

Continuous Dynamic Autonomy via Path Prediction on Semantic Topological Maps

Xuning Yang¹ and Jean Oh¹

Abstract— This paper presents a *continuous dynamic autonomy* framework for human-controlled navigation, such that the robot seamlessly navigates with semi-autonomy with human directly controlling the robot. In assisted navigation, the robot needs to first reason about the operator’s intention given the navigation context. We propose incorporating environment context into the navigation framework via online path predictions given a semantically topological navigation graph that represents general traversability. Such path predictions are then used in autonomous navigation wherever possible. Continuous dynamic autonomy ensures that the human-robot system maintains continuity in the human interaction and the reference trajectory for the robot avoiding destabilizing hand-overs while engaging human only when required. The result is a system that is maximally efficient, such that the human intervenes only necessarily, resulting in minimal operator engagement while retaining smooth navigation performance.

I. INTRODUCTION

Dynamic autonomy has been introduced to allocate control between the human vs. robot autonomy. In continuous settings such as navigation, the most efficient framework should *maximize the strengths of the human and robot wherever possible*. However, dynamic autonomy with discrete hand-offs in continuous tasks such as driving or remote piloting have shown to be destabilizing as sudden changes introduce discontinuities in the control reference. We propose a novel human-in-the-loop framework with *continuous dynamic autonomy* such that human-robot maintains continuity in the human interaction (*continuous-interaction*) and the reference trajectory for the robot (*continuous-control*) (Fig. 2). Our method is driven by two key principles: 1) Incorporate the environment information as a contextual prior for the human’s decision making during navigation and 2) Ensure continuous hand-over and trajectory generation.

Consider the two scenarios in Fig. 3: A hazardous warehouse with collapsed shelves, and a wooded area with rough terrain. Both scenarios follow a simplified task structure: navigate, inspect the scene, and return to the start. This task structure is best suited for human-in-the-loop control due to the unpredictability of the scenario. In both of these cases, some partial map may be available. However, this information is *imperfect* as unexpected obstacles changes traversability assumptions.

The partial availability of a map can provide rich, contextual information for semi-autonomy. This information is especially informative in reasoning about the operator’s intentions. In constrained environments, the paths that the

¹Xuning Yang and Jean Oh are with the Robotics Institute, Carnegie Mellon University, Pittsburgh PA 15232, USA xuningy@gmail.com, jeanoh@cmu.edu

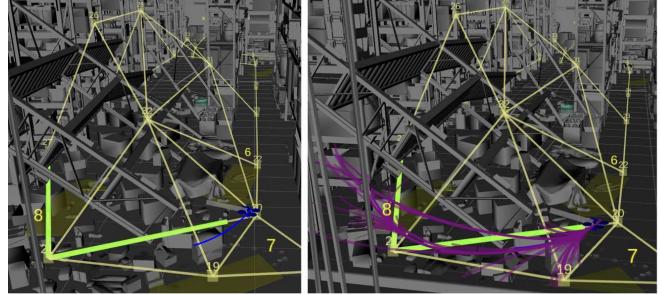


Fig. 1: Predicted path (green) on a semantically topological graph (yellow) used to generate local trajectory candidates (magenta) during dynamic autonomy.

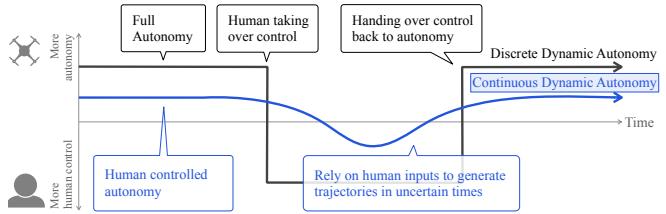


Fig. 2: Discrete dynamic autonomy is highlighted by distinct hand-offs between human and robot. We propose the idea of continuous dynamic autonomy, where assistance is provided as needed and does not require explicit communication and hand-off between human and robot.

robot can undertake are enumerable and distinct; e.g., hallways, narrow set of corridors, tunnels, or areas with difficult terrains. Therefore, the environment information for navigation purposes can be succinctly condensed into a semantically representative topological navigation map for decision making.

This leads to a key insight that *the role of the human in human-in-the-loop (HITL) navigation in constrained environments can be reduced to selecting from a set of discrete, traversable paths*. By incorporating the environment context as a semantically topological navigation graph, the robot’s autonomy can be further increased by predicting the path

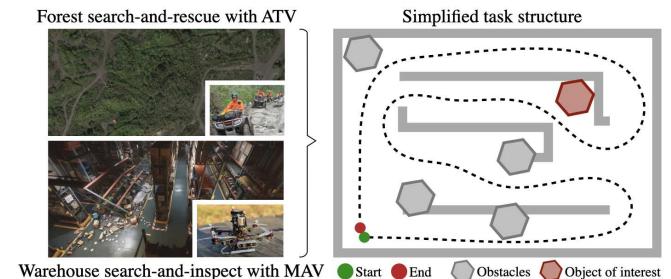


Fig. 3: Motivating scenarios for human-in-the-loop navigation. The example navigation tasks on the left follow a simplified task structure: navigate and identify/search/inspect along the path, and return to the start. Continuous dynamic autonomy seeks to assist teleoperation in these scenarios without well-defined goals.

that the human intends to follow. We formulate online global path prediction by updating beliefs about the set of discrete, distinctive paths from the navigation graph as the robot moves. In certain scenarios where the human provides an indicative input that deviates from the predictions, control of trajectory generation is handed over to the human operator until prediction is re-aligned. The outcome is that the transition between human control and robot is imperceptible.

This paper proposes *continuous dynamic autonomy* in a hierarchical HITL framework by generating path predictions on a semantically topological navigation graph and discusses continuous hand-over at the trajectory level. We evaluate our method with a pilot study in a cluttered warehouse environment in a navigation task, and show that assistance via continuous dynamic autonomy increases human-robot efficiency by shifting control from the human to the robot.

II. PRELIMINARIES

We generate trajectories in the flat output space of the multirotor, i.e., $\mathbf{x} = [x, y, z, \theta]$ [1]. A path ξ is a sequence of edges that connects a set of ordered waypoints. A trajectory is a time-parameterized function $\zeta(t)$, defined over a time interval $t \in [0, T]$ that maps a given time t to a state \mathbf{x}_t :

$$\zeta_T : [0, T] \rightarrow \mathcal{X} \quad T \in [0, \infty), \zeta(0) = \mathbf{x}(0) \quad (1)$$

Operator's inputs are given in the form of $u_h = [v_x, \omega, v_z]^\top$ via a joystick, where $v_x \in \mathcal{V}_x$ is the linear velocity, $\omega \in \Omega$ is the angular velocity, and $v_z \in \mathcal{V}_z$ is the z -velocity. A motion primitive $\gamma(t)$ is a parameterized trajectory function which generates a unique sequence of states given an initial state $\mathbf{x}_0 \in \mathcal{X}$ and an input $\mathbf{a} \in \mathcal{A}$, $\mathcal{A} = \mathcal{V}_x \times \Omega \times \mathcal{V}_z$ according to specific dynamics:

$$\gamma_{\mathbf{a}, T} : [0, T] \rightarrow \mathcal{X} \quad \mathbf{a} \in \mathcal{A}, T \in [0, \infty), \mathbf{x}(0) = \mathbf{x}_0 \quad (2)$$

$$\text{s.t. } \dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{a}) \quad (3)$$

To parameterize the operator input into a motion primitive, we have $\mathbf{a} = u_h$. We select the unicycle model for motion primitives [2] and generate trajectories such that continuity up to snap is retained.

A. Topological map representation

We seek a simplified graph representation that captures the free-space connectivity. Search-based methods such as A* utilizes shortest-distance heuristics to generate paths between start and end goal configurations. Sample-based methods such as Probabilistic Roadmap (PRM) [3] and RRT [4] results in high density vertices and edges, but fails to capture distinct topologies of the environment.

In this work, we assume the existence of a *semantically topological navigation graph* [5]: a minimally representative, undirected navigation graph where each path formed by connected vertices are homotopically distinct, and represent free-space that is traversable. This means that paths in the map must be *homotopically distinct*, i.e., every path represents a different homotopy class¹, and that they are *traversable* by

¹Two paths are said to be in the same homotopy class iff one can be smoothly deformed into the other without intersecting obstacles [6].

the robot.

B. IIA and choice theory

The above assumptions give rise to an important characteristic of the paths on a semantically topological navigation graph. The homotopically distinct paths can be assumed to be *independent given irrelevant alternatives*; that is, each choice of path is independent and are non-substitutable given the agent's preference for each choice.

The implication of the IIA assumption is that the human's choice model can be modeled using the Boltzmann rationality decision model [7], where the human is assumed to approximately optimize a reward functions given a set of choices [8–10].

$$P(o) = \frac{\exp(R(o))}{\sum_{o' \in \mathcal{O}} \exp(R(o'))} \quad (4)$$

III. METHODOLOGY

The algorithm involves two components: 1) Path prediction on a semantically topological navigation graph, and 2) continuous dynamic autonomy.

A. Path prediction on navigation graph

a) *Path extraction from Graph*: Given a semantically topological graph $G = \{V^k\}$, $k \in [1, K]$ with K vertices V^k , and the robot's current state, we extract a set of possible paths from the graph. As the robot navigates, the approaching vertex becomes the root node, V^0 , of a *directed tree* with a finite horizon. Each branch j in the navigation graph is a path, $\xi_j = [V_{j,0}, V_{j,1}, \dots, V_{j,N_j}]$ with N_j vertices, composed of $N_j - 1$ segments $s_{j,i} = \{V_{j,i-1}, V_{j,i}\}$. Note that, each one of the vertex in the path $V_{j,i} \in \xi_j$ will map to a vertex on the graph, $V^k \in G$, however, many vertices on the path may map to the same node on the graph as segments are shared between paths. See Fig. 4 for a detailed illustration of the path tree extraction.

b) *Receding horizon observation model*: To generate a prediction, we use a simplified model of vehicle behavior. At time t , given a window of M past state observations $x_{\{t-M\}:t}$, we evaluate each path with an evaluation cost function. We drop the superscript for readability.

Given a cost evaluation function c , we integrate the cost along the path:

$$C(x, \xi) = \int_{\xi} c(x, \xi) w(\xi) d\xi \quad (5)$$

Notice the addition of a weighting function $w(\xi) : [0, 1] \rightarrow \mathbb{R}^+$. This model a function of the previous state observations x , therefore further down, the previous observations x are less likely to affect choices and costs in the future. The weighting function considers the effects of time on the cost.

For a discretized path $\xi = [s_1, s_2, \dots, s_N]$ with N segments, this becomes:

$$C(x, \xi) = \frac{1}{N} \sum_{n=1}^N c(x, s_n) w(s_n) \quad (6)$$

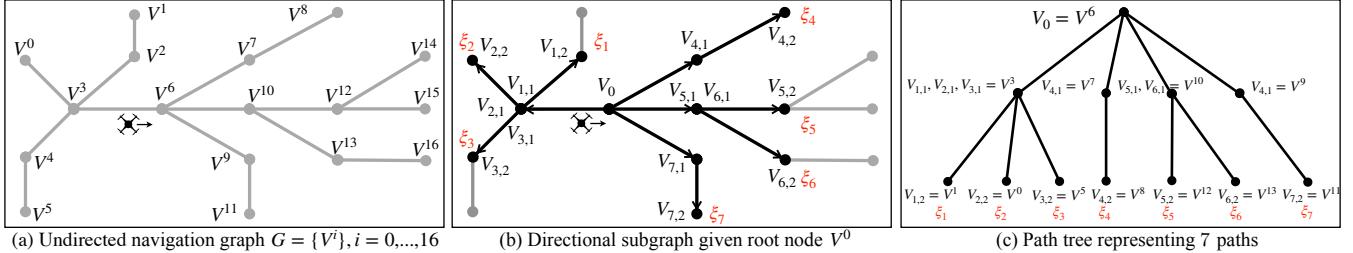


Fig. 4: Extracting a path tree from a topological graph, given the current vehicle position. From a single node, the undirected graph can be turned into a directed tree.

c) *Segment wise cost* $c(x, s)$: Given a segment of the path s and x , we sample N discretizations along the observation and the path segment, s.t., $\vec{x}_i = x_i - x_{i-1}$, and $\vec{s}_i = s(i) - s(i-1)$ with $s(i)$ indicates the i th discretization of the segment s . A point-wise projection is given by:

$$\text{proj}(x, s) = \frac{1}{D} \sum_{i=1}^D \frac{\vec{x}_i}{\|\vec{x}_i\|} \cdot \frac{\vec{s}_i}{\|\vec{s}_i\|} = \frac{1}{D} \sum_{i=1}^D \hat{x}_i \cdot \hat{s}_i \quad (7)$$

Then, shift the projection such that the cost is positive:

$$c(x, s) = (1 - \text{proj}(x, s)) \quad (8)$$

d) *Path prediction*: To compute the probability of a path $\xi \in \Xi$ given x , we use the Boltzmann's rationality decision model. Starting with Bayes rule,

$$p(\xi|x) = \frac{p(x|\xi)p(\xi)}{p(x)} \quad \xi \in \Xi \quad (9)$$

we follow the principle of max entropy to induce a distribution over the set of paths, which models the probability of a path decreasing exponentially with the cost C . The likelihood, $p(x|\xi)$, is given by:

$$p(x|\xi) \propto \exp(-C(x, \xi)) \quad (10)$$

Substituting Eq. (10) into Eq. (11) and using η as a normalization factor:

$$p(\xi|x) = \eta \cdot \exp \left(-\frac{1}{N} \sum_{n=1}^N c(x, s_n) w(s_n) \right) p(\xi) \quad (11)$$

B. Continuous Dynamic Autonomy

When humans are interacting with the robot, their actions gives great insight with respect to their underlying intention. Assuming the human is observant and not distracted, they could provide no inputs as the robot navigates, which implies that they agree with the motion of the vehicle. However, if they provide an input, this indicates that they disagree with the motion of the vehicle and would like to change it. We leverage this key insight in constructing continuous dynamic autonomy framework as follows:

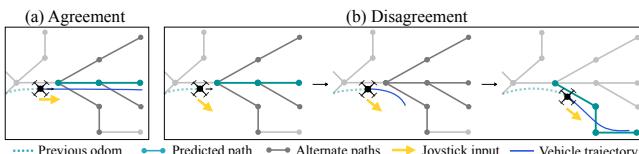


Fig. 5: Illustration of prediction agreement. As the predicted path diverges from the operator input, the input is used to directly generate a dynamically feasible motion primitive. As the prediction becomes confident again, full autonomy takes over.

e) *Prediction checking*: Predictions are continuously evaluated against the human input u_h as to whether the operator agrees with the prediction. We do so by computing the dot product of the joystick input u_h with the normalized vector representing the first segment of the path, i.e., the immediate next step:

$$u_h \cdot \vec{S}_j = u_h \cdot \frac{V_{j,1} - V_{j,0}}{\|V_{j,1} - V_{j,0}\|}$$

Then, agreement is evaluated by normalizing over all of the first segment paths stemming from the current vertex:

$$w_{\text{agreement}}(\xi_j, u_h) = \frac{(1 + u_h \cdot \vec{S}_j)/2}{\sum_j (1 + u_h \cdot \vec{S}_j)/2} \quad \xi_j \in \Xi \quad (12)$$

Ξ is the set of paths stemming from the current vertex V_0 .

f) *Trajectory Generation*: If the joystick input agrees, the robot operates fully autonomously where the predicted path ξ^* is incorporated within a hierarchical framework [11] to guide local trajectory generation. Any local trajectory generation method can be used. For this paper, we use Biased Incremental Action Sampling (BIAS) introduced in [12].

During disagreement, the operator input u_h is directly used to generate a motion primitive, following [2] with obstacle avoidance. The algorithm of continuous dynamic autonomy is illustrated in Algorithm 1.

Algorithm 1: Continuous Dynamic Autonomy

Input: Given a topological navigation graph G

- 1 **while** Robot navigating **do**
- 2 **Prediction**
- 3 Extract paths given vehicle state x_t : $\Xi = \{\xi\}$
- 4 Compute path prediction $p(\xi|x)$ via Eq. 9
- 5 Evaluate most likely path $\xi^* = \text{argmax}_{\xi} p(\xi|x)$
- 6 **Agreement**
- 7 **if** ξ^*, u_h agree **then**
- 8 Set global path $\xi_G = \xi^*$
- 9 Generate trajectory ζ
- 10 **else**
- 11 Set local trajectory to parameterized motion primitive
- 12 $\zeta = \gamma(u_h, x_t)$
- 13 **return** ζ



Fig. 8: Experiment setup: The secondary task is to identify a green box with a symbol. The operator is given a third person omniscient view, and the joystick inputs are given in the body frame of the vehicle.

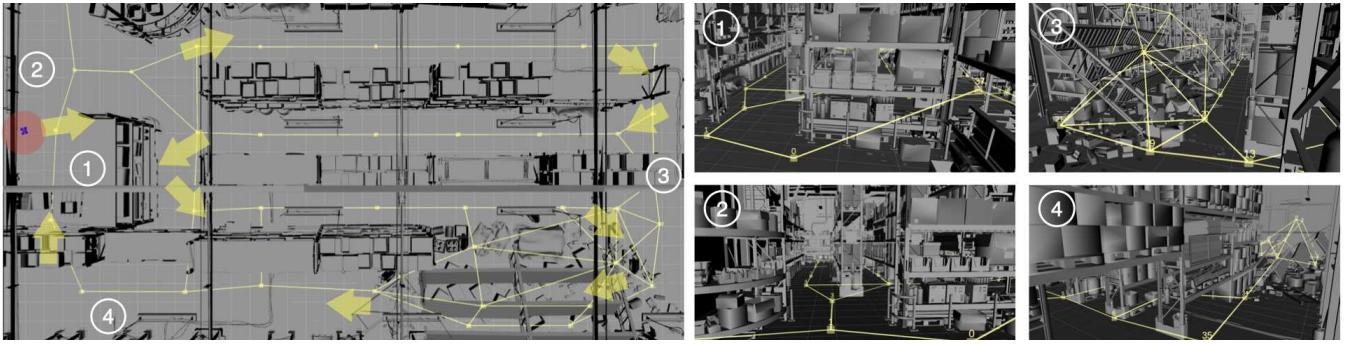


Fig. 6: Semantically topological navigation graph for the warehouse. Cutaway views of the navigation graph is shown at various viewpoints.

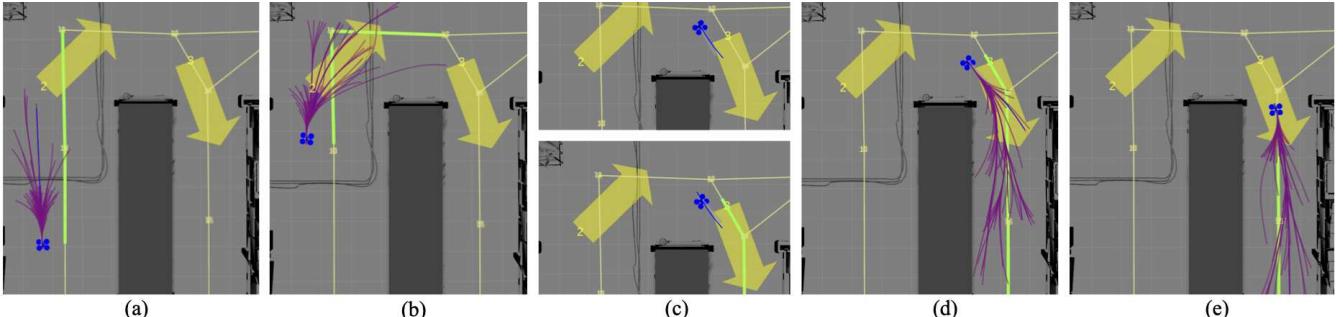


Fig. 7: A sequence of continuous dynamic autonomy switching for rounding a corner. The operator disagrees with the predicted path (green) from (b) to (c), and thus a human-parameterized trajectory (blue) is generated. As the prediction aligns again, autonomy resumes (d-e).

IV. EXPERIMENTS

A. Experiment design

The task is to navigate in a densely cluttered warehouse environment ($45\text{m} \times 22\text{m} \times 11\text{m}$) in simulation following a path described verbally to the operator and indicated by yellow arrows, and return to the red landing pad at the origin as shown in Fig. 6. The operators were also asked to look out for a randomly placed green box along the path as a secondary task, in order to simulate attention division similar to search-and-rescue. The simulated quadrotor is controlled via a joystick, specifying the forward, side, angular, and z velocities scaled according to a max velocity parameter, set to 1.5 m/s. The operator is given a third-person omniscient view with only the guiding arrows visible (Fig. 8). The generated trajectory was chosen to be *hidden* so as to simulate control without visual aids. This design choice and its implications will be discussed later in the results section. The semantically topological navigation graph used for the proposed method is shown in Fig. 6. Paths extracted from the graph are approximately 10m in length.

B. User study design

We conduct a pilot study ($n=10$) to evaluate the proposed method, continuous dynamic autonomy (DA) method against two methods of trajectory-based teleoperation: motion primitive teleoperation (MP) [2], and a velocity based teleoperation method (VEL). For all three methods, the joystick interface remains the same. However, the underlying dynamics that generates the trajectories are different: DA and MP both utilize the unicycle model, whereas for the VEL method, the yaw is decoupled from the heading.

The participants have no prior exposure to our system but have varying experience with teleoperating quadrotors. The

pilot follows a within-subjects design, where each participant used all three methods, A, B, and C. The ordering of the methods were randomized, such that A, B and C corresponded to one of VEL, MP, and DA. Prior to each trial, the participant was given a tutorial period of 3 minutes to test out the control and dynamics. As the controls required for all three systems are the same, no additional details were provided about the controls. However, note that, since the underlying dynamics that generates the trajectories for MP and DA are the same, the control during the tutorial period are exactly the same.

C. Hypotheses

The experiments in this pilot study aims to evaluate human-robot efficiency by way of operator engagement. The operator engagement is evaluated by using number of joystick inputs used per trial. Operator preference is evaluated via a survey post-trial. The hypotheses are:

H1 The system will require *less* direct human control and will navigate *mostly with autonomy* with DA.

H2 Operators will engage with the system *less* when using DA, leading to reduced number of inputs during navigation.

H3 Participants will prefer DA over direct control methods VEL and MP.

V. RESULTS

A. Assistance and Operator Effort

1) Qualitative observations: The operator behavior using DA, is shown in Fig. 9. The plot highlights (1) The number of possible path choices during operation, and (2) the operator's input and how it agreed with the predicted path. As the vehicle approaches many decision points, the operator tends to stop, and proceed with caution. This behavior is

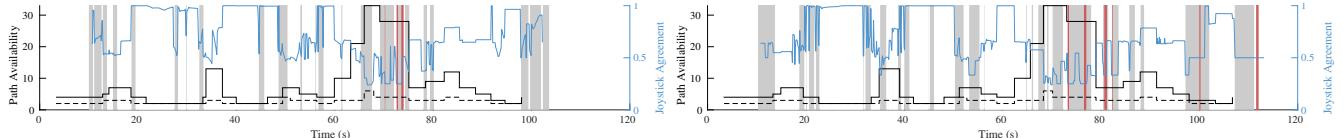


Fig. 9: Operator behavior over time for two select trials, highlighting two observations. 1) The number of path choices/immediate edge choices vs. the type of motion (shaded). Observe that the vehicle is stopped more frequently near areas of increased path choices. 2) Joystick agreement vs. type of motion. As prediction and human's inputs align, vehicle navigates mostly autonomously.

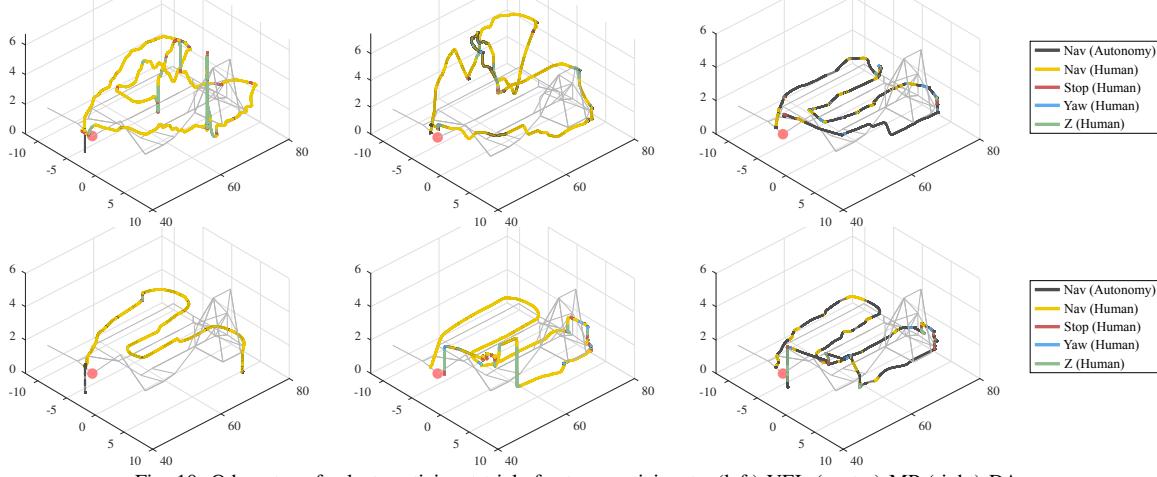


Fig. 10: Odometry of select participant trials for two participants. (left) VEL (center) MP (right) DA

observed across multiple participants with varying degrees of cautiousness (high caution is indicated by many stops). However, this is not limited to DA; it is observed across all three methods (Fig. 10).

2) Assistance: We categorize motion into 4 modes: (1) **Stop** (2) **In-place yaw**, (3) **Z** and (4) **Nav**, which means that the input provided includes non-zero inputs along the x - y plane. We primarily focus on **Nav**, as DA is invoked only during navigation. Fig. 10 shows some example odometries of the three methods. We observe that the vehicle is mostly autonomous for DA. We compute the amount of navigation done by direct human control. This result is tabulated in Table I, with an accompanying bar plot in Fig. 11. These results were assessed using a one-way repeated measures ANOVA. The results showed that the proposed method was able to reduce the human's role in navigation control from 86% to 24.5% ($F(2, 34)=88.0, p<.001$), strongly supporting **H1**. Having assistance allowed shifting navigation from human-controlled to robot, which increases the human-robot system efficiency.

TABLE I: Breakdown of odometry in each mode as a percentage of the total trajectory length. This is visually presented in Fig. 11.

	Stop (%)	Yaw (%)	Z (%)	Nav Human (%)	Nav Auton. (%)
VEL	5.1 ± 3.8	0.7 ± 0.8	2.2 ± 2.7	92 ± 5.5	N/A
MP	5.6 ± 4.7	1.4 ± 0.6	3.9 ± 1.7	89.1 ± 5.8	N/A
DA	7.5 ± 5.4	2.1 ± 1.4	4.5 ± 2.4	24.5 ± 6.4	62 ± 7.5

3) Trends in Operator Engagement: The number of inputs corresponding to each mode is shown in Table II. This results were assessed using a one-way repeated measures ANOVA. While we observe that the number of inputs received during **Nav** is lower than both of the comparison methods, this data is not supported by statistical significance and should be noted as a trend. We believe that this is due to the relatively small sample size and high variance in the number of inputs.

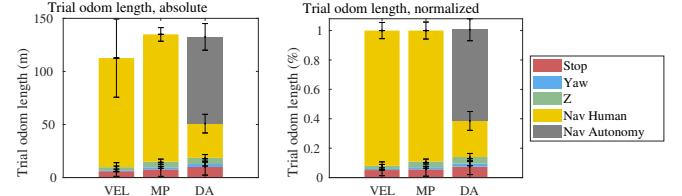


Fig. 11: Odometry length, broken down by modes. The human controlled navigation is reduced significantly, with operator directly controlling navigation approx 24.5% of the time. The tabulated result is in Tab. I.

Thus, **H2** could not be supported at this time. However, this trend combined with confirmed **H1** indicates that the system's assistance is effective at increasing human-robot efficiency. To validate this trend, we believe the proposed method could benefit with a further study with larger sample sizes to further confirm this hypothesis with statistical significance.

TABLE II: Number of inputs for the warehouse navigation task

	Stop	Yaw	Z	Nav	Total
VEL	25 ± 19	14 ± 11	18 ± 14	219 ± 100	277 ± 113
MP	47 ± 57	50 ± 48	50 ± 46	238 ± 134	384 ± 234
DA	46 ± 36	31 ± 21	43 ± 22	187 ± 48	307 ± 115

B. Preference and Qualitative Observations

We additionally evaluate operator preference given the three methods. The post-trial survey asked the following questions for each method:

- 1) I find the controls to be natural/intuitive.
- 2) I find the controls to be comfortable to use.
- 3) I was able to stabilize the vehicle with ease.
- 4) I was able to navigate the vehicle with ease.
- 5) I was able to avoid obstacles with ease.
- 6) The vehicle performed the motion that I intended for it to do.

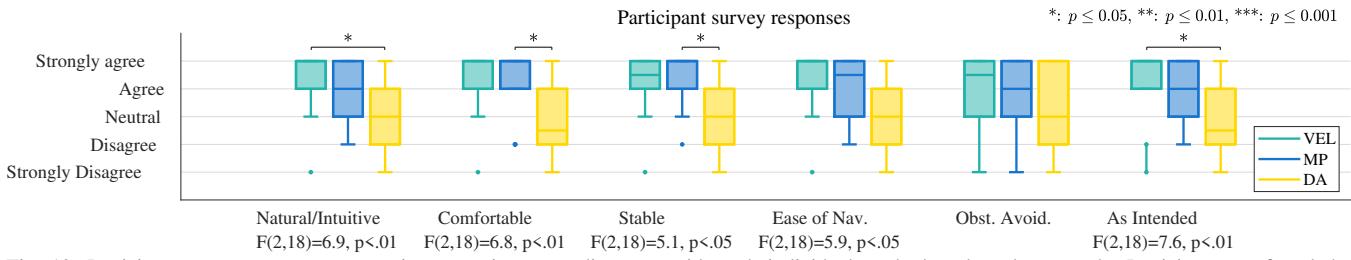


Fig. 12: Participant survey responses to various questions regarding use, with each individual method evaluated separately. Participants preferred the proposed DA method the least given its inability to respond to individual nudges.

The results are shown in Fig. 12, with a strong support against **H3**. The trends suggest that the operators preferred VEL, with the yaw decoupled from the vehicle’s heading. The proposed method, DA, was thought to be more difficult to use. This result leads to a *subjective vs. objective gap* in DA’s perceived helpfulness. We reason about this gap via the following observations:

a) Behaviors of naive vs. experienced operators: Naive operators tend to act more carefully by provide minor corrections. This is illustrated via “nudging” the system by flicking the joystick. Experienced operators are well versed with motion coupling and dynamics of the quadrotor. As the VEL system is a direct velocity parameterization of their inputs, they are more likely to prefer this method.

b) Subjective perceptions of operators: We note many subjective interpretations by the operators for the controls. For example, some participants remarked that “the controls (for DA) feels completely different than the previous (MP)” during the tutorial period. However, the controls and the underlying dynamics are exactly the same. Further, the resulting odometries of VEL and MP were qualitatively observed to be drifty and unstable – however, the operators still preferred them for controllability (Fig. 12). We hypothesize that, if we show videos of DA, MP and VEL to a separate group and ask similar questions, they would perceive that DA is more stable than VEL or MP. We leave this investigations as future work.

c) Sensitivity of controls: All three methods used the same set of parameters. Therefore, we attribute remarks on “sensitivity” to the differing trajectory generation methods: The participants expected the system to respond to minor adjustments in inputs. As the local trajectories generated by DA does not respond to minor adjustments, the participants remarked this to be unresponsive and difficult to control.

d) Interface design and behavioral changes: The generated trajectory was not visualized the operators. Therefore, many operators provided inputs based on what they perceive the robot will do at the immediate next step, even though the system’s current trajectory was safe and ideal. We hypothesize that adding visual feedback of the trajectory would change the operator’s interaction with the system and will help to increase “trust” of the robot, although we leave further investigations of these hypotheses to future work.

VI. LIMITATIONS AND FUTURE WORK

This paper presents a continuous dynamic autonomy framework, by generating path predictions on semantic topological maps. The contributions of this paper are two

folds: 1) path prediction on navigation graphs by way of a simple receding horizon model; and 2) continuous dynamic autonomy. This framework shows that complex environments with dense environment features can be eschewed in favor of simple representations that encode semantic traversability.

Yet, the work yields some surprising discoveries: While the human-robot performed efficiently with assistance, operators did not like having a system that was non responsive to small input changes. This gives us insight that in generating human-preferred trajectories, the system should respond to the small inputs in a meaningful way so as to communicate trust and acknowledgement. We leave these investigations to future work.

VII. ACKNOWLEDGEMENTS

The authors would like to thank Henny Admoni, Sanjiban Choudhury, and Alex Spitzer for insightful discussions regarding path prediction on navigation graphs; Helen Oleynikova and Fan Yang for helpful discussions regarding various topological graphs for navigation; and Ada Taylor on statistical analyses.

REFERENCES

- [1] D. Mellinger and V. Kumar, “Minimum snap trajectory generation and control for quadrotors,” in *2011 IEEE international conference on robotics and automation*. IEEE, 2011, pp. 2520–2525.
- [2] A. Spitzer, X. Yang, J. Yao, A. Dhawale, K. Goel, M. Dabhi, M. Collins, C. Boirum, and N. Michael, “Fast and agile vision-based flight with teleoperation and collision avoidance on a multirotor,” in *Int. Sym. on Exp. Robot. (ISER)*. Springer, 2018.
- [3] L. E. Kavraki, P. Svestka, J.-C. Latombe, and M. H. Overmars, “Probabilistic roadmaps for path planning in high-dimensional configuration spaces,” *IEEE transactions on Robotics and Automation*, vol. 12, no. 4, pp. 566–580, 1996.
- [4] S. M. LaValle *et al.*, “Rapidly-exploring random trees: A new tool for path planning,” 1998.
- [5] S. Thrun and A. Büken, “Integrating grid-based and topological maps for mobile robot navigation,” in *Proceedings of the national conference on artificial intelligence*, 1996, pp. 944–951.
- [6] S. Bhattacharya, “Search-based path planning with homotopy class constraints,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 24, no. 1, 2010.
- [7] A. R. Benson, R. Kumar, and A. Tomkins, “On the relevance of irrelevant alternatives,” in *Proceedings of the 25th International Conference on World Wide Web*, 2016, pp. 963–973.
- [8] H. J. Jeon, S. Milli, and A. D. Dragan, “Reward-rational (implicit) choice: A unifying formalism for reward learning,” 2020.
- [9] C. L. Baker, J. B. Tenenbaum, and R. R. Saxe, “Goal inference as inverse planning,” in *Proceedings of the Annual Meeting of the Cognitive Science Society*, vol. 29, no. 29, 2007.
- [10] O. Morgenstern and J. Von Neumann, *Theory of games and economic behavior*. Princeton university press, 1953.
- [11] X. Yang and N. Michael, “An intention guided hierarchical trajectory generation framework for trajectory-based teleoperation of mobile robots,” *2021 IEEE/RSJ International Conference on Robotics and Automation (ICRA)*, 2021.
- [12] ——, “Assisted mobile robot teleoperation with intent-aligned trajectories via biased incremental action sampling,” *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020.