Evaluating Path Planning in Human-Robot Teams

Quantifying Path Agreement and Mental Model Congruency

Brandon S. Perelman
Human Research and Engineering Directorate
US Army Research Laboratory
Aberdeen Proving Ground, MD
brandon.s.perelman.ctr@mail.mil

Shane T. Mueller
Department of Cognitive and Learning Sciences
Michigan Technological University
shanem@mtu.edu

Kristin E. Schaefer
Human Research and Engineering Directorate
US Army Research Laboratory
Aberdeen Proving Ground, MD
kristin.e.schaefer-lay.civ@mail.mil

***Abstract*—The integration of robotic systems into daily life is increasing, as technological advancements facilitate independent and interdependent decision-making by autonomous agents. Highly collaborative human-robot teams promise to maximize the capabilities of humans and machines. While a great deal of progress has been made toward developing efficient spatial path planning algorithms for robots, comparatively less attention has been paid to developing reliable means by which to assess the similarities and differences in path planning decisions and associated behaviors of humans and robots in these teams. This paper discusses a tool, the Algorithm for finding the Least Cost Areal Mapping between Paths (ALCAMP), which can be used to compare paths planned by humans and algorithms in order to quantify the differences between them, and understand the user's mental models underlying those decisions. In addition, this paper discusses prior and proposed future research related to human-robot collaborative teams. Prior studies using ALCAMP have measured path divergence in order to quantify error, infer decision-making processes, assess path memory, and assess team communication performance. Future research related to human-robot teaming includes measuring formation and path adherence, testing the repeatability of navigation algorithms and the clarity of communicated navigation instructions, inferring shared mental models for navigation among members of a group, and detecting anomalous movement.

Keywords—ALCAMP, Human-Robot Interaction, Human-Agent Teaming, Path Mapping, Trust, Shared Mental Models

I. Introduction

As autonomous, intelligent systems become more widely integrated into daily human life, the differences in decision-making processes between humans and robots will pose a challenge for successful collaboration in human-robot teams. The structures and cognitive processes underlying human decision-making are highly adaptive to produce appropriate behaviors in uncertain environments. But these processes are currently not directly reproducible in the development of artificial intelligence. Currently, robots are driven by algorithms that are logical by design, but the algorithm may not be optimized for producing the range of behaviors required in the real world. Differences in decision-making processes between human and robots often result in different solutions to problems. This has potentially serious implications for human trust in autonomous systems and the overall performance of human-robot teams.

People in general prefer not to cede decision-making authority to robots. This is in part due to the human team member's lack of knowledge and reasoning about the underlying decision-making processes of the robot, the risk associated with allowing the robot to make decisions, and mismatches in trust and self-confidence [1-3]. However, as operational environments become more uncertain, complex, and risky, independent and appropriate intelligent decision-making becomes essential for effective team performance. Robot authority should be allowed when the associated decisions are consistent with its designed capabilities that would also benefit the safety and success of the human team [4]. One such set of decisions is related to path planning and navigation.

Recent works in robotics engineering and human-robot interaction (HRI) have focused on creating algorithms for intelligent path planning and execution that are usable, trustworthy, and produce reliable movement for robots [5-7]. These algorithms are intended for implementation either in fully autonomous or mixed initiative systems designed to function in dynamic and probabilistic operational environments. The potential for a great deal of variance in the weighting of information during path planning in these high complexity environments can produce very different solutions to spatial problems. For example, in search and reconnaissance operations, this information includes terrain features [8] and weather conditions [9]. Some of this information may not be available or easily quantified for an algorithm, for example individual target differences such as stress and expertise [10]. Therefore, an autonomous robot's navigation behavior in uncertain conditions may appear to be unpredictable by human teammates because of differences in information access, mobility, or information weighting between the robot and

Research was sponsored by the US Army Research Laboratory. It was accomplished under Cooperative Agreement Number W911-NF-12-2-0019 and W911NF-16-2-0109. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Laboratory or the US Government. The US Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation herein

human. Incongruity between a human's preferred spatial solution to a given problem and the solution generated by the robot holds different implications for each of these systems. Therefore, increasing the user's situation awareness (as defined by [11]) of the robot's behaviors can assist the user in reasoning about the robot's intent. Improving situation awareness can increase appropriate trust in the robot's ability to fulfill its role in the team [3], [12].

A. *Mental Models and Path Planning*

A user's mental model is a structured, organized knowledge that describes, explains, and predicts a system's purpose, form, function, or state [13, 14], and in HRI supports a human's understanding of the capabilities and limitations of the robotic team member [15]. Often, a human team member's mental model is underdeveloped due to a lack of knowledge about the system's capabilities, limited training, or even task complexity. Therefore, expectations may not match the system's behaviors, leading to degradations in trust, even if the robot is making appropriate decisions [16]. Therefore, it is important to consider the HRI implications of implicit social cues when moving through a cluttered space [17], identifying and incorporating spatial affordances and reasoning for replanning [18], and effective communication [19] when developing the underlying path planning architectures. This is essential, as feedback and movement patterns can convey information that leads to user perceptions of the robot's intelligence [20], and the legibility and predictability of the navigation behaviors can impact perceived safety [21]. A challenge in HRI has been to objectively quantify and assess variations in these mental models. Doing so is important because the team member's mental models can directly impact trust and use of the robot [15]. Metrics of path divergence provide a means of objectively quantifying and assessing user mental models.

B. *Quantifying Navigation Algorithms*

In addition to HRI, there are a number of engineering problems related to robotics that can benefit from a quantitative metric of path divergence. These problems require quantifying an algorithm's ability to adhere to formations and paths, the repeatability of stochastic path planning algorithms, and the detection of anomalous behavior. The ability of an intelligent agent to adhere to its place within a formation has been recognized as an engineering problem for over 25 years in spacecraft [22], and more recently in unmanned aerial, e.g., [23-24] and ground [25-27] vehicle swarms. Comparing a vehicle's path with its optimal path within a swarm formation offers a performance metric of the vehicle's ability to maintain formation. These comparisons can be used to quantify hardware or software fitness for formation flight applications.

A related problem is measuring a robot's ability to adhere to a designated path. Deviations in a path can arise from global versus local decision-making in an algorithm, or environmental effects. For example, sliding is a significant cause of route deviation in off-road ground robots. Algorithms attempt to control for sliding [28], and comparing the planned path with the actual path taken by the robot can provide a useful error metric for quantifying the fitness of a hardware-software system for adhering to a designated path.

Stochastic navigation algorithms can adapt in uncertain environments, and therefore offer many advantages for robot navigation [29] and swarm control [30]. By nature, these algorithms will produce different paths on each run. Path mapping can provide a metric to quantify deviation of the traveled path from an optimal path. Calculating statistics over multiple runs would allow developers to assess an algorithm's performance in different situations, or with different input parameters. If robots are operating in close proximity to humans, then the predictability (via consistency) of their paths would be exceptionally important. Likewise, reducing uncertainty in dynamic systems will reduce the burden on other agents within the system.

Finally, detecting anomalous behavior from paths is an active area of research in commercial domains, such as a trajectory-based measure of divergence to detect illicit activity among taxi drivers [31] or detecting anomalous activity in aircrafts [32]. By comparing the planned route with the route that the agent travels, it is possible to detect anomalous behavior post-hoc or in real time and respond accordingly.

C. *Problem Summary*

This paper discusses the application of a specific algorithm, the Algorithm for finding the Least Cost Areal Mapping between Paths (ALCAMP). This algorithm compares solutions to spatial problems to produce quantitative metrics of divergence and correspondence, to research questions related to human-robot teaming. A tool to analyze and compare human and robot decision-making specifically related to path mapping provides potential benefits to the development of a quantifiable metric for identifying variations between human and robot mental models of spatial navigation, as well as a metric for quantifying algorithm development and testing. Outcomes of these findings can be used to identify and infer the underlying reasons for changes in trust, and impact future system design.

II. Path Mapping for HRI and Robotics Engineering

The topic of path mapping and analysis is a well-researched problem with a number of potential algorithmic solutions (see [33]). While there is a clear need for robust metrics for comparing routes, these traditional algorithms typically treat the paths as trajectories and do not address the specific needs of the robotics community. First, the mechanisms used to generate two paths may differ in ways that prohibit or challenge drawing a meaningful comparison using trajectory-based methods. For example, comparing traveled paths requires scaling or warping movement speeds if there is a great disparity in speed between the agents. In an extreme example, snake robots may be restricted to speeds of less than a few mph [34], whereas an MQ-1 Predator has a top speed of 135 mph [35]. In addition, the method used to collect the path data will produce different results, and trajectory-based measures cannot directly compare paths when one or both paths lacks timing information; for example, a robot's path tracked with GPS contains timing information, whereas plan generation does not produce timing information relevant for a comparison. A similar problem occurs when comparing paths that are reconstructed from memory, predicted, or are developed from another's instruction [33, 36]. Traveling and creating the path may be very slow, taking anywhere from minutes in experimental settings, to days in naturalistic settings. However, the reconstruction of a path from memory can be exceptionally rapid, requiring only seconds. The resultant paths differ greatly in terms of timing and the number of points. An algorithm used for this purpose must be robust to permit comparing paths with inconsistent or nonexistent timing data.

Second, the complex nature of the operational environment, along with the performance characteristics of the agents themselves, can produce paths that often self-intersect, cross over one another, end abruptly (in the case of link loss or destruction of the agent, for example), or produce parallel loops (in the case of aircraft performing loitering maneuvers). An algorithm useful for HRI should be robust to these characteristics of naturalistic paths through uncertain, often threatening spaces. ALCAMP is a highly robust path mapping algorithm that can be used to quantify path similarity for these types of operational environments.

III. ALCAMP

The ALCAMP algorithm (see R package [pathmapping], available from CRAN: the Comprehensive R Archive Network) is designed to map the correspondence and divergence between two paths [33]. Paths are defined as an ordered series of points in multidimensional space. Critically, paths differ from trajectories in that they ignore timing information. ALCAMP takes two paths as input, and produces an area-based measure of divergence (i.e., a single numerical value) as well as a mapping between both paths.

A. General Description of ALCAMP

ALCAMP defines a path as a series of points connected by line segments, and treats all points and segments on each path as nodes. The goal of the optimization process is to find the optimal mapping between *Paths A* and *B*. The algorithm then generates multiple *proper mappings*, in which each point on *Path A* corresponds to at least one node on *Path B*, and in which each point on *Path B* corresponds to at least one node on *Path A*. This process results in multiple point and segment connections for each node. Connections are pruned by selecting the route through a planar graph, containing all possible connections between points on one path and nodes on the other, that minimizes the average distance between points and nodes. The result is both an optimal mapping of the correspondence between the two paths, and the divergence between the paths operationalized as the area of the resultant polygons (see Fig. 1). The applications discussed herein concern the use of the divergence metric alone.

B. Computational Complexity Considerations

Computational complexity of the algorithm is important, whether the intended use for the algorithm is data analysis or application in a system. In either case, increasing the complexity of the paths will increase trial-wise run time; while

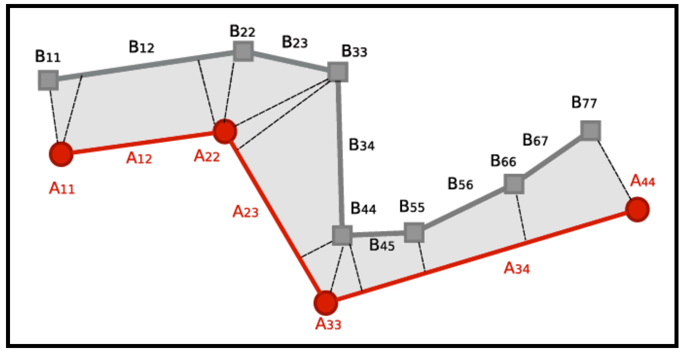

Fig. 1. The proper mapping between *Path A* and *Path B* is used to find the divergence and correspondence between the two paths. Each path consists of points, connected by segments. These features are collectively described as *Nodes*. The points on each path are mapped by the shortest Euclidean distance onto nodes of the other. The light grey shaded area indicates the Polygon *P*, the area of which constitutes the measure of divergence between the paths.

increasing the number of comparisons will increase the total run time. While this may be less important in analysis, some operational applications may require near real-time path comparisons.

When considering computation, several factors are important. The first factor is the speed of the algorithm for typical analysis including the expected run time for a single comparison and the number of the total comparisons required. The second factor is related to computational complexity (and therefore run time) scale with the problem size (for a path with *M* segments compared to a path with *N* segments). Comparing more complex paths is more computationally intensive, producing longer run times. A third and final consideration is the extent to which the algorithm is amenable to simplifying the problem (by removing redundant points, for example) without substantially distorting of the result.

Simple ways of computing the area of polygons are generally very computationally efficient even given large paths. For example, the Surveyor's Formula [37] is essentially computing the determinant of a 4*(M+N) matrix for paths *M* and *N*. However, these methods fail for any moderately-complex pair of paths (see [33], for a description of these limitations). In contrast, the ALCAMP approach is robust, but has complexity O((2M+1)*(2N+1)). This means that for large naturalistic paths with hundreds of points (or more), the time to compute a solution can become a burden. Currently, the ALCAMP algorithm is implemented in R code, and on a 2.5 ghz Intel quad core i5-2520M cpu, problems with a size of (2M+1)*(2N+1)=10,000 (around 50x50 segments) will produce a mapping in about five seconds. This is sufficient for small-to-medium problems. Implementation in a compiled language should increase the speed by an order of magnitude. The O(MxN) complexity may still be too high for comparing paths containing hundreds of points. However, run times can be further reduced by simplifying the paths themselves, and ALCAMP is amenable to simple path simplification routines.

In order to reduce run times, the ALCAMP method provides functionality for simplifying and smoothing existing paths. ALCAMP accomplishes this using shape evolution routines [38], which remove points that minimally impact the overall shape of the path. Because most paths that are sampled at a constant and high frequency are relatively smooth, most of the samples are redundant. Removing these redundant samples changes the overall path minimally, thus producing a very small effect on the overall solution, but these simplification processes have the potential to decrease the run times substantially.

IV. Applications of ALCAMP to Naturalistic Problems, and Implications for Robotics

ALCAMP has been used for a number of applications in basic and applied research. These applications have direct links to current problems in robotics. In this section we will discuss the general problem that these prior applications addressed, describe the methodology employed to solve these problems, and connect these general problems to current research questions in robotics.

A. *Measuring Divergence Between Paths to Quantify Error*

ALCAMP has been used in prior investigations to provide an error metric between two paths. Performance can be measured as the deviation of the reconstructed path from the original path. These cases represent the simplest use of ALCAMP to directly compare two paths. In these example cases below, the comparison was between an optimal path and an attempted reconstruction of that path based on (1) decision-making, (2) memory, or (3) information communicated by another person. In all cases, the technical methodology for assessing ALCAMP is identical. The difference between these two examples lies in the interpretation of the findings.

Inferring Decision-Making Capacity. In a study of human attentional control and motor performance in a parafoveal detection task, ALCAMP was used to quantify the trial-wise error in participants' motor movements by comparing their mouse movements to the optimal mouse trajectory [39]. Prior research examining the effects of task difficulty and response confidence on the resultant motor movements suggested that human motor movements reflected dynamic aspects of the decision-making process [40]. That is, human mouse movements would bow toward distracters when those distracters were ambiguous (e.g., share features with the target). However, these findings were based upon the averaged trajectories across many trials. By examining summary statistics of the trial-wise mouse movements, [39] did not find evidence for a dynamic motor policy that hedges based upon information, but rather a motor policy that executes deterministic movements once a threshold of confidence had been surpassed. Averaging the movements seen in Fig. 2 would erroneously indicate a policy that incorporates hedging.

There are many cases in which researchers may want to infer decision-making from mouse movements. For example, supervisory control using a stick-and-carrot control scheme requires the operator to continually select new destinations for the robot [41]. Likewise, even in control schemes that involve

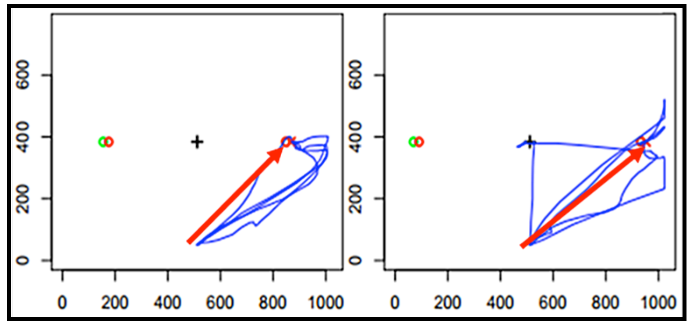

Fig. 2. This figure is an example of human-generated motor movements (blue) versus the optimal movement (red arrow). Participants were briefly shown four items which varied in color (red / green) or letter (X or O), and instructed to click the red X. If they did not see a red X, they were instructed to click the black fixation cross. Increasing the item eccentricity from the fixation cross increases task difficulty. Left panel shows the task at the tightest eccentricity, while right panel shows the task at the widest eccentricity (highest difficulty).

adjusting parameters rather than directly modifying routes, human operators generate mouse movements that may reflect aspects related to their current decision-making capacity, such as workload or confidence [42]. Quantifying the divergence of these mouse movements from the optimal shortest-path movements provides a means of examining these variables.

Path Memory. ALCAMP has also been used to assess human memory for unmanned aerial system (UAS) paths flown under supervisory control during a simulated wilderness search and rescue task [36]. During this task, participants piloted the UAS using a simple stick-and-carrot control system to search the problem space for targets using a map while also monitoring the camera sensor (see Fig. 3). Memory performance was quantified by comparing the flown route with the reconstructed route (Fig. 4). Results from this work suggested that humans encode gist information of the routes traveled by UAS under supervisory control, but fail to reproduce detailed flight paths.

Quantifying human memory for paths traveled by unmanned systems under their control is valuable for robotics research. Participants in the aforementioned study were using only a single UAS, and reproduced the routes immediately. Supervising multiple UAS requires that the operator maintain situation awareness of all UAS not currently being tasked. Put another way, the operator must store the locations and destinations of all subordinate UAS in memory. The precision

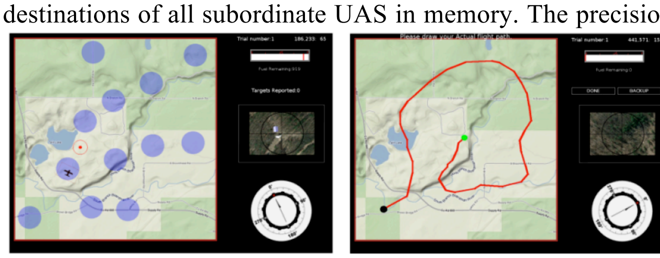

Fig. 3. Participants controlled the UAS using a stick-and-carrot control scheme (left panel): clicking the map would set a new destination for the UAS (red reticle). Once arriving at the destination, the UAS would orbit until retasked. Participants searched for targets located in half of the blue potential target locations on the map. After completing the task, participants attempted to reconstruct their route by drawing it on the map space (right panel).

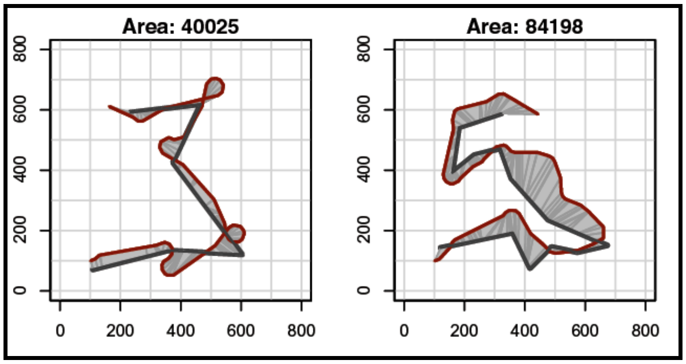

Fig. 4. ALCAMP comparisons of flown (red) and reconstructed (dark grey) paths. Left panel shows a trial on which the participant exhibited fairly good memory for the flight path, failing to encode only superficial aspects of the route (loops), versus the right panel in which the participant failed to encode large portions of the flight path. Memory performance is quantified by the area of the resultant polygon measured in pixels, shown above each panel.

with which operators can do this determines the precision with which they can localize their assets for optimal control.

Path Communication. Spatial navigation directions are communicated verbally among team members in many different environments. Reference [33] used ALCAMP to quantify team performance in the Human Communication Research Centre (HCRC) map task. In this task, *Participant A* is provided with a map containing landmarks, and a path through that map. *Participant B* is provided with the same map, but without the spatial path drawn on it. The goal of the team is for *Participant B* to reproduce the path based upon verbal instructions from *Participant A*. In this application, ALCAMP provided a means of quantifying trial-wise error between the actual and reproduced paths (Fig. 5). The results of this study show that small breakdowns in communication can have potentially grave consequences for overall team performance. One recent goal of robotics research is the development of naturalistic communication with a robotic teammate [43-44]. In these types of applications, it is imperative that the robot understand the human's intent, as verbal instructions may be vague and require interpretation of non-verbal information to disambiguate several candidate interpretations [45]. ALCAMP provides a means of quantifying the error between the human's intended route, and the route traveled by the robot.

B. *Inferring Mental Models using Multiple Comparisons*

In addition to testing hypotheses related to two groups of paths, ALCAMP can be used to create similarity spaces of multiple paths with a few additional processing steps. The first step is to create a symmetrical $n*n$ dissimilarity matrix, where n is the number of paths. Each cell in the matrix corresponds to a pairwise comparison of those paths (i.e., the area-based divergence measure generated using ALCAMP). The number of calculations in this process can be roughly halved by generating only the upper or lower triangle of the matrix, then reflecting across the diagonal.

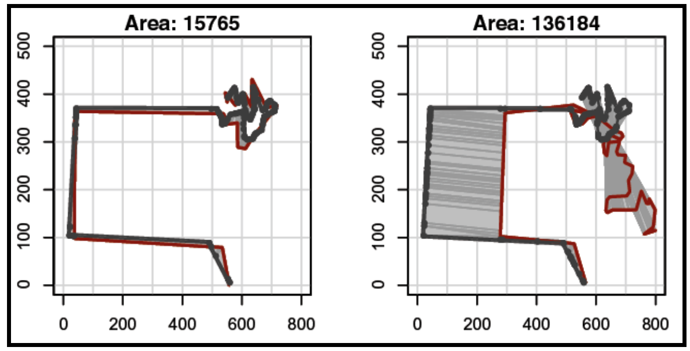

Fig. 5. This figure provides examples of two different teams' verbal communication performance on the HCRC task. The actual path viewed by *Participant A* is shown in dark grey whereas the path reproduced by *Participant B* is shown in red. The first team (left panel) performed much better than the team whose solution is shown on the right panel; this performance difference is reflected in the area of each of the polygons, shown above each panel.

The dissimilarity matrix is used as input for a multidimensional scaling (MDS) algorithm, which permits the data to be visualized in a reduced number of dimensions (ideally two or three). The dissimilarity matrix is symmetrical and the diagonal values will be zero (i.e., minimality), but there is no reason to assume that the data will satisfy the triangle inequality, i.e., divergence D for paths a, b, and c, $D(a,b) + D(b,c) > D(a,c)$. Therefore, a non-metric MDS algorithm such as Kruskal's NMDS algorithm is required. The result of this step is an n-dimensional spatial mapping of the similarity among agents' paths. This spatial mapping can be used to infer shared mental models on the basis of similarity reflected by spatial proximity among the agents' paths.

In order to infer shared mental models among the agents, a clustering algorithm such as K-means [46] or finite mixture modeling (FMM) [47-48] is used to cluster paths within the MDS solution. Each cluster contains similar paths, presumably generated according to a similar mental model for the problem space. Importantly, FMM is preferable to K-means, because MDS projects the similarity relationships into lower dimensional solutions. In so doing, it may not necessarily properly represent the similarity relationships for paths that are highly erroneous or unique. FMM can provide some robustness against this by using one Gaussian cluster to catch all of the "noisy" routes without obfuscating the clustering results.

This technique was used to infer mental models of problem solving shared among participants engaged in a simulated UAS task [49]. In that task, participants plotted a route for a UAS given two distinct sets of instructions, one designed to encourage shortest path solutions, and the other designed to encourage search behavior in which participants minimized the time expected to find a missing target among candidate locations. The results of the study showed that participants were capable of adapting to these task instructions. However, the participants exhibited several different mental models for the best way to adapt to these instructions sets.

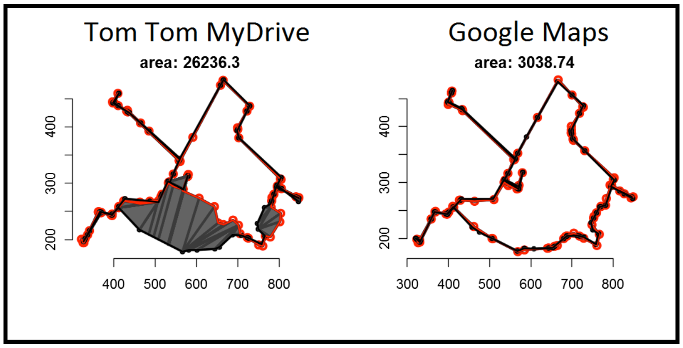

Fig. 6. Two sample mappings generated using ALCAMP for a single participant's solutions to a spatial problem without (dark grey lines) and with the assistance of each GPS navigation algorithm (red lines; either TomTom's MyDrive, left, or Google Maps, right). The area of the dark grey polygons indicates the divergence between particiapnts' paths with and without assistance (shown above each panel). The comparatively high area on the left panel indicates that the participant made more extreme path changes given assistance from TomTom's MyDrive application than from Google Maps.

C. *Current HRI Applications and Future Directions*

To address the capability to use ALCAMP to quantify path planning and its implications on trust, a study is currently under way to quantify differences in spatial problem solving between humans and common path planning algorithms used by GPS navigation systems. In this study, participants are provided with a map containing a starting location and a number of destinations. Their task is to plan a route along the roads to visit all of the destinations, ending on the starting location. Participants produce routes with and without the assistance of two different GPS navigation algorithms, Google Maps and TomTom's MyDrive application. ALCAMP is used to quantify divergence between the agents' routes.

Preliminary results from this study indicate a correlation between trust (trait-based trust, generally, as well as trust in each of the algorithms) and the extent to which each participant is willing to modify his route to more closely match that suggested by the algorithm (see Fig. 6). Furthermore, participants' ratings of quality for the routes produced by each algorithm scale with the extent to which those algorithm-generated routes match their own. Further, the study described herein will test for consensus (i.e., shared mental models) among humans for each of the spatial problems provided, and quantify aspects of each of the routes that are important to humans versus the tested algorithms. This study is conducted in parallel with a related research effort to map parameter differences between human and algorithm-generated routes, with the goal of informing algorithm development and leveraging the findings to inform system transparency manipulations.

V. Conclusion

A tool for quantifying the divergence between paths, and an associated method for identifying groups of shared mental models among a large number of agents, has the potential to address many research and engineering problems in the HRI community. Generally, prior approaches to algorithm development have focused on optimizing the robot's performance independent of the reception of that robot's behavior by human teammates. The HRI community benefits from the ability to quantify the extent to which a robot will perform the way its designers intend and human teammates expect, which is critical for calibrating appropriate trust. If a robot's teammates have strong mental models of the algorithm's planning process, they will be better suited to predict that robot's actions, and know when to suggest alternative actions for the robot when its planning process fails to incorporate important information. In cases where the robot's decision making processes and emergent behaviors may be poorly understood by human teammates (e.g., due to the complexity of the task, differences in information access, or differences in the prioritization of task parameters), the present methods can be leveraged to identify cases in which the robot's behavior will be unpredictable, warranting manipulations such as system transparency to improve user trust in the system. The methods presented herein provide a means for addressing these needs at the levels of basic research, algorithm development, system design, and system performance evaluation.

References

[1] J.Y.C. Chen and M.J. Barnes, "Human–Agent Teaming for Multirobot Control: A Review of Human Factors Issues," IEEE Transactions on Human-Machine Systems, vol. 44, no.1, pp. 13-29, 2014.

[2] K.E. Schaefer, et al., "A Meta-Analysis of Factors Influencing the Development of Trust in Automation: Implications for Human-Robot Interaction," US Army Research Laboratory, Aberdeen Proving Ground, MD, Report ARL-TR-6984, 2014.

[3] K.E. Schaefer, et al., "Relinquishing Manual Control: Collaboration Requires the Capability to Understand Robot Intent," in *Proc of the IEEE Int Conf on Collaboration Technologies and Systems,* 2017, pp. 359-366.

[4] Defense Science Board, "The role of autonomy in Department of Defense systems," Office of the Under Secretary of Defense for Acquisition, Technology and Logistics, Washington, DC, 2012.

[5] J. Minguez, "Integration of planning and reactive obstacle avoidance in autonomous sensor-based navigation," in *Proc of the IEEE/RSJ Int Conf on Intelligent Robots and Systems*, 2005, pp. 2486-2492.

[6] L. Montesano et al., "Lessons learned in integration for sensor-based robot navigation systems," Int Journal of Advanced Robotic Systems, vol. 3, no. 1, pp. 85-91, 2006.

[7] J. J. Park et al., "Robot navigation with MPEPC in dynamic and uncertain environments: From theory to practice," in *Proc of the IROS 2012 Workshop on Progress, Challenges and Future Perspectives in Navigation and Manipulation Assistance for Robotic Wheelchairs*, 2012.

[8] L. Lin and M. A. Goodrich, "UAV intelligent path planning for wilderness search and rescue," in *Proce of the IEEE/RSJ Int Conf on Intelligent Robots and Systems,* 2009, pp. 709-714.

[9] L. Evers et al. "Robust UAV mission planning," Annals of Operations Research, vol. 222, no. 1, pp. 293-315, 2014.

[10] B. S. Perelman, "A naturalistic computational model of human behavior in navigation and search tasks," PhD Dissertation, Department of Cognitive and Learning Sciences, Michigan Technological University, Houghton, MI, 2015.

[11] M. R. Endsley, "Toward a theory of situation awareness in dynamic systems," Human Factors, vol. 37, no. 1, pp. 32-64, 1995.

[12] J. Y. C. Chen et al., "Situation awareness-based agent transparency," US Army Research Laboratory, Aberdeen Proving Ground, MD, Report ARL-TR-6905, 2014.

[13] R. Rauh et al., "Preferred and alternative mental models in spatial reasoning," Spatial Cognition & Computation, vol. 5, no. 2-3, pp. 239-269, 2005.

[14] W. B. Rouse and N. M. Morris, “On looking into the black box: Prospects and limits in the search for mental models,” Psychological Bulletin, vol. 100, no. 3, pp. 349-363, 1986.

[15] E. Phillips et al., “From Tools to Teammates: Toward the Development of Appropriate Mental Models for Intelligent Robots,” in *Proc of the Human Factors and Ergonomics Society*, vol. 55, no. 1, pp. 1491-1495. 2011.

[16] K. E. Schaefer and E. R. Straub, “Will Passengers Trust Driverless Vehicles: Removing the Steering Wheel and Pedals,” *in Proc of the IEEE CogSIMA Conf*, 2016, pp. 159-165.

[17] J. C. Boerkoel Jr. and J. A. Shah, ”Planning for flexible human-robot co-navigation in dynamic manufacturing environments,” in *Proc of the 8th ACM/IEEE Int Conf on Human-Robot Interaction,* 2013.

[18] S. L. Epstein et al., ”Learning Spatial Models for Navigation,” in *Proc of the 12th Int Conf on Spatial Information Theory,* 2015, pp. 403-425.

[19] S. L. Lee et al., “Human mental models of humanoid robots,” in *Proc of the IEEE Int Conf on Robotics and Automation*, 2005, pp. 2767-2772.

[20] W. Yan et al. “A neural approach for robot navigation based on cognitive map learning,” in Proceedings of the International Joint Conference on Neural Networks, Brisbane, Australia, pp. 1-8, 2012.

[21] P. Althaus et al., “Navigation for human-robot interaction tasks,” in *Proc of the IEEE Intl Conf on Robotics and Automation*, 2004, pp. 1894-1900, 2004.

[22] C. Lichtenthäler et al., “Influence of legibility on perceived safety in a virtual human-robot path crossing task,” in *Proc of IEEE RO-MAN: The 21st IEEE Int Symposium on Robot and Human Interactive Communication,* 2012, pp. 676-681.

[23] A. B. Decou, “Multiple spacecraft optimal interferometry- Preliminary feasibility assessment,” Jet Propulsion Laboratory, 1991.

[24] E. W. Justh and P. S. Krishnaprasad, “A simple control law for UAV formation flying,” Institute for Systems Research, Maryland University College Park, MD, Report TR 2002-38, 2002.

[25] R. Cassinis et al., “Landmines detection methods using swarms of simple robots,” in *Proc of the 6th Int Conf on Intelligent Autonomous Systems,* 2000, pp. 212-218.

[26] K. M. Dawson-Howe and T. G. Williams, “The detection of buried landmines using probing robots,” Robotics and Autonomous Systems, vol. 23, no. 4, pp. 235-243, 1998.

[27] D. J. Stilwell and J. S. Bay, “Toward the development of a material transport system using swarms of ant-like robots,” in *Proc of the IEEE Int Conf on Robotics and Automation,* 1993, pp. 766-771.

[28] R. Lenain et al., “Model predictive control for vehicle guidance in presence of sliding: application to farm vehicles path tracking,” in *Proc of the IEEE Int Conf on Robotics and Automation,* 2005, pp. 885-890.

[29] G. Kewlani et al., “Stochastic mobility-based path planning in uncertain environments,” in *Proc of the IEEE/RSJ Int Con on Intelligent Robots and Systems*, 2009, pp. 1183-1189.

[30] W. Xi et al., “A stochastic algorithm for self-organization of autonomous swarms,” in Proc of the 44th IEEE Conf on Decision and Control, 2005, pp. 765-770.

[31] C. Chen et al., “Real-time detection of anomalous taxi trajectories from GPS traces,” in *Proc of the 8th Int Conf on Mobile and Ubiquitous Systems: Computing, Networking, and Services,* 2011, pp. 63-74.

[32] M. Gariel et al., “Trajectory clustering and an application to airspace monitoring,” IEEE Transactions on Intelligent Transportation Systems, vol. 12, no. 4, 1511-1524, 2011.

[33] S. T. Mueller et al., “An optimization approach for mapping and measuring the divergence and correspondence between paths,” Behavior Research Methods, vol. 48, no. 1, pp. 53-71, 2016.

[34] M. Tesch et al., “Parameterized and scripted gaits for modular snake robots,” Advanced Robotics, vol. 23, no. 9, pp. 1131-1158, 2009.

[35] U.S. Air Force (2016, November 10). MQ-1B Predator. [Online] Available: http://www.af.mil/AboutUs/FactSheets/Display/tabid/224/Article/104469/mq-1b-predator.aspx

[36] B. S. Perelman and S. T. Mueller “Examining memory for search using a simulated aerial search and rescue task,” in *Proc of the 17th Int Symp on Aviation Psychology,* 2013, pp. 302-309.

[37] A. L. F. Meister, Generalia de genesi figurarum planarum et inde pendentibus earum affectionibus, 1769.

[38] L. J. Latecki and R. Lakamper “Shape similarity measure based on correspondence of visual parts,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 22, no. 10, pp. 1185-1190, 2000.

[39] B. S. Perelman and C. W. Myers “Visual search and target selection using a boundedly optimal model of state estimation & motor control,” in Proc of the 18th Int Symp Aviation Psychology, 2015.

[40] J. Freeman et al., “Hand in motion reveals mind in motion,” Frontiers in Psychology, vol. 2, pp. 59, 2011.

[41] J. S. Metcalfe et al., “Field testing of tele-operation versus shared and traded control for military assets: an evaluation involving real-time embedded simulation and soldier assessment,” *in Proc of SPIE Defense, Security, and Sensing,* 2010, pp. 769206.

[42] M. T. Shaikh et al., “Adverb Palette: GUI-based Support for Human Interaction in Multi-Objective Path-Planning,” in *Proc of the 11th ACM/IEEE Int Conf on Human Robot Interation*, 2016, pp. 515-516.

[43] M. Marge et al., “Assessing Agreement in Human-Robot Dialogue Strategies: A Tale of Two Wizards,” in *Proc of the 16th Int Conf on Intelligent Virtual Agents,* 2016, pp. 484-488.

[44] M. Marge et al., “Applying the Wizard-of-Oz technique to multimodal human-robot dialogue,” in *Proc of IEEE RO-MAN: The 21st IEEE Int Symp on Robot and Human Interactive Communication*, 2016.

[45] H. Admoni and B. Scassellati, “Robot Nonverbal Communication as an AI Problem (and Solution),” in *Proc of the Association for the Advancement of Artificial Intelligence Fall Symp Series,* 2015, pp. 2-4.

[46] S. Lloyd, “Least squares quantization in PCM,” IEEE Transactions on Information Theory, vol. 28, no. 2, pp. 129-137, 1982.

[47] B. Gruen and F. Leisch “Fitting finite mixtures of generalized linear regressions in R,” Computational Statistics & Data Analysis, vol. 51, no. 11, pp. 5247-5252, 2007.

[48] F. Leisch, “FlexMix: A general framework for finite mixture models and latent class regression in R,” Journal of Statistical Software, vol. 11, no. 8, pp. 1-18, 2004.

[49] B. S. Perelman and S. T. Mueller “Identifying mental models of search in a simulated flight task using a pathmapping approach,” in Proc of the 18th Int Symp on Aviation Psychology, 2015.