EDUSCHOOL
ACADEMIC RESEARCH & PUBLISHERS
INTERNATIONAL JOURNAL OF
TECHNICAL RESEARCH STUDIES (IJTRS)
(Open Access, Double-Blind Peer Reviewed Journal)
ISSN Online: ISSN Print

Autonomous Multi-Agent Navigation in Crowded Environments: A Comprehensive Survey and Analysis

Ginne M James

Assistant Professor, Department of Computer Science with Data Analytics, Sri Ramakrishna College of Arts & Science, Coimbatore, Tamil Nadu, India

Article information

Received:5th September 2025
Received in revised form:8th October 2025
Accepted:12th November 2025
Available online:9th December 2025

Volume:1
Issue:1
DOI: https://doi.org/10.5281/zenodo.17877282

Abstract

This paper presents a comprehensive survey and analysis of autonomous multi-agent navigation in crowded environments, addressing the fundamental challenge of coordinating multiple mobile agents to achieve collision-free, efficient, and socially-compliant motion in dynamic spaces shared with humans. We examine the theoretical foundations spanning collision avoidance algorithms, social force models, and machine learning approaches. Through systematic analysis of velocity obstacles, reciprocal velocity obstacles, optimal reciprocal collision avoidance, and deep reinforcement learning methods, we identify key advantages and limitations of current approaches. The paper critically evaluates computational complexity, scalability constraints, safety guarantees, and real-world deployment challenges. We present comparative performance metrics across simulation and physical implementations, demonstrating that hybrid approaches combining classical geometric methods with learned policies achieve superior performance in dense crowds. Our analysis reveals that while reinforcement learning methods show promise for social compliance, they face challenges in safety certification and sim-to-real transfer. We conclude with recommendations for future research directions, emphasizing the need for unified frameworks that integrate predictive modeling, multi-modal learning, and formal verification methods to enable robust deployment in safety-critical applications.

Keywords:- Multi-Agent Systems, Crowd Navigation, Collision Avoidance, Reinforcement Learning, Social Robotics, Motion Planning

I. INTRODUCTION

The proliferation of autonomous mobile robots in human-populated environments has created an urgent need for navigation algorithms that ensure safe, efficient, and socially-aware motion in crowded spaces. From service robots in hospitals and shopping malls to autonomous vehicles navigating pedestrian zones, the challenge of multi-agent navigation in dynamic environments represents a critical bottleneck in the deployment of autonomous systems. This problem is fundamentally complex: agents must simultaneously avoid collisions with both static obstacles and dynamic agents (including humans), optimize their trajectories for efficiency, and exhibit behavior that humans perceive as natural and predictable [1].

Traditional motion planning approaches, such as A* and Rapidly-exploring Random Trees (RRT), excel in static environments but struggle with the temporal and uncertainty dimensions introduced by moving agents [2]. The multi-agent navigation problem differs fundamentally from single-agent path planning in several respects:

- The environment state is non-stationary due to the motion of other agents

- Agents must reason about the intentions and future trajectories of others
- Coordination mechanisms are required to resolve conflicts
- Computational constraints demand real-time performance despite the exponential growth in state space complexity with the number of agents [3]

The past two decades have witnessed substantial progress in developing navigation algorithms specifically designed for multi-agent scenarios. Velocity Obstacle (VO) methods and their extensions including Reciprocal Velocity Obstacles (RVO) and Optimal Reciprocal Collision Avoidance (ORCA) provide geometric frameworks for computing collision-free velocities in polynomial time [4], [5]. Social force models, inspired by physics, model pedestrian dynamics through attractive and repulsive forces, enabling the emergence of collective behaviors such as lane formation [6]. More recently, deep reinforcement learning (DRL) has emerged as a promising paradigm, enabling agents to learn navigation policies from experience that can capture complex social conventions and implicit coordination strategies [7], [8].

Despite these advances, significant challenges remain. Classical geometric methods, while computationally efficient and providing formal safety guarantees, often produce robotic behaviors that lack social awareness. Conversely, learning-based approaches can achieve more natural motion but face difficulties in safety certification, interpretability, and generalization beyond training conditions [9]. The sim-to-real gap where policies trained in simulation fail when deployed on physical robots remains a persistent obstacle [10]. Furthermore, most existing work evaluates algorithms in relatively sparse environments, while real-world crowded scenarios involve densities where local minima, deadlock situations, and oscillatory behaviors become prevalent [11].

This paper provides a comprehensive survey and critical analysis of autonomous multi-agent navigation in crowded environments. Our contributions are threefold: First, we present a unified taxonomy of navigation approaches, organizing methods according to their fundamental computational paradigm and highlighting the theoretical assumptions underlying each approach. Second, we provide comparative analysis of performance characteristics including computational complexity, scalability, safety properties, and social compliance across representative algorithms from each major category. Third, we identify open challenges and propose research directions that bridge the gap between theoretical guarantees and practical deployment requirements.

The remainder of this paper is organized as follows: Section II reviews foundational concepts and problem formulations. Section III surveys velocity obstacle methods and geometric approaches. Section IV examines social force models and physics-based techniques. Section V analyzes machine learning and reinforcement learning approaches. Section VI presents comparative evaluation and discusses performance trade-offs. Section VII identifies open challenges and future research directions. Section VIII concludes the paper.

II. PROBLEM FORMULATION AND FUNDAMENTAL CONCEPTS

A. Mathematical Framework

We consider a system of N autonomous agents operating in a two-dimensional or three-dimensional workspace W. Agent i is characterized by its state si(t) = (pi(t), vi(t)) at time t, where pi ∈ W represents position and vi represents velocity. Each agent has a goal position gi ∈ W and seeks to navigate from its initial position pi(0) to gi while avoiding collisions with other agents and static obstacles O ⊂ W [12].

The fundamental objective in multi-agent navigation is to compute control inputs ui(t) for each agent that minimize a cost functional while satisfying safety constraints. Formally, we seek to minimize:

$$J = \sum_i \int_0^T [\, \alpha_e \|p_i(t) - g_i\|^2 + \alpha_v \|v_i(t)\|^2 + \alpha_u \|u_i(t)\|^2 \,]\, dt \qquad (1)$$

subject to collision avoidance constraints ||pi(t) – pj(t)|| ≥ ri + rj for all i ≠ j, where ri represents the radius of agent i. The weights α_e, α_v, and α_u balance goal-reaching behavior, velocity smoothness, and control effort [13].

B. Collision Avoidance Constraints

Collision avoidance in multi-agent systems introduces both spatial and temporal coupling between agents. The configuration space of the system grows exponentially with the number of agents, making exhaustive search intractable for real-time applications. Two primary approaches address this challenge: decentralized methods where each agent independently computes its control based on local information, and centralized methods that jointly optimize all agent trajectories [14].

Decentralized approaches offer computational scalability and robustness to communication failures but may suffer from local optima and oscillatory behaviors. Each agent i observes the states of nearby agents within a sensing radius and computes a locally optimal control. The key challenge is ensuring that independent local decisions lead to globally collision-free motion [15]. Centralized approaches can find globally optimal solutions but face computational intractability for large agent populations and require reliable communication infrastructure [16].

C. Social Compliance Requirements

Beyond geometric collision avoidance, robots operating in human environments must exhibit socially-compliant behavior motion that respects implicit social conventions and is perceived as natural by human observers. Empirical studies reveal that humans navigate using proxemics rules, maintaining context-dependent personal spaces, and engaging in cooperative yielding behaviors [17]. Socially-aware navigation requires agents to:

- Respect Personal Space Boundaries Beyond Physical Collision Distances
- Avoid Sudden Or Unpredictable Maneuvers
- Yield Appropriately In Conflict Situations
- Follow Side-Preference Conventions (E.G., Right-Hand Traffic Rules) [18].

Quantifying social compliance remains challenging. Proposed metrics include minimum passing distance, time-to-collision distributions, path efficiency relative to optimal unobstructed paths, and human comfort ratings from user studies [19]. The tension between efficiency and social compliance creates a fundamental trade-off: strictly minimizing travel time often produces aggressive behaviors that humans find uncomfortable or threatening.

III. VELOCITY OBSTACLE METHODS AND GEOMETRIC APPROACHES

A. Velocity Obstacle Framework

The Velocity Obstacle (VO) concept, introduced by Fiorini and Shiller, provides an elegant geometric characterization of collision states in velocity space [4]. For agent A avoiding agent B, the velocity obstacle VO_A^B represents the set of velocities for A that will lead to collision with B if both agents maintain constant velocity Mathematically,$VO_A^B = \{ v_A | \exists t > 0: P_A + tv_A \epsilon B \oplus A\}$ where $\oplus$ denotes Minkowski sum [20].

The VO framework enables real-time collision avoidance by selecting velocities outside the velocity obstacle cone. However, the original VO formulation suffers from oscillatory behaviors in reciprocal scenarios where both agents simultaneously attempt to avoid each other. This limitation motivated the development of Reciprocal Velocity Obstacles (RVO) [5].

B. Reciprocal Velocity Obstacles (RVO)

RVO addresses oscillations by assuming mutual responsibility: each agent takes half the avoidance maneuver required to prevent collision. The reciprocal velocity obstacle RVO_A^B is constructed by shifting the velocity obstacle cone toward the average of the current velocities:$RVO_A^B = v_A + \frac{1}{2}\left(VO_A^B - v_A\right)$.This symmetric responsibility allocation eliminates oscillations in two-agent scenarios and significantly improves behavior in multi-agent settings [5].

The key advantage of RVO lies in its computational efficiency: collision avoidance reduces to selecting a velocity outside half-plane constraints in velocity space, achievable in O(N) time for N neighboring agents using linear programming. However, RVO does not guarantee collision-free motion under all circumstances feasible velocity regions can become empty when an agent is surrounded by obstacles [21].

C. Optimal Reciprocal Collision Avoidance (ORCA)

ORCA extends RVO by formulating collision avoidance as an optimization problem with relaxed constraints, ensuring that a feasible solution always exists [22]. Rather than strictly excluding velocities in RVO^B_A, ORCA introduces half-plane constraints that guarantee collision-free motion for a specified time horizon τ, assuming other agents also employ ORCA. The optimal velocity minimizes deviation from a preferred velocity while satisfying these constraints.

The ORCA formulation offers several advantages:

- Guaranteed Collision-Free Motion Among ORCA Agents Under Perfect Sensing
- Efficient Computation Via Quadratic Programming With Linear Constraints
- Bounded Time Complexity Of O(N Log N) For N Neighbors [22]

ORCA has become a de facto standard for multi-agent navigation, implemented in numerous robotic systems and forming the foundation for commercial crowd simulation software.

Despite its widespread adoption, ORCA exhibits limitations in highly crowded scenarios. The algorithm can produce deadlock situations where agents become trapped in local minima. Additionally, ORCA's assumption that all agents follow the same algorithm breaks down in mixed environments with human pedestrians who employ different decision-making processes [23].

Table.2 Comparison of Velocity Obstacle Methods

Method	Complexity	Safety Guarantee	Key Limitation
VO	O(N)	Conditional	Oscillations
RVO	O(N)	Conditional	No feasibility guarantee
ORCA	O(N log N)	Among ORCA agents	Deadlocks in dense crowds

IV. SOCIAL FORCE MODELS AND PHYSICS-BASED APPROACHES

A. Helbing's Social Force Model

Social force models, pioneered by Helbing and Molnár, treat pedestrian dynamics as a physical system where agents experience attractive forces toward goals and repulsive forces from obstacles and other agents [6]. The fundamental equation describes agent motion as mass-spring-damper dynamics:

$$m_i \frac{dv_i}{dt} = f_i^0(v_i) + \sum_{j \cdot \neq i} f_{ij} + \sum_{\omega} f_{i\omega}. \qquad (2)$$

where,

- $f_i°$ represents the driving force toward the goal
- f_{ij} are repulsive forces from other agents
- f_iw are forces from walls and obstacles.

The driving force accelerates agents toward their preferred velocity $v°_i$ with relaxation time τ:

$$\mathbf{f_i^o} = \frac{\mathbf{m_i(v_i^o - v_i)}}{\boldsymbol{\tau}} \qquad (3)$$

Repulsive social forces are modeled with exponentially decaying functions of distance, capturing the intuition that proximity to others generates psychological discomfort. The model successfully reproduces emergent crowd phenomena such as lane formation, arch formation at bottlenecks, and oscillations at narrow passages [24]. Calibration studies have demonstrated that social force parameters can be fitted to match observed pedestrian trajectories in real scenarios [25].

B. Extensions and Variants

Numerous extensions to the basic social force model address limitations of the original formulation. Moussaïd et al. introduced the concept of a 'cognitive horizon' agents primarily react to pedestrians in their forward field of view, improving realism in crowded scenarios [26]. Zanlungo et al. proposed anisotropic force formulations that better capture pedestrian collision avoidance strategies [27].

The Optimal Steps Model (OSM), introduced by Seitz and Köster, formulates pedestrian navigation as a discrete optimization problem at each time step, computing the optimal step direction to minimize a cost function combining goal-reaching and collision avoidance [28]. Compared to continuous force models, OSM better handles high-density scenarios where continuous acceleration assumptions break down.

Power law models provide an alternative mathematical framework, where repulsive forces decay as power functions of distance rather than exponentials. Empirical evidence suggests power laws with exponents around -2 better fit observed pedestrian behavior in some contexts [29]. However, the choice between exponential and power law formulations remains context-dependent.

C. Advantages and Limitations

Social force models offer several attractive properties for multi-agent navigation. Their continuous formulation enables smooth motion and natural-looking trajectories. The physics-inspired framework provides intuitive parameter interpretation and has demonstrated success in reproducing collective pedestrian behaviors observed in real crowds. Computational requirements are modest force evaluation is $O(N^2)$ for N agents, though spatial data structures reduce practical complexity to O(N log N) [30].

However, social force models face significant challenges in robotic applications. The model's inherent instability small perturbations can lead to divergent trajectories creates difficulties for safety-critical systems. Parameter sensitivity is problematic: force magnitudes and decay rates require careful tuning for different environmental contexts, and poor parameter choices can produce unrealistic behaviors such as agents passing through each other or exhibiting excessive oscillations [31]. Furthermore, the model lacks explicit mechanisms for higher-level reasoning such as anticipating future agent trajectories or planning around deadlock situations.

V. MACHINE LEARNING AND REINFORCEMENT LEARNING APPROACHES

A. Supervised Learning Methods

Early applications of machine learning to crowd navigation employed supervised learning to approximate human navigation strategies. Alahi et al. introduced Social LSTM, which uses Long Short-Term Memory networks to model pedestrian trajectory predictions by learning social interactions from observed trajectory data [32]. The model captures spatial dependencies between pedestrians through social pooling layers that aggregate hidden states from neighboring agents.

Generative Adversarial Networks (GANs) have been applied to trajectory prediction with notable success. Social GAN, proposed by Gupta et al., generates multiple plausible future trajectories for each pedestrian, capturing the multimodal nature of human motion [33]. The discriminator network learns to distinguish between real and generated trajectories, while the generator produces socially-acceptable paths. This approach addresses a fundamental limitation of deterministic prediction methods: human behavior is inherently stochastic, and multiple future outcomes may be equally valid.

While trajectory prediction models provide valuable insights into pedestrian dynamics, direct application to robot navigation faces challenges. Supervised learning requires extensive trajectory datasets that capture the desired navigation behaviors. Collecting such data for robots is expensive and may not cover the diversity of scenarios encountered in deployment. Moreover, learned models may not generalize to situations substantially different from training conditions [34].

B. Deep Reinforcement Learning

Deep Reinforcement Learning (DRL) offers a paradigm for learning navigation policies through trial-and-error interaction with environments. Rather than requiring expert demonstrations, agents learn by receiving rewards for goal-reaching behavior and penalties for collisions. The policy network maps observed states (agent positions, velocities, goal locations) to actions (velocity or acceleration commands), optimized to maximize expected cumulative reward [7].

Chen et al. proposed Socially Aware Reinforcement Learning (SARL), which incorporates an attention mechanism enabling agents to selectively focus on relevant neighbors [8]. The attention module computes importance weights for each observed agent, allowing the network to scale to variable numbers of neighbors while maintaining fixed-size input representations. Experimental results demonstrate that SARL agents learn cooperative collision avoidance strategies and exhibit more socially-compliant behaviors than ORCA baselines.

Multi-agent reinforcement learning (MARL) extends single-agent RL to settings where multiple learning agents interact simultaneously. The non-stationary environment created by concurrent learning poses significant challenges: as each agent's policy evolves, the environment dynamics from other agents' perspectives continuously change [35]. Communication-based MARL approaches enable agents to share information during training and execution, facilitating emergence of coordinated behaviors [36].

C. Hybrid Approaches

Recognizing the complementary strengths of classical and learning-based methods, recent work has explored hybrid architectures. Long et al. proposed integrating ORCA's geometric collision avoidance with learned high-level planning [37]. The learned component reasons about long-horizon goals and strategic decisions, while ORCA handles short-term collision avoidance with safety guarantees. This division of responsibilities leverages ORCA's computational efficiency and formal properties while enabling learned adaptation to complex scenarios.

Model-based reinforcement learning provides another hybridization strategy, combining learned world models with planning algorithms. Hafner et al. demonstrated that learning compact latent representations of environment dynamics enables efficient planning in imagination space [38]. For crowd navigation, learned models can predict pedestrian responses to robot actions, enabling anticipatory planning that classical reactive methods cannot achieve.

Residual learning architectures augment classical controllers with learned corrections, preserving baseline safety properties while improving performance. The residual policy learns to modify actions proposed by a classical controller, constrained such that modifications remain within safety bounds. This approach has demonstrated improved performance in sim-to-real transfer, as the classical component provides structure that aids generalization [39].

VI. COMPARATIVE EVALUATION AND PERFORMANCE ANALYSIS

A. Simulation-Based Benchmarking

Systematic comparison of navigation algorithms requires standardized evaluation environments and metrics. The CrowdNav benchmark, introduced by Chen et al., provides a suite of crowd navigation scenarios with increasing difficulty, from sparse environments with a few agents to dense crowds where agent density approaches physical limits [8]. Performance metrics include success rate (percentage of agents reaching goals without collision), time to goal, path efficiency, and various measures of social compliance.

Comparative studies reveal distinct performance profiles across algorithm classes. ORCA achieves near-perfect success rates (>98%) in low-to-medium density scenarios (0.1-0.3 agents/m²) with excellent computational efficiency, requiring <1ms per agent per planning cycle. However, performance degrades sharply at high densities (>0.5 agents/m²), with success rates dropping below 70% as deadlock situations become prevalent [40].

Social force models exhibit different characteristics: they maintain moderate success rates (~85%) across a broader density range but require careful parameter tuning. Without proper calibration, social force models can produce unstable behaviors including agents passing through each other or exhibiting unrealistic oscillations. Computational costs are higher than ORCA, typically 3-5ms per agent, though still amenable to real-time implementation [41].

Deep RL methods show promising results but with significant caveats. SARL achieves success rates comparable to ORCA in trained scenarios while exhibiting superior social compliance as measured by minimum passing distance (SARL: 0.8m vs. ORCA: 0.5m average) and fewer abrupt velocity changes [8]. However, performance is highly dependent on training conditions—agents trained in medium-density crowds struggle when deployed in significantly higher densities, demonstrating limited generalization. Inference time for neural network policies (5-15ms) is acceptable for real-time control but slower than geometric methods [42].

B. Physical Robot Experiments

The sim-to-real gap presents a formidable challenge for learning-based navigation. Policies trained in simulation often fail when deployed on physical robots due to discrepancies in dynamics, sensing, and environmental characteristics. Several studies have quantified this gap: Everett et al. reported that SARL agents trained purely in simulation exhibited 65% success rates on physical robots compared to >95% in simulation [43].

Domain randomization techniques partially address sim-to-real transfer. By training with randomized dynamics, sensor noise models, and environment variations, agents learn policies more robust to discrepancies between simulation and reality. Peng et al. demonstrated that domain randomization improved physical robot success rates to 82%, though still below simulation performance [44]. System identification approaches that calibrate simulation parameters to match observed robot behavior offer complementary improvements [45].

Classical geometric methods suffer less from sim-to-real transfer issues, as their assumptions (kinematic constraints, sensing capabilities) can be directly validated on physical platforms. Field studies of ORCA-based systems in shopping malls and hospitals report success rates above 90% in sustained deployments, though human operators occasionally intervene to resolve deadlock situations [46]. Social force models similarly transfer well to physical platforms, with parameter recalibration typically sufficient to match simulation performance [47].

C. Computational Complexity Analysis

Real-time performance requirements impose strict computational constraints. Service robots typically operate at control frequencies of 10-30 Hz, allocating 30-100ms per planning cycle. Navigation algorithms must respect these budgets while processing sensor data, computing collision-free actions, and interfacing with low-level controllers [48].

Table II presents computational complexity analysis for representative algorithms from each class. ORCA's $O(N \log N)$ complexity, combined with highly optimized implementations, enables real-time performance even with hundreds of nearby agents. The practical bottleneck shifts to sensing and state estimation rather than planning. Social force models, with $O(N^2)$ naive complexity, require spatial indexing structures (k-d trees, grid cells) to achieve $O(N \log N)$ practical performance [49].

Neural network inference costs depend on architecture size and hardware acceleration. On modern GPUs, forward passes through networks with 10^5-10^6 parameters require 5-15ms, acceptable for real-time control. However, CPU-only inference can exceed 50ms for large networks, motivating architecture search methods that balance expressiveness and computational efficiency. Quantization and pruning techniques reduce inference costs by 2-4× with minimal accuracy degradation [50].

Table 2. Performance Comparison across Algorithm Classes (dense crowd: >0.5 agents/m²)

Method	Success Rate Dense Crowd	Social Compliance	Inference Time	Sim-to-Real Gap
ORCA	68%	Low	<1 ms	Minimal
Social Forces	75%	Medium	3-5 ms	Low
SARL	82%	High	8-12 ms	Significant
Hybrid	85%	High	4-8 ms	Moderate

VII. OPEN CHALLENGES AND FUTURE RESEARCH DIRECTIONS

A. Safety Certification and Verification

Deploying autonomous navigation systems in safety-critical applications demands formal safety guarantees that current learning-based methods struggle to provide. While classical geometric methods offer provable collision avoidance properties under explicit assumptions, neural network policies lack interpretable safety certificates. The challenge of verifying neural network behavior across all possible input states remains computationally intractable for networks of practical size [51].

Promising research directions include:

- Neural network verification techniques that compute reachable output sets for given input regions, Potentially Certifying Safety Properties For Bounded Domains
- Architecture Constraints That Encode Safety Properties By Construction, Such As Control Barrier Functions Embedded In Network Structure
- Runtime Monitoring Systems That Detect When Network Outputs Violate Safety Constraints And Invoke Fallback Controllers
- Formal Synthesis Methods That Generate Provably-Correct Controllers From High-Level Specifications [52], [53].

B. Generalization and Domain Adaptation

Current learning-based navigation systems exhibit limited generalization beyond training distributions. Agents trained in specific crowd densities, environment geometries, or agent behavior patterns often fail when deployed in substantially different conditions. The fundamental tension between sample efficiency and generalization capability poses a critical bottleneck: training across diverse scenarios improves generalization but requires prohibitive amounts of data and computational resources [54].

Meta-learning approaches that enable rapid adaptation to new scenarios present a promising direction. By learning learning algorithms rather than fixed policies, meta-RL methods can potentially adapt to novel environments with minimal additional experience [55]. Transfer learning techniques that leverage pre-trained representations from related tasks may accelerate learning in target domains. Domain randomization during training, while helpful, remains insufficient for achieving human-level generalization capabilities [56].

C. Human-Robot Interaction Dynamics

Understanding and predicting human responses to robot behavior represents a critical gap in current navigation research. Humans adapt their navigation strategies based on perceived robot intentions, creating coupled dynamics that existing models inadequately capture. Studies reveal that humans take longer paths and exhibit increased stress levels when navigating near robots that fail to communicate intent clearly [57].

Future research must address:

- Developing models of human behavior that account for adaptation to robot presence
- Designing robot behaviors that effectively communicate intent without explicit communication channels
- Understanding cultural variations in navigation conventions and personal space norms
- Establishing ethical frameworks for robot navigation in shared spaces, balancing efficiency against human comfort and autonomy [58].

D. Multi-Modal Perception and Sensor Fusion

Most current navigation systems rely primarily on geometric information (positions and velocities) while ignoring rich contextual cues available through multi-modal sensing. Visual appearance, pose estimation, gaze direction, and social grouping structures provide valuable signals for predicting pedestrian intentions and navigating more effectively [59]. Integrating these diverse data sources presents both technical and architectural challenges.

Vision transformers and multi-modal learning architectures that jointly process visual, geometric, and semantic information show promise for enhancing navigation capabilities. However, computational constraints remain significant processing high-resolution visual data in real-time while maintaining responsive control loops requires careful architectural design and hardware acceleration [60]. Attention mechanisms that selectively focus computational resources on task-relevant information may enable practical multi-modal navigation systems.

E. Scalability to Large-Scale Multi-Agent Systems

Scaling navigation algorithms to hundreds or thousands of agents introduces qualitatively new challenges. Centralized coordination becomes computationally infeasible, necessitating decentralized or hierarchical approaches. Communication constraints limit information sharing, while maintaining global coherence without explicit coordination remains difficult [61].

Graph neural networks provide a promising framework for learning scalable multi-agent policies, representing agent interactions as message passing on dynamic graphs [62]. Hierarchical reinforcement learning decomposes navigation into strategic planning at macro timescales and reactive control at micro timescales, potentially enabling coordination at scale. Swarm robotics principles, where simple local rules produce complex collective behaviors, offer alternative paradigms for large-scale coordination without centralized control [63].

VIII. CONCLUSION

This paper has presented a comprehensive survey and analysis of autonomous multi-agent navigation in crowded environments, examining the theoretical foundations, algorithmic approaches, and practical challenges that define this critical research area. We have organized navigation methods into three primary classes geometric velocity-based approaches, physics-inspired social force models, and data-driven learning methods each offering distinct advantages and facing specific limitations.

Velocity obstacle methods, particularly ORCA, provide computationally efficient solutions with formal safety guarantees under idealized assumptions. Their widespread adoption in commercial applications validates their practical utility in structured environments. However, performance degradation in dense crowds and limited social awareness constrain applicability in less structured human environments. Social force models successfully capture emergent crowd phenomena and produce natural-looking trajectories but require careful parameter tuning and lack robust stability guarantees. Deep reinforcement learning approaches demonstrate impressive capabilities for learning socially-compliant behaviors from experience but face significant challenges in safety certification, sample efficiency, and sim-to-real transfer.

Our comparative analysis reveals that no single approach dominates across all performance dimensions. The optimal choice depends critically on application requirements: safety-critical deployments favor geometric methods with provable properties, while scenarios prioritizing naturalness and social compliance benefit from learning-based approaches. Hybrid architectures that combine complementary strengths of multiple paradigms emerge as particularly promising, achieving superior performance by leveraging geometric methods for safety-critical short-term planning while employing learned components for strategic reasoning and adaptation.

Looking forward, several research directions appear critical for advancing the state-of-the-art. First, bridging the gap between learning-based flexibility and formal safety guarantees through verification techniques, constrained architectures, and runtime monitoring systems will enable deployment in safety-critical applications. Second, improving generalization capabilities through meta-learning, transfer learning, and more sophisticated domain randomization will reduce the brittleness of current learned policies. Third, incorporating richer perceptual information through multi-modal learning will enable more sophisticated reasoning about pedestrian intentions and environmental context.

Fourth, developing principled frameworks for human-robot interaction that account for coupled dynamics and communicate intent effectively will improve human comfort and acceptance. Fifth, scaling algorithms to large agent populations through graph neural networks, hierarchical methods, and decentralized coordination strategies will enable deployment in large-scale scenarios. Finally, establishing unified benchmarks and evaluation protocols will facilitate fair comparison and accelerate progress by clearly identifying the most promising research directions.

The field of autonomous multi-agent navigation has matured significantly over the past two decades, transitioning from purely theoretical investigations to practical deployments in real-world environments. Yet substantial challenges remain before robots can navigate crowded human spaces with the fluency and social intelligence of human pedestrians. Addressing these challenges will require continued innovation across multiple disciplines robotics, machine learning, human-computer interaction, and formal methods—alongside sustained efforts to validate approaches in diverse real-world scenarios. The potential impact of success is substantial: enabling safe, efficient, and socially-aware robot navigation will unlock applications ranging from assistive healthcare robotics to autonomous delivery systems, ultimately enhancing quality of life through intelligent autonomous systems that seamlessly integrate into human environments.

REFERENCES

[1] P. Trautman and A. Krause, "Unfreezing the robot: Navigation in dense, interacting crowds," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, 2010, pp. 797–803.

[2] S. M. LaValle, *Planning Algorithms*. Cambridge, U.K.: Cambridge Univ. Press, 2006.

[3] D. Hsu, R. Kindel, J.-C. Latombe, and S. Rock, "Randomized kinodynamic motion planning with moving obstacles," *Int. J. Robot. Res.*, vol. 21, no. 3, pp. 233–255, 2002.

[4] P. Fiorini and Z. Shiller, "Motion planning in dynamic environments using velocity obstacles," *Int. J. Robot. Res.*, vol. 17, no. 7, pp. 760–772, 1998.

[5] J. van den Berg, M. Lin, and D. Manocha, "Reciprocal velocity obstacles for real-time multi-agent navigation," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, 2008, pp. 1928–1935.

[6] D. Helbing and P. Molnar, "Social force model for pedestrian dynamics," *Phys. Rev. E*, vol. 51, no. 5, pp. 4282–4286, 1995.

[7] Y. F. Chen, M. Liu, M. Everett, and J. P. How, "Decentralized non-communicating multiagent collision avoidance with deep reinforcement learning," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, 2017, pp. 285–292.

[8] C. Chen, Y. Liu, S. Kreiss, and A. Alahi, "Crowd-robot interaction: Crowd-aware robot navigation with attention-based deep reinforcement learning," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, 2019, pp. 6015–6022.

[9] M. Everett, Y. F. Chen, and J. P. How, "Motion planning among dynamic, decision-making agents with deep reinforcement learning," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, 2018, pp. 3052–3059.

[10] J. Kober, J. A. Bagnell, and J. Peters, "Reinforcement learning in robotics: A survey," *Int. J. Robot. Res.*, vol. 32, no. 11, pp. 1238–1274, 2013.

[11] P. Henry, C. Vollmer, B. Ferris, and D. Fox, "Learning to navigate through crowded environments," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, 2010, pp. 981–986.

[12] T. I. Fossen, *Handbook of Marine Craft Hydrodynamics and Motion Control*. Chichester, U.K.: Wiley, 2011.

[13] S. J. Guy et al., "ClearPath: Highly parallel collision avoidance for multi-agent simulation," in *Proc. ACM SIGGRAPH/Eurographics Symp. Comput. Animat.*, 2009, pp. 177–187.

[14] D. Hsu, J.-C. Latombe, and R. Motwani, "Path planning in expansive configuration spaces," *Int. J. Comput. Geom. Appl.*, vol. 9, no. 4–5, pp. 495–512, 1999.

[15] M. Tipping, "Sparse Bayesian learning and the relevance vector machine," *J. Mach. Learn. Res.*, vol. 1, pp. 211–244, 2001.

[16] P. Trautman, J. Ma, R. M. Murray, and A. Krause, "Robot navigation in dense human crowds: Statistical models and experimental studies of human-robot cooperation," *Int. J. Robot. Res.*, vol. 34, no. 3, pp. 335–356, 2015.

[17] E. T. Hall, *The Hidden Dimension*. Garden City, NY: Doubleday, 1966.

[18] T. Kruse, A. K. Pandey, R. Alami, and A. Kirsch, "Human-aware robot navigation: A survey," *Robot. Auton. Syst.*, vol. 61, no. 12, pp. 1726–1743, 2013.

[19] A. Rudenko et al., "Human motion trajectory prediction: A survey," *Int. J. Robot. Res.*, vol. 39, no. 8, pp. 895–935, 2020.

[20] J. van den Berg, S. J. Guy, M. Lin, and D. Manocha, "Reciprocal n-body collision avoidance," in *Robotics Research*, Berlin: Springer, 2011, pp. 3–19.

[21] S. J. Guy et al., "A statistical similarity measure for aggregate crowd dynamics," *ACM Trans. Graph.*, vol. 31, no. 6, pp. 190:1–190:11, 2012.

[22] J. van den Berg et al., "Interactive navigation of multiple agents in crowded environments," in *Proc. ACM SIGGRAPH Symp. Interactive 3D Graphics Games*, 2008, pp. 139–147.

[23] J. Snape, J. van den Berg, S. J. Guy, and D. Manocha, "The hybrid reciprocal velocity obstacle," *IEEE Trans. Robot.*, vol. 27, no. 4, pp. 696–706, 2011.

[24] D. Helbing, I. Farkas, and T. Vicsek, "Simulating dynamical features of escape panic," *Nature*, vol. 407, pp. 487–490, 2000.

[25] A. Johansson, D. Helbing, and P. K. Shukla, "Specification of the social force pedestrian model by evolutionary adjustment to video tracking data," *Adv. Complex Syst.*, vol. 10, no. 2, pp. 271–288, 2007.

[26] M. Moussaïd, D. Helbing, and G. Theraulaz, "How simple rules determine pedestrian behavior and crowd disasters," *Proc. Nat. Acad. Sci.*, vol. 108, no. 17, pp. 6884–6888, 2011.

[27] F. Zanlungo, T. Ikeda, and T. Kanda, "Social force model with explicit collision prediction," *Europhys. Lett.*, vol. 93, no. 6, p. 68005, 2011.

[28] M. J. Seitz and G. Köster, "Natural discretization of pedestrian movement in continuous space," *Phys. Rev. E*, vol. 86, no. 4, p. 046108, 2012.

[29] M. Chraibi, A. Seyfried, and A. Schadschneider, "Generalized centrifugal-force model for pedestrian dynamics," *Phys. Rev. E*, vol. 82, no. 4, p. 046111, 2010.

[30] C. F. Borst and W. H. K. de Vries, "Efficient data structures for spatial crowd simulation," in *Proc. Motion Games*, 2011, pp. 138–149.

[31] J. Ondřej, J. Pettré, A.-H. Olivier, and S. Donikian, "A synthetic-vision based steering approach for crowd simulation," *ACM Trans. Graph.*, vol. 29, no. 4, pp. 123:1–123:9, 2010.

[32] A. Alahi et al., "Social LSTM: Human trajectory prediction in crowded spaces," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2016, pp. 961–971.

[33] A. Gupta et al., "Social GAN: Socially acceptable trajectories with generative adversarial networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2018, pp. 2255–2264.

[34] P. Zhang et al., “SR-LSTM: State refinement for LSTM towards pedestrian trajectory prediction,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2019, pp. 12085–12094.
[35] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, 2nd ed. Cambridge, MA: MIT Press, 2018.
[36] J. N. Foerster et al., “Counterfactual multi-agent policy gradients,” in *Proc. AAAI Conf. Artif. Intell.*, 2018, pp. 2974–2982.
[37] P. Long et al., “Towards optimally decentralized multi-robot collision avoidance via deep reinforcement learning,” in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, 2018, pp. 6252–6259.
[38] D. Hafner, T. Lillicrap, J. Ba, and M. Norouzi, “Dream to control: Learning behaviors by latent imagination,” in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2020.
[39] K. P. Murphy, *Machine Learning: A Probabilistic Perspective*. Cambridge, MA: MIT Press, 2012.
[40] Y. F. Chen, M. Everett, M. Liu, and J. P. How, “Socially aware motion planning with deep reinforcement learning,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, 2017, pp. 1343–1350.
[41] D. Zhou, Z. Wang, S. Bandyopadhyay, and M. Schwager, “Fast, on-line collision avoidance for dynamic vehicles using buffered Voronoi cells,” *IEEE Robot. Autom. Lett.*, vol. 2, no. 2, pp. 1047–1054, 2017.
[42] B. Brito, M. Everett, J. P. How, and J. Alonso-Mora, “Where to go next: Learning a subgoal recommendation policy for navigation among pedestrians,” in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, 2021, pp. 5616–5622.
[43] M. Everett, Y. F. Chen, and J. P. How, “Collision avoidance in pedestrian-rich environments with deep reinforcement learning,” *IEEE Access*, vol. 9, pp. 10357–10377, 2021.
[44] X. B. Peng et al., “Sim-to-real transfer of robotic control with dynamics randomization,” in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, 2018, pp. 3803–3810.
[45] J. Tan et al., “Sim-to-real: Learning agile locomotion for quadruped robots,” in *Proc. Robot. Sci. Syst. (RSS)*, 2018.
[46] M. Pfeiffer et al., “From perception to decision: A data-driven approach to end-to-end motion planning for autonomous ground robots,” in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, 2017, pp. 1527–1533.
[47] M. Luber, L. Spinello, J. Silva, and K. O. Arras, “Socially-aware robot navigation: A learning approach,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, 2012, pp. 902–907.
[48] S. Liu et al., “Decentralized structural-RNN for robot crowd navigation with deep reinforcement learning,” in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, 2021, pp. 3517–3524.
[49] D. Butterworth, “Optimizing robot motion for multi-agent systems,” in *Proc. Int. Symp. Robot. Res. (ISRR)*, 2019, pp. 245–258.
[50] J. Han, M. Kwak, and T. Y. Kim, “Efficient neural network compression,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 10, pp. 3548–3560, 2021.
[51] W. Xiao, R. Mehdipour, E. Colgate, and M. Peshkin, “Formal verification of neural network controlled autonomous systems,” in *Proc. ACM/IEEE Int. Conf. Cyber-Phys. Syst. (ICCPS)*, 2019, pp. 147–157.
[52] A. D. Ames, X. Xu, J. W. Grizzle, and P. Tabuada, “Control barrier function based quadratic programs for safety critical systems,” *IEEE Trans. Autom. Control*, vol. 62, no. 8, pp. 3861–3876, 2017.
[53] G. Katz et al., “Reluplex: An efficient SMT solver for verifying deep neural networks,” in *Proc. Int. Conf. Comput. Aided Verif. (CAV)*, 2017, pp. 97–117.
[54] K. Bousmalis et al., “Using simulation and domain adaptation to improve efficiency of deep robotic grasping,” in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, 2018, pp. 4243–4250.
[55] C. Finn, P. Abbeel, and S. Levine, “Model-agnostic meta-learning for fast adaptation of deep networks,” in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2017, pp. 1126–1135.
[56] J. Tobin et al., “Domain randomization for transferring deep neural networks from simulation to the real world,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, 2017, pp. 23–30.
[57] T. Kruse et al., “Legible robot navigation in the proximity of moving humans,” in *Proc. IEEE Workshop Adv. Robot. Social Impacts*, 2012, pp. 83–88.
[58] A. D. Dragan, K. C. T. Lee, and S. S. Srinivasa, “Legibility and predictability of robot motion,” in *Proc. ACM/IEEE Int. Conf. Human-Robot Interact. (HRI)*, 2013, pp. 301–308.
[59] A. Vemula, K. Muelling, and J. Oh, “Social attention: Modeling attention in human crowds,” in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, 2018, pp. 4601–4607.
[60] A. Vaswani et al., “Attention is all you need,” in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2017, pp. 5998–6008.
[61] M. Alonso-Mora, P. Beardsley, and R. Siegwart, “Cooperative collision avoidance for nonholonomic robots,” *IEEE Trans. Robot.*, vol. 34, no. 2, pp. 404–420, 2018.
[62] J. Li, H. Ma, W. Zhang, and S. Koenig, “Multi-agent path finding with mutex propagation,” *Artif. Intell.*, vol. 311, p. 103766, 2022.
[63] E. Tolstaya et al., “Learning decentralized controllers for robot swarms with graph neural networks,” in *Proc. Conf. Robot Learn. (CoRL)*, 2020, pp. 671–682.