

# A multimodal mission-oriented model of hybrid control for robotic systems<sup>\*</sup>

Andrii Yarmilko<sup>1,\*</sup> and Stanislav Marchenko<sup>1,†</sup>

<sup>1</sup> Bohdan Khmelnytsky National University of Cherkasy, 81 Shevchenko Blvd., Cherkasy, 18031, Ukraine

## Abstract

This paper presents a structured multimodal mission-oriented model of hybrid control for robotic systems operating under incomplete information, varying channel quality, real-time constraints, and operator involvement. The control process is described as a coordinated sequence of signal- and feature-level processing, semantic interpretation, situational fusion, mission-level selection, and delegation. Unlike conventional command or policy blending, shared control is organized through admissible event routing among operator, autonomous, mission, and safety handlers. The multimodal state comprises navigation, visual, communication, operator, system-computational, and energy channels, with channel-state estimates, data availability, and uncertainty considered separately. The model was evaluated using synthetic data in a discrete-time simulation of a declaratively specified multi-stage mission. The experiments covered nominal execution, isolated channel degradations, compound conflict situations, and controlled safety events. Their purpose was to assess the basic functional consistency of the model and its sensitivity to different degradation factors. Within the adopted simulation setting, the results demonstrated context-dependent channel relevance, correct mission and fallback transitions, rejection of conflicting operator initiative, and stable delegation among manual, shared, autonomous, and safety-oriented modes.

## Keywords

hybrid control, robotic systems, mission graph, shared autonomy, event routing, multimodal fusion, effective reactivity, operator initiative, safety fallback

## 1. Introduction

The need for structural and system-level optimization of mission-oriented robotic systems has increased attention to synergy and deeper information cooperation between the human operator and the robotic machine [1]. This also applies to unmanned platforms intended for various purposes. At the current stage, progress in human-machine systems is expected within the hybrid intelligence paradigm [1, 2]. Consequently, research focuses on synthesizing new approaches to organizing such interaction, methods for their technical realization, and their implementation in applied systems. According to the concept of hybrid intelligence [2], a human-machine system should rationally combine human and machine capabilities while preserving the leading role of the human within the system. The evident shift in mission-control interaction toward cooperation in performing intelligent operations raises the issue of effectively organizing the interaction between human mental and cognitive functions and the artificial intelligence of the unmanned platform's machine components. Mission-oriented human-robot systems require a shared representation of the operational context, flexible allocation of control functions, and support for human-machine cooperation under changing environmental and system conditions. Under traditional control approaches based on maximizing control over system parameters, the amount of information processed by the human operator generally exceeds physiologically feasible levels. At the same time, the tasks listed above cannot be solved by absolutizing the intellectual independence

<sup>\*</sup>ADT-ISMDDC 2026: Advanced Digital Technologies: Intelligent Systems, Modeling, and Data-Driven Control, September 22–24, 2026, Odesa, Ukraine

<sup>1\*</sup> Corresponding author.

<sup>†</sup> These authors contributed equally.

✉ a-ja@ukr.net (A. Yarmilko); sv.marchenko1989@gmail.com (S. Marchenko)

id 0000-0003-2062-2694 (A. Yarmilko); 0000-0003-1655-3091 (S. Marchenko)



© 2026 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

(autonomy) of the unmanned platform. Accordingly, the optimal role of the human operator in a hybrid unmanned-mission control system is to compensate for limitations in the software that supports the platform's intelligent functions. The unmanned device itself should minimize the operator's physical, psychological, and emotional overload while current real-time control tasks are being solved. The selection of information-interaction channels between the human and machine components of a hybrid control system also requires specific justification. Important quality criteria for the communication structure of such a system include physiological and psycho-emotional limitations on human information perception and the operator's psychological responses to the content and quality of interaction with the unmanned platform. In open environments, objects within the mission area are most rationally referenced through visual landmarks because the visual perception channels of the human operator and the robot are the most closely related. They enable informative features to be extracted within the control area and interpreted during the synthesis of control decisions with the lowest total resource expenditure by both the operator and the platform's artificial intelligence. Nevertheless, remote control of unmanned systems, particularly in complex and emotionally demanding applications, has a pronounced adverse effect on operators' mental health and performance [3, 4]. This necessitates an additional objective analysis of the phenomenon in the context of ensuring control-loop stability. In this context, investigating mission control of a robotic system under variable channel quality, incomplete information, real-time constraints, and operator participation is an important research task. Addressing it requires a mathematical model of multimodal hybrid control that combines performance, reactivity, mission execution, operator cognitive state, communication stability, and safety compliance within a single formalized system.

The objective of this article is to construct and perform scenario-based validation of a mathematical model for multimodal mission-oriented hybrid control of a robotic system. The model should connect event-level reactivity, channel estimates, semantic state, the mission graph, safety constraints, operator initiative, and the delegation policy into a sequential system of mappings, thereby enabling its configurations to be assessed according to the admissibility criteria for unmanned-mission control modes. The experiments are intended to assess the model's basic functional consistency and its sensitivity to isolated and compound degradations rather than to estimate the performance of a particular robotic platform.

## **2. Theoretical background and related works**

### **2.1. Hybrid control, shared autonomy, and human-robot teaming**

Hybrid control of robotic systems is developing at the intersection of classical control theory, human-robot interaction, shared autonomy, mixed-initiative control, and runtime supervision. Human-robot interaction research emphasizes that robotic autonomy should be regarded as a variable allocation of functions among the human, the automatic controller, and the autonomous planner [5]. This conclusion is consistent with the concept of levels of automation, in which the human operator can act as a source of goals, supervisor, arbiter, or active executor depending on risk, task complexity, and information availability [6]. Recent reviews of practical human-robot teaming additionally emphasize three interrelated aspects of interaction: action coordination, communication, and cognitive compatibility [7]. For mission control, this triad has direct mathematical significance because each aspect becomes a separate observation channel and a constraint on the admissibility of a control decision. Shared autonomy emerged in response to situations in which an autonomous system has sufficient information to provide local assistance while the operator retains a strategic or contextual advantage. One of the basic mathematical approaches is action or policy blending, in which the resulting command is computed as a weighted combination of operator control and an autonomous proposal [8]. This scheme is straightforward to interpret and performs well for tasks with a relatively stable goal; however, its

sensitivity to intention-recognition errors creates a need for probabilistic models of the user's state. A POMDP (partially observable Markov decision process) formulation of shared autonomy treats user intention as a hidden state and combines belief updating with action planning under incomplete information [9]. Its advantage is the rigorous treatment of uncertainty, whereas its practical limitations arise from the dimensionality of the goal space, the complexity of belief-state approximations, and the need for rapid updates within the control loop. Model-free shared autonomy and reinforcement-learning approaches reduce the number of prior assumptions about the intention model but shift the complexity to data, rewards, and safety validation [10]. In open-environment robotic systems, this trade-off has mixed value: learning-based arbitration can improve behavior in scenarios with complex interaction patterns, but guaranteeing deadlines, the safety envelope, and mission validity requires an external constraint layer. Reviews of POMDPs in shared autonomy confirm that partial observability is natural for human-in-the-loop systems, although most implementations focus on relatively constrained manipulation or assistive-robotics tasks [11]. For mission control of mobile robotic systems, this model class must be supplemented with temporal, resource, communication, and mission constraints.

A separate research direction develops autonomy-infused teleoperation and perceptive shared autonomy, in which the autonomous system not only blends commands but also augments the operator's actions with knowledge of the map, risk zones, the expected trajectory, perception-model uncertainty, and available local-planning modes [12]. This approach is important for multimodal mission control because operator initiative must be evaluated through the available navigation, visual, communication, operator, system, and energy channels. A new wave of shared autonomy based on generalist robot policies and VLA models demonstrates that action-level steering can improve task success and reduce manual intervention; however, these results leave open the question of rigorously combining policy steering with runtime safety [13]. Mathematical methods for hybrid control can also be systematized by how they coordinate authority. Compensatory methods use weighted sums, utility functions, Bayesian belief updates, soft penalties, MPC, or learning-based value functions. They are suitable when deterioration in one channel can be partially compensated for by high confidence in another channel. Non-compensatory methods use thresholds, guard conditions, lexicographic ordering, minimax/Chebyshev criteria, control barrier functions, or safety filters. Their use is justified for constraints for which averaging is inadmissible, including collision risk, geofence violation, critical energy, a missed hard deadline, or loss of the command link during a takeover-critical phase. A hybrid-intelligent system executing a mission requires a composition of compensatory and non-compensatory rules linked to the mission phase and event class.

## **2.2. Mission-oriented specification and semantic mission execution**

The mission-oriented approach in robotics aims to separate a high-level objective from low-level motion commands. Current mission-specification research shows that robotic systems use state machines, behavior trees, hierarchical task networks, temporal logic, BPMN-like descriptions, DSLs, and task-oriented mission languages [14]. Comparative analyses of robotic-mission formalisms emphasize the absence of a single standard that would support equally well goal hierarchies, reactivity, integration with humans and external devices, and runtime reconfiguration [15]. Mission Specification Patterns for mobile robots describe recurring task patterns such as reach, patrol, avoid, visit, coverage, surveillance, search, return, and monitor [16]. They show that a formal mission specification must separate a routing task from a search or coverage task. A flight or movement from point A to point B performs a transit function, whereas a search phase requires the definition of a search area, a coverage metric, visual or sensory conditions for candidate detection, a target-confirmation threshold, and rules for repeated inspection. Therefore, the search stage should be represented as a semantic phase with its own state variables. Research on language-specified missions and heterogeneous robot teaming shows that contemporary missions are increasingly specified at the intent level, while execution requires semantic reasoning,

opportunistic communication, shared maps, and active information acquisition [17]. Air-ground collaboration in unknown environments illustrates the need to connect a semantic-metric map, mission actions, and plan updates when the specification changes. It follows that the mission state must depend on a combination of spatial information, perception evidence, communication availability, target conditions, and resource constraints. Mathematically, mission-oriented control is most often implemented using state graphs, hybrid automata, timed automata, temporal logic, HTNs, or behavior trees. A state graph provides a transparent interpretation of phases and transitions; timed automata add temporal constraints; temporal logic formalizes properties such as eventually, always, and until; an HTN fixes the task hierarchy; and behavior trees provide reactive composition of subtasks. For hybrid control, the key requirement is to preserve an unambiguous relationship between mission semantics and the admissible control mode. A mission graph with guard conditions, dynamic deadlines, coverage variables, and fallback transitions therefore provides a convenient compromise.

### **2.3. Multimodal perception, channel reliability, and missing modalities**

Multimodal modeling of robotic systems is based on the assumption that no single channel fully represents the system state. Reviews of multi-sensor fusion for autonomous systems distinguish sensor-level, feature-level, and decision-level fusion, emphasizing the trade-off among raw-data volume, feature informativeness, processing latency, and computational cost [18]. In robotic vision, multimodality includes not only the conventional combination of cameras and sensors but also semantic scene understanding, VLM/LLM-assisted reasoning, navigation, localization, and domain adaptation [19]. Multimodal processing is likewise necessary for mission-oriented hybrid control because, in addition to perception tasks, it includes intrinsically modal operator and communication channels, system reactivity, and energy state. Filtering methods such as the Kalman filter, extended/unscented Kalman filters, and sequential Monte Carlo remain fundamental for estimating the navigation state because they provide recursive estimation in the presence of a dynamic model and noisy measurements [20, 21]. Channels with unknown correlations and conflicting estimates require conservative fusion. Covariance intersection is used when correlations between estimates are unknown and fusion must not artificially reduce uncertainty [22]. Evidence-based fusion, particularly Dempster-Shafer theory, is useful for discrete hypotheses and conflicting evidence, although its direct use in a high-frequency loop requires complexity constraints [23]. The operator channel requires separate analysis because it includes a hidden cognitive state. Psychophysiological measurements such as HRV, EDA, EEG, gaze, PERCLOS, response time, and task performance are widely used to estimate workload and fatigue [24]. Systematic reviews emphasize that the results depend on the task type, sensor set, observation window, and interpretation method [25]. Multimodal datasets in human-robot collaboration combine physiological, behavioral, audio, facial, and task-related signals [26]. Missing modalities must be considered in real control systems because physiological sensors may be unavailable, invalid, or organizationally unacceptable. The operator state in the model should therefore also include a modality mask and observation uncertainty [27].

### **2.4. Reactivity, real-time event processing, and energy-aware execution**

Reactivity in embedded and cyber-physical systems is associated with processing events within temporal constraints. Classical real-time scheduling theory provides the basis for analyzing periodic and sporadic tasks using fixed-priority scheduling, rate-monotonic scheduling, and earliest deadline first [28, 29]. Weakly-hard real-time models describe streams in which a limited number of deadline misses preserves acceptable control quality [30]. Such models are particularly appropriate for video frames, informational telemetry, and auxiliary updates, whereas hard safety events must retain a separate non-compensatory status [31]. In real-time robotic systems, reactivity includes queue waiting, event processing, dispatching, network latency, feature processing, decision-making, mission transition, and command execution. Experimental studies of mission

computers for robotic platforms show that energy-saving modes reduce consumption but introduce wake-up delays and alter the available reactivity budget [32]. The energy channel must therefore be included in the model as a factor affecting inference, the system tick, event processing, and admissible control modes.

## **2.5. Safety assurance, runtime guards, and mission-valid delegation**

Safety assurance for autonomous systems is currently being developed through control barrier functions, safety filters, reachability, runtime assurance, and fallback supervision. Safety filters provide a unified intervention scheme in which a control proposal is checked against the safe set and projected onto or replaced with an admissible action [33]. Control barrier functions formalize the invariance of the safe region and enable control synthesis under state/input constraints, disturbances, and sampled-data effects [34]. Runtime assurance adds an architectural layer in which an advanced or learning-based controller may operate only within a monitored safety envelope; if the condition is violated, the system switches to a verified fallback controller [35].

For mission-oriented hybrid control, these methods have a dual role. They impose hard constraints on physical safety and also constrain delegation. If an operator command or autonomous recommendation violates the safety envelope, delegation must be rejected regardless of trust, expected utility, or mission benefit. If the operator channel degrades, the system may retain the operator in the loop at the information level but must not delegate hard real-time functions that require a guaranteed response to the operator. Thus, the general model must combine safety filters, real-time reactivity, and human-autonomy teaming within a common mathematical requirement.

## **3. Unresolved issues**

The preceding analysis indicates the need for a consistent mathematical model that combines mission semantics, event-driven reactivity, multimodal channel evidence, operator initiative, safety constraints, and a delegation policy. In mission control, the time horizon depends on the active phase, spatial situation, channel state, operator availability, computational resources, energy reserve, and event class. The model must therefore distinguish the event deadline, the mission-transition horizon, the reactivity budget of a control mode, event-routing admissibility, and the time required to process operator initiative. It should also support the investigation of the key research questions (RQ):

- RQ1: How should the multimodal state of a robotic system be represented for mission-oriented hybrid control under varying information-channel quality, incomplete observations, and data uncertainty? This question concerns how navigation, visual, communication, operator, system, and energy features can be combined into a unified mission-usable representation of the system state.
- RQ2: How can the mission graph, current-phase semantics, sub-mission execution progress, and fallback decisions be connected within a unified mission-execution logic? This question concerns the transition from describing a mission as a set of phases and transitions to a runtime process in which the system determines an appropriate next step, evaluates progress, and switches to fallback behavior when the situation deteriorates.
- RQ3: How should the reactive suitability of a hybrid control system be evaluated with regard to temporal constraints, events of different criticality, and safety requirements? This question focuses on the system's ability to process events in time, distinguish hard and soft temporal constraints, maintain safe behavior, and change the control mode when reactive conditions deteriorate.

- RQ4: How should operator initiative be incorporated into mission-oriented hybrid control with regard to operator state, communication quality, mission validity, and safety? This question addresses the acceptance, deferral, or overriding of operator actions in situations in which human initiative may be beneficial, ambiguous, or risky for mission execution.
- RQ5: How can stable delegation among manual, shared, autonomous, and safety-oriented modes be ensured under channel degradation and mixed conflicting scenarios? This question concerns switching stability, avoidance of excessive oscillations, preservation of mission progress, and transition to more conservative behavior under compound degradations.

#### 4. General composition of a multimodal mission-oriented model of hybrid control

The proposed model defines mission-oriented hybrid control as an ordered composition of policies that transforms heterogeneous events, channel observations, and operator actions into a mission-admissible control mode. The policy sequence fixes the causal order of information transformation from the primary event to the selection of an admissible control mode. A discrete time instant is denoted by  $t_k = k\Delta t$ , where  $k \in N_0$  is the step number and  $\Delta t$  is the dispatcher-step duration. At step  $k$ , the model receives the set of raw events  $E_k$ , asynchronous channel observations  $Y_k$ , the current mission state  $M_k$ , and, when present, operator initiative  $I_k^{op}$ . The complete computational chain is given in (1).

$$(E_k, Y_k, M_k, I_k^{op}) \xrightarrow{P_{sig}} (F_k, \mu_k^{sig}) \xrightarrow{P_{feat}} Z_k^0 \xrightarrow{\Phi_X} (J_k^{cand}, X_k, \Omega_k) \xrightarrow{P_{fus}} A_k \xrightarrow{P_{mis}} D_k^{mis} \xrightarrow{P_{del}} D_k^{del}, \quad (1)$$

where  $P_{sig}$  is the signal-level policy;  $F_k$  is the set of validated event facts;  $\mu_k^{sig}$  is the vector of aggregated signal metrics;  $P_{feat}$  is the feature-level policy;  $Z_k^0$  is the set of base channel evidence; the superscript 0 indicates that the evidence is evaluated against the reference channel requirement before a specific mission transition is selected;  $\Phi_X$  is the semantic interpreter;  $J_k^{cand}$  is the set of transitions whose structural triggers are satisfied;  $X_k$  is the collection of semantic vectors for these transitions;  $\Omega_k$  is the collection of physical and temporal contexts;  $P_{fus}$  is the situational fusion policy;  $A_k$  is the set of candidate-transition estimates;  $P_{mis}$  is the mission policy;  $D_k^{mis}$  is the mission decision;  $P_{del}$  is the delegation and event-routing policy; and  $D_k^{del}$  is the final mode decision.

The base sets of the model are defined in (2).

$$\begin{aligned} C &= \{nav, vis, comm, op, sys, eng\}, \\ Q &= \{HUMAN, SHARED1, SHARED2, SHARED3, AUTO, SAFE\}, \\ V &= \{transit, search, confirm, track, return, landing, landed\}, \end{aligned} \quad (2)$$

where  $C$  is the set of information and resource channels: navigation, visual, communication, operator, system-computational, and energy;  $Q$  is the set of event-processing and routing modes; and  $V$  is the set of mission-graph phases.

The notations *SHARED1*, *SHARED2*, and *SHARED3* define different tables of terminal handlers for event classes. The policy interfaces are formally defined by the mappings in (3).

$$P_{sig}: E_k \times A_k^{sig} \rightarrow F_k \times \mu_k^{sig}, P_{feat}: Y_k \times \mu_k^{sig} \times \Theta_{cal} \rightarrow Z_k^0, \quad (3)$$

$$\begin{aligned}
\Phi_X: M_k \times F_k \times Z_k^0 \times O_k^{mis} &\rightarrow J_k^{cand} \times X_k \times \Omega_k, \\
P_{fus}: X_k \times \Theta_{fus} &\rightarrow A_k, P_{mis}: M_k \times A_k \times \Omega_k \times R^{mis} \rightarrow D_k^{mis}, \\
P_{del}: D_k^{mis} \times F_k \times X_k \times A_k \times I_k^{op} \times R^{del} &\rightarrow D_k^{del},
\end{aligned}$$

where  $\Lambda_k^{sig}$  is the current load of the signal dispatcher;  $\Theta_{cal}$  denotes the identified parameters of the channel models;  $O_k^{mis}$  contains direct mission measurements, including the completed route fraction, coverage, target posterior, duration of valid tracking, and distance to base;  $\Theta_{fus}$  denotes parameters of calibrated fusion metamodels;  $R^{mis}$  denotes declarative mission requirements; and  $R^{del}$  denotes the reliability, timeliness, and safety requirements for the modes. The symbol  $R$  in  $R^{mis}$  and  $R^{del}$  denotes a set of requirements and is unrelated to  $R_{eff}$ .

The current mission state and direct mission measurements are defined in (4).

$$\begin{aligned}
M_k = \langle v_k, \tau_k^v, p_{path,k}, p_{cov,k}, p_{conf,k}, p_{track,k}, p_{ret,k}, p_k^{mis}, \zeta_k, status_k \rangle, \\
O_k^{mis} = \langle \hat{p}_{path,k}, \hat{p}_{cov,k}, \hat{p}_{conf,k}, \hat{p}_{track,k}, \hat{p}_{ret,k}, \hat{p}_k^{mis}, \\
D_{route,k}^{rem}, D_{energy,k}^{rem}, D_{hazard,k}^{rem}, D_{phase,k}^{rem}, d_{home,k} \rangle,
\end{aligned} \tag{4}$$

where  $v_k \in V$  is the active phase;  $\tau_k^v$  is the time spent in that phase;  $p_{path,k}$ ,  $p_{cov,k}$ ,  $p_{conf,k}$ ,  $p_{track,k}$ , and  $p_{ret,k}$  are accumulated within-phase indicators;  $p_k^{mis}$  is overall progress;  $\zeta_k$  is mission strain; and  $status_k$  is the textual status. The hatted quantities in  $O_k^{mis}$  are direct measurements from the current step, from which  $P_{mis}$  updates the accumulated state  $M_{k+1}$ ; they are not produced by the channel-estimation models.

The operator initiative used in (1) has the structure defined in (5).

$$I_k^{op} = \langle type_k^{init}, j_k^{init}, q_k^{init}, e_k^{init}, t_k^{init}, ctx_k^{hmi} \rangle, \tag{5}$$

where  $type_k^{init}$  is the request type;  $j_k^{init}$  is the proposed transition or  $\emptyset$ ;  $q_k^{init}$  is the requested mode;  $e_k^{init}$  is the initiative-carrying event;  $t_k^{init}$  is time; and  $ctx_k^{hmi}$  is the HMI context. The initiative is considered only after  $D_k^{mis}$  has been formed. To eliminate data ambiguity between policies, four composite structures defined in (6) are used.

$$\begin{aligned}
Z_k^{c,0} &= \langle s_k^{c,0}, a_k^c, u_k^{c,0} \rangle, \\
X_k(j) &= \langle x_k(j), h_k(j), n_k(j), r_k(j), e_k(j) \rangle, \\
A_k(j) &= \langle \Sigma_k(j), U_k(j), C_k^{conf}(j), L_j, C_j, A_k^{crit}(j), G_k^{hard}(j) \rangle, \\
D_k^{mis} &= \langle M_{k+1}, J_k^{cand}, J_k^{act}, j_k^i, D_k^{mis}(j_k^i), F_k^{mis}, A_k(j_k^i) \rangle,
\end{aligned} \tag{6}$$

where  $s_k^{c,0}$  is the base probability that the channel requirement is satisfied;  $a_k^c$  denotes data availability and freshness;  $u_k^{c,0}$  is the standard uncertainty of the available support;  $x_k$ ,  $h_k$ ,  $n_k$ ,  $r_k$ , and  $e_k$  are, respectively, the spatial-visual, operator, communication, system-computational, and energy semantic components of transition  $j$ ;  $\Sigma_k(j)$  is the calibrated probability of transition success;  $U_k(j)$  is its standard uncertainty;  $C_k^{conf}(j)$  is diagnostic conflict;  $L_j$  is the list of relevant semantic components;  $C_j$  is the list of physical channels from which they originate;  $A_k^{crit}(j)$  is the

minimum support among the components declared critical;  $G_k^{hard}(j)$  is the non-compensatory admissibility condition;  $J_k^{act}$  is the set of transitions with an open guard;  $j_k^i$  is the selected transition; and  $F_k^{mis}$  is the fallback transition or  $\emptyset$ .

The context  $\Omega_k(j)$  is separated from the probabilistic vector  $X_k(j)$  by (7).

$$\Omega_k(j) = \langle \zeta_k, d_k^{safe}, T_{pipe,k}^{95}, D_{phase,k}^{rem}, D_{route,k}^{rem}, D_{energy,k}^{rem}, D_{hazard,k}, m_k^E(j), G_k^{sig,H} \rangle, \quad (7)$$

Vector  $\zeta_k = \langle lat_k, lon_k, o_k^{risk}, w_k^{op}, T_k^{op}, T_k^{lat}, T_k^{jit}, p_k^{loss}, B_k, U_k^{cpu}, T_k^{inf}, F_k^{fps} \rangle$  contains raw physical and operational quantities retained for context and traceability but not treated as additional fusion components. Here,  $lat_k$  and  $lon_k$  are coordinates;  $o_k^{risk}$  is the obstacle-risk estimate;  $w_k^{op}$  and  $T_k^{op}$  are operator workload and response time;  $T_k^{lat}, T_k^{jit}, p_k^{loss}$ , and  $B_k$  are communication latency, jitter, loss ratio, and bandwidth;  $U_k^{cpu}$  is computational load; and  $T_k^{inf}$  and  $F_k^{fps}$  are inference latency and effective frame rate.  $d_k^{safe}$  is the margin to the safety-envelope boundary;  $T_{pipe,k}^{95}$  is the 95th percentile of the system-pipeline time;  $D_{phase,k}^{rem}, D_{route,k}^{rem}, D_{energy,k}^{rem}$ , and  $D_{hazard,k}$  are residual temporal budgets;  $m_k^E(j)$  is the energy reserve; and  $G_k^{sig,H}$  is the result of the signal-level hard check. Components  $\Omega_k(j)$  are used as physical constraints in  $P_{mis}$  and  $P_{del}$ . Raw quantities have a source index or physical unit and are introduced only into  $E_k, Y_k, O_k^{mis}$ , or  $I_k^{op}$ . The superscript 0 is used only for base evidence  $Z_k^{c,0}$  that has not yet been bound to a transition. After  $\Phi_X$ , all probabilistic quantities must have transition index  $j$ . After  $P_{fus}$ , they are stored within  $A_k(j)$ . Symbol  $D$  denotes a time budget or deadline;  $T$  denotes realized or predicted time;  $p$  or  $s$  denotes probability;  $u$  or  $U$  denotes standard uncertainty; and  $G$  denotes a binary non-compensatory condition.

## 5. Experimental configuration and scenario matrix

The experimental configuration is built around an external mission description. The graph topology, geometric goals, temporal constraints, priorities, guards, and fallback states are specified in JSON markup. After loading and structural validation, the hybrid-control policies receive the active state and candidate transitions. All experimental data are synthetic. At each 0.5s simulation step, the simulator updates the platform and mission state and generates observations for the navigation, visual, communication, operator, system-computational, and energy channels. Scenario parameters determine the imposed channel degradations, while stochastic perturbations provide variability between runs. The channel and fusion parameters were obtained from a separate synthetic calibration set and remained fixed during evaluation. Scenarios V1 and V2 additionally use controlled injections of a proximity hazard and an actuator fault. No real-sensor, platform-in-the-loop, or hardware-in-the-loop data were used. The configuration contains 14 states and 23 explicitly declared transitions. The main route covers the preflight check, takeoff, flight to the search area, search, target tracking, return, landing, and the terminal LANDED state. Route Replanning, Search Repositioning, and Target Reacquisition form local adaptation branches. Compact Telemetry is a separate communication-adaptation state. Safe Loiter and Aborted implement the safety branch of this configuration. Figure 1 presents one executable graph in two coordinated panels. Panel (a) shows the main topology and local adaptation states without intersecting cross-cutting edges, whereas panel (b) details transitions to Compact Telemetry, the global energy/deadline rule, and the safety transition with source  $* = S \setminus \{safe_{loiter}, landed, aborted\}$ . The semantics of most transitions are shown directly in Figure 1. Additional clarification is required for the transitions that terminate communication

adaptation. Transition T16 directs the system from Compact Telemetry to Return Home when persistent link degradation, the remaining time budget, or the available energy reserve no longer permits safe continuation of the current task. After communication is restored, T22 returns the system to Coverage Search if the search context remains valid, whereas T23 resumes Target Tracking only when both communication and visual evidence are sufficient. Safe Loiter remains the fallback behavior for all three transitions. Their exact deadlines, priorities, and fallback parameters are retained in the machine-readable JSON mission configuration and are used directly by the execution policies.

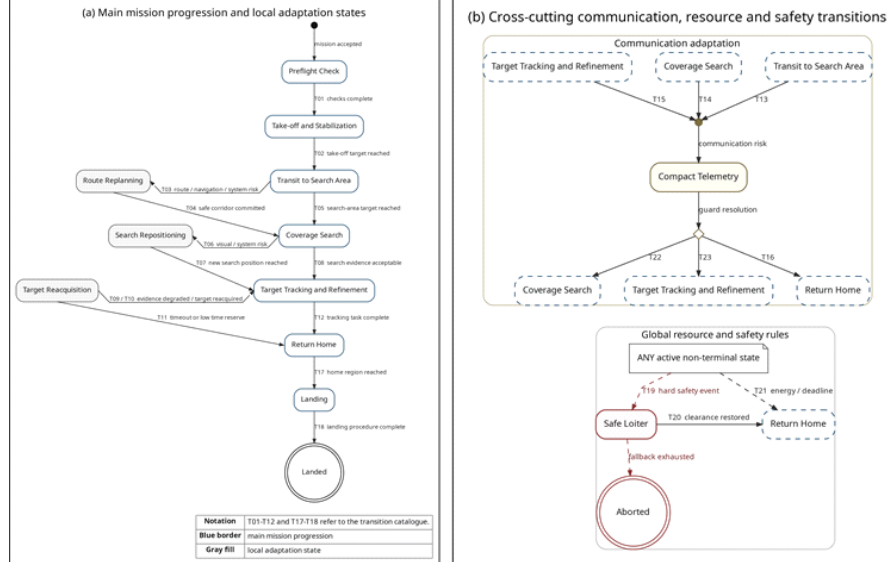


**Figure 1:** Declarative graph of the multistage mission: (a) main sequence and local adaptation states; (b) cross-cutting communication, resource, and safety transitions.

The primary scenario set  $S_{main} = \{S0, \dots, S7\}$  verifies nominal execution, isolated degradations, and a mixed-initiative conflict. A separate set  $S_{safe} = \{V1, V2\}$  is used to validate safety transitions. The experimental scenarios are described in Table 1.

**Table 1**

Scenarios of the Multi-Run Simulation Experiment for Unmanned-Mission Execution

| ID | Scenario                       | Graph aspect under test                                    |
|----|--------------------------------|------------------------------------------------------------|
| S0 | Nominal mission                | Reference execution of the nominal and optional branches   |
| S1 | Communication degradation      | T13-T16/T22-T23 communication-adaptation branch            |
| S2 | Operator overload              | Shared-routing and global return rule T21                  |
| S3 | Computational overload         | System-risk branch T03-T04 and processing-time suitability |
| S4 | Visual occlusion               | Search/tracking evidence and adaptive visual branches      |
| S5 | Energy constraint              | Energy evidence and resource return rule T21               |
| S6 | Mixed-initiative conflict      | Operator initiative within graph-constrained routing       |
| S7 | Compound degradation           | Joint channel degradation and fallback competition         |
| V1 | Proximity hazard during search | Explicit safety transition T19 during the search segment   |
| V2 | Actuator fault during return   | Explicit safety transition T19 during return               |

Ten runs with independent seeds were performed for each scenario. The base simulation step is 0.5 s; however, execution terminates earlier upon reaching LANDED or ABORTED, or upon reaching the 75-second mission limit. At every step, the active state, selected transition, its criticality,  $\Sigma$ ,  $R_{eff}$ , control mode, fallback, channel estimates, events, physical progress, and safety indicators are logged.

## 6. Experimental evaluation and performance assessment

The results are analyzed in three interrelated dimensions: mission completion, activation of the declarative graph edges, and phase-dependent control-mode selection. The nominal scenario, visual occlusion, energy constraint, and V2 ended in landing in every run (Figure 2). The partial success of the other scenarios is explained by reaching the time limit in Landing or Return. Final progress is therefore interpreted together with `terminal_phase`: a value close to one is not treated as LANDED unless the landing guard has been satisfied.

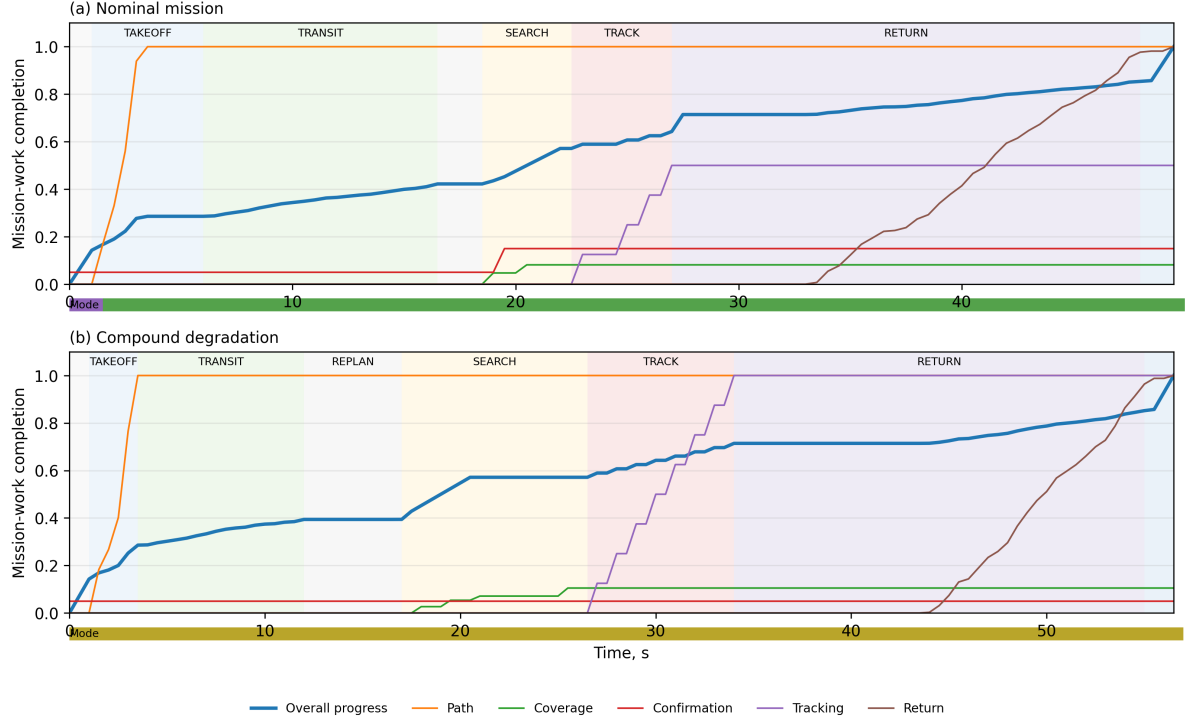


**Figure 2:** Mission-conditioned execution dynamics: phases, physical progress components, and control modes in the nominal (a) and compound (b) scenarios.

Mean reactivity decreases from 0.624 in S0 to 0.512 in S7, while the fifth percentile decreases from 0.345 to 0.133. This decrease does not automatically trigger SAFE: the SAFE share is zero in S0–S7 because these scenarios contain no hard safety trigger. Communication degradation and operator overload increase the share of shared routing, whereas the nominal and system-computational scenarios are executed predominantly in AUTO (Table 2). V1 triggered SAFE in nine of ten runs: one run did not reach the state in which the injection was scheduled before the time limit expired. Recovery through T20 was recorded in all nine episodes. V2 reached active RETURN in every run; therefore, T19 and T20 were activated ten times. The entry latency in both scenarios equals one dispatcher step under the adopted discretization, and no safety-envelope violations were recorded. Panel (b) of Figure 3, together with Figure 2, shows that the current graph state determines the available outgoing transitions, while the selected transition determines the relevant channel evidence and admissible event routes. Thus, mission context affects delegation before modes are compared. In all scenarios, the route component and return are close to full completion, whereas confirmation and tracking are substantially more sensitive to channel quality.

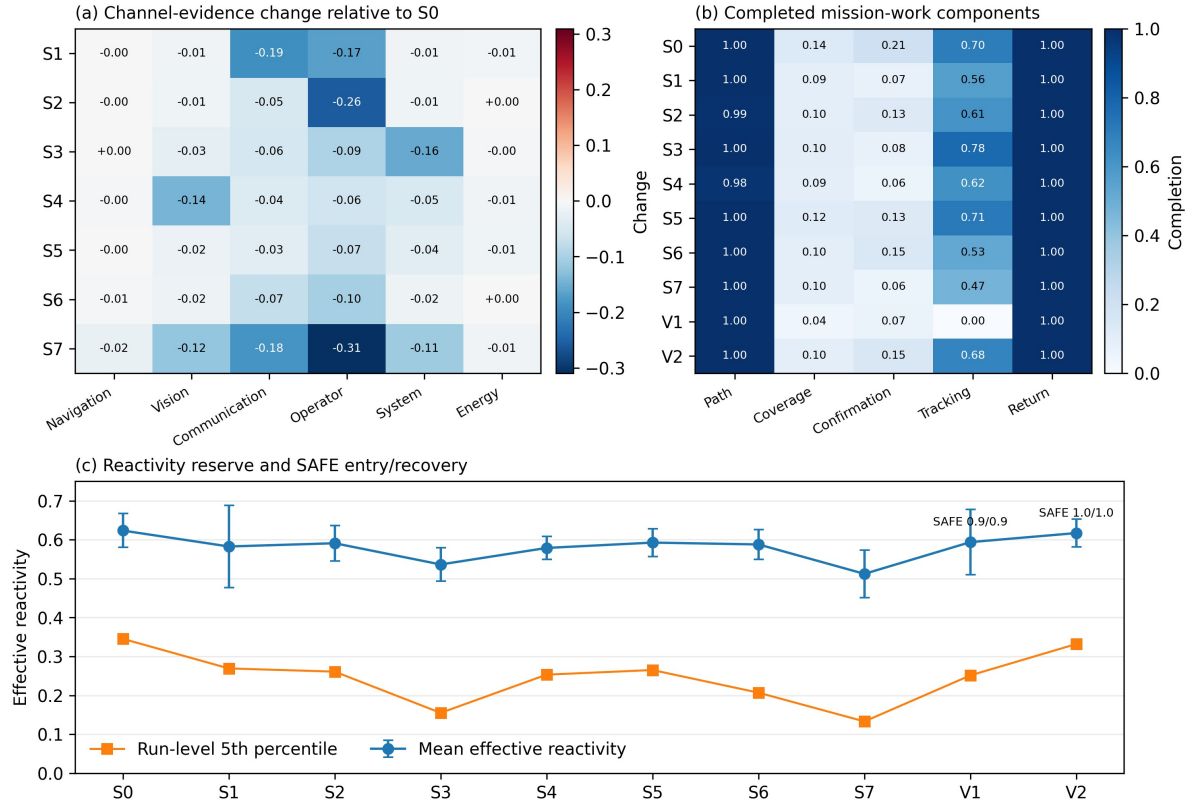


**Figure 3:** Cross-scenario evidence for multimodality, mission progress, and reactive suitability.

**Table 2**

Reactivity and Mode Shares by Scenario

| S  | Mean $R_{eff}$ | P05 $R_{eff}$ | Mean $\Sigma$ | Shared | Auto  | SAFE  | Mean Switches |
|----|----------------|---------------|---------------|--------|-------|-------|---------------|
| S0 | 0.624          | 0.345         | 0.819         | 0.321  | 0.679 | 0.000 | 0.7           |
| S1 | 0.583          | 0.269         | 0.813         | 0.510  | 0.490 | 0.000 | 0.7           |
| S2 | 0.591          | 0.261         | 0.816         | 0.710  | 0.290 | 0.000 | 0.3           |
| S3 | 0.536          | 0.155         | 0.798         | 0.320  | 0.680 | 0.000 | 0.7           |
| S4 | 0.579          | 0.253         | 0.813         | 0.612  | 0.388 | 0.000 | 0.4           |
| S5 | 0.593          | 0.265         | 0.812         | 0.319  | 0.681 | 0.000 | 0.7           |
| S6 | 0.588          | 0.207         | 0.815         | 0.612  | 0.388 | 0.000 | 0.4           |
| S7 | 0.512          | 0.133         | 0.797         | 0.514  | 0.486 | 0.000 | 0.5           |
| V1 | 0.594          | 0.251         | 0.810         | 0.493  | 0.471 | 0.035 | 2.2           |
| V2 | 0.617          | 0.332         | 0.816         | 0.243  | 0.721 | 0.036 | 2.7           |

The terminal state is considered separately from integral progress: a progress value close to one does not replace the fact that the transition to LANDED has been executed. Panel (c) of Figure 3 shows a reduction in the reactive margin without conflating degradation with an emergency mode. No hard-deadline misses, soft violations, or safety-envelope violations occur throughout the series. SAFE is activated only in V1 and V2; the number of entries equals the number of recoveries for the episodes that actually occurred. RQ3 is therefore supported within the simulation setting. Panel (a) of Figure 4 summarizes conflicting operator requests in S6 and S7. In S6, 57 explicit decisions were recorded: 54 override and 3 defer; in S7, 13 decisions were recorded: 12 override and 1 defer. The current stochastic series does not include a separate compatible-initiative scenario; therefore, the absence of accept decisions should not be interpreted as evidence that the corresponding branch is inoperative. The experiment confirms filtering of conflicting initiative, but RQ4 is supported only partially overall. Full validation requires an additional scenario in which the operator request is structurally and statistically admissible.

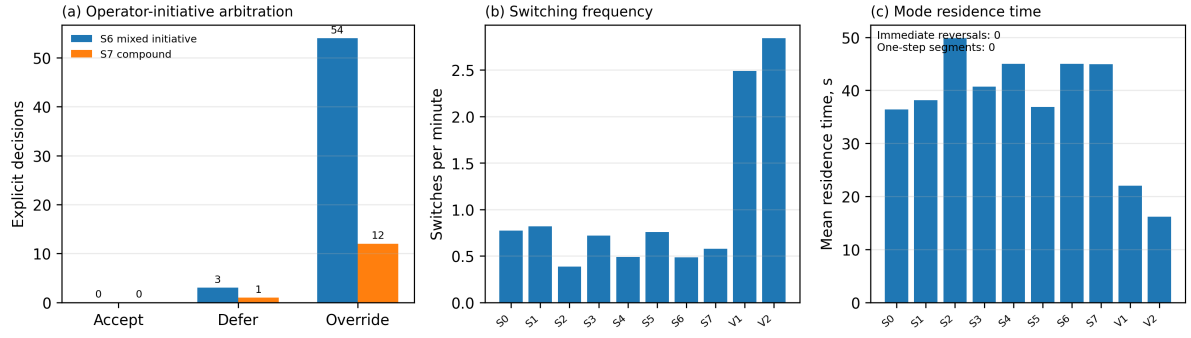


**Figure 4:** Operator-initiative arbitration and delegation stability: decisions, switching frequency, and residence time.

Panels (b)–(c) of Figure 4 show that, in S0–S7, the mean number of switches is 0.3–0.7 per run and the mean dwell time is approximately 36–50 s. The number of switches is naturally higher in V1 and V2 because each safety episode adds an entry into SAFE and a recovery. Across all ten scenarios, the number of immediate reverse switches and one-step mode segments is zero. Thus, statistical mode comparison and residence time suppress oscillations without blocking meaningful safety transitions. RQ5 is therefore experimentally supported.

## 7. Conclusions

This article presents an integrated mission-oriented model of hybrid control for a robotic system in which event-driven reactivity, multimodal channel evidence, semantic state, the mission graph, operator initiative, safety constraints, and the delegation policy are combined into a single sequence of formal policies. The proposed model defines shared control through event routing among the signal, feature, mission, operator, autonomous, and safety processing levels. Consequently, the control mode is determined by the event class, active mission transition, time budget, channel state, operator availability, reactive suitability, and mission-execution safety conditions. This constitutes the basis of the study's scientific novelty.

An additional element of novelty is the decomposition of the robotic system's multimodal state into channel evidence containing a channel-state estimate, data freshness, and evidence uncertainty. This representation preserves the physical meaning of the navigation, visual, communication, operator, system-computational, and energy channels and enables their subsequent semantic interpretation in the context of the active mission phase. On this basis, the concept of mode-dependent effective reactivity  $R_{eff}(j, q, k)$  is introduced; it determines the admissibility of a specific control mode for the active mission transition with consideration of event deadlines, routing, operator or autonomous participation, and safety/fallback logic. The proposed model provides a consistent basis for analyzing mission progress, reactive admissibility, operator intervention, channel conflict, and delegation stability under degradation of individual modalities. The obtained results are simulation-based. Phase-level analysis confirms the central proposition: the significance of a channel is determined by the transition context. Communication and operator evidence are most critical for CONFIRM; visual evidence is most critical for SEARCH and TRACK; system-computational availability contributes to all reactive transitions; and energy evidence determines whether the plan continues or an energy return is initiated. Consequently, the mode changes with mission semantics even under a similar overall channel state.

The multi-run simulation series confirmed traversal of the main unmanned-mission test route, use of the route-replanning and compact-telemetry branches, and correct entry into Safe Loiter with recovery through T20 in the controlled safety scenarios. Transitions T06–T11 and T23 did not receive scenario coverage. For these transitions, only syntactic correctness, computability, and local

executability of the guard conditions were confirmed using specially constructed test states. So, the scenario-based evaluation fully supported RQ3 and RQ5 within the adopted simulation setting, whereas RQ1 and RQ4 received only partial support. RQ2 was implemented at the structural and logical levels of the model, but its complete experimental validation requires scenario coverage of all transitions. Hardware-level validation was not performed at this stage because it requires platform-specific integration and synchronization of the six information channels, followed by re-identification of their measurement and timing models. PIL/HIL evaluation is therefore considered a subsequent validation stage. This study formalizes and evaluates the hybrid-control model itself, whereas its progression toward hybrid intelligence requires the integration of intelligent multimodal data-processing methods, including perception-data processing, into the semantic interpretation, situational fusion, and mission-decision policies. Further work should include targeted scenarios for these branches, platform-in-the-loop and hardware-in-the-loop re-identification, and automatic generation of SCXML/UPPAAL representations from the same mission description file. The practical value of the model stems from its use as a consistent specification for the development and validation of mission-oriented hybrid-control software. For each mission phase, it helps identify critical information channels, define how operator initiative should be handled, establish mode-switching rules, and specify the conditions for activating fallback or safety behavior. Scenario-based simulation of degradations, conflicts, and timing violations supports the identification of undesirable decision causes and the assessment of configuration robustness. Delegation policies, timing budgets, and event-routing rules can therefore be adjusted to a particular mission and platform before field trials.

## Declaration on Generative AI

During the preparation of this manuscript, the authors made use of AI-based tools for grammar refinement and proofreading of the text. Following the use of these tools, the authors carefully reviewed and manually revised the material and accept full responsibility for its content.

## References

- [1] D. Dellermann, P. Ebel, M. Söllner, J. M. Leimeister, Hybrid intelligence, *Business & Information Systems Engineering* 61(5) (2019) 637-643. DOI: 10.1007/s12599-019-00595-2.
- [2] A. Yarmilko, S. Marchenko, Conceptual model of hybrid control of robotic systems with delegation of authority, in: *Conference Proceedings of the VIII International Scientific-Practical Conference "Information Technology for Education, Science and Technics" (ITEST-2026)*, ChSTU, Cherkasy, Ukraine, 2026, pp. 420-422. DOI: 10.5281/zenodo.21843513.
- [3] J. M. Campbell, Psychological effects on UAV operators and proposed mitigation strategies to combat PTSD, Master's thesis, Naval Postgraduate School, Monterey, CA, 2021. URL: <https://apps.dtic.mil/sti/trecms/pdf/AD1150884.pdf>.
- [4] J. W. Kieffer, S. Stahlman, Mental health disorders, behavioral health problems, fatigue and sleep outcomes in remotely piloted aircraft/manned aircraft pilots, and remotely piloted aircraft crew, U.S. Air Force, 1 October 2003-30 June 2019, *Medical Surveillance Monthly Report*, 28(8) (2021) 14-21. PMID: 34622757.
- [5] M. A. Goodrich, A. C. Schultz, Human-robot interaction: a survey, *Foundations and Trends in Human-Computer Interaction* 1(3) (2007) 203-275. DOI: 10.1561/11000000005.
- [6] R. Parasuraman, T. B. Sheridan, C. D. Wickens, 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. DOI: 10.1109/3468.844354.

- [7] S. Singh, E. Rantanen, J. Heard, Human-robot teaming: A comprehensive survey on collaboration, communication and cognition, *ACM Transactions on Human-Robot Interaction* 15(2) 43, 2026. DOI: 10.1145/3776548.
- [8] A. D. Dragan, S. S. Srinivasa, A policy-blending formalism for shared control, *The International Journal of Robotics Research* 32(7) (2013) 790-805. DOI: 10.1177/0278364913490324.
- [9] S. Javdani, H. Admoni, S. Pellegrinelli, S. S. Srinivasa, J. A. Bagnell, Shared autonomy via hindsight optimization for teleoperation and teaming, *The International Journal of Robotics Research* 37(7) (2018) 717-742. DOI: 10.1177/0278364918776060.
- [10] S. Reddy, A. D. Dragan, S. Levine, Shared autonomy via deep reinforcement learning, *Robotics: Science and Systems XIV* (2018). DOI: 10.15607/RSS.2018.XIV.005.
- [11] S. Shaldambayeva, K. Zhumakhanova, A. Sandygulova, M. Rubagotti, A. Shintemirov, Partially Observable Markov Decision Processes in Shared Autonomy Applications: A Survey, *IEEE Access* 13 (2025) 186937-186951. DOI: 10.1109/ACCESS.2025.3625601.
- [12] Lee J. et al. SPIRIT: Perceptive shared autonomy for robust robotic manipulation under deep learning uncertainty, *arXiv:2603.05111*, 2026. DOI: 10.48550/arXiv.2603.05111.
- [13] C. Zhou, J. Yang, D. J. Weber, Z. Erickson, SAPS: Shared autonomy for policy steering by blending teleoperation with a pretrained VLA, *arXiv:2606.15568*, 2026. DOI: 10.48550/arXiv.2606.15568.
- [14] M. Luckcuck, M. Farrell, L. A. Dennis, C. Dixon, M. Fisher, Formal specification and verification of autonomous robotic systems: A survey, *ACM Computing Surveys* 52(5) (2019) 100. DOI: 10.1145/3342355.
- [15] G. Filippone, S. Pettinari, P. Pelliccione, Formalisms for robotic mission specification and execution: A comparative analysis, *arXiv:2603.15427*, 2026. DOI: 10.48550/arXiv.2603.15427.
- [16] C. Menghi, C. Tsigkanos, P. Pelliccione, C. Ghezzi, T. Berger, Mission specification patterns for mobile robots, *IEEE Transactions on Software Engineering* 50(11) (2024) 2992-3022. DOI: 10.1109/TSE.2024.3453601.
- [17] F. Cladera, Z. Ravichandran, J. Hughes, V. Murali, C. Nieto-Granda, M. A. Hsieh, Air-ground collaboration for language-specified missions in unknown environments, *IEEE Transactions on Field Robotics* 2 (2025) 626-642. DOI: 10.1109/TFR.2025.3584019.
- [18] H. Qian, M. Wang, M. Zhu, H. Wang, A review of multi-sensor fusion in autonomous driving, *Sensors* 25(19) (2025) 6033. DOI: 10.3390/s25196033.
- [19] X. Han, S. Chen, Z. Fu, Z. Feng, L. Fan, D. An, C. Wang, L. Guo, W. Meng, X. Zhang, R. Xu, S. Xu, Multimodal fusion and vision-language models: A survey for robot vision, *Information Fusion* 126(PB) (2025) 103327. DOI: 10.1016/j.inffus.2025.103327.
- [20] S. J. Julier, J. K. Uhlmann. Unscented filtering and nonlinear estimation, *Proceedings of the IEEE* 92(3) (2004) 401-422. DOI: 10.1109/JPROC.2003.823141.
- [21] A. Doucet, N. de Freitas, N. Gordon, An introduction to sequential Monte Carlo methods, in: *Sequential Monte Carlo methods in practice*, Springer, New York, 2001, pp. 3-14. DOI: 10.1007/978-1-4757-3437-9\_1.

- [22] S. J. Julier, J. K. Uhlmann, General decentralized data fusion with covariance intersection. Handbook of multisensor data fusion, CRC Press, Boca Raton, 2001.
- [23] K. Sentz, S. Ferson, Combination of evidence in Dempster-Shafer theory, Report SAND2002-0835, Sandia National Laboratories, Albuquerque, NM, 2002. DOI: 10.2172/800792.
- [24] Z. Yin, J. Zhang, Recognition of mental workload levels under complex human-machine collaboration by using physiological features and adaptive support vector machines, IEEE Transactions on Human-Machine Systems 44(2) (2014) 200-214. DOI: 10.1109/THMS.2013.2295853.
- [25] R. L. Charles, J. Nixon, Measuring mental workload using physiological measures: A systematic review, Applied Ergonomics 74 (2019) 221-232. DOI: 10.1016/j.apergo.2018.08.028.
- [26] A. Bussolan, S. Baraldo, O. Avram, P. Urcola, L. Montesano, L. M. Gambardella, A. Valente, MultiPhysio-HRC: A multimodal physiological signals dataset for industrial human-robot collaboration, Robotics 14(12) (2025) 184. DOI: 10.3390/robotics14120184.
- [27] R. Wu, H. Wang, H.-T. Chen, G. Carneiro, Deep multimodal learning with missing modality: A survey, arXiv:2409.07825, 2024. DOI: 10.48550/arXiv.2409.07825.
- [28] C. L. Liu, J. W. Layland, Scheduling algorithms for multiprogramming in a hard-real-time environment, Journal of the ACM 20(1) (1973) 46-61. DOI: 10.1145/321738.321743.
- [29] J. P. Lehoczky, L. Sha, Y. Ding, The rate monotonic scheduling algorithm: Exact characterization and average case behavior, in: Proceedings of the IEEE Real-Time Systems Symposium (1989) 166-171. DOI: 10.1109/REAL.1989.63567.
- [30] G. Bernat, A. Burns, A. Liamosi, Weakly hard real-time systems, IEEE Transactions on Computers 50(4) (2001) 308-321. DOI: 10.1109/12.919277.
- [31] K. Salamun, I. Pavić, H. Džapo, I. Čuljak, Weakly hard real-time model for control systems: A survey, Sensors 23(10) (2023) 4652. DOI: 10.3390/s23104652.
- [32] O. Liubimov, I. Turkin, V. Cheranovskiy, L. Volobuieva, UAV mission computer operation mode optimization focusing on computational energy efficiency and system responsiveness, Computation 12(12) (2024) 235. DOI: 10.3390/computation12120235.
- [33] K.-C. Hsu, H. Hu, J. F. Fisac, The safety filter: A unified view of safety-critical control in autonomous systems, Annual Review of Control, Robotics, and Autonomous Systems 7 (2024) 47-72. DOI: 10.1146/annurev-control-071723-102940.
- [34] K. Garg, J. Usevitch, J. Breeden, M. Black, D. Agrawal, H. Parwana, D. Panagou, Advances in the theory of control barrier functions: Addressing practical challenges in safe control synthesis for autonomous and robotic systems, Annual Reviews in Control 57 (2024) 100945. DOI: 10.1016/j.arcontrol.2024.100945.
- [35] J. D. Schierman, M. D. DeVore, N. D. Richards, M. A. Clark, Runtime assurance for autonomous aerospace systems, Journal of Guidance, Control, and Dynamics 43(12) (2020) 2205-2218. DOI: 10.2514/1.G004862.