

A Survey of Decision-Making and Planning Methods for Self-Driving Vehicles

Finley Holt

2025-12-25

Table of contents

1 A Survey of Decision-Making and Planning Methods for Self-Driving Vehicles	2
1.1 Overview	2
1.2 Decision Impact for Flyby-F11	2
1.2.1 ADOPT - High Confidence	3
1.2.2 CONSIDER - Needs Validation	3
1.2.3 AVOID - Evidence Against	4
1.2.4 INVESTIGATE - Open Questions	4
1.2.5 Approach Comparison Matrix	5
1.2.6 Industry Lessons	5
1.3 Knowledge-Driven Decision and Planning Methods	6
1.3.1 Decision-Making Processes	7
1.3.2 Planning Processes	8
1.4 Data-Driven Decision and Planning Methods	8
1.4.1 1. Imitation Learning	8
1.4.2 2. Reinforcement Learning (RL)	8
1.4.3 3. Inverse Reinforcement Learning (IRL)	9
1.5 Hybrid Decision and Planning Methods	9
1.5.1 Motivation for Hybrid Approaches	9
1.5.2 Implementation Approaches	9
1.5.3 Challenges in Hybrid Development	10
1.6 Comparative Analysis: Knowledge vs. Data-Driven Approaches	10
1.6.1 Knowledge-Driven Limitations	10
1.6.2 Data-Driven Limitations	10
1.7 Experimental Platforms and Validation	11
1.7.1 Datasets	11
1.7.2 Simulation Platforms	11
1.8 Current Industry Deployment	11
1.8.1 Commercial Applications	11
1.8.2 Deployment Challenges	11
1.9 Challenges and Future Perspectives	12
1.9.1 Current Challenges	12
1.9.2 Future Research Directions	12
1.10 Relevance to Drone Autonomy	12

1.10.1	Direct Applicability	12
1.10.2	Drone-Specific Considerations	13
1.10.3	Integration with Flyby-F11 Approach	13
1.10.4	Alternative Approaches for Flyby-F11	14
1.11	Key Findings Summary	14
1.11.1	Methodological Insights	14
1.11.2	Technical Contributions	14
1.11.3	Implications for Drone Autonomy	15
1.12	Conclusion	15
1.13	References	15

1 A Survey of Decision-Making and Planning Methods for Self-Driving Vehicles

Authors: Jun Hu, Yuefeng Wang, Shuai Cheng, Jinghan Xu, Ningjia Wang, Bingjie Fu, Zuotao Ning, Jingyao Li, Hualin Chen, Chaolu Feng, Yin Zhang

Source: <https://www.frontiersin.org/journals/neurorobotics/articles/10.3389/fnbot.2025.1451923/full>

DOI: 10.3389/fnbot.2025.1451923

Year: 2025

Journal: Frontiers in Neurorobotics

Citation: Hu J, Wang Y, Cheng S, Xu J, Wang N, Fu B, Ning Z, Li J, Chen H, Feng C and Zhang Y (2025) A survey of decision-making and planning methods for self-driving vehicles. *Front. Neurorobot.* 19:1451923.

1.1 Overview

This comprehensive survey provides a systematic review of decision-making and planning algorithms in autonomous driving, covering both knowledge-driven and data-driven approaches, as well as emerging hybrid methods. The paper is particularly significant for neurorobotics applications, as it addresses how neural network-based approaches combined with robotic systems can improve autonomous vehicle decision-making and planning.

The autonomous driving decision-making and planning problem is divided into two key phases:

1. **Behavior Decision-Making:** Addresses responses to temporary events such as abnormal driving behaviors of other vehicles, sudden pedestrian crossings, and emergency vehicle avoidance. The decision system must exhibit high adaptability and predictive capability for potential future scenarios.
2. **Motion Planning:** Generates detailed trajectories based on current vehicle state and behavior decision outputs, ensuring smoothness and comfort while adhering to dynamic constraints such as speed and acceleration.

1.2 Decision Impact for Flyby-F11

This survey provides critical guidance for Flyby-F11's autonomous navigation architecture by systematically evaluating knowledge-driven, data-driven, and hybrid approaches to decision-making

and planning. The findings directly inform our technology choices and implementation strategy.

1.2.1 ADOPT - High Confidence

- 1. Hybrid Architecture (Knowledge + Data-Driven) - Evidence:** Survey identifies hybrid methods as “compelling direction for future research” and most promising approach - **Application:** Integrate ontology-based knowledge with reinforcement learning for Flyby-F11 - **Benefits:** Balances safety guarantees with adaptive learning, addresses limitations of pure approaches - **Implementation:** Use ontology to constrain and guide RL exploration while enabling optimization
- 2. MDPs/POMDPs for Uncertainty Modeling - Evidence:** Widely validated in autonomous driving for handling partial observability and hidden road users - **Application:** Model decision-making under sensor occlusion, GPS-denied navigation - **Benefits:** Structured framework for probabilistic reasoning, handles hidden states effectively - **Implementation:** Use POMDPs for mission-level planning when environment information is incomplete
- 3. Hierarchical Planning Architecture - Evidence:** Successful separation of high-level strategic planning from low-level control - **Application:** Ontology-based high-level mission planning, RL for trajectory optimization - **Benefits:** Clear separation of concerns, modularity, interpretability at each level - **Implementation:** Mission planner (ontology) generates goals, local planner (RL) executes trajectories
- 4. Sampling-Based Motion Planning (RRT, PRM) - Evidence:** Proven effective in high-dimensional spaces without explicit discretization - **Application:** Real-time obstacle avoidance in 3D environments - **Benefits:** Scales to drone’s 6-DOF state space, handles dynamic constraints - **Implementation:** Use for local trajectory generation in cluttered environments
- 5. Simulation-Based Validation - Evidence:** Industry standard for safe testing before real-world deployment - **Application:** PX4 SITL + Gazebo for algorithm development and scenario testing - **Benefits:** Safe exploration of edge cases, rapid iteration, cost reduction - **Implementation:** Already deployed in project-drone, extend to Flyby-F11 mission scenarios

1.2.2 CONSIDER - Needs Validation

- 1. Game-Theoretic Multi-Agent Approaches - Rationale:** Relevant if Flyby-F11 operates in multi-UAV scenarios or coordinated swarms - **Questions:** Is multi-agent interaction a primary mission requirement? Computational overhead acceptable? - **Validation Needed:** Test in simulation with multiple drone scenarios, measure coordination benefits - **Risk:** Added complexity may not justify benefits for single-drone missions
- 2. Inverse Reinforcement Learning (IRL) - Rationale:** Could help infer reward functions from high-level mission objectives - **Questions:** Do we have sufficient expert demonstrations? Can ontology provide better prior than IRL? - **Validation Needed:** Compare ontology-derived rewards vs. IRL-inferred rewards in simulation - **Risk:** Computational cost, potential bias from incomplete expert data
- 3. End-to-End Deep Learning - Rationale:** Tesla’s approach shows promise but lacks interpretability - **Questions:** Can we maintain safety guarantees with E2E learning? Edge compute constraints? - **Validation Needed:** Benchmark against modular pipeline on Jetson Orin NX - **Risk:** Black-box behavior, difficult verification, sample inefficiency

4. Continuous/Lifelong Learning - Rationale: Adapt to deployment environment over time - **Questions:** Can we prevent catastrophic forgetting? Safety during online adaptation? - **Validation Needed:** Test incremental learning with safety constraints active - **Risk:** Stability concerns, requires robust monitoring and rollback mechanisms

1.2.3 AVOID - Evidence Against

- 1. Pure Rule-Based Systems - Evidence:** Survey highlights scaling challenges, inability to handle novel situations, conservative decision-making - **Weakness:** Cannot adapt to GPS-denied, communications-limited environments Flyby-F11 will encounter - **Alternative:** Use rules as constraints within hybrid framework, not sole decision mechanism - **Exception:** May be appropriate for critical safety checks and hard limits
- 2. Pure Imitation Learning - Evidence:** Limited by training data quality, distribution shift problems, poor generalization - **Weakness:** Requires extensive expert demonstrations for drone missions we haven't flown yet - **Alternative:** Use ontology + RL instead, which can explore beyond demonstrated behaviors - **Exception:** May augment training with limited expert data if available
- 3. Pure Black-Box Reinforcement Learning - Evidence:** Poor interpretability, difficult safety guarantees, sample inefficiency - **Weakness:** Unacceptable for safety-critical drone operations, especially in NDAA-compliant missions - **Alternative:** Constrain RL with ontology to ensure interpretability and safety bounds - **Exception:** May use for non-critical optimization (e.g., energy efficiency tuning)
- 4. Overly Complex State Transition Models - Evidence:** Computational resource constraints mentioned as key challenge - **Weakness:** Jetson Orin NX has limited compute; must prioritize real-time performance - **Alternative:** Balance model expressiveness with computational efficiency - **Exception:** Offline planning may use more complex models if real-time execution is lightweight

1.2.4 INVESTIGATE - Open Questions

- 1. Optimal Balance Between Knowledge and Data Components - Question:** What percentage of decision-making should be ontology-driven vs. RL-optimized? - **Approach:** Systematic experiments varying constraint tightness in simulation - **Metrics:** Safety compliance, mission success rate, adaptation speed, computational cost - **Timeline:** Investigate during Flyby-F11 simulation development phase
- 2. Sample Efficiency with Ontology Constraints - Question:** How much does ontology-guided exploration improve RL sample efficiency? - **Approach:** Compare constrained vs. unconstrained RL training convergence - **Metrics:** Episodes to reach performance threshold, total training time, safety violations - **Timeline:** Early experimentation in project-drone, validate on Flyby-F11
- 3. Transferability Across Mission Types - Question:** Can ontology + RL trained on one mission generalize to different mission profiles? - **Approach:** Train on inspection missions, test on delivery/surveillance missions - **Metrics:** Transfer learning performance, retraining requirements, mission success rates - **Timeline:** After initial mission-specific validation
- 4. Real-Time Performance on Edge Hardware - Question:** Can hybrid architecture meet <100ms decision cycle on Jetson Orin NX? - **Approach:** Profile ontology reasoning + RL inference on target hardware - **Metrics:** Latency percentiles, CPU/GPU utilization, memory footprint - **Timeline:** Hardware-in-the-loop testing when Flyby-F11 platform available

5. Explainability for Mission Assurance - Question: Can we generate human-understandable explanations for autonomous decisions? - **Approach:** Leverage ontology structure to provide semantic reasoning traces - **Metrics:** Operator comprehension, audit trail completeness, failure diagnosis utility - **Timeline:** Integrate with mission monitoring dashboard development

1.2.5 Approach Comparison Matrix

Dimension	Knowledge-Driven	Data-Driven	Hybrid (Ontology + RL)
Safety Guarantees	Strong - Verifiable rules and constraints	Weak - No formal bounds, “black box” behavior	Strong - Ontology enforces safety, RL optimizes within bounds
Adaptability	Weak - Cannot handle novel situations	Strong - Learns from experience, discovers new strategies	Strong - RL adapts while respecting ontology constraints
Sample Efficiency	N/A - No learning required	Weak - Requires vast training data	Medium - Ontology guides exploration, reduces search space
Interpretability	Strong - Explicit rules, traceable logic	Weak - Neural networks lack transparency	Medium-Strong - Ontology provides semantic explanations
Generalization	Weak - Brittle in unexpected scenarios	Medium - Limited by training distribution	Strong - Ontology enables reasoning, RL handles variations
Development Cost	High - Extensive expert knowledge required	High - Large dataset collection, labeling	Medium - Ontology effort upfront, RL training automated
Computational Cost	Low - Simple rule evaluation	High - Neural network inference	Medium - Ontology reasoning + lightweight RL policy
Maintenance	High - Manual rule updates required	Medium - Retraining with new data	Low - Ontology updates logical, RL adapts automatically
Real-Time Performance	Excellent - Fast rule matching	Variable - Depends on model size	Good - Optimized ontology + efficient RL policy
Verification & Validation	Straightforward - Test rule coverage	Difficult - Exhaustive scenario testing	Medium - Verify ontology, validate RL in constrained space

Key Insight: Hybrid approach (Ontology + RL) combines strengths of both paradigms, addressing critical weaknesses while maintaining acceptable trade-offs across all dimensions.

1.2.6 Industry Lessons

Waymo's Knowledge-Driven Approach

Strategy: - Heavy reliance on rule-based systems and predefined scenarios - Extensive mapping and localization infrastructure - Conservative decision-making with safety margins - Geographic containment to well-mapped areas (Phoenix, Miami)

Lessons for Flyby-F11: - Proves safety-critical systems benefit from explicit knowledge representation - Geographic constraints demonstrate importance of environmental knowledge (analogous to mission-specific ontologies) - Conservative approach ensures safety but limits operational flexibility - Infrastructure dependence (HD maps) analogous to our ontology development effort

What to Adopt: - Formal safety verification methods - Scenario-based testing frameworks - Incremental deployment strategy (start constrained, expand gradually)

What to Avoid: - Pure rule-based approach that cannot adapt to novel situations - Over-reliance on specific infrastructure (Flyby-F11 must handle GPS-denied, communications-limited environments)

Tesla's Data-Driven Approach

Strategy: - End-to-end deep learning with neural networks - Fleet learning from millions of vehicles - Rapid iteration and over-the-air updates - Pushing boundaries of neural network capabilities

Lessons for Flyby-F11: - Demonstrates power of learning from real-world experience - Shows scalability benefits of data-driven approaches - Highlights challenges of black-box systems (interpretability, safety verification) - Illustrates gap between ambitious goals and deployment reality (still not true L4 autonomy)

What to Adopt: - Continuous learning and improvement philosophy - Use of simulation for edge case generation - Rapid iteration cycles - Neural networks for perception tasks

What to Avoid: - Pure end-to-end learning without interpretability - Over-promising capabilities before validation - Insufficient safety verification before deployment - Black-box decision-making in safety-critical contexts

Hybrid Path for Flyby-F11

Synthesis: - Combine Waymo's safety rigor with Tesla's adaptive learning - Use ontology as "safety rails" (Waymo-like formal constraints) - Use RL as "optimization engine" (Tesla-like adaptive learning) - Ensure interpretability through ontology-based explanations - Enable mission flexibility while guaranteeing safety compliance

Competitive Advantages: - Faster adaptation than pure rule-based systems (Waymo's limitation) - Better safety guarantees than pure learning systems (Tesla's gap) - Edge-based autonomy without infrastructure dependence (both companies' constraint) - Mission-specific customization through ontology engineering - Suitable for NDAA-compliant, communications-denied operations

Deployment Strategy: - Start with tightly constrained missions (Waymo's geographic containment philosophy) - Gradually relax ontology constraints as validation accumulates - Continuously learn and improve (Tesla's fleet learning philosophy) - Maintain formal safety verification throughout (Waymo's strength)

1.3 Knowledge-Driven Decision and Planning Methods

Knowledge-driven methods separate the decision-making phase from the path planning phase to enhance modularity, manageability, and efficiency. This distinction allows the decision module to focus on high-level strategies while the path planning module translates these strategies into specific driving paths.

1.3.1 Decision-Making Processes

1.3.1.1 1. Rule-Based Decision Systems

Rule-based systems make decisions using predefined rules and logic, relying on expert knowledge and experience.

Key Approaches: - **If-Then Rule Systems:** Process perception information from traffic environment to generate corresponding driving behaviors - **Multi-Level Collision Mitigation:** Systems that evaluate remaining reaction time (TTR) to decide whether to intervene - **Advanced Predictive Mechanisms:** Analyze current state of traffic flow and vehicle behavior patterns using probabilistic models

Notable Implementations: - Zhao et al. (2021): Rule-based system selecting optimal driving strategy based on surrounding traffic conditions and risk assessment - Hillenbrand et al. (2006): Time-based decision-making approach providing flexible trade-off between benefits and risks - Noh and An (2017): Highway decision framework with situation assessment and strategy decision-making components

Challenges: - Difficulty handling novel situations not covered by rules - Scaling challenge as rule sets grow complex - Overly conservative decision-making - Hard to tune for optimal performance - Increased time complexity due to rule accumulation

1.3.1.2 2. State Transition and Management Models

State transition models describe how a system moves between different states based on actions and observations, using frameworks like Markov Decision Processes (MDPs) and Partially Observable MDPs (POMDPs).

Markov Decision Processes (MDPs): - Provide structured framework for decision-making under uncertainty - Enable vehicles to evaluate potential actions based on probabilistic models - Optimize decision-making when not all variables are fully observable

Partially Observable MDPs (POMDPs): - Handle scenarios with hidden road users and occluded conditions - Model decision problems in urban autonomous driving - Particularly useful for target object search in partially unknown environments

Notable Implementations: - Huang Z. et al. (2024): Online learning-based behavior prediction model using transformer-based architecture with recurrent neural memory - Schörner et al. (2019): Hierarchical framework for multi-interaction environments under occluded conditions - Chen and Kurniawati (2024): Context-aware decision-making algorithm modeling problems as POMDP

Multi-Agent Extensions: - Multi-agent MDPs (MMDPs) facilitate coordinated strategies in environments with multiple vehicles, pedestrians, and traffic systems - Enable anticipation and response to actions of other agents - Lead to more reliable and intelligent driving solutions

1.3.1.3 3. Game-Theory Based Decision Models

Game-theory approaches treat autonomous driving as a multi-agent system, analyzing and predicting other traffic participants' behaviors to form cooperative or competitive driving strategies.

1.3.2 Planning Processes

1.3.2.1 1. Search-Based Planning

Search-based algorithms find paths through discrete state spaces using graph search techniques.

Common Algorithms: - **A***: Uses heuristics to guide search toward goal - **Dijkstra's Algorithm**: Finds shortest path in weighted graphs - **Hybrid A***: Combines discrete search with continuous motion primitives

Applications: Grid-based path finding, optimal route planning in known environments

1.3.2.2 2. Sampling-Based Planning

Sampling-based methods explore state space by randomly sampling configurations and building trees or graphs.

Common Algorithms: - **RRT (Rapidly-exploring Random Trees)**: Efficiently explores high-dimensional spaces - **PRM (Probabilistic Roadmap)**: Builds roadmap of collision-free paths

Advantages: Effective in high-dimensional spaces, doesn't require explicit discretization

1.3.2.3 3. Optimization-Based Planning

Optimization-based techniques formulate path planning as an optimization problem, minimizing cost functions subject to constraints.

Approaches: - Trajectory optimization techniques - Constraint satisfaction methods - Cost function minimization for smoothness, safety, and efficiency

1.4 Data-Driven Decision and Planning Methods

Data-driven approaches enhance decision accuracy and adaptability by training and optimizing decision models using vast amounts of real driving data. These methods don't require pre-defined explicit rules and can effectively learn and emulate human driver behaviors.

1.4.1 1. Imitation Learning

Imitation learning trains systems by observing expert behavior and demonstrations.

Key Characteristics: - Learns from expert demonstrations - Bypasses need for explicit reward function design - Can quickly acquire reasonable behaviors from human experts

Challenges: - Requires large volumes of high-quality expert demonstrations - Limited by training data quality and diversity - Difficult to generalize beyond demonstrated scenarios - Distribution shift problems when encountering novel situations

1.4.2 2. Reinforcement Learning (RL)

Reinforcement learning enables systems to learn through trial and error by interacting with environment and receiving rewards.

Key Strength: RL's ability to explore and optimize policies in complex, dynamic decision-making tasks makes it particularly promising for autonomous driving motion planning challenges.

Advantages: - Can discover novel strategies beyond human demonstrations - Optimizes for long-term rewards - Adapts to dynamic environments

Critical Challenges: - **Sample Efficiency:** One of the key challenges for RL in autonomous driving - **Safety During Exploration:** Difficult to explore safely in real-world environments - **Reward Function Design:** Challenging to specify appropriate rewards - **Limited by Training Data:** Relying solely on expert demonstrations or prior-based reward shaping can limit the algorithm's ability to explore optimal policies - **Bias Issues:** Especially when prior information is incomplete or contains significant bias

1.4.3 3. Inverse Reinforcement Learning (IRL)

Inverse reinforcement learning infers reward functions from expert demonstrations, addressing the challenge of reward function design.

Approach: - Observes expert behavior to infer underlying reward function - Once reward function is learned, standard RL can be applied - Combines advantages of imitation learning and reinforcement learning

Applications: Learning from human driver behavior, understanding underlying objectives

1.5 Hybrid Decision and Planning Methods

CRITICAL FINDING: Research has identified a clear trend and compelling direction for future research: **the integration of knowledge-driven and data-driven approaches into hybrid methods.**

1.5.1 Motivation for Hybrid Approaches

Hybrid methods aim to combine the strengths of both knowledge-driven and data-driven approaches:

Knowledge-Driven Component Benefits: - Ensures decisions comply with traffic regulations and safety standards - Provides systematic constraints and guidance within well-defined framework - Offers interpretability and verifiability - Guarantees vehicle remains in safe state within predefined range of rules

Data-Driven Component Benefits: - Improves system's adaptability to complex environments - Enhances prediction accuracy by extracting patterns from extensive driving data - Enables learning behaviors beyond explicit programming - Significantly enhances generalizability across different environments

Combined Advantages: - More flexible, robust, and interpretable decision planning - Balances safety guarantees with adaptive learning - Addresses limitations of pure approaches - Enables both rule compliance and novel situation handling

1.5.2 Implementation Approaches

Integration Strategies: 1. **Expert Knowledge with Imitation Learning:** Using knowledge-driven constraints to validate/filter expert demonstrations 2. **Rule-Based Frameworks with**

RL: Combining predefined safety rules with reinforcement learning for optimization 3. **Inverse RL with Domain Knowledge:** Leveraging ontological knowledge to help infer reward functions from mission objectives 4. **Hierarchical Architectures:** Knowledge-based search for high-level planning, learning-based methods for low-level control

1.5.3 Challenges in Hybrid Development

Design Complexity: - Requires precise integration of fundamentally different techniques - Demands strong theoretical knowledge from algorithm designers - Necessitates continuous tuning and optimization in practice

Implementation Challenges: - Balancing control between knowledge-driven and data-driven components - Determining appropriate boundaries between rule-based and learned behaviors - Managing computational resources for both components - Ensuring consistent behavior across transitions

Validation Difficulties: - Testing both rule compliance and learned behaviors - Verifying safety in novel situations - Ensuring interpretability of combined system - Validating performance across diverse scenarios

1.6 Comparative Analysis: Knowledge vs. Data-Driven Approaches

1.6.1 Knowledge-Driven Limitations

Scalability Issues: - Difficulty handling novel situations not covered by predefined rules - Scaling challenge as rule sets grow increasingly complex - Hard to tune for optimal performance across diverse scenarios - Increased time complexity due to rule accumulation

Adaptability Constraints: - Overly conservative decision-making in uncertain situations - Cannot learn from experience or adapt to new patterns - Limited flexibility in dynamic environments - Requires manual updates for new scenarios

Maintenance Burden: - Extensive expert knowledge required for rule creation - Difficult to maintain consistency across large rule sets - Challenge in prioritizing conflicting rules - Time-consuming to update and validate changes

1.6.2 Data-Driven Limitations

Sample Efficiency: - One of the key challenges for reinforcement learning - Requires vast amounts of training data - Expensive and time-consuming data collection - Difficulty obtaining diverse, representative samples

Training Dependencies: - Limited by training data quality and coverage - Solely relying on expert demonstrations or prior-based reward shaping can limit the algorithm's ability to further explore optimal policies - Bias issues, especially when prior information is incomplete or contains significant bias - Distribution shift problems when encountering scenarios different from training

Safety and Interpretability: - Poor interpretability of learned models ("black box" problem) - Difficult to ensure consistent and safe decisions - Challenge in verifying correctness across all scenarios - Safety risks during exploration phase - Lack of guaranteed bounds on behavior

Generalization Challenges: - Difficulty generalizing beyond demonstrated scenarios - May not transfer well to new environments - Sensitive to differences between training and deployment

conditions - Requires retraining for significantly different contexts

1.7 Experimental Platforms and Validation

The paper discusses crucial resources for autonomous driving research and development:

1.7.1 Datasets

Purpose: Support training and testing of decision-making and planning algorithms

Key Characteristics: - Contain diverse traffic scenarios and conditions - Include sensor data from multiple modalities - Provide ground truth annotations - Enable comparative evaluation of algorithms

Applications: - Training data-driven models - Validation of knowledge-driven rules - Benchmarking algorithm performance - Testing generalization capabilities

1.7.2 Simulation Platforms

Purpose: Testing algorithms and simulating complex traffic environments

Critical Role: - Enable safe testing before real-world deployment - Provide controlled, repeatable test scenarios - Allow exploration of edge cases and dangerous situations - Support rapid iteration and development - Reduce costs compared to physical testing

Key Features: - Physics-based vehicle dynamics - Sensor simulation (cameras, LiDAR, radar) - Traffic scenario generation - Multi-agent interaction modeling - Hardware-in-the-loop capabilities

Importance for Autonomous Driving: - Essential for transitioning algorithms from theory to practice - Enable comprehensive testing across diverse conditions - Support validation of safety-critical systems - Facilitate comparison of different approaches

1.8 Current Industry Deployment

1.8.1 Commercial Applications

Waymo: - Offers autonomous taxi services in Phoenix, Arizona - Expanding to Miami, Florida - Partnership with Moove for fleet operations - Approximately 200 autonomous vehicles deployed in Phoenix - Focuses on specific geographic areas and traffic conditions

Tesla: - Autopilot and Full Self-Driving (FSD) systems - Fleet learning program for data collection - End-to-end (E2E) deep learning strategy - Integration of neural networks and reinforcement learning - Still working toward true Level 4 (L4) autonomous driving

Common Methodologies in Industry: - Rule-based systems - State transition models (MDPs, POMDPs) - Game-theoretic approaches - Sophisticated algorithms for road environment processing - Time-based decision-making approaches

1.8.2 Deployment Challenges

Technical Challenges: - Interactions with human drivers in mixed traffic - Responses to unexpected situations - Adaptability in complex traffic environments - Safety and reliability issues - System robustness in edge cases

Regulatory and Operational: - Regulatory compliance and licensing - Ethical and legal responsibilities - Adherence to socially accepted moral standards - Traffic regulation compliance during emergencies - Market acceptance challenges

1.9 Challenges and Future Perspectives

1.9.1 Current Challenges

Environmental Perception Uncertainties: - Sensor limitations in adverse weather - Occlusion and partial observability - Dynamic scene understanding - Multi-modal sensor fusion

Traffic Participant Unpredictability: - Human driver behavior modeling - Pedestrian intention prediction - Multi-agent interaction complexity - Cooperative vs. competitive scenarios

Algorithm Limitations: - Computational resource constraints - Real-time performance requirements - Robustness in unforeseen situations - Stability and safety guarantees

Ethical and Legal Considerations: - Decision-making in unavoidable accident scenarios - Liability and responsibility assignment - Compliance with varying regional regulations - Public trust and acceptance

1.9.2 Future Research Directions

The paper identifies several promising directions for advancing autonomous driving:

1. **Multi-Sensor Fusion:** - Integration of diverse sensor modalities - Robust perception in challenging conditions - Redundancy for safety-critical applications
2. **Deep Learning Advances:** - Improved feature encoding and representation - End-to-end learning architectures - Transfer learning across domains - Efficient neural network architectures
3. **Behavior Prediction Models:** - Enhanced accuracy of traffic participant predictions - Long-term trajectory forecasting - Intent recognition and reasoning - Uncertainty quantification
4. **Scenario Simulation:** - More realistic virtual environments - Edge case generation - Adversarial scenario testing - Scenario diversity for training
5. **Advanced Reinforcement Learning:** - Improved sample efficiency - Safe exploration strategies - Multi-agent reinforcement learning - Hierarchical RL architectures
6. **Synthetic Data Generation:** - Augmenting real-world data - Generating rare scenarios - Domain randomization - Photo-realistic rendering
7. **Continuous Learning:** - Lifelong learning systems - Online adaptation - Incremental learning without catastrophic forgetting - Learning from deployment experience
8. **Explainable AI (XAI):** - Interpretable decision-making - Transparent neural networks - Causal reasoning - Human-understandable explanations

1.10 Relevance to Drone Autonomy

1.10.1 Direct Applicability

Shared Challenges: - Decision-making in dynamic, uncertain environments - Multi-agent interaction and collision avoidance - Real-time trajectory planning under constraints - Safety-critical

operation requirements - Sensor fusion and perception uncertainties

Transferable Methodologies: - Hybrid knowledge-driven and data-driven approaches apply to drone systems - MDPs and POMDPs for decision-making under uncertainty - Reinforcement learning for adaptive control - Sampling-based planning (RRT, PRM) for path planning - Optimization-based trajectory generation

1.10.2 Drone-Specific Considerations

3D Environment Complexity: - Autonomous vehicles operate primarily in 2D plane - Drones require full 3D motion planning and obstacle avoidance - Additional degrees of freedom increase state space complexity - Vertical dimension adds planning challenges

Dynamic Constraints: - Drones have different dynamics than ground vehicles - More sensitive to disturbances (wind, turbulence) - Energy/battery constraints more critical - Faster response times often required

Sensor Modalities: - Different sensor suites (less reliance on road infrastructure) - Visual odometry and SLAM more critical for drones - IMU and altitude sensors essential - Depth perception for obstacle avoidance

Mission Profiles: - Inspection, surveillance, delivery missions vs. transportation - Communications-denied operations more common - GPS-denied navigation requirements - Higher autonomy levels often required (no human in vehicle)

1.10.3 Integration with Flyby-F11 Approach

Strong Validation for Ontology-Constrained RL:

1. **Hybrid Architecture Recommendation:** The survey's key finding strongly supports integrating knowledge-driven (ontology) and data-driven (RL) approaches
2. **Ontology as Knowledge-Driven Component:**
 - Provides safety rules and mission structure
 - Enables semantic reasoning about mission objectives
 - Offers interpretable, verifiable constraints
 - Ensures compliance with operational requirements
3. **RL as Data-Driven Component:**
 - Enables optimization and adaptation
 - Handles situations beyond explicit rules
 - Learns from experience
 - Discovers novel strategies
4. **Addressing Complementary Limitations:**
 - Ontology constraints guide RL exploration (improves sample efficiency)
 - RL handles novel situations beyond explicit rules (enhances adaptability)
 - Ontology provides structured prior knowledge (reduces training bias)
 - Combined approach balances safety guarantees with learning flexibility

1.10.4 Alternative Approaches for Flyby-F11

Based on this survey, additional architectures to consider:

1. **Imitation Learning + Ontology:** - Use ontology to validate and filter expert demonstrations - Ensure learned behaviors comply with safety constraints - Leverage expert knowledge while maintaining formal guarantees
2. **Inverse RL + Ontology:** - Ontology helps infer reward functions from high-level mission objectives - Formal mission specifications guide reward learning - Combines mission-level reasoning with low-level optimization
3. **Hierarchical Planning + Learning:** - Ontology-based search for high-level mission planning - RL for low-level control and trajectory optimization - Clear separation of concerns between strategic and tactical layers
4. **Game-Theoretic Multi-Agent Approaches:** - Model drone interactions in multi-UAV scenarios - Apply game theory for cooperative mission planning - Relevant for swarm operations or communication-denied coordination

1.11 Key Findings Summary

1.11.1 Methodological Insights

1. **No Single Best Approach:** Neither pure knowledge-driven nor pure data-driven methods can address all challenges in autonomous systems
2. **Hybrid Methods as Future Direction:** Integration of both approaches provides balanced framework leveraging strengths of each
3. **Importance of Interpretability:** Safety-critical applications require transparent, verifiable decision-making processes
4. **Sample Efficiency Critical:** Real-world deployment requires algorithms that learn efficiently from limited data
5. **Safety Must Be Guaranteed:** Cannot rely solely on learned behaviors; need formal safety constraints

1.11.2 Technical Contributions

1. **Comprehensive Taxonomy:** Clear categorization of knowledge-driven, data-driven, and hybrid approaches
2. **Comparative Analysis:** Detailed evaluation of advantages and limitations of each method category
3. **Industry Context:** Discussion of real-world deployment challenges and current commercial implementations
4. **Validation Framework:** Emphasis on simulation platforms and datasets for algorithm development
5. **Future Research Roadmap:** Identification of key challenges and promising research directions

1.11.3 Implications for Drone Autonomy

1. **Validates Hybrid Approach:** Strong evidence supporting ontology-constrained RL architecture
2. **Addresses Key Challenges:** Sample efficiency, safety, interpretability all critical for drone operations
3. **Mission-Specific Adaptation:** Framework allows customization for different mission profiles
4. **Real-World Deployment Path:** Emphasis on simulation and validation aligns with our development strategy
5. **Scalability Considerations:** Hybrid methods provide path to handle increasing complexity

1.12 Conclusion

This comprehensive survey provides strong theoretical and empirical support for hybrid decision-making and planning approaches in autonomous systems. The paper's central finding—that integrating knowledge-driven and data-driven methods is the most promising direction—directly validates the ontology-constrained reinforcement learning architecture being developed for the Flyby-F11 platform.

The survey emphasizes that neither pure rule-based systems nor pure learning-based systems can adequately address the full spectrum of challenges in autonomous navigation. Rule-based systems offer safety guarantees and interpretability but lack adaptability; learning-based systems provide flexibility and optimization but struggle with sample efficiency, safety assurance, and interpretability. Hybrid approaches combining both methodologies offer the best path forward.

For drone autonomy applications, this research provides crucial insights into balancing formal safety constraints with adaptive learning capabilities—a critical requirement for operations in GPS-denied, communications-limited, and dynamic environments. The ontology-constrained RL approach for Flyby-F11 embodies these principles by using formal ontological knowledge to guide and constrain reinforcement learning, ensuring both mission compliance and adaptive optimization.

1.13 References

Hu J, Wang Y, Cheng S, Xu J, Wang N, Fu B, Ning Z, Li J, Chen H, Feng C and Zhang Y (2025) A survey of decision-making and planning methods for self-driving vehicles. *Front. Neurorobot.* 19:1451923. doi: 10.3389/fnbot.2025.1451923