Project Title **Solvers for Agents under Ignorance (SAIGE)**
Project Quality and Innovation

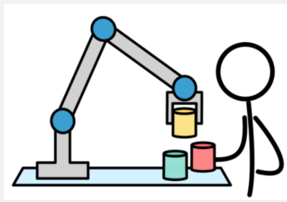
Gap in Knowledge. Automation is no longer just about reproducing laborious tasks, it increasingly involves programs making decisions, e.g., a robot interacting with a human, and algorithms on social media. Such decision-making programs, called *agents*, decide what to do next based on their experience so far in order to achieve their objectives. They excel at making decisions “under risk”[1,2,3] which is when probabilities can be calculated or sampled, aka “measurable uncertainty”[4]. However, they have trouble making good decisions “under ignorance” which is when the available data is too scarce to be aggregated by probabilities, aka, “unmeasurable uncertainty”[4]. Many critical environments, such as underground or seabed inspection, display a great degree of unmeasurable uncertainty, e.g., due to unpredictable human behaviour or unfamiliar terrain. Applying techniques for measurable uncertainty to situations of unmeasurable uncertainty is not scientifically well-founded. Thus, another approach is needed.

Aim. Develop algorithms and tools for constructing and verifying agents that make decisions under ignorance.

Novelty. In SAIGE, we will develop solvers that can handle unmeasurable uncertainty, and so advance the field of autonomous decision-making. This marks a departure from the tradition of relying on probabilistic assumptions[1,2,3], and so presents a unique benefit to problems for which data is not as abundant or even missing. SAIGE will do this by integrating Planning and Temporal-logic Synthesis with Decision Theory. This is foundational, tackling the issue of how to model and apply Decision-Theoretic principles for handling ignorance to Planning; as well as practical, creating new solvers and verifiers.

Challenges. A *solver* (aka, *planner*) is a tool that, given a formal (logical) representation of the world an agent is situated in, of its capabilities, and of its objectives, finds a high-level control *policy* that tells the agent which action to do in every situation in order to try achieve its objectives[5]. Solvers are well established for handling decisions under risk: they typically assume a probabilistic environment and return optimal (or near-optimal) policies, e.g., that maximise the expected reward[1,2,3]. Instead, we aim to handle agents under ignorance. This is challenging because it is non-trivial to provide crisp, principled, and efficiently computable formulations of policies that are effective under ignorance. The following Collaborative Manufacturing scenario[6,7] clarifies some of these challenges.

Mischief in Manufacturing. The workspace consists of a set of blocks, each of which is on the work surface or on top of another block, and a robot that can move one block at a time. Opposite the robot there is a mischievous person who will flatten block towers, snatch blocks out of the robot’s hand, and even throw blocks at the robot: while it is constrained by physical principles, the person acts according to no discernible rules. The robot may not snatch blocks back, may not touch the person, and cannot keep anything out of the person’s reach. Even if the robot only has a single goal, say to achieve a tower of all the blocks, how should it act?



This scenario is typical of decision making under ignorance: the environment (here, person) may be capricious, and so the agent (here, robot) cannot guarantee that it can achieve its goal, because of sabotage. However, in certain situations, the agent may find that it can achieve its goal, because of serendipity (e.g., the person may be idle long enough for the robot to build the tower, or the person may even provide some help to build the tower). Obviously not all actions are sensible: a robot that always undoes any of the person’s moves at building a tower is obviously inferior to one that takes advantage of opportunities presented by the person. Are there any general mathematical rules for the robot to follow? The key insight is that we can build solvers that are guaranteed to produce agent policies whose behaviours conform to given *decision-making rules*. These rules are founded in *Decision Theory*[8], the mathematical theory of decision making that makes sense of how individuals make, or should make, decisions under ignorance. Classic rules include: “maximin”, “dominance” (aka “admissibility”), “minimax regret”, “maximax”, “the optimism-pessimism criterion”, the “principle of insufficient reason”, and their combinations. Such rules can eliminate certain policies that are not sensible, and suggest others that are. For instance, in the *Mischief* scenario, the robot policy that always undoes any of the person’s moves at building a tower is *inadmissible*[9], since it is dominated by the policy that simply remains idle, which itself is dominated by a policy that, intuitively, tries to build the tower.

Methods. SAIGE solvers will be built using two closely-related[10,11] declarative disciplines for sequential decision making, known for their compositionality, reuse, transparency, and formal guarantees: (Nondeterministic) Planning[5,12] and Temporal-Logic Synthesis[13], hereafter called *Planning*. Solvers go hand in hand with *verifiers* that, given a policy — whether constructed by a solver, by hand, or by other automated methods such as learning — determine whether or not the policy obeys a given Decision-Theoretic rule.

Research Problem 1 — Establish foundations of SAIGE. Decision Theory teaches us that a decision-maker has preferences on outcomes that are, in general, *pre-ordered*. This allows an agent to also be indifferent or undecided about their preference between any two outcomes; in planning under risk, preferences are represented by an interval-based utility function and so are completely ordered, i.e., the agent is never undecided about their preference between two outcomes (!). For each Decision-Theoretic rule, we will establish if it admits a history-based characterisation (as has been done for the dominance principle[14,9] and can easily be adapted to minimax and maximax, but has not been settled, e.g., for minimax regret), and use this to determine when such policies exist. We will specify preferences using preference-based logics[15] instantiated with linear temporal logic (LTL) and its finite-trace variants promoted in AI by the PIs[16,17,18,19,20,21,22], and we will devise optimal algorithms for constructing and verifying policies.

Research Problem 2 — Build and test SAIGE solvers and verifiers. We will build solvers and verifiers using the *method of compilation*, i.e., we will show how to efficiently encode a given planning problem and Decision-Theoretic solution concept into a logic engine, such as a satisfiability solver (SAT). This will allow us to handle quite large state spaces[23]. Moreover, there is evidence that SAT-based compilation can be more robust (for standard solution concepts) on domains in which an "optimistic" assumption for dealing with ignorance is counterproductive[23]. We will empirically evaluate SAIGE on domains employed in Planning competitions for evaluating nondeterministic planners[24]. To do this we will apply performance measures such as coverage, time, memory, and policy size, and metrics for plan quality[25].

Team. CI Rubin's expertise is in Computational Logic for AI; PI Vardi (Knuth prize) has made seminal contributions and tech transfer in verification, planning, synthesis, and multi-agent systems; PI De Giacomo's (ERC advanced) is in Knowledge Representation and Reasoning. CI Rubin has collaborated with each PI on topics in this proposal: verification[26,27] and decision-theoretic rules for planning (see Pilot Study).

Previous work and Pilot Study. Certain Decision-Theoretic rules have already been considered in Planning: the *dominance principle* rules out clearly suboptimal solutions, and scattered studies explore the viability of *regret minimisation* to encourage agent collaboration with humans[28] and agent exploration of potentially huge reward[29]. The dominance principle featured in our *Pilot study*[30,31,9,32,33,34,35]. Intuitively, a policy is *dominated* (aka, *inadmissible*) if there is another policy that does at least as well in all environments and strictly better in some environments. Generally, *non-dominated* (aka, *admissible*) policies guarantee that if the environment is not displaying worst-case behaviour during execution (e.g., if the environment makes a mistake at a crucial time, or if it is agnostic to the agent), then the agent will achieve its goal in that execution. There are typically many admissible strategies, and further guidance is needed to select one, i.e., by using other Decision-Theoretic rules such as minimax regret[36]. The Pilot Study focused on a single Boolean-valued goal (typically expressed in LTL). While previous work in Planning has considered richer preferences on outcomes[15], it typically does not consider both nondeterministic environments and non-complete preferences, the history-based characterisation from the Pilot Study that guarantees the existence of admissible strategies is missing, amd it does not seamlessly integrate other Decision-Theoretic principles.

Scholarly Benefits. The ambition of SAIGE is to affect the dominant paradigm in Planning to incorporate Decision-Theoretic criteria for dealing with agent ignorance, rather than relying on probabilistic assumptions that are not robust under ignorance[37]. This will open up new challenges in related fields such as Control and Robotics.

Social and industrial benefits. Autonomous Systems is a *Critical Technology Field in the National Interest*, and indeed the impact of this project will be felt wherever Planning is adopted in challenging environments[38], e.g., underground and marine robotics[39,40]. The project will form the basis of future work that addresses the *National Science and Research Priority* of protecting and restoring Australia's environment through improved planning; transport via autonomous vehicles; safe mining automation in remote sites; and agriculture outdoors in all weather conditions.

References.

[1] C. Boutilier, T. L. Dean, and S. Hanks. In: *J. Artif. Intell. Res.* (1999).**[2]** S. Thiébaux, C. Gretton, J. K. Slaney, D. Price, and F. Kabanza. In: *J. Artif. Intell. Res.* 25 (2006), pp. 17–74.**[3]** A. Camacho, R. T. Icarte, T. Q. Klassen, R. A. Valenzano, and S. A. McIlraith. In: *IJCAI*. 2019.**[4]** F. H. Knight. Houghton Mifflin, 1921.**[5]** H. Geffner and B. Bonet. Springer, 2013.**[6]** M. Schoppers. In: *IJCAI*. 1987.**[7]** K. He, A. M. Wells, L. E. Kavraki, and **M. Vardi**. In: *ICRA*. 2019.**[8]** M. D. Resnik. U of Minnesota Press, 1987.**[9]** B. Aminof, **G. De Giacomo**, and **S. Rubin**. In: *IJCAI*. 2021.**[10]** N. Robinson, C. Gretton, D. N. Pham, and A. Sattar. In: *ICAPS*. 2009.**[11]** A. Camacho, M. Bienvenu, and S. A. McIlraith. In: *ICAPS*. 2019.**[12]** P. Haslum, N. Lipovetzky, D. Magazzeni, and C. Muise. Synthesis Lectures on AI and ML. Springer, 2019.**[13]** A. Pnueli and R. Rosner. In: *POPL*. 1989.**[14]** D. Berwanger. In: *STACS*. 2007.**[15]** J. A. Baier and S. A. McIlraith. In: *AI Mag.* (2008).**[16]** C. Gretton. In: *PRICAI*. 2014.**[17]** **G. De Giacomo** and **M. Vardi**. In: *IJCAI*. 2013.**[18]** **G. De Giacomo** and **M. Vardi**. In: *IJCAI*. 2015.**[19]** **G. De Giacomo** and **M. Vardi**. In: *IJCAI*. 2016.**[20]** **G. De Giacomo**, A. Di Stasio, F. Fuggitti, and **S. Rubin**. In: *IJCAI*. 2020.**[21]** L. Bonassi, **G. De Giacomo**, M. Favorito, F. Fuggitti, A. E. Gerevini, and E. Scala. In: *ICAPS*. 2023.**[22]** S. Almagor, U. Boker, and O. Kupferman. In: *J. ACM* (2016).**[23]** T. Geffner and H. Geffner. In: *ICAPS*. 2018.**[24]** C. Muise, S. A. McIlraith, and J. C. Beck. In: *AAAI*. 2024.**[25]** A. Gerevini, P. Haslum, D. Long, A. Saetti, and Y. Dimopoulos. In: *AIJ* (2009).**[26]** D. Bustan, **S. Rubin**, and **M. Vardi**. In: *CAV*. 2004.**[27]** B. Aminof, L. Cooper, **S. Rubin**, **M. Vardi**, and F. Zuleger. In: *KR*. 2024.**[28]** K. Muvvala, P. Amorese, and M. Lahijanian. In: *ICRA*. 2022.**[29]** J. Zhao, K. Zhu, S. Li, and X. Yin. In: *IFAC* (2023).**[30]** B. Aminof, **G. De Giacomo**, A. Lomuscio, A. Murano, and **S. Rubin**. In: *IJCAI*. 2020.**[31]** B. Aminof, **G. De Giacomo**, A. Lomuscio, A. Murano, and **S. Rubin**. In: *KR*. 2021.**[32]** B. Aminof, **G. De Giacomo**, **S. Rubin**, and F. Zuleger. In: *IJCAI*. 2022.**[33]** B. Aminof, **G. De Giacomo**, and **S. Rubin**. In: *AAAI*. 2023.**[34]** B. Aminof, **G. De Giacomo**, **S. Rubin**, and F. Zuleger. In: *LICS*. 2023.**[35]** B. Aminof, **G. De Giacomo**, G. Parretti, and **S. Rubin**. In: *IJCAI*. 2024.**[36]** M. Cadilhac, G. A. Pérez, and M. van den Bogaard. In: *FOSSACS*. 2019.**[37]** A. M. Wells, Z. K. Kingston, M. Lahijanian, L. E. Kavraki, and **M. Vardi**. In: *ICRA*. 2021.**[38]** M. Hanheide, M. Göbelbecker, et al. In: *Artif. Intell.* (2017).**[39]** A. Akbari, P. S. Chhabra, U. Bhandari, and S. Bernardini. In: *IROS*. 2020.**[40]** A. Akbari and S. Bernardini. In: *IEEE Robotics Autom. Lett.* (2021).