npj | wireless technology

Article

https://doi.org/10.1038/s44459-025-00006-x

Enhancing intelligence in multi-agent systems with edge-assisted causal knowledge aggregation

Check for updates

Muhammad Waqas Nawaz[1] ✉, Muhammad Mahtab Alam[2], Rafiq Swash[3], Qammer H. Abbasi[1], Muhammad Ali Imran[1] & Olaoluwa Popoola[1]

Dynamic and uncertain environments pose major challenges for multi-agent autonomous systems, particularly in achieving robust simultaneous localization and mapping (SLAM) and efficient knowledge sharing across robots. Conventional data-driven methods often overlook underlying causal structures, resulting in spurious correlations and limited generalization. To address this, we present CASK—an edge-assisted causal knowledge aggregation framework that fuses structured causal inference with data-driven learning to improve adaptive decision-making. A key feature is a time-based normalization mechanism that ensures mapping consistency across varying operational speeds, enabling speed-independent transfer of spatial knowledge between heterogeneous agents. We validate CASK through simulations and real-world experiments using autonomous ground vehicles, a class of mobile robots. Results show substantial gains over state-of-the-art methods: up to 20% higher success at low speeds, 40% at high speeds, 50% lower trajectory deviation, and 45% fewer re-planning steps. These findings demonstrate how causal inference combined with mobile edge computing enables scalable, reliable, and generalizable autonomy in multi-agent systems.

SLAM problem is widely considered one of the most crucial challenges in developing fully autonomous ground vehicles[1]. The development of autonomy has accelerated over the past 20 years due to advancements in artificial intelligence, machine learning (ML), and sensor technology[2]. A robust SLAM solution would unlock new applications in multi-agent robotics, making it crucial for real-world deployment. Data-driven approaches have proven successful, as evidenced by new vehicle models with intelligent driving assistance and commercial L4 robotics operations[3–5]. However, efficiently computing, storing, and exploiting SLAM posterior distributions at scale remains challenging. Rising communication and storage costs limit purely centralized cloud solutions for large fleets of Internet of Things (IoT) devices[6].

Mobile edge computing (MEC) helps overcome these limitations by providing high data rates, low latency, and scalable connectivity beyond 5G. As 6G networks are standardized and deployed, they will further strengthen on-device intelligence and real-time decision-making in autonomous systems[7]. These advances are accelerating the autonomous-robotics ecosystem by enabling tighter perception–planning loops and resilient coordination[8]. Autonomous systems used in vacuuming, search and rescue, and transportation depend on accurate environment maps for effective navigation, often without GPS[1]. Generalization across environments is difficult: transfer learning[9], multi-task learning[10], and meta-learning[11] offer reuse of prior knowledge but typically assume that the target distribution is close to the source. When this assumption fails, models can learn spurious correlations and degrade under distribution shift. We therefore focus on principled mechanisms that preserve the structure needed for cross-environment generalization rather than emphasizing broad notions such as artificial general intelligence(AGI). Although these learning paradigms are often discussed together, their distinct algorithms can lead to confusion—particularly when applied in varying environments. This complexity underscores the need to address task composition and hierarchy[12], as developing models that reliably predict outcomes in unseen environments remains a central challenge.

Counterfactual reasoning is essential for understanding how systems would behave under hypothetical interventions. Yet formalizing counterfactuals is challenging within standard logic, algebra, or probability, as it requires a language capable of separating invariant mechanisms from contingent beliefs[13]. In scientific practice, extrapolating findings across contexts is common, but purely statistical models remain reliable only when underlying conditions are unchanged; once mechanisms shift, predictive

[1]James Watt School of Engineering, University of Glasgow, Glasgow, Scotland, UK. [2]Thomas Johann Seebeck Department of Electronics, Tallinn University of Technology, Tallinn, Estonia. [3]College of Engineering, Design and Physical Sciences, Brunel University of London, London, England, UK.
✉e-mail: m.nawaz.1@research.gla.ac.uk

performance deteriorates. This limitation underscores the need for a robust framework to generalize knowledge across diverse environments. When the target environment diverges substantially from the source, traditional transfer strategies—such as online/offline learning, weight sharing[14], or meta-learning—often underperform because they fail to encode the causal structure governing environmental variability. Without capturing this structure, transferred knowledge risks introducing bias and spurious correlations, ultimately reducing adaptability and decision accuracy in real-world deployments. Causal inference offers the theoretical foundation to address this gap by distinguishing statistical correlations from structural relationships that remain invariant across domains[15]. Using causal diagrams and selection variables, it is possible to explicitly model how mechanisms differ between environments[16], while do-calculus provides the formal conditions for transporting knowledge reliably[17]. Although these tools have seen application in embodied autonomy[18], they have largely been limited to specific prediction or explanation tasks—such as motion intent estimation or lane-change detection—rather than being integrated into SLAM-based knowledge aggregation pipelines[19,20].

Our approach builds on Pearl's structural causal models and directed acyclic graphs (DAGs)[13]. Since causal relationships can vary across domains, we begin by defining the underlying structure of interest, with an emphasis on physical influence—where manipulating causes leads to observable changes in effects. To operationalize this, we focus on experimental methods that enable local modelers to capture interactions among endogenous states, quantify the impact of exogenous inputs, and embed invariant causal knowledge into their models. Building upon our earlier work[7,14,21], we extend this direction by identifying three key limitations that prior approaches have not fully addressed: (i) the lack of a systematic procedure to identify and preserve invariant cause–effect relations across environments; (ii) the absence of a normalization mechanism to ensure mapping consistency when agents operate at different velocities; and (iii) the limited validation of scalability across heterogeneous robotic platforms under edge-assisted knowledge sharing.

To overcome these challenges, we propose CASK—an edge-assisted causal structured knowledge aggregation framework—designed to identify invariant causal structures in dynamic environments for effective knowledge integration. To the best of our knowledge, no existing algorithm targets this problem. CASK integrates causal selection diagrams into the MEC pipeline to share only the relations likely to transport across environments and applies a time-based normalization to stabilize occupancy representations for speed-consistent mapping. This design enhances scalability in multi-agent systems, improves robustness, and boosts system intelligence. By integrating environmental perception with edge causal inference (ECI), CASK enables do-conditional distributions that support adaptive decision-making and robust performance across varied operating contexts.

In summary, the contributions of this paper are:

- *Causal knowledge framework*: A theoretical model for invariant mapping of observational variables within a parent–child cause–effect structure, incorporating an on-device modeler to form a meta-observational knowledge layer.
- *Edge selection diagrams*: Integration of observational knowledge into causal selection diagrams at the edge, where the Edge Knowledge Modeler (EKM) acts as an intervention layer.
- *Time-based normalization*: A method to stabilize occupancy representations for speed-consistent mapping, improving reliability of shared knowledge in dynamic settings.

The proposed framework is validated through simulation and offline experiments on a UGV and a mobile robot, showing improved generalization, reduced relearning, and efficient resource use. Performance is assessed with three metrics: trajectory deviation Δ_{traj} (difference between planned and executed paths), success rate Γ (percentage of successful navigations), and re-planning steps ψ (number of path adjustments due to environmental changes). Traditionally, selection diagrams assess data transportability in offline analysis. With our Scotland 5G Centre[22] acting as MEC, we enable real-time adaptation and dynamic knowledge aggregation with negligible latency.

Generalization has played a pivotal role across various machine learning (ML) paradigms, often under the umbrella of transfer learning (TL), with the common objective of improving learning efficiency and adaptability[23]. In supervised learning, deep transfer learning (DTL) has been employed to address the challenge of insufficient training data for target tasks through domain-based and feature-space-based methods[24]. However, DTL methods face significant limitations for general, autonomous AI systems in multi-agent because: (1) They rely on human intuition to select source and target domains. (2) They struggle to address the unknown changes in data distributions caused by agent-environment interactions. In contrast, reinforcement learning (RL)[25] provides a framework in which agents learn by interacting with the environment, taking actions, altering states, and receiving rewards. Transfer learning in RL (RTL) involves reusing knowledge—such as policies, reward functions, or value functions, learned from one task to perform another or similar task. However, RTL methods are constrained by reliance on human intuition and goal-entangled, task-specific knowledge. These limitations extend to deep reinforcement learning (DRL), particularly when target tasks change unpredictably[26,27].

The concept of *environmental covariate shift* refers to situations where the probability distribution of certain environment related variables changes across domains, while the conditional relationships between variables representing causal mechanisms are assumed to remain invariant[28–30]. This assumption can break down in scenarios where static layouts differ or dynamic environmental factors alter these distributions, thereby affecting the conditions under which heterogeneous multi-agents operate. Rojas-Carulla et al.[31] demonstrated that optimal predictors for transfer domains can be constructed using only the causal parents of a variable, aligning with Pearl's assertion that causal relations represent invariant physical mechanisms[32,33]. Unlike traditional machine learning, causal reasoning offers several advantages, including transparency, testability, the ability to predict through interventions, and robustness to missing data[34].

Causal inference, which involves deducing cause-effect relationships among variables[13], often relies on observational data in causal discovery methods, but this approach is limited, especially for agents learning interactively. In contrast, our work operates within the interventional regime, where variables can be manipulated to observe their effects on the system. In summary, SLAM generalization across environments remains challenging, with traditional transfer methods often failing under distribution shift. Counterfactual reasoning and causal inference offer a principled way to capture invariant structures, yet their use in SLAM-based knowledge aggregation is limited. Our proposed CASK framework addresses this gap through MEC-enabled causal selection diagrams and time-based normalization, improving scalability, robustness, and adaptability in multi-agent systems.

Results

This hierarchical knowledge-sharing framework is validated in two distinct environments, $E(\pi_1)$ and $E(\pi_2)$, using unmanned ground vehicles (UGVs): the Jackal Clearpath J-100 equipped with a VLP-32C Velodyne LiDAR for r_1, and the TurtleBot4 equipped with an RP LiDAR-A1 for r_2. Both robots ran the ROS SLAM Toolbox (https://github.com/SteveMacenski/slam_toolbox) for 2D occupancy grid mapping and localization, integrating LiDAR and odometry inputs in real time. Offline and real-time tests were conducted to evaluate the framework's ability to generalize across environments, focusing on knowledge transfer efficiency and navigation performance. In both environments, the robots navigated a 2D closed-loop path, with environmental data—either occupancy grids, trajectories, or both—stored at the edge for subsequent knowledge aggregation, as shown in Fig. 1.

Each map cell (c_i) was assigned a probability value: static objects ($\mathcal{S}_t = 1$ if occupied, else 0) and dynamic objects ($D_t = 1$ if occupied, else 0), based on LiDAR and odometry readings. An occupancy threshold of 0.65, as in ref. 35, determined whether a cell was occupied or free. Further, an edge

Fig. 1 | Initial robot positions for two experiments conducted in James Watt South Building, University of Glasgow, using a UGV and a mobile robot.

server facilitated communication between simulations, enabling message exchange with minimal latency. The goal was to extract and transfer causally learned models across environments using our proposed algorithm, enhancing system perception and adaptability. Offline simulations ensured system compatibility. Data from each environment was uploaded to the edge modeler, which identified perceptual variables and stored them as a S^*. Upon a transition request, the edge accessed the selection node (S^r) for the target environment and shared it with the requesting node. The target node integrated this perceptual awareness using Eq. (20), enabling adaptive navigation in dynamic environments.

Evaluation

The performance of both systems is evaluated by comparing the acquired map knowledge, shaped by the causal structure model (20), using maximum mean discrepancy (20). To assess the impact of environmental knowledge on navigation efficiency, evaluation begins in the training environment E_1 with r_1, and the learned causal knowledge is subsequently tested on r_2 in unseen environments E_1 and vice versa. Figure 2 presents a comparative analysis of system performance across two environments. In Fig. 2a, r_1, trained in E_1, uploads collected data to the edge. The edge modeler identifies and stores invariant environmental mappings in a knowledge-sharing matrix, ensuring seamless adaptation across environments, as formulated in Eq. (22). The controller (C_q) facilitates inter-environment communication, ensuring seamless transitions. Upon receiving a command, the edge modeler processes the target environment query and transmits structured knowledge to r_2, enabling it to execute the desired trajectory (Fig. 2c). The comparative analysis between the ground-truth and causally inferred trajectory (Fig. 2d) highlights the system's ability to generalize in an unfamiliar environment with minimal deviation, enabled by edge-assisted knowledge aggregation. Notably, r_2 bypasses local path construction, leveraging precomputed observation mappings from the edge. This structured transfer of knowledge allows r_2 to dynamically adapt its trajectory, ensuring efficient and informed navigation.

Furthermore, the system continuously updates its model to account for any environmental changes, ensuring adaptive and precise path planning based on real-time observations. The above goal is achieved by continuously updating the occupancy grid map for both static and dynamic objects during navigation. The occupancy grid map $\mathcal{G}_{\text{edge}}$ is stored at the edge, where each cell c_i has an occupancy probability $p(c_i)$. The edge system provides the initial environment model, and as the trajectory progresses, new observations are incorporated into the edge system, ensuring up-to-date information for future navigation. Figure 3, compares the results of the two methods (1) without knowledge aggregation and (2) proposed with edge-assisted knowledge aggregation under max-speed conditions of two autonomous systems, respectively. The results show a trajectory performance of the causal learned knowledge versus a true model trained in E_1, E_2. In Fig. 3a, when learned knowledge of r_1 is applied to a new r_2 with a novel edge aggregated input mapping, the zero-knowledge learned robot r_2 shows a negligible prediction error and deviation from the true trajectory due to its ability to generalize provided occupancy map effectively into the new environment and vice versa. In contrast Fig. 3b, the causal learned model maintains low prediction errors, closely following the optimal path without building the occupancy map.

Performance under varying speeds

This section evaluates the proposed system's real-time capabilities under low-speed and high-speed operation, addressing the need for robust navigation performance in dynamic environments as outlined in the problem definition. To this end, we systematically test the proposed methods in both low-speed and high-speed scenarios, conducting an in-depth analysis of their planning results and performance differences through simulation experiments.

The study examines three navigation strategies: (1) a static occupancy map using ROS, (2) time-based normalization for speed-consistent mapping, and (3) edge-assisted causal knowledge aggregation. The static map remains unchanged despite speed variations, while (2) dynamically adjusts occupancy probabilities to ensure consistency: $p_{\text{time}}(c_i) = \alpha \mathcal{S}_t + \beta D_t$. where α represents the weight assigned to spatial information $\mathcal{S}_t$, capturing the occupancy state of the environment over time, and β represents the weight assigned to the dynamic factor D_t, accounting for temporal changes such as moving obstacles or velocity variations. This ensures real-time responsiveness by adapting occupancy probabilities to speed variations. By

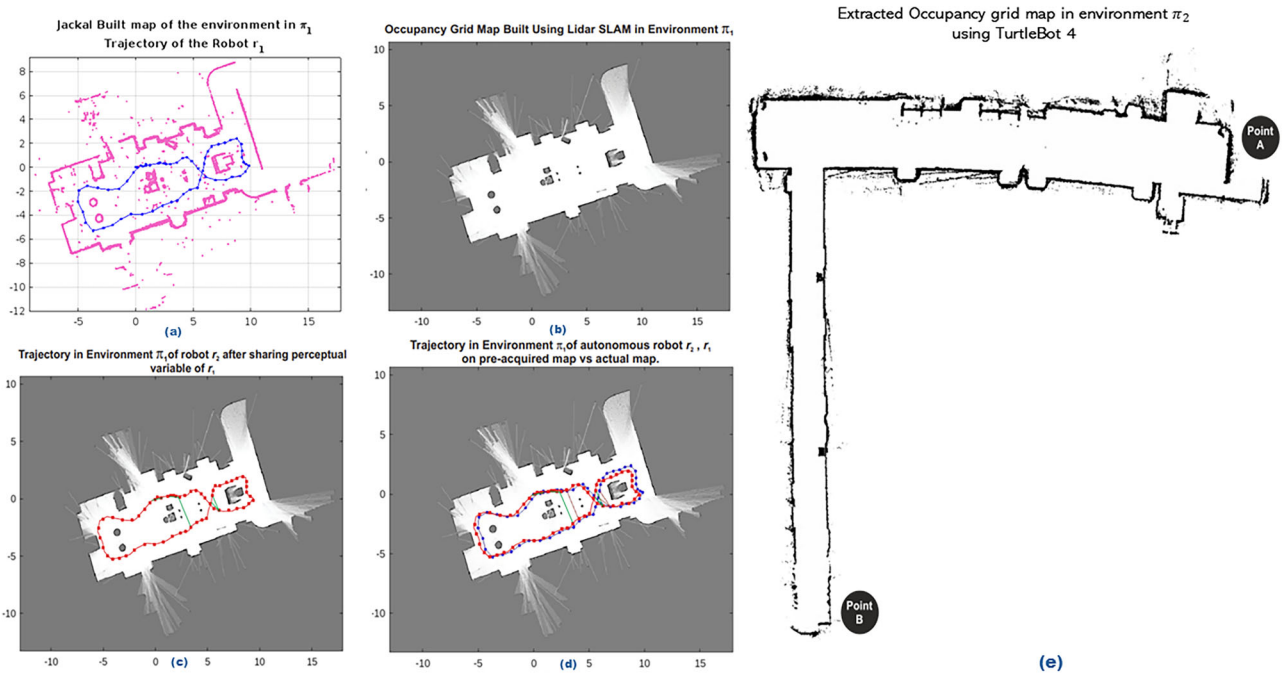

Fig. 2 | A comparison of robot trajectories and mapping strategies on an *X*–*Y* surface. a Jackal robot trajectory in blue, **b** knowledge separation at the edge showing environmental data sharing, **c** shared knowledge with TurtleBot 4 visualised in red, **d** compared trajectories in closed-loop path planning, and **e** extracted occupancy map in environment π_2.

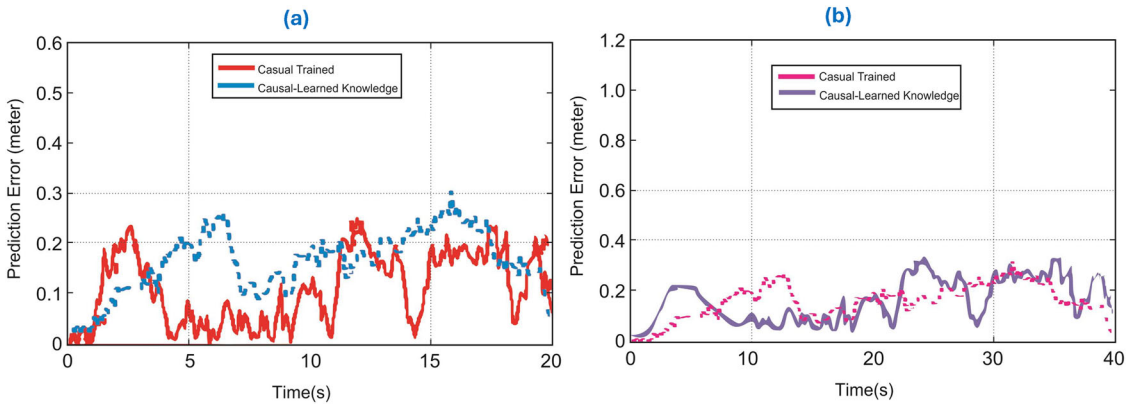

Fig. 3 | A comparison between the squared prediction error MMD. a Causal Trained r_1 vs Causal learned r_2 in unseen environment π_1. **b** Causal Trained r_2 vs Causal learned r_1 in unseen environment π_2.

stabilizing occupancy over time, (2) enhances (3), providing reliable spatial-temporal data for more accurate and adaptive knowledge aggregation across environments.

Given that different robots operate at varying speeds, the proposed time-based map normalization ensures that shared knowledge remains relevant across different velocity profiles. If an occupancy grid is initially collected at speed v and another robot navigates at speed v', the occupancy probability updates as $p'(c_i|\mathcal{S}, D_t, v') = \alpha\mathcal{S}_t + \beta D_t$, ensuring consistency in navigation. The edge-assisted knowledge-sharing mechanism further extends this approach by integrating real-time sensor updates into the occupancy grid, using the formulation: $p_{\text{edge}}(c_i) = \alpha\mathcal{S}_t + \beta D_t + I_n[\text{sensorupdate}]$ where I, represents the influence of real-time knowledge updates. This optimization improves accuracy by dynamically refining occupancy data based on real-time observations, ensuring a more robust and adaptive navigation strategy.

To validate the approach, real-time data was collected from a UGV and a mobile robot operating at different speeds, providing a diverse dataset for training the simulation model. This real-world dataset was then used to test the proposed method in simulation, analyzing how the occupancy grid adapts to different speed factors. As described in our experimental design, static objects are represented as $\mathcal{S}_t = 1$ if occupied, otherwise 0, while dynamic objects are represented as $D_t = 1$ if occupied, otherwise 0. This approach was implemented in our real-time experiment to ensure accurate environmental perception. However, to assess the robustness of our model, we introduced occupancy label-swap augmentation, where occupancy and dynamics labels were randomly swapped from 1 to 0 and vice-versa. This modification simulates environmental transitions by altering spatial distributions, ensuring that while $\pi^*(\mathcal{S}_t)$ and $\pi^*(D_t)$ remain approximately consistent across different environments, the conditional probability $\pi^*(D_t|\mathcal{S}_t)$ varies. This augmentation process evaluates the adaptability of our model in dynamically changing environments. Figure 4 shows the speed variations over time for mobile and UGV autonomous systems, compared to the target trajectory and normalized causal simulated approach. The normalized causal simulated speed closely follows the target trajectory, with

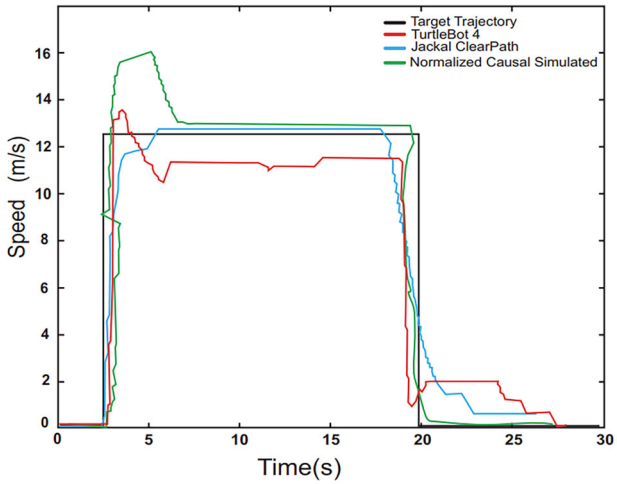

Fig. 4 | Simulated Control Experiment of Speed and Time parameters.

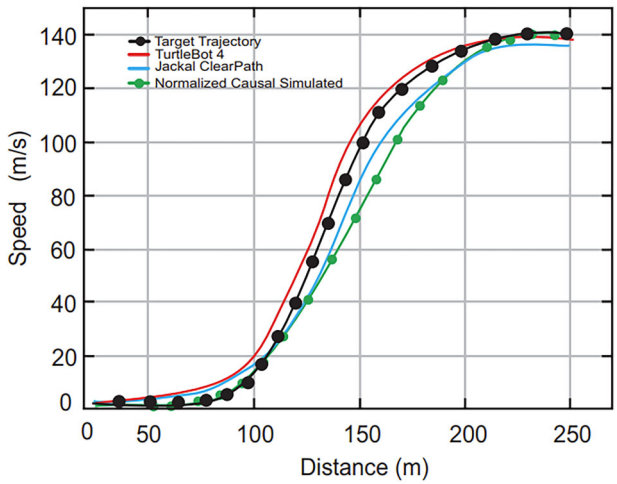

Fig. 5 | Trajectory tracking across varying speeds and distances. CASK (green) maintains accuracy in unseen environments without rebuilding maps from scratch.

Jackal ClearPath adapting faster than TurtleBot4. The data also includes maximum speed data collected for both platforms, extending to higher speeds in the simulation. The causal knowledge-sharing model smooths out fluctuations, enhancing speed regulation and improving trajectory-following accuracy.

Figure 5 shows that, through causal knowledge aggregation from autonomous systems operating at different speeds and distances, the proposed CASK model (green) maintains a high level of path-following accuracy even in previously unseen environments. By leveraging speed-consistent, time-normalized occupancy mapping and edge-assisted knowledge sharing, the model preserves navigation intelligence across varying conditions, enabling robust trajectory tracking where traditional SLAM approaches tend to drift. In contrast, conventional methods need to perceive and reconstruct the environment from the beginning in each deployment, which is computationally inefficient.

Table 1 shows that static occupancy maps cause greater trajectory deviations at high speeds due to their limited adaptability. Time-based normalization improves adaptability but lacks real-time updates, limiting its performance. In contrast, the edge-assisted approach achieves the lowest Δtraj, the highest Γ, and the lowest ψ_n, validating the effectiveness of integrating causal knowledge aggregation with speed-aware normalization. The bold values in the table highlight these best-performing metrics—lowest trajectory deviation, highest success rate, and fewest re-planning steps—demonstrating the superiority of the edge-assisted method across all speed

Table 1 | Performance comparison of different knowledge-sharing approaches under forward varying speed conditions

Approach	Speed (m/s)	Trajectory deviation (m)		Success rate (%)		Re-planning steps	
		Low	High	Low	High	Low	High
Static Occupancy Map	10	0.42	0.78	75	52	5.2	9.8
	40	0.55	1.02	68	45	6.0	11.2
Time-Based Normalization	10	0.35	0.62	82	67	4.1	7.5
	40	0.40	0.75	78	60	4.5	8.1
Edge-Assisted Knowledge Aggregation	10	**0.21**	**0.38**	**95**	**89**	**2.3**	**3.9**
	40	**0.28**	**0.50**	**91**	**85**	**2.6**	**4.2**

conditions. These findings confirm that the CASK approach significantly enhances system intelligence, navigation accuracy, and robustness in dynamic environments. Compared to traditional methods, CASK leverages MEC for real-time adaptation and efficient knowledge transfer, addressing limitations in prior works[28–30] that rely on static models and minimal assumptions.

Discussion

The proposed CASK framework demonstrates strong effectiveness in enabling knowledge transfer and robust navigation across heterogeneous robots and environments. By combining causal inference with edge-assisted knowledge aggregation, the system reduces trajectory deviation and re-planning steps while achieving consistently higher success rates compared to static mapping and time-based normalization. Importantly, knowledge extracted in one environment and deployed in another maintains predictive accuracy under varying robot speeds and dynamic conditions, highlighting the adaptability of the framework. This performance is driven by the integration of speed-consistent occupancy normalization and edge-level causal modeling, ensuring that shared knowledge remains valid across diverse platforms without the need for computationally expensive relearning.

The experiments, conducted on both UGV and mobile robot platforms, confirm that the framework generalizes beyond simulation, with CASK (green trajectories) closely tracking ground truth in previously unseen environments, whereas conventional SLAM-based approaches often drift or require complete map reconstruction. Moreover, by embedding a causal modeler between odometry acquisition and SLAM, the system performs causal-relational analysis before map generation, further enhancing robustness. While the validation is limited to indoor scenarios and moderate-scale environments, results consistently demonstrate improved trajectory prediction, reduced reliance on local path planning, and enhanced generalization. Collectively, these findings establish CASK as an effective mechanism for structured knowledge aggregation, improving adaptability, robustness, and efficiency in dynamic real-world deployments.

Methods

Notations and preliminaries

Figure 6 illustrates the overall structure of the proposed CASK framework. Each environment $E_r \in \pi_k$ (for $k = 1, \ldots, n$) is equipped with a local modeler $\mathcal{M}_{E_r}$, which captures *endogenous variables* $V = \{v_1, \ldots, v_{|V|}\}$ (e.g., robot pose, velocity, occupancy) and *exogenous variables* $U = \{u_1, \ldots, u_{|U|}\}$ (e.g., odometry u_t, perception z_t, weather). Relations $\mathbb{R}$ and structural functions F describe their evolution, with variables denoted in uppercase (e.g., $Z, X, \mathbb{C}_k, Y$) and values in lowercase. A causal state-update is $x(t) = f(x(0), u(0{:}\,t), v(0{:}\,t))$ and incrementally $x(t+1) = \tilde{f}(x(t), u(t), v(t))$.

The local models are transmitted to the edge modeler $\mathcal{M}_{\text{edge}}$, which integrates observational and interventional data via do-calculus and selection diagrams. The interventional query layer then computes transportable causal quantities that are used by the controller $\mathcal{C}_q$ for adaptive decision-making. This layered architecture enables heterogeneous agents to share

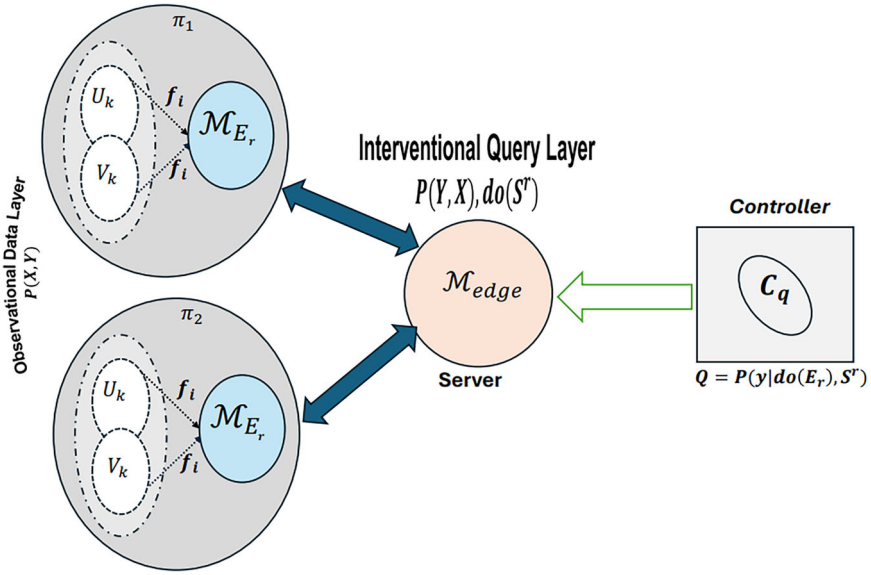

Fig. 6 | A Core CASK architecture for a learned causal environment.

causal knowledge efficiently, ensuring scalability and robustness in dynamic environments.The following definitions formalize the components of this architecture.

Structural Causal Model (SCM): An SCM is defined as $M = \langle U, V, F, P(U)\rangle$, where:

- U: exogenous variables,
- V: endogenous variables determined by $U \cup V$,
- F: structural functions f_i: $\mathrm{Dom}(U \cup \mathrm{PA}_i) \rightarrow V_i$ with PA_i as parents of V_i,
- $P(U)$: probability distribution over U.

Causal Diagram: A causal diagram G is a DAG where:

- Solid nodes: endogenous variables,
- Hollow nodes: exogenous variables,
- Directed edge $V_c \rightarrow V_e$: direct influence,
- Bidirected edge $V_a \leftrightarrow V_b$: shared exogenous parent.*Example:* In SLAM, robot pose is a solid node, while rain affecting LiDAR is a hollow node.

Interventions and Do-Calculus: An intervention $do(X = x)$ represents actively setting the variable X to the value x, replacing its original assignment in the structural equations F. In graphical terms, this means removing all incoming edges into X in the causal graph. We denote the modified graph as $G_{\overline{X}}$, where the overline "–" indicates an intervention that cuts incoming edges to the variable.

Do-calculus: provides three transformation rules for manipulating interventional and observational probabilities under certain conditional independence constraints in the modified graph:

- Rule 1 (Insertion/Deletion of Observations): $(y|do(x), z, w) = P(y|do(x), w)$ if Y is independent of Z given X and W in the intervened graph $G_{\overline{X}}$. *Meaning:* If Z provides no extra information about Y once X and W are known after intervening on X, then Z can be removed from the conditioning set.
- Rule 2 (Action/Observation Exchange): $P(y|do(x), do(z), w) = P(y|do(x), z, w)$ if Y is independent of Z given X and W in $G_{\overline{X}\underline{Z}}$. Here, $\underline{Z}$ indicates that Z is treated as observed rather than intervened upon. *Meaning:* In this case, an intervention on Z can be replaced by simply conditioning on its observed value.
- Rule 3 (Insertion/Deletion of Actions): $P(y|do(x), do(z), w) = P(y|do(x), w)$ if Y is independent of Z given X and W in $G_{\overline{X},\overline{Z(W)}}$. *Meaning:* If Z has no effect on Y once X and W are known (after intervening on X and possibly W), then the intervention on Z can be removed.

Selection Diagrams and Transportability: Given SCMs M and M^* for environments π and π^*, a selection diagram $\mathcal{D}$ adds nodes $S_i \rightarrow V_i$ wherever mechanisms or $P(U_i)$ differ. A causal quantity R is *directly transferable* if $R(\pi^*) = R(\pi)$. In particular, $P(y|do(x))$ is transferable if $(S \perp Y|X)_{G_{\underline{X}}}$, yielding $P(y|do(x), s) = P(y|do(x))$.

Problem definition

Knowledge transfer across environments is challenging because models often fail to capture the causal structures governing environmental variations. To illustrate robustness under such changes, we consider autonomous ground vehicles transferring causal knowledge across domains in three representative scenarios (Fig. 7).

Scenario 1—Static layout differences (linked to contribution 1: causal knowledge framework). Consider an autonomous system in Environment E_1, estimating the causal effect of state X on performance Y for different layouts Z, shown in Fig. 7a. When transferring to Environment E_2, the distribution $P(x, y, z)$ differs from $P^*(x, y, z)$. If layout-specific effects remain invariant, the causal effect in E_2 can be computed as:

$$P^*(y|\,\mathrm{do}(x)) = \sum_z P(y|\mathrm{do}\,(x), z)P^*(z). \tag{1}$$

Identifying when this invariance holds is crucial for reliable knowledge sharing while preserving causal relationships.

Scenario 2—Unmeasured environmental factors (linked to contributions 1 and 2: causal knowledge framework, edge selection diagrams). Here, Z is a proxy (e.g., noise density, exogenous variables) correlated with an unobserved factor such as environmental complexity (Fig. 7b). If Z has no direct or indirect effect on X or Y, disparities in $P(z)$ across domains can be ignored:

$$P^*(y|\,\mathrm{do}(x)) = P(y|do\,(x)). \tag{2}$$

The challenge lies in determining when such proxies can safely be excluded without degrading transfer performance.

Scenario 3—Dynamic mediating variables (linked to contribution 3: time-based normalization). In this case, Z is a dynamic mediator (e.g., lighting) influenced by X and affecting Y (Fig. 7c). If $P(z) \neq P^*(z)$ for each

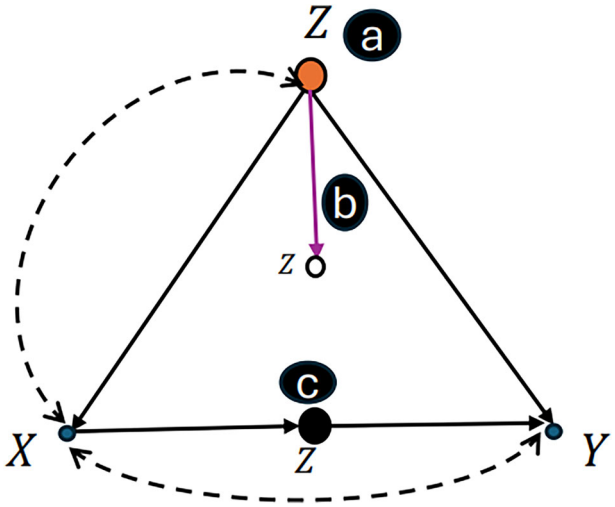

Fig. 7 | Causal selection diagrams for three scenarios. a Static layout differences, **b** unmeasured environmental factors, and **c** dynamic mediating variables.

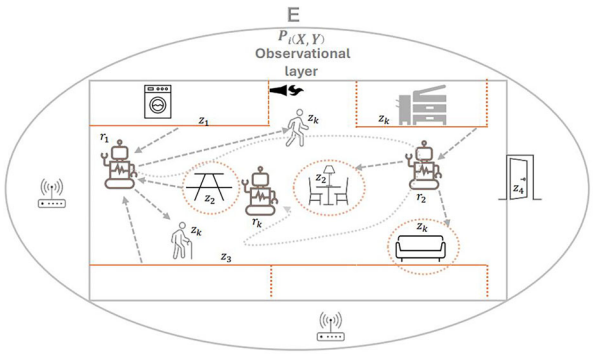

Fig. 8 | Causal-relational observation framework in multi-agent environments. Agents operate within an environment (E), collecting observations ($P_i(X, Y)$) across regions ($\{z_1, z_2, z_3, z_4, z_k\}$) while navigating and interacting. These observations enable the formation of causal-relational models linking perception and action for adaptive behaviour.

X, the causal effect in E_2 becomes:

$$P^*(y|\,\mathrm{do}(x)) = \sum_z P^*(y|\mathrm{do}(x), z)P^*(z|\mathrm{do}\,(x)), \tag{3}$$

which reduces to Eq. (1) if Z is unaffected by X:

$$P^*(y|\,\mathrm{do}(x)) = \sum_z P(y|\mathrm{do}\,(x), z)P^*(z|x). \tag{4}$$

Capturing such mediators is essential for speed-consistent and robust mapping.

These scenarios demonstrate that knowledge aggregation for transportability is fundamentally *causal*, not merely *statistical*, and directly motivate the contributions described in "Introduction".

Task-environment & local autonomous system dynamics

The on-device modeler $\mathcal{M}_{local}$, serves as the first layer in the hierarchical knowledge-sharing framework, operating at the autonomous agent level. In an asymmetric world, some dependent variables are observable ($V_o \in V$), while others are unobserved ($U_u \in U$), linked through local causal factors like map layout, static obstacles, path constraints, and fixed entry/exit points, illustrated in Fig. 8. In highly asymmetric environments, knowledge is location-dependent, whereas in self-similar environments, it remains consistent. Our nonparametric causal analysis relies on the graph G to encode assumptions and compute key quantities. Navigation is influenced by environmental variables: Z_0 (prior perception), Z_1 (room dimensions), Z_2 (static obstacles), Z_3 (path layout), Z_4 (entry/exit points), and $\mathbb{C}$ (external factors like lighting and noise). Unmeasured variables Z_0 and $\mathbb{C}$ introduce confounding effects, where Z_0 shapes the current environment, and $\mathbb{C}$ affects perception and navigation. Figure 9 illustrates the measured quantities in G, including z_1, z_2, z_3, z_4, which directly influence navigation (x) and outcome (y). z_1 affects z_2, z_3, and z_4, which in turn impact y. Unmeasured variables, shown with dashed arrows, include z_0 (historical perception) and $\mathbb{C}$ (external factors), both influencing z_1, z_2, z_3, and z_4. The absence of a direct link between z_1 and y indicates that its effect is mediated through z_2, z_3, and z_4. The total effect of x on y can be consistently estimated from observed distributions of x, z_1, z_2, z_3, z_4, and y. Assuming discrete variables, this effect is given by:

$$P(Y|\tilde{x}) = \sum_{z_1}\sum_{z_2}\sum_{z_3}\sum_{z_4}\Big[P(y|z_2, z_3, z_4, x)\cdot P(z_2|z_1, x) \cdot \sum_{\tilde{x}}\big(P(z_3|z_2, z_1, \tilde{x})\cdot P(z_1, \tilde{x})\big)\Big] \tag{5}$$

Here, $\tilde{x}$ serves as a summation index over X.

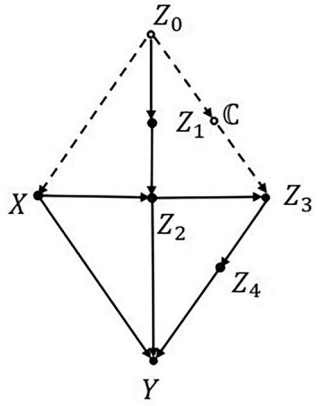

Fig. 9 | Causal graph (G) illustrating the effect of autonomous navigation ((X)) on outcome ((Y)). Directed edges represent causal dependencies among variables ($\{Z_0, Z_1, Z_2, Z_3, Z_4, \mathbb{C}\}$), depicting how latent factors mediate the influence of navigation decisions on observed outcomes.

G as models of intervention

By analyzing these causal dependencies, the autonomous system quantifies how navigation commands influence success in dynamic environments, considering spatial, static, and perceptual constraints. This leads to:

$$\begin{aligned} P(z_0, x, z_1, \mathbb{c}, z_2, z_3, z_4, y) \;=\; & P(z_0)\cdot P(x|z_0)\cdot P(z_1|z_0) \\ & \cdot P(\mathbb{c}|z_0)\cdot P(z_2|x, z_1) \\ & \cdot P(z_3|z_2, \mathbb{c})\cdot P(z_4|z_3) \\ & \cdot P(y|x, z_2, z_3, z_4)\cdot \epsilon_i \end{aligned} \tag{6}$$

This recursive decomposition of the joint distribution, similar to Bayesian networks[36], represents probability as a product of conditional dependencies in G, where ϵ_i $(1 \le i \le n)$ are mutually independent with arbitrary distributions. Graphically, F_i is represented as a parent node of X_i,

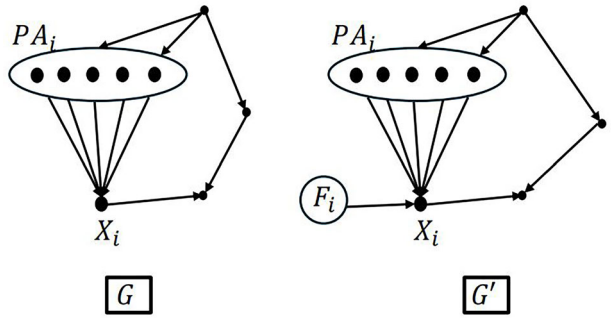

Fig. 10 | External intervention F_i represented by an augmented network G'.

with interventions encoded through standard conditional independence by conditioning on $F_i = f_i$, as depicted in Fig. 10. This process models intervention as an external force F_i, altering f_i, and modifying system behavior. The influence of perceptual variables Z_i and all unobserved factors including ϵ_i into u_i, are expressed in a deterministic nonparametric background model:

$$X_i = I(\mathrm{pa}_i, f_i, u_i), \tag{7}$$

where pa_i denotes the parents of the variable x_i, and $P(x_i|\mathrm{pa}_i)$ represents the conditional probability of x_i given its parents. where I is a three argument function: $I(q, l, m) = f_l(q, m)$ when $l = f_l$. In this context, the focus is primarily on the physical impact of the perceptual variables, which represent all unobserved factors Z_i, within a deterministic, nonparametric framework involving background variables (U_t, X_t) pairs:

$$x_i = f_i(\mathrm{pa}_i, u_i), \quad \text{for } i = 1, 2, \ldots, n, \text{and } u \in U \tag{8}$$

The joint probability distribution of the variables is given by:

$$P(x_1, x_2, \ldots, x_n) = \prod_i P(x_i|\mathrm{pa}_i) \tag{9}$$

The intervention modifies specific functions while keeping others unchanged, updating the probability function accordingly. The effect of an atomic intervention $do(U_i = u_i')$ is:

$$P(x_1, ..x_n \mid do(U_i = u_i')) = \begin{cases} \prod_{j \neq i} P(x_j|pa_j) & \text{if } u_i = u_i' \\ 0 & \text{if } u_i \neq u_i' \end{cases} \tag{10}$$

This removes $P(x_i|pa_i)$ from the factorization, severing parent-child links while preserving the network structure. Applying $do(U = u\prime)$ yields:

$$\begin{aligned} P(z_0, z_1, \mathbb{c}, z_2, z_3, y|do(U_x = u')) = P(z_0)P(z_1|z_0) \\ P(\mathbb{c}|z_0)P(z_2|z_1)P(z_3|z_2, \mathbb{c})P(y|x', z_2, z_3) \end{aligned} \tag{11}$$

Ensuring interventions maintain network consistency post-modification, the updated probability becomes:

$$P(u_1, \ldots, u_n|\hat{U}_i) = \begin{cases} \frac{P(u_1, \ldots, u_n)}{P(x_i'|pa_i)} & \text{if } u_i = u_i', \\ 0 & \text{if } u_i \neq u_i'. \end{cases} \tag{12}$$

Key question: How can knowledge aggregation across environments be achieved using do-expressions, and how can this mechanism be controlled and transferred?

To address this, we utilize MEC as an interventional layer $\mathcal{M}_{edge}$, where environmental differences are captured through a selection diagram S (diamond nodes in G), preserving perceptual disparities and ensuring consistent navigation across autonomous systems. Extending $P(x_1, \ldots, x_n)$ to compound interventions $do(S = s)$ leads to:

$$P(u_1, ..u_n|\hat{s}) = \begin{cases} \prod_{i|U_i \notin S} P(x_i|pa_i) & \text{if } u_1, \ldots \cdot u_n \\ \text{consistentwith } s, & \\ 0 & \text{otherwise} \end{cases} \tag{13}$$

Joint edge-causal modeling for global generalization

Once the on-device modeler $\mathcal{M}$ processes local sensor inputs to the edge, the $\mathcal{M}_{edge}$ extracts relevant perceptual and state variables. It stores structured causal knowledge in selection diagrams (S^*), which represent invariant mappings across different environments. This knowledge is made accessible for future agent requests, ensuring scalability and reusability. Each x_k is determined by monitoring x_{k-1}, z_k, and z_{k-1}, following the probability distribution $P(x_k|x_{k-1}, z_k, z_{k-1})$. The system's overall performance is captured by the joint probability: $P(y, z_1, \ldots, z_n, x_1, \ldots, x_n)$ where y represents the outcome. Notably, z_k (perception) depends only on the previous observation z_{k-1} and the prior state x_{k-1}.

Based on Eq. (13), the performance of $P(\pi)$ in the new environment, denoted as $P(\pi^*)$, will be determined by the distribution of the adapted strategy S^*:

$$\begin{aligned} P^*(y, x_1, \ldots, x_n, z_1, z_2, \ldots, z_n) = P^*(y|x_1, \ldots, x_n, z_1, \ldots z_n) \\ \cdot \prod_{k=1}^{n} P^*(x_k|x_{k-1}, z_k, z_{k-1}) \\ \cdot \prod_{k=1}^{n} P^*(z_k|z_{k-1}, x_{k-1}) \end{aligned}$$

The first two terms remain invariant for state estimation in any environment and the third one can be known using the edge intervention, we have:

$$\begin{aligned} P^*(y) = \sum_{z_1, \ldots, z_n} \sum_{x_1, \ldots, x_n} \underbrace{P(y|z_1, \ldots, z_n, x_1, \ldots x_n)}_{term-1} \\ \underbrace{\prod_{k=1}^{n} P(x_k|x_{k-1}, z_k, z_{k-1})}_{term-2} \underbrace{\prod_{k=1}^{n} P^*(z_k|z_{k-1}, x_{k-1})}_{term-3^*} \end{aligned} \tag{14}$$

The *term-3** encapsulates the distribution of perceptual variables within $P(\pi^*)$, acting as the core of collective intelligence. It enables autonomous navigation in unknown environments by dynamically adapting perceptual insights and optimizing decision-making strategies based on observed states. In the most cases, where S^* (*term-3**) for individual environment $P^*(y) = P(y|\hat{z}_1, \hat{z}_2, \ldots, \hat{z}_n)$ is deterministic and time-invariant, Z_k becomes a function of Z_{k-1}, X_k, and X_{k-1}:

$$z_k = g(z_{k-1}, x_k, x_{k-1}). \tag{15}$$

Then the summation over $z_1, \ldots, z_n$ can be performed, yielding:

$$\begin{aligned} P^*(y) = \sum_{x_1, \ldots, x_n} P(y|x_1, x_2, \ldots, x_n, g_1, g_2, \ldots, g_n) \\ \cdot \prod_{k=1}^{n} P(x_k|x_{k-1}, g_{k-1}) \end{aligned} \tag{16}$$

where g_k is defined recursively as:

$$g_1 = g(x_1), \quad g_k = g(g_{k-1}, x_k, x_{k-1}). \tag{17}$$

In the special case of a strategy S^* composed of elementary actions $do(S_k = s_k)$, the function g degenerates into a constant, z_k, and we obtain:

$$P^*(y) = P(y|\hat{z}_1, \hat{z}_2, \ldots, \hat{z}_n) \tag{18}$$

which can also be obtained from Eq. (13).

$$P^*(y) = \sum_{x_1,\ldots,x_n} P(y|x_1, x_2, \ldots, x_n, z_1, z_2, \ldots, z_n) \cdot \prod_k P(x_k|x_{k-1}, z_{k-1}) \tag{19}$$

This involves weighting the z-specific effects by $P^*(x|z)$, which needs to be estimated in the target environment. Additionally, this simplifies to Eq. (1), yielding $P(x_{k-1}|x_k, \hat{z}_k) = P(x_{k-1}|x_k, z_k)$ if $P^*(z)$ provided. We first show how to convert the derived observation value into do-free terms to depart from typical do-calculus assumptions. This transforms causal expressions into conditional probabilities based on observed variables, enabling flexible modeling while preserving the causal structure.

Let N_k denote the perceptual knowledge plan-an ordered sequence of $P(\pi^*)$ where no elements are descendants of any variables in $P(\pi)$. The sequence $P^*(\hat{z}_1, \hat{z}_2, \ldots, \hat{z}_n)$ represents value assignments to perceptual variables in $P(\pi)$, with $\hat{z}_k$ setting Z_k to z_k. This sequence controls the system, where each $g_k(x_k)$ assigns Z_k based on X_k while ensuring no descendants of Z_k appear in $g_k(X_k)$. A control problem is identifiable in an unseen environment if the expression $P(y|\hat{z}_1, \hat{z}_2, \ldots, \hat{z}_n)$ can be reduced, via do-calculus, to a form where S appears only as a conditioning variable in do-free terms. This reduction ensures correct causal inference of interventions on perceptual variables, enabling effective decision-making and robust control in dynamic environments. We can now state the main theorem:

Proposition 1 The probability $P(Y|\hat{z_1}, \ldots, \hat{z_n})$ is identifiable if, for every $1 \le k \le n$, there exists a set X_k of covariates satisfying the following sequential back-door conditions: $X_k \subseteq N_k$, where X_k consists of non-descendants of $\{Z_k, Z_{k+1}, \ldots, Z_n\}$, and $(Y \perp\!\!\!\perp Z_k | Z_1, \ldots, Z_{k-1}, X_1, X_2, \ldots, X_k)_{G_{\underline{Z_k}\overline{Z}_{k+1},\cdots,\overline{Z}_n}}$.

When these conditions are satisfied, the causal effect of the knowledge-sharing Eq. (18) can be reduced to do-free terms in target environment with edge inference:

$$P(y|\hat{z_1}, \ldots, \hat{z_n}) = \sum_{x_1,\ldots,x_n} P(y|x_1, \ldots, x_n, z_1, \ldots z_n) \cdot \prod_{k=1}^{n} P(x_k|x_1, \ldots, x_{k-1}, z_1, \ldots, z_{k-1})$$

The *proof* given here is based on the inference rules of *definition 3.4*, which reduces the causal effect formula to hat-free expression to test the identifiability of the control problem:

Step 1: Using the condition $X_k \subseteq N_k$, we have:

$$P(x_k|x_1, \ldots, x_{k-1}, z_1, \ldots, z_{k-1}, \hat{z}_k, \hat{z}_{k+1}, \ldots, \hat{z}_n) = P(x_k|x_1, \ldots, x_{k-1}, z_k, \ldots, z_{k-1})$$

This holds because $P^*(X_1, \ldots, X_k, Z_1, \ldots Z_{k-1})$ consists of non-descendants of any node in $P(Z_k, Z_{k+1}, \ldots, Z_n)$, and Rule 3 of do-calculus (*definition 3.3*) eliminates irrelevant variables.

Step 2: Further conditioning using the above theorem, permits us to invoke rule 2 and write:

$$= P(y|x_1, \ldots, x_k, z_1, \ldots, z_{k-1}, z_k, \hat{z}_{k+1}, \ldots, \hat{z}_n)$$

Under the assumption of the sequential back-door conditions, this simplifies to external intervention when: $P(y|x,\ \mathrm{do}\,(\hat{z_1}))$

$$\begin{aligned}
P(y|\hat{z}_1, \ldots, \hat{z}_n) &= \sum_{x_1} P(y|x_1, \hat{z}_1, \hat{z}_2, \ldots, \hat{z}_n) \cdot P(x_1|\hat{z}_1, \ldots, \hat{z}_n) \\
&= \sum_{x_1} P(y|x_1, z_1, \hat{z}_2, \ldots, \hat{z}_n) P(x_1) \\
&= \sum_{x_2} \sum_{x_1} P(y|x_1, x_2, z_1, \hat{z}_2, \ldots, \hat{z}_n) \cdot P(x_1) P(x_2|x_1, z_1, \hat{z}_2, \ldots, \hat{z}_n) \\
&= \sum_{x_2} \sum_{x_1} P(y|x_1, x_2, z_1, z_2, \hat{z}_3, \ldots, \hat{z}_n) \cdot P(x_1) P(x_2|x_1, z_1) \\
&\vdots \\
P(y|\hat{z}_1, \ldots, \hat{z}_n) &= \sum_{x_1,\ldots,x_n} P(y|x_1, \ldots, x_n, z_1, \ldots, z_n) \prod_{k=1}^{n} P(x_k|x_1, \ldots, x_{k-1}, z_1, \ldots, z_{k-1})
\end{aligned}$$

This establishes the G-identifiability of the global causal relationship in Eq. (18), optimizing SLAM while maintaining system control. Consequently, we arrive at the final knowledge-sharing maximization problem, formulated as follows:

$$\max_{\{z_1,\ldots,z_n\}} \mathbb{E}[y] = \sum_{x_1,\ldots,x_n} P(y|x_1, \ldots, x_n, z_1, \ldots z_n) \underbrace{\prod_{k=1}^{n} P^*(x_k|x_{k-1}, z_k, z_{k-1})}_{S^*} \tag{20}$$

This final form demonstrates the decomposition of the probability into a summation over Z_k in the continuous trajectory x_k, fulfilling the conditions for identifiability. This yields an effective decision procedure and further prove our Eq. (14) that knowing *term-3** potentially enhances the system reliability. To analyze causality testing, we use the maximum mean discrepancy (MMD)[37]. Since MMD is designed for i.i.d (independen and identical distribution) data, we extend its validity by deriving conditions for non-i.i.d. cases and introducing a test statistic:

$$\mathrm{MMD}\,(\mathbf{y}_i^P, \mathbf{y}_j^{P^*}) = \left\| \mathbb{E}_{\mathbf{y}_i \sim P}[\phi(\boldsymbol{x}_i^P)] - \mathbb{E}_{\mathbf{y}_i \sim P^*}[\phi(\boldsymbol{x}_i^{P^*})] \right\|_{\mathcal{H}} \tag{21}$$

Using Eq. (20), we compare true vs. knowledge-sharing performance via Eq. (20). The MMD test remains valid with a characteristic kernel if initial conditions $\mathbf{Z}^P(0)$, $\mathbf{Z}^{P^*}(0)$ and input trajectories $\mathbf{x}^P$, $\mathbf{x}^{P^*}$ are i.i.d. A positive MMD indicates distinct trajectory distributions and causal influence through knowledge aggregation.

Edge knowledge modeler (EKM) $\mathcal{M}_{edge}$

The edge system organizes data using a causal graph G, where each parent-child relationship defines a deterministic function, accounting for structural discrepancies (S^*). This enables efficient retrieval and adaptation to dynamic queries. A relation R is identifiable in a multi-environment system if it can be inferred from knowledge uploaded by autonomous systems across environments $\{\pi_1, \ldots, \pi_K\}$ to a target environment π^*:

$$R = \{R(\pi_1), \ldots, R(\pi_K), R(\pi^*)\}, \tag{22}$$

where $R(\pi_k)$ represents the functional relationships derived from perceptual and state variables. Each system independently collects and uploads critical variables such as state transitions $P(x_k|x_{k-1},\ z_{k-1})$ and perceptual

observations $P(z_k|z_{k-1}, x_{k-1})$, forming a shared repository in the edge system. This repository enables seamless knowledge transfer, allowing models from $\{\pi_1, \ldots, \pi_K\}$ to inform systems in π^*. Meta-identifiable relations help bridge structural differences, ensuring consistent navigation and decision-making. Perceptual discrepancies (S^r) are stored in the edge system as an identity matrix:

$$I_n = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{bmatrix}.$$

Diagonal elements $I_{ii} = 1$ represent vehicle autonomy, while off-diagonal elements $I_{ij} = 0$ indicate independence. Upon receiving a transition request via controller (C_q), the EKM processes interventional queries to facilitate knowledge transfer. It identifies perceptual variables (S^r) relevant to the target environment and retrieves applicable causal submodels from prior stored experiences. Using causal identifiability, it updates the identity matrix I_n with a knowledge-sharing matrix S, where off-diagonal elements represent shared knowledge:

$$S = \begin{bmatrix} 0 & s_{12} & \cdots & s_{1n} \\ s_{21} & 0 & \cdots & s_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ s_{n1} & s_{n2} & \cdots & 0 \end{bmatrix} \tag{23}$$

The knowledge is then transferred to the target agent, delivering structured observational data that enables adaptation to the new environment. The updated model becomes: $M = I_n + S$:

$$M_{\text{updated}} = \begin{bmatrix} 1 & s_{12} & \cdots & s_{1n} \\ s_{21} & 1 & \cdots & s_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ s_{n1} & s_{n2} & \cdots & 1 \end{bmatrix} \tag{24}$$

Once the target agent receives knowledge from the edge, it applies the acquired information to navigate efficiently. It adjusts its navigation strategy using learned causal dependencies and modifies planned paths to minimize uncertainty. The agent continuously updates its local models based on real-time observations, ensuring dynamic adaptation to the environment.

Data availability

The datasets generated and analyzed during the current study, including occupancy grid maps, robot trajectories, and simulation results, are available from the corresponding author upon reasonable request. Due to hardware-specific constraints, real-world UGV and TurtleBot4 data can be shared in processed form to ensure reproducibility.

Received: 23 June 2025; Accepted: 16 September 2025;
Published online: 07 January 2026

References

1. Grisetti, G., Kümmerle, R., Stachniss, C. & Burgard, W. A tutorial on graph-based slam. *IEEE Intell. Transp. Syst. Mag.* **2**, 31–43 (2010).
2. Thrun, S. et al. Stanley: the robot that won the DARPA Grand Challenge. *J. Field Robot.* **23**, 661–692 (2006).
3. Yin, T., Zhou, X. & Krahenbuhl, P. Center-based 3d object detection and tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 11784–11793 (2021).
4. Li, Z. et al. Bevformer: learning bird's-eye-view representation from multi-camera images via spatiotemporal transformers. In *European Conference on Computer Vision*, 1–18 (2022).
5. Liu, Z. et al. Bevfusion: Multi-task multi-sensor fusion with unified bird's-eye view representation. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, 2774–2781 (2023).
6. Al-Dulaimy, A. et al. The computing continuum: from IoT to the cloud. *Internet Things* **27**, 101272 (2024).
7. Nawaz, M. W. et al. 6g edge-networks and multi-UAV knowledge fusion for urban autonomous vehicles. *Phys. Commun.* **67**, 102479 (2024).
8. Gupta, S., Tripathi, A. K. & Parameswaran, V. Attention transfer-based deep distilled architecture for 6g driven-smart vehicle transportation system. In *IEEE Transactions on Intelligent Transportation Systems* (2024).
9. Weiss, K., Khoshgoftaar, T. M. & Wang, D. A survey of transfer learning. *J. Big Data* **3**, 1–40 (2016).
10. Zhang, Y. & Yang, Q. An overview of multi-task learning. *Natl Sci. Rev.* **5**, 30–43 (2018).
11. Vilalta, R. & Drissi, Y. A perspective view and survey of meta-learning. *Artif. Intell. Rev.* **18**, 77–95 (2002).
12. Cody, T. Homomorphisms between transfer, multi-task, and meta-learning systems. In *International Conference on Artificial General Intelligence* (Springer International Publishing, 2022).
13. Pearl, J. *Bibliography* (Cambridge University Press, 2009).
14. Nawaz, M. W. et al. K-DUMBs IoRT: Knowledge driven unified model block sharing in the internet of robotic things. In *2023 IEEE 97th Vehicular Technology Conference (VTC2023-Spring)* (2023).
15. Pearl, J. & Bareinboim, E. Transportability of causal and statistical relations: a formal approach. In *Proceedings of the Twenty-Fifth National Conference on Artificial Intelligence (AAAI)*, 247–254 (2011).
16. Bareinboim, E. & Pearl, J. Causal transportability with limited experiments. In *Proceedings of the Twenty-Seventh AAAI Conference on Artificial Intelligence* (2013).
17. Bareinboim, E. & Pearl, J. Meta-transportability of causal effects: a formal approach. In *Proceedings of the Sixteenth International Conference on Artificial Intelligence and Statistics (AISTATS)* (2013).
18. Howard, R., Peter, M. & Kunze, L. Extending Structural Causal Models for Autonomous Vehicles to Simplify Temporal System Construction & Enable Dynamic Interactions Between Agents. In *Proc. of Machine Learning Research*, Vol. 275 (eds Huang, B. & Drton, M.) 1477–1505 (PMLR, 2025).
19. Li, J. et al. Exploring the causality of end-to-end autonomous driving. Preprint at https://arxiv.org/abs/2407.06546 (2024).
20. Tang, C., Srishankar, N., Martin, S. & Tomizuka, M. Grounded relational inference: domain knowledge driven explainable autonomous driving. *IEEE Trans. Intell. Transp. Syst.* **25**, 10617–10635 (2024).
21. Nawaz, M. W., Imran, M. A. & Popoola, O. Temporal hierarchical clustering for knowledge aggregation in connected vehicular networks with federated multi-task learning. In *2024 IEEE Wireless Communications and Networking Conference (WCNC)* (2024).
22. Glasgow, U. Scotland 5G Centre Project Glasgow. https://scotland5gcentre.org/s5gcprojectglasgow/ (2019).
23. Yan, P. et al. A comprehensive survey of deep transfer learning for anomaly detection in industrial time series: methods, applications, and directions. *IEEE Access* (2024).
24. Tan, C. et al. A survey on deep transfer learning. In *Artificial Neural Networks and Machine Learning - ICANN* (2018).
25. Luo, F.-M. et al. A survey on model-based reinforcement learning. *Sci. China Inf. Sci.* **67**, 121101 (2024).
26. Ke, Z. et al. Enhancing transferability of deep reinforcement learning-based variable speed limit control using transfer learning. *IEEE Trans. Intell. Transp. Syst.* **22**, 4684–4695 (2020).
27. Sheikhlar, A., Thórisson, K. R. & Eberding, L. M. Autonomous cumulative transfer learning. In *International Conference on Artificial General Intelligence*, 306–316 (2020).
28. Shi, L. & Liu, W. Adversarial self-training improves robustness and generalization for gradual domain adaptation. In *Advances in Neural Information Processing Systems*, Vol. **36** (2024).

29. Singhal, P., Walambe, R., Ramanna, S. & Kotecha, K. Domain adaptation: challenges, methods, datasets, and applications. *IEEE Access* **11**, 6973–7020 (2023).
30. Bouvier, V., Very, P., Hudelot, C. & Chastagnol, C. Hidden covariate shift: a minimal assumption for domain adaptation. Preprint at https://arxiv.org/abs/1907.12299 (2019).
31. Rojas-Carulla, M. et al. Invariant models for causal transfer learning. *J. Mach. Learn. Res.* **19**, 1–34 (2018).
32. Rahimi, A., Luan, P.-C., Liu, Y., Rajic, F. & Alahi, A. Sim-to-Real Causal Transfer. A Metric Learning Approach to Causally-Aware Interaction Representations. In *Proc. 2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* 17271–17281 https://doi.org/10.1109/CVPR52734.2025.01610 (2025).
33. Guo, S. et al. Causal de finetti: on the identification of invariant causal structure in exchangeable data. In *Advances in Neural Information Processing Systems*, Vol. **36** (2024).
34. Dahal, P. et al. Robuststatenet: Robust ego vehicle state estimation for autonomous driving. *Robot. Auton. Syst.* **172**, 104585 (2024).
35. Grisetti, G., Stachniss, C. & Burgard, W. Improving grid-based slam with rao-blackwellized particle filters by adaptive proposals and selective resampling. In *Proceedings of the 2005 IEEE International Conference on Robotics and Automation* (2005).
36. Heckerman, D. In *Learning in Graphical Models*, 301–354 (1998).
37. Gretton, A. et al. A kernel two-sample test. *J. Mach. Learn. Res.* **13**, 723–773 (2012).

Acknowledgements

The authors acknowledge support from the University of Glasgow and the Scotland 5G Centre. We also thank colleagues at the James Watt School of Engineering for their technical guidance and provision of laboratory facilities. This work was supported in part by EPSRC Projects, CHEDDAR under Grant EP/X040518/1 and in part by CHEDDAR uplift under Grant EP/Y037421/1.

Author contributions

Muhammad Waqas Nawaz conceptualised the study and developed the methodology. M.W. Nawaz performed the simulations and data analysis. M.W. Nawaz, M.M. Alam, and Rafiq Swash conducted the literature review and drafted the background section. M.W. Nawaz and Olaoluwa Popoola wrote the main manuscript text and supported the framework, implementation and real-world deployment. M.A. Imran, Qammer H. Abbasi, and Olaoluwa Popoola contributed to the theoretical modelling and experimental design. All authors reviewed and approved the final manuscript.

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to Muhammad Waqas Nawaz.