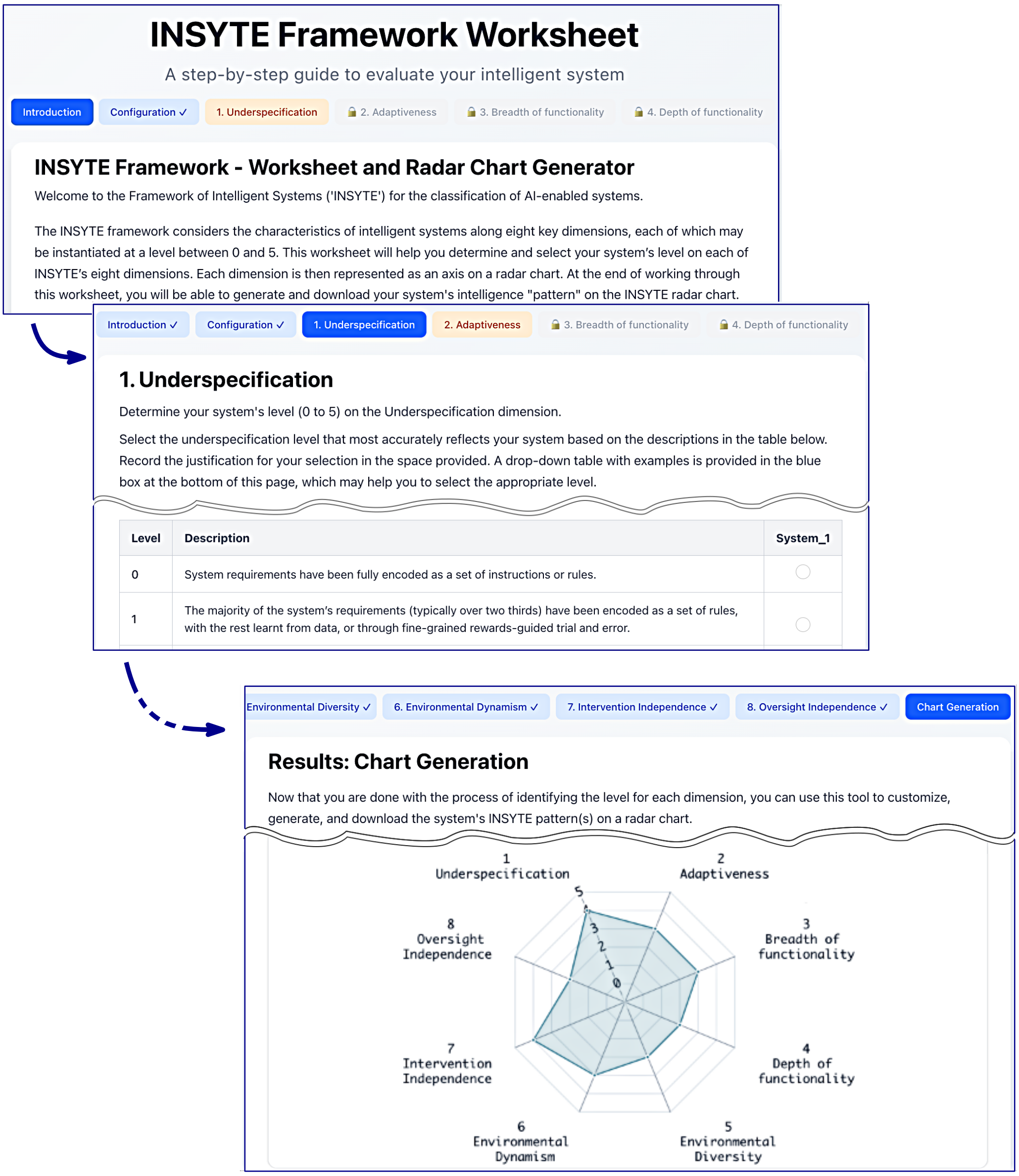
This concludes the level descriptions, which are core to the second component of the INSYTE framework: the tool-supported process for selecting the appropriate level for each of the eight dimensions of a given system's overall intelligence and autonomy.

4 TOOL SUPPORT

To support the adoption of INSYTE, we developed and have made freely available online a worksheet that guides users through our process for determining an AI-enabled system's level on each dimension of the framework, and then allows users to generate its 'pattern' on a radar chart [60]. We call this the 'online tool'. We implemented this online tool using the Next.js React framework[1] to achieve improved performance and a faster initial loading experience, and Tailwind CSS styling[2] for a user-friendly, customised design. To ensure confidentiality, all data from the users' assessment of their intelligent systems are stored only in their browser's local storage.

[1] https://nextjs.org/

[2] https://tailwindcss.com/


INSYTE Framework Worksheet
A step-by-step guide to evaluate your intelligent system
Introduction
Configuration ✓
1. Underspecification
2. Adaptiveness
3. Breadth of functionality
4. Depth of functionality
Introduction ✓
Configuration ✓
1. Underspecification
2. Adaptiveness
3. Breadth of functionality
4. Depth of functionality
Environmental Diversity ✓
6. Environmental Dynamism ✓
7. Intervention Independence ✓
8. Oversight Independence ✓
Chart Generation
1
Underspecification
2
Adaptiveness
3
Breadth of functionality
4
Depth of functionality
5
Environmental Diversity
6
Environmental Dynamism
7
Intervention Independence
8
Oversight Independence
5
4
3
2
1
0

INSYTE Framework - Worksheet and Radar Chart Generator

Welcome to the Framework of Intelligent Systems ('INSYTE') for the classification of AI-enabled systems.

The INSYTE framework considers the characteristics of intelligent systems along eight key dimensions, each of which may be instantiated at a level between 0 and 5. This worksheet will help you determine and select your system's level on each of INSYTE's eight dimensions. Each dimension is then represented as an axis on a radar chart. At the end of working through this worksheet, you will be able to generate and download your system's intelligence "pattern" on the INSYTE radar chart.

1. Underspecification

Determine your system's level (0 to 5) on the Underspecification dimension.

Select the underspecification level that most accurately reflects your system based on the descriptions in the table below. Record the justification for your selection in the space provided. A drop-down table with examples is provided in the blue box at the bottom of this page, which may help you to select the appropriate level.

Level	Description	System_1
0	System requirements have been fully encoded as a set of instructions or rules.	○
1	The majority of the system's requirements (typically over two thirds) have been encoded as a set of rules, with the rest learnt from data, or through fine-grained rewards-guided trial and error.	○

Results: Chart Generation

Now that you are done with the process of identifying the level for each dimension, you can use this tool to customize, generate, and download the system's INSYTE pattern(s) on a radar chart.

Fig. 2. INSYTE tool: introduction, underspecification assessment, and radar chart generation

As shown at the top of Figure 2, this online tool first provides an introduction to the INSYTE framework. At this early stage, the eight dimensions of the classification framework and their roles are briefly overviewed, and the users are introduced to the classification process mentioned in Section 3.1.2. Next (not shown in Figure 2), the users are offered the option to configure the number of system variants they will classify and compare (the default is one system variant). The users are then taken through the step-by-step process for selecting the most appropriate level between 0-5 of each dimension for the system variant(s) under consideration, thereby classifying their system variant(s) on each dimension of the framework.

For each dimension, the worksheet provides:

(1) A summary of what the users need to do in order to determine the level(s) of their system variant(s) on that dimension.
(2) The set of dimension-specific factors to consider when making their selection(s).
(3) Descriptions of levels 0–5 for the dimension, with level-selection options for each system variant.
(4) A reminder to record the justification for the level(s) selected for the system variant(s).
(5) Dedicated space for recording the rationale for these selection(s).
(6) Examples of capabilities that three concrete AI and autonomous systems of different types (an embodied AI-enabled system, a traditional non-embodied AI system, and an agentic AI system) would need at different levels on the dimension being assessed.

The partial worksheet page in the middle of Figure 2 depicts items (1)–(3) for the Underspecification dimension of INSYTE, with a single system variant named 'System_1' being classified. As users complete the assessment of a dimension of their system variant(s), they are guided to progress to the next dimension. On completing the assessment of all INSYTE dimensions, the online tool reaches the 'Chart Generation' stage, a part of which is shown at the bottom of Figure 2. At this stage, the users can:

(1) Customise, generate and download the INSYTE radar chart(s) for the assessed system variant(s) – with the possibility to produce separate or overlayed charts for multiple system variants.
(2) Read and download a report that combines the justifications provided for their level selections on each dimension of the framework.

These tool outputs are provided in formats that makes their integration into the users' technical reports or papers straightforward.

5 FRAMEWORK EVALUATION

5.1 Evaluation methodology

Our primary evaluation objectives were to assess the usability and usefulness of the INSYTE framework. We also sought to explore its relevance to a wide range of application domains and system types.

Internal testing of the tool-supported framework allowed us to refine it before starting the evaluation activity. In this phase, two co-authors, who had not been involved in defining the eight dimensions or their level descriptions, applied the online worksheet to well-known systems like ChatGPT [16] and Roomba [64], leading to refinements in both the framework (clearer level descriptions) and the online tool (better placement of the examples).

A total of sixteen participants then completed the evaluation activity across two rounds of systematic evaluation. Our evaluators were researchers and industry practitioners actively involved in the design, development, safety engineering, deployment, and safety assurance of AI and AI-enabled systems. We selected our evaluators from a diverse set of disciplines and sectors and ensured that we engaged with users of varying levels of experience. The application domain, participants' professions and years of experience, and system types evaluated, are presented in Figure 3. The category of 'researcher' in the second pie chart of the figure includes academic researchers in robotics, data analysis, human-computer interaction, navigation, safety engineering, verification, and clinical informatics. Evaluators were encouraged to consider variants of the same system. For instance (referring to the bottom half of Figure 3), INSYTE was applied by evaluators to both a

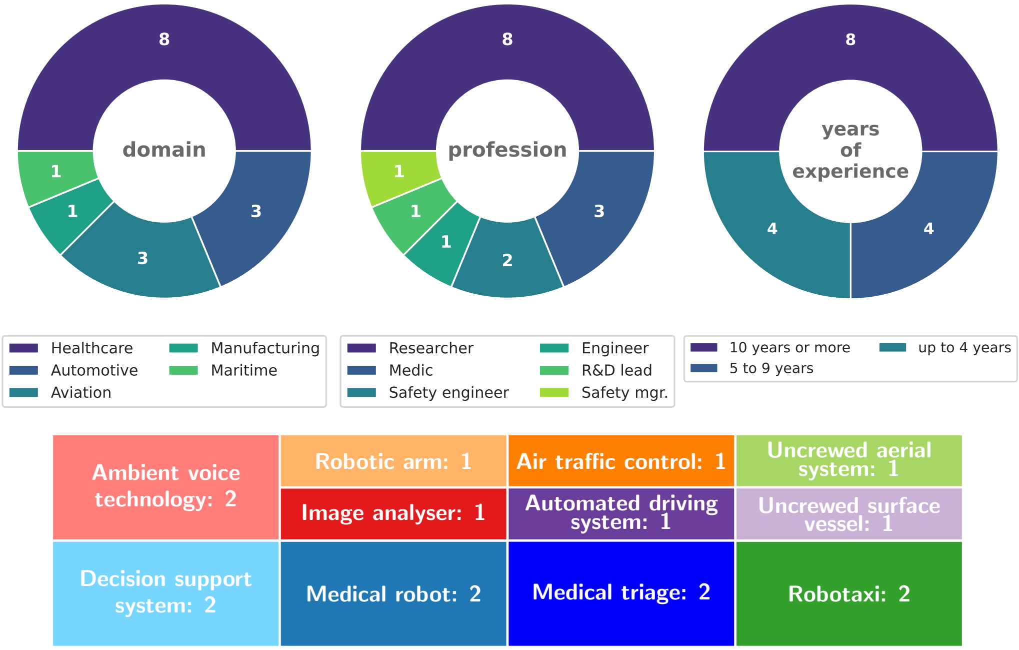


Fig. 3. The application domain, participants' profession and years of experience (top), and evaluated systems (bottom)

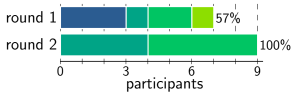
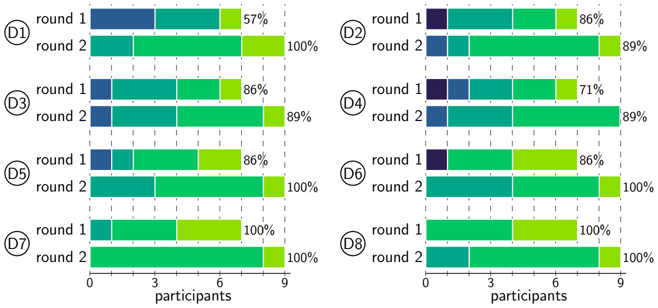
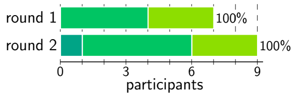
decision-support and an autonomous variant of an air traffic control system, and to both fixed and mobile variants of a medical robot (collaborative assistive care). Agentic AI systems were also represented in the evaluation activity - by the ambient voice technology systems, enabled by large language models (LLMs), and by an advanced, agentic variant of the automated driving system.

The first round of evaluation was completed by seven evaluators, whose feedback prompted refinements to the framework and tool: we fine-tuned the level descriptions for clarity, updated the tool's text with further guidance on the dimension-specific factors users should consider when making their selections, and incorporated a broader range of illustrative examples (encompassing embodied, traditional, and agentic AI). Nine external evaluators then completed the second round of evaluation.

The evaluation activity had three parts. First, evaluators used the online worksheet to apply the INSYTE framework to an AI or AI-enabled system. Second, evaluators compiled their system's INSYTE patterns using the online radar chart generation tool. They also downloaded the 'justifications report', which summarised their rationales for level selections, and emailed these documents (the INSYTE patterns and justifications report) to the lead author. Third, evaluators completed a 10-question questionnaire. The first three questions were context-setting questions asking the participants to specify the application domain for their evaluated system or system variants (Question 1), as well as their profession (Question 2) and years of experience (Question 3). Questions 4 to 6 solicited feedback on the *usability* of the INSYTE framework. Questions 7 to 9 solicited feedback on the *usefulness* of the INSYTE framework. Question 10 asked an open question about any perceived limitations of the framework that had not been covered in previous answers.

The evaluation activity was approved by the University of York's Physical Sciences Ethics Committee; participants gave informed consent to the use of anonymised answers in this paper.

Table 6. Questionnaire: Usability of the INSYTE framework

Question	**Response**† (showing %age of ‘easy’/‘quite easy’/‘some effort’ answers)
Q4. How easy was it to apply the INSYTE framework to the system using the PROCESS provided, overall and for each axis? Please explain your answer. *(Free text)*	
Q5. How easy was it to instantiate the eight dimensions of the INSYTE framework? What improvements would you suggest to ease the instantiation of the eight dimensions? *(Free text)*	
Q6. How easy was it to generate the system’s INSYTE pattern using the TOOL provided? Please explain your answer. *(Free text)*	

†Key: (D1) ··· (D8) INSYTE dimensions 1–8 ■ difficult ■ quite difficult ■ some effort ■ quite easy ■ easy

5.2 Evaluation results

Participant responses to the context-setting questions (Questions 1–3), summarised at the top of Figure 3, show that INSYTE’s evaluation encompassed AI systems across diverse domains – predominantly healthcare – and involved a variety of professionals, half with a decade or more of experience. The actual AI systems being evaluated were even more diverse (see the bottom half of Figure 3), with those from the healthcare domain, for instance, ranging from medical triage of patients and ambient voice technology to (clinical) decision support systems and medical robots.

The results of the INSYTE evaluation are presented in two parts. First, we share our evaluators’ feedback on the *usability* of the tool-supported process for working through the INSYTE framework (Questions 4–6 of the evaluation questionnaire). Responses to the closed questions on usability are given in Table 6. These are explained further in Section 5.2.1, along with the free-text answers given by evaluators. Notably, in their free-text answers, evaluators gave particularly positive feedback on INSYTE’s inclusion of a radar chart for the visual representation of a system’s classification. Second, we share the evaluators’ feedback on the *usefulness* of the INSYTE framework (Questions 7-9 of the evaluation questionnaire) in Table 7 and Section 5.2.2.

5.2.1 Usability of INSYTE. Closed questions on the usability of the INSYTE framework asked evaluators to rank the ease of using the process provided (Q4), of instantiating each of INSYTE’s eight dimensions (Q5), and of generating the radar chart using the online tool (Q6). Participants selected from ‘difficult,’ ‘quite difficult,’ ‘some effort,’ ‘quite easy,’ and‘easy.’ In Table 6, we have broken down our evaluators’ responses to these closed questions into first and second round results, showing how the improvements made to the framework and online worksheet after the first round of evaluation let to more positive evaluations of user-friendliness in the second round. Percentages in this table represent the combined ’easy,’ ’quite easy,’ and ’some effort’ answers: we deemed any of these to indicate acceptable usability, as thoughtfully working through the INSYTE framework and considering each of the eight dimensions inherently requires some effort, which the process cannot fully eliminate.

In the free-text answers to Q4, second round evaluators who found the process ‘quite easy’ to work through said: *"overall this is very clear and relatively straightforward"*; *"... the process provided was manageable and logically structured"*; *"the process was clear and logically structured"*; *"the axis descriptions and level breakdowns made it straightforward to assess each dimension with minimal ambiguity. Mapping my system’s capabilities to the framework felt intuitive."* There was a query about where to draw the boundary around the system: *"I wasn’t clear on the system boundaries for the ’AI system’, e.g. is the user of a clinical decision support tool part of the system?"*. We hope that the clarification in the terminology section of this paper, as well as the observation that applying INSYTE necessarily involves some judgement, helps to address this query.

Second round evaluators who found the process took ‘some effort’ said: *"a bit tricky to choose between levels but overall ... I never ended up picking randomly between two contiguous levels"*. Some evaluators found it took effort to select between Levels 2 and 3 on some dimensions. This was particularly identified for the functionality dimensions: *"Breadth of Functionality and Depth of Functionality, required more careful consideration, particularly in distinguishing between closely related levels."* One evaluator said that effort would be reduced if the framework was applied by a cross-functional team: *"A thorough understanding of the system is essential for effectively using this framework ... a cross-functional team of experts is necessary to maximize its potential"*.

In response to the closed question on how easy it was to instantiate each of the eight dimensions (Q5), we can see in Table 6 that, by the second round of evaluation, the following dimensions achieved fully acceptable usability (i.e., 100 percent of round two evaluators ranked this at ‘some effort’ or easier): 1 (underspecification), 5 (environmental diversity), 6 (environmental dynamism), 7 (operational independence from intervention) and 8 (operational independence from oversight). Evaluators generally found the operational independence and environmental diversity dimensions the easiest to apply, in particular independence from intervention. Dimensions 2 (adaptiveness), 3 (breadth of functionality) and 4 (depth of functionality) scored highly on usability (i.e., 89 per cent of round two evaluators ranked their ease of instantiation at ‘some effort’ or easier). Most rated these three dimensions ‘quite easy’ to apply.

Free-text answers to Q5 provide further insight. One evaluator in round two found their system’s adaptiveness ‘quite difficult’ to assess because it is only evident after the system has encountered uncertainties in operation. To clarify, the framework should be applied to the system as it is intended to operate. INSTYE can then serve as a useful benchmark to evaluate its adaptiveness (and the other seven dimensions) in operation, comparing it against pre-deployment intentions - something discussed in Section 6.2.4. Another evaluator found the functional breadth and functional depth dimensions ‘quite difficult’ to apply because the meanings of the terms within their level descriptions, specifically "tasks," "outputs," and "decision criteria," were not explained. Explanations of these terms have now been added to the fuller description of these dimensions in Section 3.2.2, We encourage users to consult the paper in addition to using the tool support when applying INSYTE to their systems. Evaluators also suggested including more worked examples to improve the ease of instantiating each dimension; we have added three worked examples in Appendix A.

In Q6, we sought feedback on the usability of the tool; that is, the ‘Chart Generation’ stage of the online tool. This was unanimously considered acceptably user-friendly. One respondent said, *"It was really easy to generate the INSYTE pattern using the tool provided. I liked how there was an option to display the charts overlaid or side-by-side, and I thought it was nice how you could choose different colour options."* Another said, *"the process of using the tool was clear and intuitive"*, another that the tool was *"user-friendly and intuitive"*, and another that it was *"a very easy tool to use."* One respondent answered that using the tool involved ‘some effort’ but their free-text explanation pointed to a difficulty with choosing between levels, rather than specifically generating an INSYTE pattern after working through the process.

Feedback in the free-text answers was particularly positive on the visual aspect of the INSYTE framework – the third component of the framework, namely the INSYTE radar pattern – with evaluators saying: *"It puts a given system ‘on a page’ so provides a useful visual from which to discuss a given system"*; *"The visualization is very good, and it helps actually for me to understand better the system I designed"*; *"The pictorial representation could provide a quick appraisal of an AI tool, and could help focus attention on strength and weakness of AI tool"*; *"The layout of dimensions and the immediate visual feedback on the radar chart made it easy to validate and reflect on each selection."*

Key insights from the answers to the usability questions (Q4 to Q6):

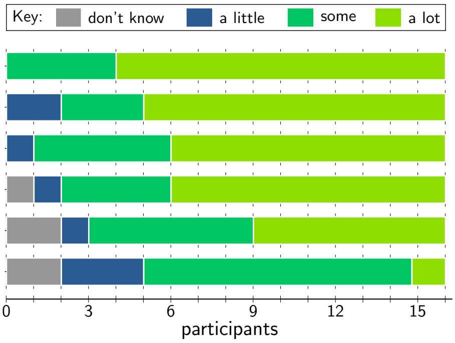
- Across the board, the INSYTE framework was found acceptably usable;
- The improvement in the feedback from round one to round two evaluation suggests that clear level descriptions and wide-ranging examples help users work through the process of selecting the most appropriate level for each dimension of the INSYTE framework; even so, users need to exercise judgement in their selection of the levels;
- Thoroughly worked examples may further enable users to select the most appropriate level for each of INSYTE dimensions, though we acknowledge people could rely too heavily on such examples; we offer three worked examples in Appendix A;
- Some of INSYTE’s more technical dimensions require effort for non-technical stakeholders to apply to a system. A cross-functional team of experts is best placed to apply the INSYTE framework to establish the correct classification of an AI or AI-enabled system;
- A system’s instantiation of some the dimensions may change between intended use and unintended use, as well as between pre-deployment and operation;
- Generating a system’s INSYTE pattern using the radar chart generation tool was straightforward and intuitive, and the visual representation of a system was very well-received.

5.2.2 Usefulness of INSYTE. Questions 7 and 8 assessed the usefulness of INSYTE. Closed questions asked evaluators to comment on its potential usefulness to their own organisation or profession (Q7), and to rank its potential value for six specific uses on the following scale (Q8): ‘don’t know,’ ‘a little,’ ‘some,’ and ‘a lot.’ Table 7 aggregates the responses from the two rounds of evaluation, since we did not take steps to increase the benefits or potential uses of INSYTE between the evaluation rounds.

As shown in Table 7, fifteen out of the sixteen people who took part in the evaluation activity could see the potential usefulness of INSYTE in their line of work (Q7). For context, the evaluator who answered ‘no’ stated that INSYTE would be valuable for standards bodies and regulators rather than for an AI software development company (where they themselves worked). Evaluators specifically praised INSYTE’s structure, clarity, multi-dimensionality, the value of its whole system perspective for assessing complexity "in the round," and its foundation for standardised representations of AI and AI-enabled autonomous systems.

For Q8, evaluators assessed six specific uses for the INSYTE framework. As can be seen in Table 7, for all uses, the predominant response indicated ‘some’ to ‘a lot’ of potential value. There was consensus that the principal value of INSYTE lies in its potential to support cross-stakeholder communication. In the free-text elaboration of this question, our evaluators said: *"The framework’s structured categories and clear dimensions make it well-suited for aligning understanding between engineers, designers, safety experts, and non-technical stakeholders"*; that it *"Provides a structured language for collaboration between designers, engineers, users, and regulators"*; and that it *"is particularly helpful for communicating between technical and non-technical teams."* A safety engineer

Table 7. Questionnaire: Usefulness of the INSYTE framework

Question number	**Response**
Q7 Do you see the INSYTE framework being potentially useful to your organisation/profession/projects? If yes, who and why? If no, why not? *(Free text)*	94% yes (15 participants) 6% no (1 participant)
Q8 Do you think the INSYTE framework has potential value for the following uses, currently or in the long term? 1. As a tool for cross-stakeholder communication 2. To inform design and development decisions 3. To inform deployment decisions 4. To inform safety engineering or safety assurance 5. As a classification system for regulatory and/or certification purposes 6. To inform decisions about liability If possible, please elaborate below on why you think it has value for the uses you selected. *(Free text)*	
Q9 Are there any other uses you think INSYTE could have? *(Free text)*	see Section 5.2.2
Q10 Would you like to raise any limitations with the framework that you have not mentioned already? *(Free text)*	see Section 5.2.2

in the automotive sector highlighted INSYTE as a tool that could be used to inform members of the public: *"It helps the general public visualise the level of autonomy of a system and understand potential limitations"*.

Most evaluators recognised INSYTE's potential to inform design, development, and deployment decisions. A robotics expert and academic noted it *"helps actually for me to understand better the system I designed."* A functional safety engineer highlighted its utility in identifying system strengths and weaknesses: *"You can see the shortcomings of the system and where it excels, so you can decide where to focus development."* Evaluators also saw value in aligning development with requirements, particularly for *"comparing variants (e.g., rule-based vs. adaptive) and aligning development with assurance, verification, or safety goals in assistive robotics and autonomous systems."* For deployment, INSYTE was considered a promising platform to *"assess operational readiness and adaptability to different environments"* and *"evaluate different systems for different use cases."*

Many also saw INSYTE's potential in safety engineering and assurance, primarily because of its emphasis on system characteristics that influence risk but may be overlooked. This is particularly evident in scenarios like *"an 'administrative' AI tool in a healthcare setting[, which] may seem low risk at first glance but could still pose significant dangers if it operates with high autonomy and influences clinical decisions without adequate human oversight"*. Furthermore, INSYTE can reveal "bad" combinations, such as *"a rigidly specified system being potentially deployed in a very dynamic world"*. These qualities could enable INSYTE to *"support risk analysis and proactive mitigation strategies, integrating tools like FMEA [Failure Modes and Effects Analysis]"*. Its whole system, contextual perspective is crucial for effective risk mitigation, as *"it is useful to formally consider these [dimensions] in the round when assessing the risk associated with a system or where scrutiny may need to be directed"*. Ultimately, INSYTE is seen as *"Useful for safety assurance [. . .] particularly thinking of multi-stakeholder and multi-organisation safety committees which convene over the life of a system."*

While fewer evaluators focused on INSYTE's regulatory and certification potential, those who did emphasised it strongly. A medical sector evaluator noted that INSYTE's *"structure and nuance for assessing AI, which could support more sophisticated and context-aware regulatory decisions,"* could significantly advance current AI regulation, as *"there aren't many regulators that attend to this breadth of issues within their regulatory scope."* An aviation sector participant highlighted INSYTE's ability to establish *"standardised criteria for compliance and benchmarking."* Furthermore, an automotive domain evaluator suggested INSYTE could serve as a *"criticality assessment, similar to Tool Classification Level or ASIL [(Automotive Safety Integrity Level)] in ISO 26262,"* allowing standard bodies to *"tailor the requirements on the development and testing of the system based on the criticality level."* Regulators, they added, could then *"tailor their level of assessment before granting a permit with regard to the criticality level of the system being assessed."*

The answers on whether INSYTE could help with the determination of liability were mixed, with most saying they could see it having 'some' value for this purpose. One evaluator, from the aviation domain, remarked that it could help *"document system performance, failure modes, and risk assessments"* which, in turn, could support legal accountability. Another participant, a medic, highlighted the risk to clinicians of being placed in the position of being a safeguard on a machine, in a peripheral role, lacking active involvement in or understanding of the system, but liable for its outcomes. Sometimes called the problem of clinicians acting as 'liability sink' [71], this issue could be spotlighted by INSYTE, showing through the other dimensions a system that would be intrinsically difficult to supervise. It should be noted that none of the evaluators came from a legal background. The authors' views on how INSYTE might inform decision on liability is given in the discussion in Section 6.2.7 below.

In the free-text responses to Q9, the evaluators identified some uses of the INSYTE framework that we had not included as options. One medic highlighted that it would likely be useful in an after-the-event safety investigation. Another medic thought the framework could be used in the training of staff to make them aware of potential pitfalls of the system – or complexities they should have a deeper awareness of. An engineer in the automotive sector thought that INSYTE patterns could potentially be used as a summary of an upfront specification from organisations, indicating the 'shape' of the system they are expecting to receive, or are prepared to approve. Another evaluator thought the system could be used to establish adopter readiness for a given system, and whether they have the appropriate controls in place.

Finally, in the free-text responses to Q10, six evaluators responded. There were two clear themes. First, instantiating the framework requires knowledge about, understanding of, and experience of the system. Cross-expert and cross-stakeholder teams can help to address this. Second, the inclusion of multiple examples is important to help users instantiate the framework.

Key insights from the answers to the usefulness questions (Q7 to Q10):

- The INSYTE framework has significant value, particularly for cross-stakeholder communication, decision-making over the system lifecycle, safety assurance, and regulation;
- Evaluator responses were more mixed on the value of INSYTE to inform decisions about legal liability, indicative of the ongoing international efforts to establish the details of such liability;
- The INSYTE framework provides a clear, well-structured basis for deeper reflection and nuanced system comparisons;
- The whole-system classification offers a valuable, holistic perspective on system complexity;
- It allows risk-relevant characteristics, and combinations of characteristics, to be highlighted where otherwise they might be overlooked;
- The visual representation of a system on a radar chart is a highly compelling way to appraise a system overall as well as focus on its key characteristics.

6 DISCUSSION AND CONCLUDING REMARKS

Our aim for the INSYTE framework has been to address the shortcomings of current classification frameworks, which are rigid, coarse-grained, prone to misuse, and often limited to certain systems. INSYTE offers a more detailed way to classify agentic AI, and applies to all AI-enabled systems — from decision-support to fully autonomous, traditional to advanced — across all sectors. A key feature of INSYTE is its single-view visual representation of a system: the INSYTE pattern, displayed on an eight-axis radar chart. Radar charts enable interested parties to gain *"an immediate sense of the big picture, as well as the detail for each individual variable."* [107]. These INSYTE patterns support open communication among stakeholders, regardless of their technical background. They are sufficiently detailed to yield insights for system designers and developers, yet intuitive enough for non-technical stakeholders to grasp a system's characteristics at a glance.

6.1 Using the INSYTE framework

Stakeholders applying the framework will need *knowledge* and *understanding* of the system they are classifying. Instantiating an INSYTE pattern should not require anyone to disclose proprietary information, but it does require knowledge of the class of algorithms, the types of models, and the intended functionality of the system, as well as sufficient understanding of a system's complexity. As such, it is likely that INSYTE patterns will be most accurate when the framework is applied by cross-functional teams, with each member bringing different expertise.

The articulated level descriptions for each dimension furnish a measure of objectivity for applying the framework. Even so, working through the process of applying INSYTE to a specific system still involves the exercise of *careful judgement*. For example, judgements need to be made about where to draw the boundaries of the system, about the intended functionality of the system, and about its intended operating environment. The inclusion of a justification table in the online worksheet – whereby users complete a brief explanation for the level selected for each dimension – provides a basis for transparently communicating these judgements to recipients and observers. Example justification reports are provided in Appendix A.

6.2 Benefits and uses of INSYTE

The INSYTE framework is intended to serve both descriptive and normative roles. Descriptively, it classifies AI and AI-enabled autonomous systems and facilitates cross-disciplinary discussion. Normatively, it supports communication, informs design and deployment, augments safety assurance, and structures system assessment and regulation. The feedback from our evaluators confirms these intended uses. Here, we elaborate on the potential usefulness of INSYTE, based on our own reflections, as shaped by the lessons drawn from the evaluation of the framework.

6.2.1 A structure to illuminate the connections between system characteristics. By determining, for each of the eight dimensions, at what level a given system should be described, INSYTE users explicate what the system is intended to do (functionality dimensions), how (design dimensions), where (environment dimensions), and how autonomously (operational independence dimensions). The framework then offers a basis for a comprehensive synthesis of how these multiple variables combine, represented in the system's INSYTE pattern. INSYTE patterns can help to: illuminate connections between the eight characteristics, including interdependencies and reciprocal effects; inform discussions about ideal and non-ideal couplings; and consider trade-offs, determining where added complexity yields only negligible benefits. We anticipate that the paper and its supporting online tool will stimulate and structure such future work. In this way, INSYTE may help designers, engineers, and developers to explore the implications of different design concepts. Indeed, after working through this process, designers might use INSYTE as a model or template to work towards, rather than just a framework to retrospectively apply to a system to establish its classification.

6.2.2 A tool for cross-stakeholder communication and understanding. The evaluation of INSYTE validated this potential use of the framework. As an illustration from the medical sector, health IT systems deployed in the English National Health Service (NHS) must comply with two clinical risk management standards: the NHS Digital Clinical Safety standards DCB0129 [28] and DCB0160 [29]. These standards require developers and clinicians to come together in workshops to analyse clinical risks associated with the device and discuss mitigation. From one author's personal experience, when the device is an AI-based system, a disproportionate amount of time in these workshops is spent addressing knowledge gaps and establishing a shared baseline of understanding about a system. Participants can have 'surprises' about a system's characteristics even once the workshops are well underway. This challenge extends beyond healthcare to other regulated safety-critical domains. In these settings, knowledge imbalances are common, and INSYTE can play a valuable role in facilitating quicker, more informed discussions. Crucially, its use does not require sharing proprietary information, allowing developers and manufacturers to support these activities without compromising commercial advantage.

6.2.3 A tool to illustrate research and development trajectories. The INSYTE framework can illustrate the difference between past, current, and planned designs for systems. INSYTE patterns can show the comparative sophistication of different systems. They can also illustrate the increasing sophistication of variants of the same system, as the healthcare and infrastructure management examples show in Appendix A. The visualisation offers a simple basis for designers and engineers to say to others - perhaps in the board room - *"the plan is to move to Level 5 in this dimension,"* or to show purchasers the more advanced characteristics new variants of a system have.

While INSYTE can illustrate trajectories of research and development, it is important to understand that 'higher' on any dimension does not inherently mean 'better' [84]. 'Better' is a function of optimal, safe performance in the intended context. Moreover, achieving the highest level on every dimension isn't always feasible, and even human experts rarely reach Level 5 across all aspects. Within the automotive domain, the Levels of Autonomy have been criticised for engendering a culture that constantly strives for higher system autonomy [55, 117]. We caution against interpreting the INSYTE model in the same way.

6.2.4 A tool to support the continuous monitoring of a system in operation. Displaying a system's INSYTE pattern on its user interface could powerfully support continuous operational monitoring. Users, such as clinicians or pilots, could refer to it to see how a system's characteristics and their combinations affect performance, enabling them to provide structured feedback for iterative development. Furthermore, the INSYTE interface could be used to reveal deviations between a system's intended and actual use, potentially helping to reduce the gap between work-as-done and work-as-imagined [54]. This may help users realign operations by intervening more or less often, narrowing functionality, or altering the operating environment of a system in real time. INSYTE could also provide real-time updates on a system's adaptive autonomy —its *planned* variation in operational independence during a mission [38]. It could even evolve to allow users to directly adjust system levels on its eight dimensions, with appropriate safety guardrails.

6.2.5 A framework to support safety engineering and safety assurance. To reduce risk to "as low as reasonably practicable" (ALARP) [52, 74] and to assure this in a safety case (an evidence-based argument that a system is acceptably safe in a particular setting [12, 67, 106]), safety engineers need to take a contextual, 'whole system' perspective. Risk does not emerge in a vacuum; it often arises at the intersection of a system's design, functionality, and operating environment [19]. For instance, dangerous behaviour could arise from interactions between a system's adaptiveness, its diverse outputs, and a dynamic environment. INSYTE offers a template for thinking about hazards, risks, and mitigations at these boundaries, enabling risk to be considered "in the round."

Safety engineers could identify specific risks associated with particular INSYTE patterns and then establish what mechanisms are available to reduce risk arising from those patterns. For example, we might reduce intended environmental diversity or dynamism, increase or decrease human oversight, or manage the inherent complexity associated with certain functionalities. The high level of the INSYTE framework's abstraction makes it applicable to a wide range of sectors. To make it useful for specific industries, INSYTE could be adapted to align with established sector-specific safety methods. For example, in the automotive sector, it could help us formally describe the safety features of a driving situation based on physical models of driving dynamics.

INSYTE patterns can also indicate when the expectations on the human-in-the-loop are too demanding [9, 71]. For example, systems that score highly on depth of functionality (axis 4) are unlikely to be ones that users, even expert users, are able to verify as functioning correctly in real-time. Thus, INSYTE can spotlight over-reliance on the human to prevent hazards and mitigate risk during operation [86], indicating where alternative solutions should be considered.

6.2.6 A better system of regulatory classification. Several nations and jurisdictions are developing risk-based regulatory regimes for AI. Some, like the EU, Canada, South Korea, and Brazil, are working towards risk-based "hard law" with legally binding regulations [14, 91, 92, 125]. Others, such as the UK and Japan, are pursuing risk-based "soft law" through non-binding guidelines [26, 48]. These risk-based frameworks classify AI systems by the criticality of their application domain, with systems in safety-critical sectors classified as high-risk. However, the INSYTE framework can show when a high-risk system (based on the criticality of application domain) is in fact low on dimensions like underspecification or environmental dynamism, and is therefore less *likely* to cause (unforeseen) harm, incurs less *uncertainty*, and is more *controllable*, than a high-risk system embodying higher levels on these dimensions. INSYTE's dimensions can therefore help to improve upon simplistic risk-based legislative categories; INSYTE patterns could even be used to calculate a global risk score for regulators and insurers.

Many AI regulations are also principles-based, focusing on human values and ethical principles, such as fairness [26, 93, 125]. Consideration of a system's level on the underspecification and environment dimensions of the INSYTE framework, alongside use of tools like model cards [85], could help to determine how rigorously a system should be tested for bias,for instance in underspecified systems where data shift is a concern. Furthermore, some sector-specific bodies, like in the maritime domain [63], base regulatory oversight on a system's autonomy level. INSYTE can offer a more nuanced perspective on autonomy, identifying risk-relevant characteristics that may otherwise be overlooked.

6.2.7 A tool to assist courts when considering liability. There are two ways in which the INSYTE framework could also be useful when considering designer liability and product liability. First, it could help define a designer's duty by identifying when characteristics — like underspecification, adaptiveness, or environmental diversity — make harm unusually likely for a system's product class. Reaching specific INSYTE thresholds might even trigger regulatory re-certification for updated AI. We postulate that this might usefully complement the thresholds in Frontier AI Safety Frameworks, which are heavily focused on the capability of the model in one particular aspect, e.g., cybersecurity [7]. Second, the framework could assist legal professionals in retrospectively assessing harm foreseeability after an incident, indicating if risk mitigation was inadequate given the system's INSYTE pattern.

6.3 Concluding remarks

The INSYTE framework enables a nuanced, granular classification of AI and AI-enabled autonomous systems, which aligns with the widely used OECD definition of a deployed AI system. The ability to classify and assess a range of such systems on INSYTE's eight dimensions, and to represent this on a radar chart, offers fertile ground for future research and to support multiple activities concerning their communication, design, development, deployment, and regulation. As frontier AI technologies continue to evolve, their applications continue to diversify, and the standards and regulations governing their use are being developed and agreed upon, our framework will require further assessment and may need fine-tuning to maintain its applicability to the entire spectrum of AI systems enabled by such advances – a task that we plan to monitor closely and address as needed.

ACKNOWLEDGEMENTS

This work has been supported by the University of York's Centre for Assuring Autonomy (funded by a Lloyd's Register Foundation grant), EPSRC projects EP/W011239/1 'Assuring Responsibility for Trustworthy Autonomous Systems' and EP/V026747/1 'UKRI Trustworthy Autonomous Systems Node in Resilience', and ARIA Safeguarded AI project 'Universal Stochastic Modelling, Verification & Synthesis Framework'. The authors are grateful to Alice Cunningham, Yameen Hayat, Jeffry Hogg, Olaolu Olabintan, Yasmeen Rafiq, Georgia Sowerby, Robyn Spence, Ioannis Stefanakos, Jihong Zhu, and the other (anonymous) participants in the evaluation of the INSYTE framework.

AUTHOR CONTRIBUTIONS

ZP – conceptualisation (lead); writing - original draft (lead); methodology; supervision; investigation (lead); visualisation; writing - review & editing; funding acquisition. RC – conceptualisation (lead); writing - original draft; methodology; supervision; software; investigation (lead); visualisation; writing - review & editing; funding acquisition. EL – conceptualisation; writing - original draft; software (lead); validation; writing - review & editing. VH – conceptualisation; investigation; software; validation; formal analysis; writing - review & editing. PR – conceptualisation; writing - original draft; investigation; software; writing - review & editing. SB - conceptualisation; writing - original draft; validation; writing - review & editing; funding acquisition. IH – conceptualisation; writing - original draft; supervision; writing - review & editing; funding acquisition. TL – conceptualisation; investigation; writing - review & editing. JAMcD – conceptualisation; writing - original draft; writing - review & editing; funding acquisition. JM – conceptualisation; writing - original draft; investigation; writing - review & editing. HM – conceptualisation; investigation; writing - review & editing. PN – conceptualisation; writing - review & editing; funding acquisition. PM – conceptualisation; writing - original draft; writing - review & editing; funding acquisition. CP – conceptualisation; writing - original draft; software; visualisation; writing - review & editing. IS – conceptualisation; writing - original draft; investigation; writing - review & editing. JZ – writing - original draft; validation; writing - review & editing.

REFERENCES

[1] Deepak Bhaskar Acharya, Karthigeyan Kuppan, and B Divya. 2025. Agentic AI: Autonomous intelligence for complex goals–A comprehensive survey. *IEEE Access* 13 (2025), 18912 – 18936.

[2] Sadeq Almeaibed, Saba Al-Rubaye, Antonios Tsourdos, and Nicolas P Avdelidis. 2021. Digital twin analysis to promote safety and security in autonomous vehicles. *IEEE Communications Standards Magazine* 5, 1 (2021), 40–46.

[3] César Álvarez, María Isabel González, and Alejandro Gracia. 2020. Flight procedures automation: Towards flight autonomy in manned aircraft. In *2020 IEEE/AIAA 39th Digital Avionics Systems Conference (DASC)*. IEEE, San Antonio, Texas, USA, 1–8.

[4] American Bureau of Shipping. 2019. ABS Advisory on Autonomous Functionality. https://ww2.eagle.org/content/dam/eagle/advisories-and-debriefs/abs-advisory-on-autonomous-functionality.pdf

[5] Markus Anderljung, Joslyn Barnhart, Anton Korinek, Jade Leung, Cullen O'Keefe, Jess Whittlestone, Shahar Avin, Miles Brundage, Justin Bullock, Duncan Cass-Beggs, et al. 2023. Frontier AI regulation: Managing emerging risks to public safety. arXiv preprint arXiv:2307.03718.

[6] Eric Anderson, Timothy Fannin, and Brent Nelson. 2018. Levels of aviation autonomy. In *2018 IEEE/AIAA 37th Digital Avionics Systems Conference (DASC)*. IEEE, London, England, UK, 1–8.

[7] Artificial Intelligence Safety Institute (AISI). 2024. Conference on Frontier AI Safety Frameworks. https://www.aisi.gov.uk/work/conference-on-frontier-ai-safety-frameworks

[8] Assuring Autonomy International Programme. 2024. AAIP Demonstrator Projects. https://www.york.ac.uk/assuring-autonomy/about/aaip/demonstrators/.

[9] Lisanne Bainbridge. 1983. Ironies of automation. *Automatica* 19, 6 (1983), 775–779.

[10] Jenay M Beer, Arthur D Fisk, and Wendy A Rogers. 2014. Toward a framework for levels of robot autonomy in human-robot interaction. *Journal of Human-Robot Interaction* 3, 2 (2014), 74–99.

[11] Danielle S Bitterman, Hugo JWL Aerts, and Raymond H Mak. 2020. Approaching autonomy in medical artificial intelligence. *The Lancet Digital Health* 2, 9 (2020), e447–e449.

[12] Robin Bloomfield and John Rushby. 2024. Confidence in assurance 2.0 cases. In *The Practice of Formal Methods: Essays in Honour of Cliff Jones, Part I*. Springer, Cham, Switzerland, 1–23.

[13] Michael Bonner, Robert Taylor, Keith Fletcher, and Christopher Miller. 2000. Adaptive automation and decision aiding in the military fast jet domain. In *Proceedings of Human Performance, Situation Awareness, and Automation*. Her Majesty's Stationery Office (HMSO), 154–159.

[14] Brazilian Chamber of Deputies. 2020. Projeto de Lei n. 21/2020. https://www.camara.leg.br/proposicoesWeb/prop_mostrarintegra?codteor=1853928&filename=PL%2021/2020

[15] James R Bright. 1957. Myths and fallacies of automation. *SAE Transactions* 65 (1957), 769–779.

[16] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems (NeurIPS 2020)*, Vol. 33. 1877–1901.

[17] Bureau Veritas. 2019. Guidelines for Autonomous Shipping. Guidance Note NI 641 DT R01 E. https://erules.veristar.com/dy/data/bv/pdf/641-NI_2019-10.pdf

[18] Simon Burton, Radu Calinescu, and Raffaela Mirandola. 2024. Resilience and Antifragility of Autonomous Systems (Dagstuhl Seminar 24182). *Dagstuhl Reports* 14, 4 (2024), 142–163.

[19] Simon Burton, Ibrahim Habli, Tom Lawton, John McDermid, Phillip Morgan, and Zoe Porter. 2020. Mind the gaps: Assuring the safety of autonomous systems from an engineering, ethical, and legal perspective. *Artificial Intelligence* 279 (2020), 103201.

[20] Central Commission for the Navigation of the Rhine. 2022. International Definition of Levels of Automation in Inland Navigation. https://ccr-zkr.org/files/documents/AutomatisationNav/DefinitionAutomatisation_en.pdf

[21] Alan Chan, Carson Ezell, Max Kaufmann, Kevin Wei, Lewis Hammond, Herbie Bradley, Emma Bluemke, Nitarshan Rajkumar, David Krueger, Noam Kolt, et al. 2024. Visibility into AI agents. In *The 2024 ACM Conference on Fairness, Accountability, and Transparency (FAccT '24)*. ACM, Rio de Janeiro, Brazil, 958–973.

[22] Alan Chan, Rebecca Salganik, Alva Markelius, Chris Pang, Nitarshan Rajkumar, Dmitrii Krasheninnikov, Lauro Langosco, Zhonghao He, Yawen Duan, Micah Carroll, et al. 2023. Harms from increasingly agentic algorithmic systems. In *Proceedings of the 2023 ACM Conference on Fairness, Accountability, and Transparency (FAccT '23)*. ACM, Chicago, USA, 651–666.

[23] M. Channon. 2019. Automated and Electric Vehicles Act 2018: An Evaluation in Light of Proactive Law and Regulatory Disconnect. *European Journal of Law and Technology* 10 (2019), 26.

[24] Bruce Clough. 2002. Metrics, schmetrics! How do you track a UAV's autonomy?. In *1st UAV Conference*. AIAA, Portsmouth, Virginia, USA, 3499.

[25] Alexander D'Amour, Katherine Heller, Dan Moldovan, Ben Adlam, Babak Alipanahi, Alex Beutel, Christina Chen, Jonathan Deaton, Jacob Eisenstein, Matthew D Hoffman, et al. 2022. Underspecification presents challenges for credibility in modern machine learning. *Journal of Machine Learning Research* 23, 226 (2022), 1–61.

[26] Department for Science, Innovation & Technology. 2023. A pro-innovation approach to AI regulation. https://www.gov.uk/government/publications/ai-regulation-a-pro-innovation-approach/white-paper

[27] Det Norske Veritas. 2018. DNVGL-CG-0261: Autonomous and Remotely Operated ships. https://standards.globalspec.com/std/13057765/dnvgl-cg-0264

[28] NHS Digital. 2018. DCB0129: Clinical Risk Management: its Application in the Manufacture of Health IT Systems.

[29] NHS Digital. 2018. DCB0160: Clinical Risk Management: its Application in the Deployment and Use of Health IT Systems.

[30] Zane Durante, Qiuyuan Huang, Naoki Wake, Ran Gong, Jae Sung Park, Bidipta Sarkar, Rohan Taori, Yusuke Noda, Demetri Terzopoulos, Yejin Choi, et al. 2024. Agent AI: Surveying the horizons of multimodal interaction. arXiv preprint arXiv:2401.03568.

[31] Phillip J Durst, Wendell Gray, and Michael Trentini. 2011. A non-contextual model for evaluating the autonomy level of intelligent unmanned ground vehicles. In *Proceedings of the 2011 Ground Vehicle Systems Engineering and Technology Symposium*. National Defense Industrial Association (NDIA), Dearborn, Michigan, USA, 1–6.

[32] EarthSense, Inc. 2020. Levels of Autonomy for Field Robots. https://www.earthsense.co/news/2020/7/24/levels-of-autonomy-for-field-robots Accessed: November 30, 2024.

[33] Mica R Endsley. 2017. From here to autonomy: lessons learned from human–automation research. *Human factors* 59, 1 (2017), 5–27.

[34] Mica R Endsley. 2018. Level of automation forms a key aspect of autonomy design. *Journal of Cognitive Engineering and Decision Making* 12, 1 (2018), 29–34.

[35] Mica R Endsley and David B Kaber. 1999. Level of automation effects on performance, situation awareness and workload in a dynamic control task. *Ergonomics* 42, 3 (1999), 462–492.

[36] Naeem Esfahani and Sam Malek. 2013. Uncertainty in self-adaptive software systems. In *Software Engineering for Self-Adaptive Systems II: International Seminar, Dagstuhl Castle, Germany, October 24-29, 2010 Revised Selected and Invited Papers*. Springer, Cham, Switzerland, 214–238.

[37] Federal Ministry for Economic Affairs and Energy (BMWi). 2019. Technology Scenario 'Artificial Intelligence in Industrie 4.0'. https://www.plattform-i40.de/IP/Redaktion/EN/Downloads/Publikation/AI-in-Industrie4.0.pdf?__blob=publicationFile&v=5

[38] Alireza Fereidunian, Matti Lehtonen, Hamid Lesani, Caro Lucas, and Mikael Nordman. 2007. Adaptive autonomy: Smart cooperative cybernetic systems for more humane automation solutions. In *2007 IEEE International Conference on Systems, Man and Cybernetics*. IEEE, Montreal, QC, Canada, 202–207.

[39] Paul Festor, Ibrahim Habli, Yan Jia, Anthony Gordon, A Aldo Faisal, and Matthieu Komorowski. 2021. Levels of autonomy and safety assurance for AI-based clinical decision systems. In *Computer Safety, Reliability, and Security. SAFECOMP 2021 Workshops: DECSoS, MAPSOD, DepDevOps, USDAI, and WAISE, York, UK, September 7, 2021, Proceedings 40*. Springer, Cham, Switzerland, 291–296.

[40] Michael Fisher, Viviana Mascardi, Kristin Yvonne Rozier, Bernd-Holger Schlingloff, Michael Winikoff, and Neil Yorke-Smith. 2021. Towards a framework for certification of reliable autonomous systems. *Autonomous Agents and Multi-Agent Systems* 35 (2021), 1–65.

[41] Eduard Fosch-Villaronga, Pranav Khanna, Hadassah Drukarch, and Bart HM Custers. 2021. A human in the loop in surgery automation. *Nature Machine Intelligence* 3, 5 (2021), 368–369.

[42] Johannes Fottner, Dana Clauer, Fabian Hormes, Michael Freitag, Thies Beinke, Ludger Overmeyer, Simon Nicolas Gottwald, Ralf Elbert, Tessa Sarnow, Thorsten Schmidt, et al. 2021. Autonomous systems in intralogistics: state of the Art and future research challenges. *Logistics Research* 14, 1 (2021), 1–41.

[43] Jörgen Frohm, Veronica Lindström, Johan Stahre, and Mats Winroth. 2008. Levels of automation in manufacturing. *Ergonomia-an International Journal of Ergonomics and Human Factors* 30, 3 (2008).

[44] Shihao Fu. 2025. L2+ ADAS Outpaces L3 in Europe, US$4B by 2042. https://www.idtechex.com/en/research-article/l2-adas-outpaces-l3-in-europe-us-4b-by-2042/32860

[45] Thomas Gamer, Mario Hoernicke, Benjamin Kloepper, Reinhard Bauer, and Alf J Isaksson. 2020. The autonomous industrial plant–future of process engineering, operations and maintenance. *Journal of Process Control* 88 (2020), 101–110.

[46] Tom M Gasser and Daniel Westhoff. 2012. BASt-study: Definitions of automation and legal issues in Germany.

[47] Global Mining Guidelines Group. 2024. Guideline for the Implementation of Autonomous Systems in Mining (Version 2). https://gmggroup.org/wp-content/uploads/2024/08/GUIDELINE_Implementation-of-Autonomous-Systems-1.pdf

[48] Japanese Government. 2021. AI Governance in Japan: Report 2021. https://www.soumu.go.jp/main_content/000769212.pdf. https://www.soumu.go.jp/main_content/000769212.pdf.

[49] Vincenzo Grassi, Raffaela Mirandola, and Diego Perez-Palacin. 2024. A conceptual and architectural characterization of antifragile systems. *Journal of Systems and Software* 213] (2024), 112051.

[50] David J Gunkel. 2012. *The machine question: Critical perspectives on AI, robots, and ethics.* MIT Press, Cambridge, MA.

[51] Tamás Haidegger. 2019. Autonomy for surgical robots: Concepts and paradigms. *IEEE Transactions on Medical Robotics and Bionics* 1, 2 (2019), 65–76.

[52] Health and Safety Executive. 2025. ALARP at a glance. https://www.hse.gov.uk/enforce/expert/alarpglance.htm

[53] Brendan Hertel, Ryan Donald, Christian Dumas, and S Reza Ahmadzadeh. 2022. Methods for combining and representing non-contextual autonomy scores for unmanned aerial systems. In *2022 8th International Conference on Automation, Robotics and Applications (ICARA)*. IEEE, New York, NY, USA, 135–139.

[54] Erik Hollnagel. 2017. Why is work-as-imagined different from work-as-done? In *Resilient Health Care*. Vol. 2. CRC Press, London, UK, 279–294.

[55] Debbie Hopkins and Tim Schwanen. 2021. Talking about automated vehicles: What do levels of automation do? *Technology in Society* 64 (2021), 101488.

[56] Hui-Min Huang. 2007. Autonomy levels for unmanned systems (ALFUS) framework: safety and application issues. In *Proceedings of the 2007 Workshop on Performance Metrics for Intelligent Systems*. ACM, Washington D.C., USA, 48–53.

[57] Hui-Min Huang, Kerry Pavek, Brian Novak, James Albus, and E Messin. 2005. A framework for autonomy levels for unmanned systems (ALFUS). In *Proceedings of the AUVSI's unmanned systems North America*. Association for Unmanned Vehicle Systems International (AUVSI), Baltimore, Maryland, USA, 849–863.

[58] Carlos C Insaurralde and David M Lane. 2012. Autonomy-assessment criteria for underwater vehicles. In *2012 IEEE/OES Autonomous Underwater Vehicles (AUV)*. IEEE, New York, NY, USA, 1–8.

[59] Institute of Marine Engineering, Science and Technology. 2019. Autonomous Shipping: Putting the Human Back in the Headlines. https://safety4sea.com/wp-content/uploads/2019/09/IMAREST-Autonomous-shipping-Putting-the-human-back-in-the-headlines-2019_09.pdf

[60] INSYTE project. 2025. INSYTE system classification worksheet. https://automation-radar-generator.vercel.app/.

[61] International Electrotechnical Commission. 2014. *Railway applications - Urban guided transport management and command/control systems - Part 1: System principles and fundamental concepts*. Standard IEC 62290-1. International Electrotechnical Commission, Geneva, Switzerland.

[62] International Electrotechnical Commission (IEC). 2017. IEC 60601-4-1:2017 - Medical electrical equipment - Part 4-1: Guidance and interpretation - Medical electrical equipment and medical electrical systems employing a degree of autonomy. Published by British Standards Institute, 2017.

[63] International Maritime Organization (IMO). 2021. MSC.1/Circ.1638. Outcome of the regulatory scoping exercise for the use of maritime autonomous surface ships (MASS).

[64] J.L. Jones. 2006. Robots at the tipping point: the road to iRobot Roomba. *IEEE Robotics & Automation Magazine* 13, 1 (2006), 76–78.

[65] David B Kaber. 2018. A conceptual framework of autonomous and automated agents. *Theoretical Issues in Ergonomics Science* 19, 4 (2018), 406–430.

[66] Atoosa Kasirzadeh and Iason Gabriel. 2025. Characterizing AI agents for alignment and governance. arXiv preprint arXiv:2504.21848.

[67] Tim Kelly. 1998. *Arguing Safety, a Systematic Approach to Managing Safety Cases.* Ph. D. Dissertation. Department of Computer Science, University of York.

[68] Kenneth W Kolence and Philip J Kiviat. 1973. Software unit profiles & Kiviat figures. *ACM SIGMETRICS Performance Evaluation Review* 2, 3 (1973), 2–12.

[69] Noam Kolt. 2025. Governing AI agents. arXiv preprint arXiv:2501.07913.

[70] Stefan Kugele, Ana Petrovska, and Ilias Gerostathopoulos. 2021. Towards a taxonomy of autonomous systems. In *Lecture Notes in Computer Science (LNCS) Volume 12857, European Conference on Software Architecture (ECSA)*. Springer, Cham, Switzerland, 37–45.

[71] Tom Lawton, Phillip Morgan, Zoe Porter, Shireen Hickey, Alice Cunningham, Nathan Hughes, Ioanna Iacovides, Yan Jia, Vishal Sharma, and Ibrahim Habli. 2024. Clinicians risk becoming 'liability sinks' for artificial intelligence. *Future Healthcare Journal* 11, 1 (2024), 100007.

[72] Audrey Lee, Turner S Baker, Joshua B Bederson, and Benjamin I Rapoport. 2024. Levels of autonomy in FDA-cleared surgical robots: a systematic review. *NPJ Digital Medicine* 7, 1 (2024), 103.

[73] Thomas Lee, Susan Mckeever, and Jane Courtney. 2021. Flying free: A research overview of deep learning in drone navigation autonomy. *Drones* 5, 2 (2021), 52.

[74] Nancy G Leveson. 2016. *Engineering a safer world: Systems thinking applied to safety.* MIT Press, Cambridge, MA.

[75] Lloyd's Register. 2017. LR Code for Unmanned Marine Systems. https://www.lr.org/en/knowledge/press-room/press-listing/press-release/2024/new-code-to-certify-unmanned-vessels-announced/

[76] John R Lucas. 1961. Minds, machines and gödel. *Philosophy* 36, 137 (1961), 112–127.

[77] Michael Luck and Mark d'Inverno. 1995. A formal framework for agency and autonomy. In *ICMAS*, Vol. 95. AAAI Press/MIT Press, Washington, DC/Cambridge, MA, US, 254–260.

[78] Matt Luckcuck, Marie Farrell, Louise A Dennis, Clare Dixon, and Michael Fisher. 2019. Formal specification and verification of autonomous robotic systems: A survey. *ACM Computing Surveys (CSUR)* 52, 5 (2019), 1–41.

[79] Ronald Lumia and James S Albus. 1988. Teleoperation and autonomy for space robotics. *Robotics and Autonomous Systems* 4, 1 (1988), 27–33.

[80] J. Marson and K. Ferris. 2021. The Lexicon of Self-Driving Vehicles and the Fuliginous Obscurity of 'Autonomous Vehicles'. *Statute Law Review* 44 (2021), 1.

[81] John McDermid, Yan Jia, and Ibrahim Habli. 2024. Upstream and Downstream AI Safety: Both on the Same River? arXiv preprint arXiv:2501.05455.

[82] John A. McDermid, Radu Calinescu, Ibrahim Habli, Richard Hawkins, Yan Jia, John Molloy, Matt Osborne, Colin Paterson, Zoe Porter, and Philippa Ryan Conmy. 2024. The Safety of Autonomy: A Systematic Approach. *Computer* 57, 4 (2024), 16–25. https://doi.org/10.1109/MC.2023.3317329

[83] Kwan Mei-Ko. 1962. Graphic programming using odd or even points. *Chinese Math* 1 (1962), 237–277.

[84] Margaret Mitchell, Avijit Ghosh, Alexandra Sasha Luccioni, and Giada Pistilli. 2025. Fully autonomous AI agents should not be developed. arXiv preprint arXiv:2502.02649.

[85] Margaret Mitchell, Simone Wu, Andrew Zaldivar, Parker Barnes, Lucy Vasserman, Ben Hutchinson, Elena Spitzer, Inioluwa Deborah Raji, and Timnit Gebru. 2019. Model cards for model reporting. In *Proceedings of the 2019 ACM Conference on Fairness, Accountability, and Transparency (FAccT '19)*. ACM, Atlanta, GA, USA, 220–229.

[86] Helen E Monkhouse, Ibrahim Habli, and John McDermid. 2020. An enhanced vehicle control model for assessing highly automated driving safety. *Reliability Engineering & System Safety* 202 (2020), 107061.

[87] Debasmita Mukherjee, Kashish Gupta, Li Hsin Chang, and Homayoun Najjaran. 2022. A survey of robot learning strategies for human-robot collaboration in industrial settings. *Robotics and Computer-Integrated Manufacturing* 73 (2022), 102231.

[88] National Institute of Standards and Technology. 2023. Artificial Intelligence Risk Management Framework (AI RMF 1.0). https://doi.org/10.6028/NIST.AI.100-1

[89] Norwegian Forum for Autonomous Ships. 2017. Definitions for Autonomous Merchant Ships. https://nfas.autonomous-ship.org/wp-content/uploads/2020/09/autonom-defs.pdf

[90] OECD. 2024. AI System Definition Update. https://oecd.ai/en/wonk/ai-system-definition-update. Accessed: 2024-06-20.

[91] Parliament of Canada. 2022. Bill C-27: Digital Charter Implementation Act, 2022. https://www.parl.ca/legisinfo/en/bill/44-1/c-27

[92] National Assembly of the Republic of Korea. 2024. Act Fostering the AI Industry and Establishing a Foundation for Trustworthy AI. https://www.assembly.go.kr/portal/bbs/B0000051/view.do?nttId=2095056&menuNo=600101&sdate=&edate=&pageUnit=10&pageIndex=1

[93] Organisation for Economic Co-operation and Development. 2022. OECD Framework for the Classification of AI Systems. https://oecd.ai/en/classification

[94] Bizhao Pang, CH John Wang, and Kin Huat Low. 2021. Framework of Level-of-Autonomy-based concept of operations: UAS capabilities. In *2021 IEEE/AIAA 40th Digital Avionics Systems Conference (DASC)*. IEEE, New York, NY, USA, 1–10.

[95] Raja Parasuraman, Thomas B Sheridan, and Christopher D Wickens. 2000. A model for types and levels of human interaction with automation. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans* 30, 3 (2000), 286–297.

[96] David Lorge Parnas and Jan Madey. 1995. Functional documents for computer systems. *Science of Computer Programming* 25, 1 (1995), 41–61.

[97] Roger Penrose. 1989. *The emperor's new mind:Concerning computers, minds, and the laws of physics.* Oxford University Press, Oxford, UK.

[98] Diego Perez-Palacin and Raffaela Mirandola. 2014. Uncertainties in the modeling of self-adaptive systems: A taxonomy and an example of availability evaluation. In *Proceedings of the 5th ACM/SPEC International Conference on Performance Engineering*. ACM, Dublin, Ireland, 3–14.

[99] Jonathan Peter Powell, Anna Fraszczyk, Chun Nok Cheong, and Ho Ki Yeung. 2016. Potential benefits and obstacles of implementing driverless train operation on the Tyne and Wear Metro: a simulation exercise. *Urban Rail Transit* 2 (2016), 114–127.

[100] Marco Protti and Riccardo Barzan. 2007. UAV Autonomy–Which level is desirable?–Which level is acceptable? Alenia Aeronautica Viewpoint. In *Platform Innovations and System Integration for Unmanned Air, Land and Sea Vehicles (AVT-SCI Joint Symposium)*. RTO, Neuilly-sur-Seine, France, 1–12.

[101] Ryan W Proud, Jeremy J Hart, and Richard B Mrozinski. 2003. *Methods for determining the level of autonomy to design into a human spaceflight vehicle: a function specific approach.* Technical Report. NASA.

[102] Andres J Ramirez, Adam C Jensen, and Betty HC Cheng. 2012. A taxonomy of uncertainty for dynamically adaptive systems. In *2012 7th International Symposium on Software Engineering for Adaptive and Self-Managing Systems (SEAMS)*. IEEE, New York, NY, USA, 99–108.

[103] Dale Richards and Alex Stedmon. 2016. To delegate or not to delegate: A review of control frameworks for autonomous cars. *Applied Ergonomics* 53 (2016), 383–388.

[104] Ørnulf Jan Rødseth, Håvard Nordahl, and Åsa Hoem. 2018. Characterization of autonomy in merchant ships. In *2018 OCEANS-MTS/IEEE Kobe Techno-Oceans (OTO)*. IEEE, Kobe, Japan, 1–7.

[105] Ørnulf Jan Rødseth, Lars Andreas Lien Wennersberg, and Håvard Nordahl. 2022. Levels of autonomy for ships. In *Journal of Physics: Conference Series*, Vol. 2311. 012018. IOP Publishing, Bristol, UK, 1–9.

[106] John Rushby. 2015. *The interpretation and evaluation of assurance cases.* Technical Report. SRI International, SRI-CSL-15-01.

[107] M Joan Saary. 2008. Radar plots: a useful way for presenting multivariate health care data. *Journal of Clinical Epidemiology* 61, 4 (2008), 311–317.

[108] SAE International/ISO. 2021. J3016. Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles. https://www.sae.org/standards/content/j3016_202104/

[109] Luca Save and Beatrice Feuerberg. 2012. Designing human-automation interaction: a new level of automation taxonomy. In *In Human Factors: a view from an integrative perspective. Proceedings HFES Europe Chapter Conference.* Human Factors and Ergonomics Society (HFES), Toulouse, France, 43–55.

[110] Matteo Schiaretti, Linying Chen, and Rudy R Negenborn. 2017. Survey on autonomous surface vessels: Part I-A new detailed definition of autonomy levels. In *Computational Logistics: 8th International Conference, ICCL 2017*, Vol. Proceedings 8. Springer, Southampton, UK, 219–233.

[111] John Searle. 1984. *Minds, Brains, and Science.* Harvard University Press, Cambridge, MA, Chapter 2.

[112] SESAR Joint Undertaking. 2020. Automation in Air Traffic Management. Available online at: https://www.sesarju.eu/automation.

[113] Yonadav Shavit, Sandhini Agarwal, Miles Brundage, Steven Adler, Cullen O'Keefe, Rosie Campbell, Teddy Lee, Pamela Mishkin, Tyna Eloundou, Alan Hickey, et al. 2023. Practices for Governing Agentic AI Systems.

[114] Thomas B Sheridan, William L Verplank, and TL Brooks. 1978. Human/computer control of undersea teleoperators. In *NASA Ames Research Center, The 14th Annual Conference on Manual Control.* NASA Conference Publication (NASA-CP-2060), Washigton DC, USA.

[115] Monika Simmler and Ruth Frischknecht. 2021. A taxonomy of human–machine collaboration: capturing automation and technical autonomy. *AI & Society* 36, 1 (2021), 239–250.

[116] Ian Sommerville. 2015. *Software Engineering* (10th ed.). Pearson.

[117] Erik Stayton and Jack Stilgoe. 2020. It's time to rethink levels of automation for self-driving vehicles. *IEEE Technology and Society Magazine* 39, 3 (2020), 13–19.

[118] Mark Sujan, Dominic Furniss, David Embrey, Matthew Elliott, David Nelson, Sean White, Ibrahim Habli, and Nick Reynolds. 2019. Critical barriers to safety assurance and regulation of autonomous medical systems. In *Proceedings of the 29th European Safety and Reliability Conference (ESREL 2019).* Hannover, Germany, 4257–4262.

[119] Krti Tallam. 2025. Alignment, agency and autonomy in frontier AI: A systems engineering perspective. arXiv preprint arXiv:2503.05748.

[120] Eric R Teoh. 2020. What's in a name? Drivers' perceptions of the use of five SAE Level 2 driving automation systems. *Journal of Safety Research* 72 (2020), 145–151.

[121] Eric J Topol. 2019. High-performance medicine: the convergence of human and artificial intelligence. *Nature Medicine* 25, 1 (2019), 44–56.

[122] Bianca Torossian, Frank Bekkers, Tim Sweijs, Michel Roelen, Alen Hristov, and Salma Atalla. 2020. *The military applicability of robotic and autonomous systems.* Hague Centre for Strategic Studies.

[123] AM Turing. 1950. Computing Machinery and Intelligence. *Mind* 59, 236 (1950), 433.

[124] UK Parliament, House of Lords. 2018. Automated and Electric Vehicles Bill, Volume 791: debated on Tuesday 5 June 2018. https://hansard.parliament.uk/lords/2018-06-05/debates/C7E8FE14-B881-423E-AC35-1FF9839A1852/AutomatedAndElectricVehiclesBill Accessed: 2024-07-09.

[125] European Union. 2024. EU AI Act.

[126] United States. Department of Defense. 2005. *Unmanned Aircraft Systems Roadmap 2005-2030.* Technical Report. United States Department of Defense. https://rosap.ntl.bts.gov/view/dot/18248 Accessed: 2024-08-24.

[127] United States Department of Transportation (USDOT). 2016. Federal Automated Vehicles Policy: Accelerating the Next Revolution in Roadway Safety. https://www.transportation.gov/AV/federal-automated-vehicles-policy

[128] United States Department of Transportation (USDOT). 2017. Automated Driving Systems: A Vision for Safety. https://www.nhtsa.gov/technology-innovation/automated-driving-systems-vision-safety

[129] Marialena Vagia, Aksel A Transeth, and Sigurd A Fjerdingen. 2016. A literature review on the levels of automation during the years. What are the different taxonomies that have been proposed? *Applied Ergonomics* 53 (2016), 190–202.

[130] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in Neural Information Processing Systems* 30 (2017).

[131] YueChao Wang and JinGuo Liu. 2012. Evaluation methods for the autonomy of unmanned systems. *Chinese science bulletin* 57 (2012), 3409–3418.

[132] Danny Weyns. 2020. *An introduction to self-adaptive systems: A contemporary software engineering perspective.* John Wiley & Sons, West Sussex, UK.

[133] Danny Weyns, Radu Calinescu, Raffaela Mirandola, Kenji Tei, Maribel Acosta, Nelly Bencomo, Amel Bennaceur, Nicolas Boltz, Tomas Bures, Javier Camara, et al. 2023. Towards a research agenda for understanding and managing uncertainty in self-adaptive systems. *ACM SIGSOFT Software Engineering Notes* 48, 4 (2023), 20–36.

[134] Will Williamson, Ed Waltz, Jihane Mimih, and Jim Scrofani. 2020. Autonomy levels for small satellite clusters. In *2020 IEEE Conference on Cognitive and Computational Aspects of Situation Management (CogSIMA).* IEEE, Virtual Conference, 1–4.

[135] Ludwig Wittgenstein. 1969. *The Blue and Brown Books.* Harper & Row, New York, NY, USA.

[136] Holly A Yanco and Jill Drury. 2004. Classifying human-robot interaction: an updated taxonomy. In *2004 IEEE International Conference on Systems, Man and Cybernetics.* IEEE, The Hague, Netherlands, 2841–2846.

[137] Guang-Zhong Yang, James Cambias, Kevin Cleary, Eric Daimler, James Drake, Pierre E Dupont, Nobuhiko Hata, Peter Kazanzides, Sylvain Martel, Rajni V Patel, et al. 2017. Medical robotics—Regulatory, ethical, and legal considerations for increasing levels of autonomy. *Science Robotics* 2, 4 (2017), eaam8638.

[138] Andre Ye. 2021. Underspecification: The Dangerously Underdiscussed Problem Facing Machine Learning. https://medium.com/dataseries/underspecification-the-dangerously-underdiscussed-problem-facing-machine-learning-4882292c67c1

[139] Larry Young, Jeffrey Yetter, and Mark Guynn. 2005. System analysis applied to autonomy: Application to high-altitude long-endurance remotely operated aircraft. In *Infotech@ aerospace.* American Institute of Aeronautics and Astronautics (AIAA), Reston, VA, USA, 7103.

APPENDIX A: INSYTE APPLICATION EXAMPLES

The eight dimensions are each represented as an axis on the radar chart that the INSYTE framework uses. Here, we illustrate how the INSYTE framework culminates in the visual representation of a system's combinatorial position on each of these axes (its "INSYTE pattern"). These worked examples come from the ongoing research and development activities of some of the paper's authors and their research collaborators. Both for transparency and to ensure the examples are a useful resource for stakeholders applying INSYTE to their own systems, we include the justifications report for each set of INSYTE patterns.

A1. Automotive sector example: Automated Driving Components

Imagine that a technology corporation designs, develops, and manufactures automated driving components for vehicle manufacturers to assemble and integrate into a modular automated driving system. Its offering comprises three components that exemplify increasing complexity.

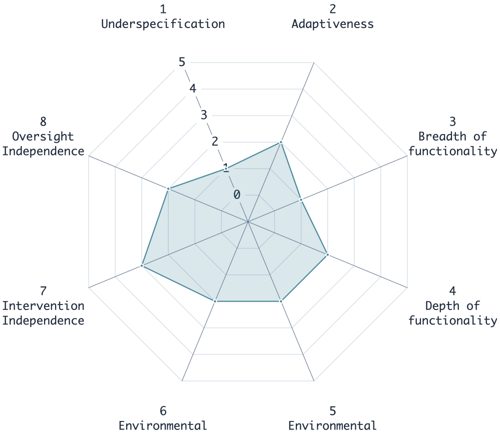


Fig. A.1. INSYTE pattern for the AEB component

First, there is Automated Emergency Braking (AEB), which performs emergency braking at different speeds, relying on machine-learnt perception models for the detection of vehicles, pedestrians, and cyclists. The INSYTE pattern for this component is given in Figure A.1.

Second, there is Highway Pilot Overtaking (HPO), which performs variants of overtaking and lane-level driving behaviour, proactively adapting to foreseen uncertainties on the highway, such as varying traffic density, speed changes, and lane availability. Third, there is Traffic Intersections (TI), which navigates complex, dynamic intersections such as T-junctions, roundabouts, four-way stops, and signal-free shared spaces. The INSYTE pattern for the HPO and TI components are given in Figure A.2. The supporting justifications report is given in Table A.1.

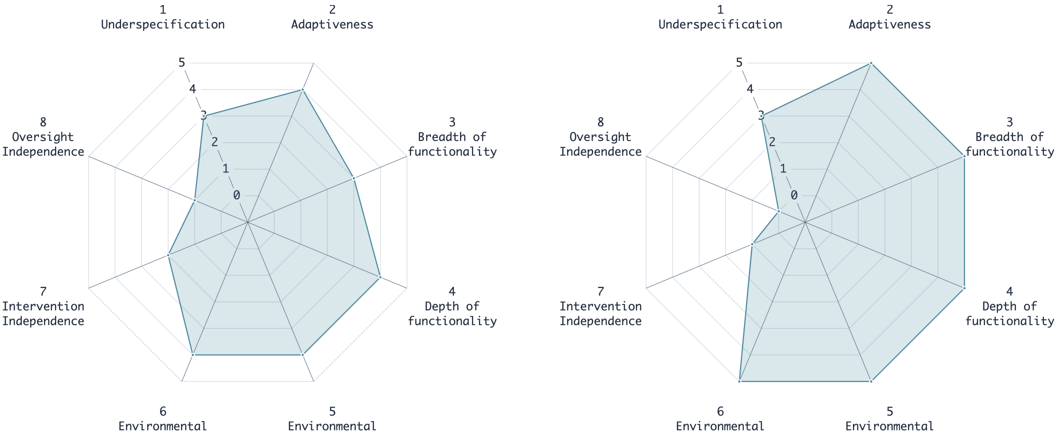


Fig. A.2. INSYTE patterns for the HPO (left) and TI (right) components

Table A.1. Justification Report for AEB, HPO, and TI

System Justification Report			
Dimension	**Automated Emergency Breaking (AEB)**	**Highway Pilot Overtaking (HPO)**	**Traffic Intersections (TI)**
1 – Underspecification	Level 1 AEB is primarily rule-based in its decision logic, but incorporates machine-learned perception models for object detection.	Level 3 HPO decisions are typically driven by learned models trained on driving data, although some constraints remain rule-encoded.	Level 3 TI decisions rely on learned policies to navigate complex, dynamic intersections, with only partial replaince on explicit traffic rules.
2 – Adaptiveness	Level 2 AEB reacts to multiple known types of uncertainty (e.g., sudden braking, obstacle appearance) using predefined tactics like braking thresholds and TTC calculations, but does not adapt dynamically beyond that.	Level 4 HPO system proactively adapts to multiple foreseen uncertainties (e.g., varying traffic density, speed changes, lane availability) using both predefined and dynamic planning tactics, such as trajectory replanning and vehicle intent prediction.	Level 5 TI continuously adapts to a wide variety of complex, multi-agent environments and can improve its behaviour through online learning or policy updates when encountering unforeseen traffic configurations or behaviour patterns.
3 – Breadth of functionality	Level 1 AEB performs variants of one task type - collision mitigation - across different speeds and object types, but it fulfils only a narrow decision criterion (emergency braking).	Level 3 HPO performs multiple related tasks such as lane change, speed adaptation, and safe distance maintenance - variants of overtaking and lane-level driving behaviour	Level 5 TI performs many variants of differenttask types (stopping, yielding, merging, unprotected turns, crosswalk negotiation, emergency manoeuvring, and multi-agent interaction) across varied intersection forms It also fulfils multiple decision criteria: safety, legality, time efficiency, and compliance with local driving norms..
4 – Depth of functionality	Level 2 AEB performs simple but fast decision-making using fixed logic and sensor inputs; depth is limited to threshold-based or simple ML-enhanced detection.	Level 4 HPO must continuously predict other vehicles' behaviour, assess dynamic risk, plan smooth trajectories, and optimise for timing, all in real time.	Level 5 TI integrates deep, multi-layered reasoning (complex multi-agent negotiations, prediction of intent and behaviour, uncertainty-aware planning, and hierarchical task decomposition). It dynamically reasons over conflicting goals, while handling ambiguous traffic patterns.
5 – Environmental diversity	Level 2 AEB typically deals with a few types of environmental element (vehicles, pedestrians, cyclists) and has limited interaction locic (e.g., brake if TTC < threshold).	Level 4 HPO operates in diverse highway environments with many elements (cars, trucks, lane types, road signs) and undertakes complex interactions with them (e.g., merging, overtaking, and risk-aware negotiation)	Level 5 TI must handle unbounded environment types (e.g., various intersection layouts, agent types, signal systems) with a vast range of possible interactions between vehicles, pedestrians, cyclists, and other road users
6 – Environmental dynamism	Level 2 AEB deals with sudden changes (e.g., vehicles stopping), but typically only one or two factors (e.g., speed or distance) vary at a time, and the rest remain stable.	Level 4 HPO faces dynamic traffic flow, merging, overtaking, and lane changes - often involving high speed or high magnitude changes, with medium frequency	Level 5 TI must handle high-frequency, high-speed, and high-magnitude changes simultaneously (e.g, sudden pedestrian crossings, vehcile right-of-way shifts, signal phase changes), especially in dense urban intersections.
7 – Intervention independence	Level 3 AEB operates autonomously during emergencies, but it is not involved in extended driving so it requires no intervention except for occasional resets or disengagements.	Level 2 HPO handles overtaking without continuous human intervention, but may require occasional input (e.g., for unclear traffic behaviour, unusual merging situations) to ensure safe execution.	Level 1 Due to high environmental complexity, the TI system cannot reliably operate without frequent intervention or guidance, especially in ambiguous or non-standard intersection scenarios.
8 – Oversight independence	Level 2 AEB typically functions without human oversight, but its interventions (e.g., sudden braking) are occasionally flagged for review	Level 1 HPO is only allowed to operate under active human supervision. Regular operator monitoring (e.g., during development testing or supervised trials) ensures overtaking remains within safe bounds.	Level 0 TI operates in highly dynamic and uncertain environments. It is is still in an early development stage or under close test control, requiring continuous human monitoring during operation for safety assurance.

A2. Healthcare sector example: Clinical Conversational Assistant (CLARA)

CLARA is a cloud-based clinical conversational assistant that conducts automated natural-language voice calls with patients, replacing human interaction. It conducts task-based clinical conversations, guides patients through clinical pathways (e.g., checking for post-operative complications), and flags cases requiring human follow-up. Advanced versions of CLARA can generate call summaries and initiate and execute downstream actions. The INSYTE diagrams clarify how different versions of such an agent may exhibit different dimensions of autonomy, depending on their classification.

The first version, CLARA 1, completes a single task-based clinical conversation (e.g., checking for pre-defined complications following a specific type of routine surgery). It relies on fine-tuned text classifiers that use natural-language processing (NLP) models, such as Bidirectional encoder representations from transformers (BERT), to classify patient inputs into pre-defined 'intents'. Every conversational 'pathway' is scripted. The second version, CLARA 2, is powered by a single Large Language Model (LLM)-based agent that receives instructions via a prompt. It completes a single task-based clinical conversation in a more nuanced and adaptive way; it also provides explanations of its outputs (i.e., decision whether significant or insignificant symptoms), conversation summaries, and a recommendation on the urgency of a patient review. CLARA 3 is powered by a hybrid system. It combines the strengths of LLMs (prompted to do specific tasks like "understand the patient" or "generate a coherent next sentence") with deterministic, logic-based rules and algorithms to conduct the conversation. This design leverages LLMs for fluent speech and understanding a range of speech without extensive training data, while ensuring the verifiability of well-specified components. CLARA 4 is an 'agentic' AI system. Multiple LLM-based agents conduct the task-based clinical conversation, while individual agents are provided with 'tools' so that CLARA 4 can also order tests, book scans, make follow-up appointments, and prescribe medicine as actions arising from the conversation.

The INSYTE patterns for these four versions are presented in Figures A.3, with the supporting justifications report in Figure A.2.

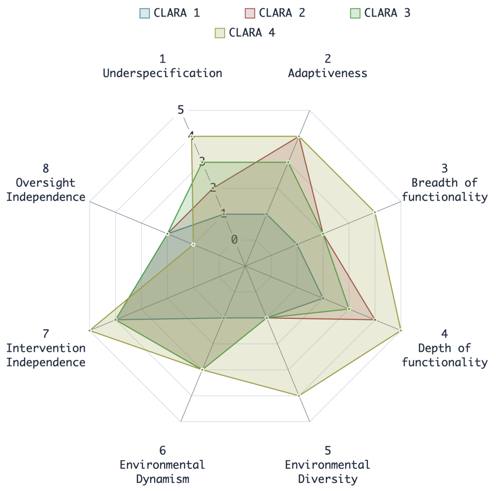


Fig. A.3. INSYTE patterns for four variants of CLARA, displayed as an overlapping diagram

Table A.2. Justification Report for the four versions of CLARA

System Justification Report				
Dimension	**CLARA 1**	**CLARA 2**	**CLARA 3**	**CLARA 4**
1 – Underspecification	**Level 1** System is primarily rule-based, but classification of patient responses is partly learned from data. Conversation flows are pre-specified.	**Level 2** Single-prompt LLM-based system has been encoded within minimal rules. Most of the clinical conversation has been learned from unlabelled data with some ‘few-shot’ examples.	**Level 3** Hybrid ’deterministic backbone’ explicitly codifies requirements into sub-goals. Beyond that, system learns to classify intents and generate responses based on a few examples.	**Level 4** Agentic LLM-based system mostly infers intended function from high-level objectives, with some learning from data and trial and error.
2 – Adaptiveness	**Level 1** System reacts to single type of uncertainty (unclear patient utterances) based on predefined adaptation tactics (repeat, restate, fallbacks).	**Level 4** System proactively reacts to multiple, foreseen types of uncertainty (e.g. patient chit-chat) using highly dynamic adaptation tactics (e.g. engaging to build rapport, but steering to key symptoms) based on a pre-defined goal.	**Level 3** System reacts to a few, foreseen uncertainty types (e.g. patient chit-chat) using both predefined (e.g. engaging once, then returning to next specified task) and dynamic (e.g. chit-chat about any appropriate topic) tactics.	**Level 4** System proactively reacts to multiple, foreseen types of uncertainty (e.g. patient chit-chat, changes in scan availabilities) using highly dynamic adaptation tactics based on a pre-defined goal.
3 – Breadth of functionality	**Level 1** System conducts a single clinical conversation, involving a few output types (e.g., classifications of symptom significance, reasoning, responses) and one decision criterion (flag for human follow up).	**Level 2** System conducts variants of a clinical conversation, involving a few output types and task types (conversation, explanations, call summaries), and a few decision criteria (flag for follow up, uncertain outcomes).	**Level 2** System conducts variants of a clinical conversation, involving a few output types and task types (conversations, explanations, call summaries), and a few decision criteria (flag for follow up, uncertain outcomes).	**Level 4** System performs variants of many different task types (conversations, explanations, call summaries, and follow-up actions, such as ordering scans and tests, and prescribing medicine).
4 – Depth of functionality	**Level 2** Task involves many routine sub-tasks (extract information, classify responses, match intents, generate and synthesise text/speech, decide symptom significance, and flag patient for follow-up).	**Level 4** Task involves simultaneous sub-tasks and interconnected decisions (e.g., next question, red flags, answers, conversation end). Complexity of steps hard to define as ‘black box’ system.	**Level 3** Task involves many sub-tasks, some of which are routine (text extraction and rule-based conversation coordination) and others complex (reasoning and coordination of multiple sub-components (agents)).	**Level 5** Task involves many very complex sub-tasks and steps (e.g., booking MRI scan involves multi-step planning and compliance with guidelines and protocols). The integration of sub-tasks requires deep processing.
5 – Environmental diversity	**Level 1** Few types of interacting element in the environment (system, patient, clinical pathway, overseeing clinician). The clinical conversation is the main source of environmental complexity.	**Level 1** Few types of interacting element in the environment (system, patient, clinical pathway, overseeing clinician). The clinical conversation is the main source of environmental complexity.	**Level 1** Few types of interacting element in the environment (system, patient, clinical pathway, overseeing clinician). The clinical conversation is the main source of environmental complexity.	**Level 4** Many types of interacting element in the environment (system, patient, clinical pathway, overseeing clinician, other systems, other healthcare workers, other clinical guidelines).
6 – Environmental dynamism	**Level 1** System released as a medical product and is constant. Conversational environments are restricted in change frequency and magnitude due to pre-scripted flows.	**Level 3** System releases may occur more frequently because it requires significantly less data to address new patient cohorts, move into new hospitals or clinics, or quickly adjust to big changes in patient behaviour.	**Level 3** System releases may occur more frequently because it requires significantly less data to address new patient cohorts, move into new hospitals or clinics, or quickly adjust to big changes in patient behaviour.	**Level 3** System can operate in new environments easily, but regulations limit rapid, frequent changes. However, the magnitude of those changes can be high (e.g., interpreting X-rays or making referrals).
7 – Intervention independence	**Level 4** Clinician identifies who receives CLARA call and follows up with patients flagged. No intervention during the call itself.	**Level 4** Clinician identifies who receives CLARA call and follows up with patients flagged. No intervention during the call itself.	**Level 4** Clinician identifies who receives CLARA call and follows up with patients flagged. No intervention during the call itself.	**Level 5** Clinician identifies who receives CLARA call and follows up with patients flagged. No intervention during the call, but clinician "signs-off" downstream tasks as required by regulation.
8 – Oversight independence	**Level 2** Varies by clinical risk of specific pathway, but regular clinician review required by regulation, especially for key decision factors (e.g., symptom classification). The system also flags unclassifiable cases.	**Level 2** Varies by clinical risk of specific pathway, but regular clinician review required by regulation, especially for key decision factors (e.g., symptom classification). The system also flags unclassifiable cases.	**Level 2** Varies by clinical risk of specific pathway, but regular clinician review required by regulation, especially for key decision factors (e.g., symptom classification). The system also flags unclassifiable cases.	**Level 1** Varies by clinical risk of specific pathway, but regular clinician review required by regulation, especially for key decision factors (e.g., symptom classification) and execution of downstream tasks. The system also flags unclassifiable cases.

A3. Infrastructure Management sector example: Solar-farm Condition Analysis and Error Recognition Robot (SCANNER)

The SCANNER robot (Solar-farm Condition ANalysis aNd Error Recognition Robot) an autonomous robot for inspecting solar farms. A maintenance service company owns a number of these four-wheeled robots. They are deployed from the company's warehouse to various solar farm sites, where their mission is to identify issues like dirty or malfunctioning solar panels, weed overgrowth, and structural defects such as corrosion. The robots have sets of onboard sensors: one set for navigation and one for inspection. Each robot has a stored map (occupancy grid) of the solar farm. The robots are equipped with software for mission planning, autonomous navigation (including obstacle avoidance), panel inspection and to issue alerts if they need assistance. When a panel defect is detected, the robots store relevant sensor and geolocation data for later analysis.

There are four versions of SCANNER. The first version, SCANNER 1, operates with a pre-defined mission. It is given an ordered set of waypoints and is tasked with planning and navigating that exact path. Human operators remotely monitor the robot's telemetry data and can intervene or teleoperate it if needed, using WiFi or the global carrier network (4G or 5G). The second version, SCANNER 2, is given an unordered set of waypoints as a connected (directionless) graph and must plan a path that visits each connection (Chinese postman problem [83]). This allows it to survey a defined section or the entire solar farm. It is also remotely monitored. The third version, SCANNER 3, has the same functionality but it operates without real-time human oversight. It has built-in safety margins and fallback procedures to handle hazardous situations, and can call for help in an emergency using WiFi or the global carrier network. The most advanced version, SCANNER 4, is only given an outline of a solar farm section. It autonomously determines the most effective way to cover that area and perform a complete inspection. It also has built-in safety margins and fallback procedures and can initiate an emergency call for assistance. The INSYTE patterns for these four versions are shown in Fig. A.4 and the justification report is in Fig. A.3.

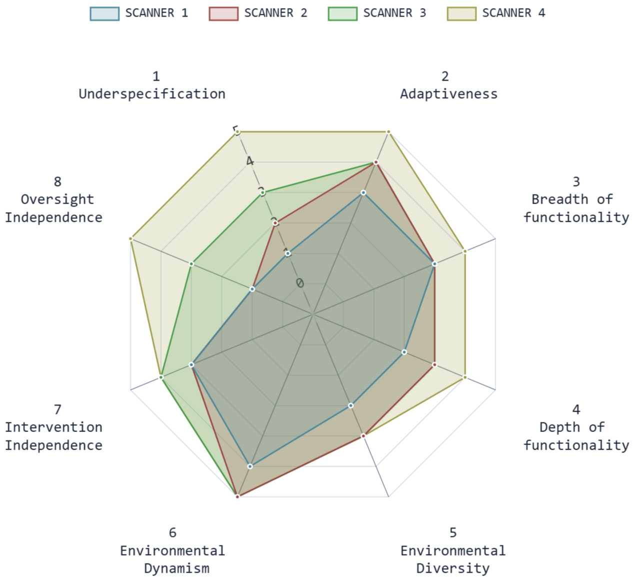


Fig. A.4. INSYTE patterns of four variants of the SCANNER robot

Table A.3. Justification Report for four versions of SCANNER

System Justification Report				
Dimension	**SCANNER 1**	**SCANNER 2**	**SCANNER 3**	**SCANNER 4**
1 – Underspecification	**Level 1** System navigates between ordered waypoints using simple path-planning (Dijkstra's algorithm) and rule-based collision avoidance algorithms. Panel inspection uses supervised classification (labelled data).	**Level 2** System navigates between unordered waypoints using a traditional algorithm (e.g., A* graph search) and ML-based collision avoidance algorithms. Panel inspection uses supervised classification (labelled data).	**Level 3** System navigates between unordered waypoints using a traditional algorithm (e.g., A* graph search) and ML-based collision avoidance algorithms. Panel inspection uses supervised classification (labelled data).	**Level 5** System given a high-level objective (survey area within specified boundary), and plans route using RL algorithms, having been trained in a partially specified environment. Panel inspection is supervised (labelled) and unsupervised (unlabelled).
2 – Adaptiveness	**Level 3** System given map of main obstacles. It sometimes reacts to multiple known types of uncertainty including new obstacles (e.g., wildlife), changing light levels or sensor faults using predefined rules.	**Level 4** System given map of main obstacles. It sometimes proactively adjusts its path between unordered waypoints to avoid new obstacles or uneven terrain. It adapts to changing light levels and sensor faults using dynamic adaptation tactics. It can update the ML models after each inspection mission.	**Level 4** System given map of main obstacles. It sometimes proactively adjusts its path between unordered waypoints to avoid new obstacles or uneven terrain. It adapts to changing light levels and sensor faults using dynamic adaptation tactics. It can update the ML models after each inspection mission.	**Level 5** System only has a partial map of obstacles to avoid. It proactively plans and adjusts its route using dynamic adaptation tactics, avoids new obstacles, and adapts to light level changes,sensor faults and changing terrain. For high environmental uncertainty, it reacts with safe fallbacks. It updates the ML models after each inspection mission.
3 – Breadth of functionality	**Level 3** System does several task types (path following between ordered waypoints, obstacle avoidance, collecting sensor data to allow human oversight, and collecting and analysing sensor data to inspect panels with variants for hotspots, dirt, cracks and corrosion variants).	**Level 3** System doess several task types (path planning, obstacle avoidance, collecting sensor data to allow human oversight, collecting and analysing sensor data for hotspots, dirt, cracks and corrosion variants and post-hoc model updating).	**Level 3** System does several task types (path planning, obstacle avoidance, detecting failures and transmitting alerts, collecting sensor data to allow post hoc human oversight, collecting and analysing sensor data for hotspots, dirt, cracks and corrosion variants, and model updating).	**Level 4** System performs several task types (overall route and adaptive path planning, obstacle avoidance, detecting and responding to sensor failures (multiple variants), transmitting alerts, collecting and analysing sensor data (multiple variants), and collecting sensor data to update the models post hoc).
4 – Depth of functionality	**Level 2** System performs simple but fast decision-making using fixed rules, ML classification and sensor inputs. Depth is limited to thresholding and simple ML-enhanced detection.	**Level 3** System performs fast decision-making with multiple sub-tasks to adapt its path to avoid collisions using sensor inputs, including predicting dynamic obstacle trajectories and optimising its trajectory. Inspection uses ML classification.	**Level 3** System performs fast decision-making with multiple sub-tasks to adapt its path to avoid collisions using sensor inputs, including predicting dynamic obstacle trajectories and optimising its trajectory. Inspection uses ML classification.	**Level 4** System proactively plans a route adapted to the current situation using deep reasoning RL and performs fast decision-making to adapt its path to avoid collisions using prediction of intent and behaviour, and uncertainty-aware planning. It uses uncertainty-aware classification for inspection.
5 – Environmental diversity	**Level 2** System encounters: panels, cables, terrain, limited flora and wildlife. It does not interact with other systems or humans.	**Level 3** System encounters: panels, cables, diverse terrain, diverse flora and wildlife. It interacts with human maintenance workers.	**Level 3** System encounters: panels, cables, diverse terrain, diverse flora and wildlife. It interacts with human maintenance workers.	**Level 3** System encounters: panels, cables, diverse terrain, diverse flora and wildlife. It interacts with human maintenance workers.
6 – Environmental dynamism	**Level 4** System deals with some quick and medium magnitude changes in the environment (wildlife and light-level changes). Other changes are slow.	**Level 5** System deals with some quick and medium/high magnitude changes in the environment (movement of humans, robots, wildlife, light changes). Other changes are slow.	**Level 5** System deals with some quick and medium/high magnitude changes in the environment (movement of humans, robots, wildlife, light changes). Other changes are slow.	**Level 5** System deals with some quick and medium/high magnitude changes in the environment (movement of humans, robots, wildlife, light changes). Other changes are slow.
7 – Intervention independence	**Level 3** Human operator intervenes rarely, to correct navigation mistakes.	**Level 3** Human operator intervenes rarely, to correct navigation mistakes.	**Level 4** Human operator only intervenes in an emergency situation, when requested by the system.	**Level 4** Human operator only intervenes in an emergency situation, when requested by the system.
8 – Oversight independence	**Level 1** Human operator regularly monitors the sensor data gathered by the robot.	**Level 1** Human operator regularly monitors the sensor data gathered by the robot.	**Level 3** Human operator reviews the sensor data after each mission, to check whether the onboard ML models need updating.	**Level 5** Human operator only downloads sensor data after an unsuccessful mission, to identify system failures and check whether ML models need updating.