Constraint-Based Navigation for Safe, Shared Control of Ground Vehicles

by

Sterling J Anderson

Bachelor of Science in Mechanical Engineering
Brigham Young University, 2007

Master of Science in Mechanical Engineering
Massachusetts Institute of Technology, 2009

Submitted to the
Department of Mechanical Engineering
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy in Mechanical Engineering

at the

Massachusetts Institute of Technology

February, 2013

Signature of Author

Department of Mechanical Engineering
December 31, 2012

Certified By

Karl Iagnemma
Principal Research Scientist
Thesis Supervisor

Accepted By

Professor David Hardt
Chairman, Committee on Graduate Studies

Constraint-Based Navigation for Safe, Shared Control of Ground Vehicles

by

Sterling J Anderson

Submitted to the Department of Mechanical Engineering
on December 31, 2012, in partial fulfillment of the requirements for the degree of
Doctor of Philosophy in Mechanical Engineering

Abstract

Human error in machine operation is common and costly. This thesis introduces, develops, and experimentally demonstrates a new paradigm for shared-adaptive control of human-machine systems that mitigates the effects of human error without removing humans from the control loop. Motivated by observed human proclivity toward navigation in fields of safe travel rather than along specific trajectories, the planning and control framework developed in this thesis is rooted in the design and enforcement of constraints rather than the more traditional use of reference paths.

Two constraint-planning methods are introduced. The first uses a constrained Delaunay triangulation of the environment to identify, cumulatively evaluate, and succinctly circumscribe the paths belonging to a particular homotopy with a set of semi-autonomously enforceable constraints on the vehicle's position. The second identifies a desired homotopy by planning – and then laterally expanding – the optimal path that traverses it. Simulated results show both of these constraint-planning methods capable of improving the performance of one or multiple agents traversing an environment with obstacles.

A method for predicting the threat posed to the vehicle given the current driver action, present state of the environment, and modeled vehicle dynamics is also presented. This threat assessment method, and the shared control approach it facilitates, are shown in simulation to prevent constraint violation or vehicular loss of control with minimal control intervention. Visual and haptic driver feedback mechanisms facilitated by this constraint-based control and threat-based intervention are also introduced.

Finally, a large-scale, repeated measures study is presented to evaluate this control framework's effect on the performance, confidence, and cognitive workload of 20 drivers teleoperating an unmanned ground vehicle through an outdoor obstacle course. In 1,200 trials, the constraint-based framework developed in this thesis is shown to increase vehicle velocity by 26% while reducing the occurrence of collisions by 78%, improving driver reaction time to a secondary task by 8.7%, and increasing overall user confidence and sense of control by 44% and 12%, respectively. These performance improvements were realized with the autonomous controller usurping less than 43% of available vehicle control authority, on average.

Thesis Supervisor: Karl Iagnemma
Title: Principal Research Scientist

ACKNOWLEDGEMENTS

I would like to thank Dr. Karl Iagnemma for guiding this research. Working with Karl has felt more like a partnership than a typical advisor-advisee relationship, and I've appreciated the autonomy and confidence that's afforded me immensely.

I would also like to thank Professors John Leonard and Emilio Frazzoli for their contributions as members of my thesis committee, and all of the guys at Quantum Signal, LLC, for their help in setting up the experimental vehicle and conducting experiments. James Walker in particular has been invaluable to this effort. From troubleshooting communication problems to wrangling test drivers and dealing with interns and exploded ice packs, James is a big part of this testing's success. Thanks also to Sisir Karumanchi, Steve Peters, Rob Lupa, Kevin Melotti, Phil Munie, Bryan Johnson, Dan Rice, Victor Perlin, Steve Rohde, Mitch Rohde, and all of the test drivers for their help.

I would also like to thank the United States Army Research Office (contract W911NF-11-1-0046) and DARPA DSO (contract W911NF-11-C-0101) for providing financial support for this work.

Finally, I would like to thank my family. For everything.

TABLE OF CONTENTS

LIST OF FIGURES

1

CHAPTER 1: INTRODUCTION

1.1 BACKGROUND AND MOTIVATION

Humans make mistakes. When humans control dynamic systems, the rate and ramifications of those mistakes increase. Whether it occurs while driving a car, flying an airplane, teleoperating an unmanned vehicle, controlling industrial machinery, or manipulating medical devices, human error can lead to costly and often deadly consequences. In 2010, over 32,000 people were killed and another 2.2 million injured in motor vehicle accidents in the United States alone [1]. During Operation Iraqi Freedom (2003 – 2011), vehicle crashes killed more than twice as many service members as the next leading cause of non-combat fatalities [2]. Even the manufacturing sector is susceptible to human error; forklift operation alone is estimated to claim 100 lives and cause 94,500 injuries each year in the United States [3].

Perhaps nowhere is the effect of human error more evident than in teleoperated systems. Operators of Unmanned Ground Vehicles (UGVs) and Unmanned Aerial Vehicles (UAVs) must not only cope with the challenges inherent to manned navigation, but must also perform many of its same functions with a restricted field of view, limited depth perception, potentially disorienting camera viewpoints, and significant time delays [4]. Remotely operating a vehicle under these conditions while monitoring the vehicle's health status, its task completion, and the condition of the environment is challenging and prone to high failure rates – even for trained operators. In studies conducted to date, the average mean time between UGV failure ranges from six to twenty hours – implying that in their present state, many of today's UGVs cannot complete even one standard (12-20

hour) shift without experiencing a mechanical failure or human/controller-caused accident [5]. In UAVs, the mishap rate is estimated at 12-100 times greater than that of their manned counterparts, with 71% of collisions attributed to human error [6], [7].

Roughly categorized, human error is caused by deficiencies in recognition, decision, or performance [8]. Recognition errors result from a human's inability to properly perceive or comprehend a situation. Common causes of recognition errors include inattention, inadequate surveillance, and internal or external distractions, among others. Decision errors arise when a human takes an improper course of action or fails to act when action is necessary. In ground vehicle applications, these errors are typically made when the human drives the vehicle faster or more aggressively than the circumstances (vehicle, environment, etc.) allow. Finally, performance errors refer to miscalculations in low-level control functions, often due to insufficient reaction time. Drivers of ground vehicles often make these errors by overcompensating or applying improper directional control. In a nationally representative sample of 5,471 crashes conducted from 2005 to 2007, the United States National Highway Traffic Safety Administration (NHTSA) attributed roughly 41 percent of vehicle crashes to recognition errors, 34 percent to decision errors, and 10 percent to performance errors [9]. While insufficient training and challenging scenarios or environments can exacerbate each of these error types, the human-machine control allocation and methods can also significantly affect the combined system's ability to respond quickly and appropriately to safety imperatives [10].

In this thesis, we address the challenge of reducing the frequency and effects of human error via shared-adaptive control. While the implementation presented in this thesis is designed specifically for manned and unmanned ground vehicles, the framework that results is also relevant, and can be adapted to, human-machine systems generally.

1.2 PREVIOUS WORK

In recent years, improvements in sensing, control, and computation capabilities have facilitated the development of driver assistance, autonomous, and shared-adaptive control systems designed to aid, replace, or correct human operator and thereby reduce or eliminate the effect of human error [11]–[13]. While distinct in their intended outcomes,

the inability of these systems to effectively share control with a human driver or capitalize on human-machine synergies has its root in a common, basic building block: each assumes the presence of a single path that the vehicle should follow. Regardless of the planning method used to obtain it (sampling, graph searches, potential fields, etc.) [14]–[16] or the control method employed to track it (PID schemes, linear-quadratic regulators, nonlinear fuzzy controllers, model predictive controllers, etc.) [17]–[20], this path identifies, evaluates, and seeks to limit the vehicle to just one of the many trajectory options available to the human operator. The subsections that follow describe how three broad categories of modern vehicle control architectures use these paths and how this usage affects their ability to effectively aid the human driver.

1.2.1 DRIVER ASSISTANCE SYSTEMS

In recent years, the focus of vehicle safety has shifted from measures designed to reduce the effects of collisions on vehicle occupants (eg. seat belts, air bags, roll cages, and crumple zones) to driver assistance systems designed to prevent those collisions from happening altogether. Driver assistance systems generally fall into one of two categories: *reactive* safety systems such as antilock brakes, traction controllers, electronic stability controllers, and lane-assist systems monitor the current state of the vehicle and apply low-level control actions to meet some safety-critical criteria. For example, stability controllers monitor the lateral acceleration, yaw, and wheel rotational speeds, and apply asymmetric torques to the wheels when estimated lateral or longitudinal tire slip exceeds a prespecified threshold. In order to avoid collisions, these systems rely on the human's ability to 1) foresee, 2) judge, and 3) respond appropriately to impending hazards to trigger intervention. This reliance on driver actions renders reactive safety systems vulnerable to human recognition and decision errors; for drivers who do not recognize and correctly respond to hazards, these systems can do very little. As estimated in NHTSA's study these errors make up a sizable portion (~75%) of vehicle accidents [9], [21].

Predictive safety systems, on the other hand, consider not only the current state of the ego vehicle, but also the predicted state evolution of the vehicle and environmental hazards through a finite preview horizon. These systems then preemptively assist the

driver in identifying, assessing the threat posed by, and in some cases avoiding an impending hazard. Recent work in predictive safety has resulted in systems that use audible warnings [22], haptic alerts [23], [24] and steering torque overlays [25] to help the driver avoid collisions [26]–[28], instability [29], or lane departure [12], [30].

In contrast to the strategic planning of a human driver that inherently considers multiple hazards, active safety systems take a more tactical approach, seeking only to avoid the most imminent threat. To estimate which of many possible scenarios (i.e. lane/road departure, frontal collision, loss of control, etc.) is most imminent and determine the appropriate type and degree of driver assistance, active safety systems use various threat assessment metrics. Threat metrics described in the literature predominantly use time-, distance-, and acceleration-based measures [31]–[38]. Time-based threat measures project time to collision (TTC) based on current speeds, positions, trajectories, and (in some formulations) other vehicle states [32]–[34]. Distance-based metrics are generally calculated using prevailing range and vehicle speeds and employ constant velocity/acceleration assumptions and simple hazard geometry [36], [37]. Acceleration-based metrics assess the threat of a given maneuver based on the minimum (and often assumed constant) lateral or longitudinal acceleration that a simple avoidance maneuver would require, given the current position, velocity, and acceleration of both host and hazard [31], [35]. Figure 1.1 illustrates the threat that might be assessed by time- and acceleration-based measures in a scenario with moving hazards.

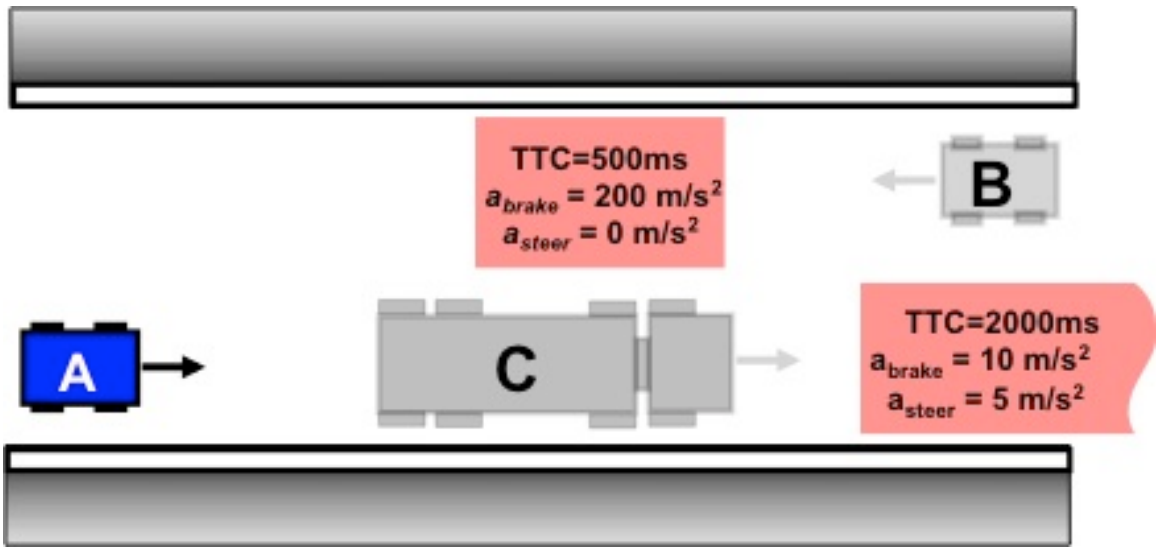


Figure 1.1. Illustration of time- and acceleration-based threat assessment metrics

While these threat metrics have been shown to provide useful estimates of the danger posed by a simple maneuver, they are not well suited to consider multiple hazards, complex vehicle dynamics, or complicated environmental geometry with its attendant constraints. The geometrically-simple (straight-line or constant-radius-curve) avoidance maneuvers they assume may also misestimate the true threat posed by scenarios in which the physically-achievable vehicle trajectory would require a curve of varying radii or non-constant velocity/acceleration, such as cases in which a lane boundary requires that the trajectory straighten out after passing an obstacle.

The threat-assessment method presented in [31] illustrates common deficiencies in existing approaches. This method estimates the lateral acceleration required to avoid a single obstacle via a constant radius evasive maneuver given the host vehicle's current position, velocity, and heading, and compares this value to the longitudinal acceleration required to avoid the obstacle by braking. When either of these threat metrics reaches a predefined threshold, corresponding countermeasures may be implemented to assist the human driver. As Figure 1.2 illustrates, this approach's failure to consider 1) the driver's intended maneuver, and 2) the effect of present evasive actions on future threat scenarios can make its threat assessment inaccurate. For example, a driver assistance system seeking to prevent a rear-end collision with obstacle C might assess threat based on a leftward passing maneuver while the human or a more strategic controller might instead

choose to stop or pass on the right. In addition to this threat assessment ambiguity, the simplified geometry of the assumed lateral avoidance maneuver (blue dashed arc of Figure 1.2) fails to consider both the lateral acceleration required to straighten out in the opposing lane as well as the vehicle's ability to successfully complete the passing maneuver before colliding with Obstacle B (illustrated by black solid line).

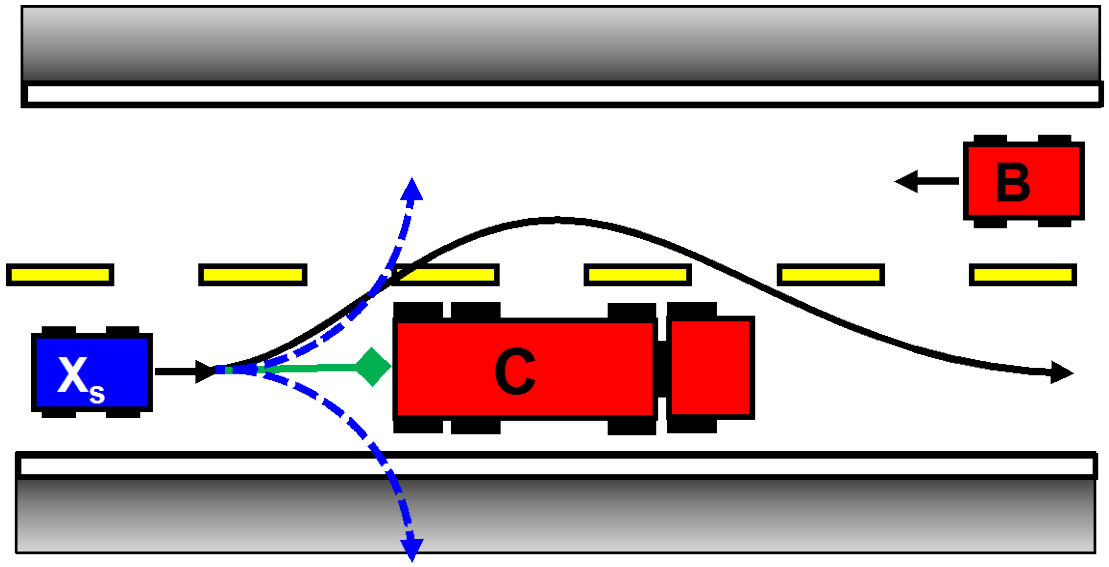


Figure 1.2. Illustration of steering (blue dashed), and braking (green solid) maneuvers commonly assumed by threat assessment methods and the more realistic (black) maneuver required to avoid Obstacles C and B.

The local focus of existing active safety systems, together with the inherent difficulty of assimilating distinct sources of threat into a single, actionable metric has led to solutions that operate purely or primarily in one dimension. Whether governing longitudinal dynamics (traction controllers, anti-lock braking systems, or adaptive cruise controllers), monitoring lateral dynamics (lane-assist systems), or assisting in stability control (yaw/roll stability controllers), these systems largely fail to consider threats from a holistic or integrated perspective. Consequently, when placed on the same vehicle, their warnings and/or control inputs can be suboptimal at best or contradictory at worst.

1.2.2 AUTONOMOUS SYSTEMS

Made possible in recent years by developments in onboard sensing, lane detection, obstacle recognition, and drive-by-wire capabilities, and promoted by competitions such as the DARPA grand challenge, autonomous planning and control frameworks for ground vehicle navigation seek to control a vehicle without requiring – or generally accepting – inputs from the human operator. The typical architecture for these systems consists of a perception layer, a strategic motion planning layer, and a tactical execution layer [39]. The perception layer commonly uses a combination of onboard sensors, such as radar, LIDAR, and camera-based feedback to identify, localize, and predict the motion of environmental hazards such as road edges and collision threats [40]. Within the workspace thus mapped by the perception layer, the motion planner designs a hazard-free trajectory to a desired goal or waypoint using any of several motion planning techniques configured to satisfy vehicular task requirements. Common methods used to design these paths include sampling, graph search, and potential fields [14]–[16]. The paths that result from the motion planning layer are often geometric in nature, and do not consider the dynamic or kinematic constraints of the vehicle. As such, tactical re-planning is typically performed at a lower, execution layer to locally "smooth" or convert the path plan into a control reference compatible with the vehicle's kinematic and dynamic constraints [11], [41]–[43]. This control reference is then tracked using any of a number of low-level controllers [17]–[19]. When model-based methods such as Model Predictive Control (MPC) or closed-loop RRT are used, replanning and control are performed in the same calculation [44]–[47]. In [48], for example, the authors use a model-based, finite-horizon constrained optimal controller to simultaneously generate and track an optimal trajectory that satisfies lane constraints and control limitations.

MIT's DARPA Urban Challenge vehicle illustrates the path-based hierarchy many of today's autonomous systems [11]. As shown in Figure 1.3, its RRT-based motion planner samples potential branches from feasible regions of the drivability map generated from onboard sensing. The motion planner then generates kinodynamically feasible trajectories to each of this tree's nodes by forward-simulating the vehicle's dynamics assuming pure-pursuit steering and proportional-integral (PI) speed control.

Finally, it checks the resulting path against the drivability map and uses these same controllers to track the trajectory generated in simulation.

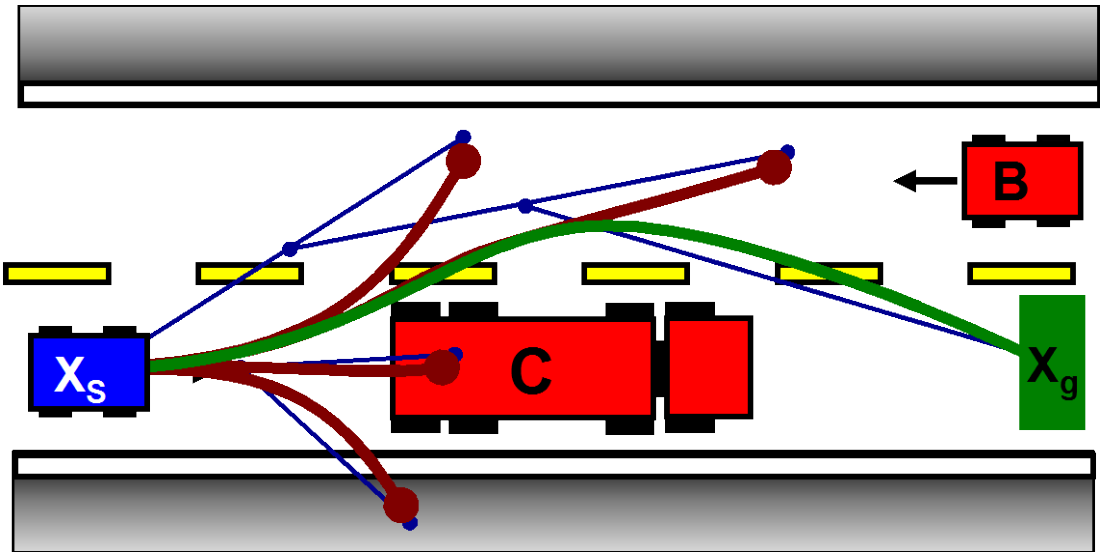


Figure 1.3. Illustration of the closed-loop RRT motion planner showing motion control points (blue), infeasible paths (red), and feasible paths (green)

While many variations of path-based planning and control have proven effective in autonomous implementation, their inability to account for the planning preferences and control inputs of a human operator in either stage of the navigation task make them ill-suited for human-in-the-loop or "semi-autonomous" control. Incorporating a human into the control loop, particularly when the vehicle command was designed to track a specific reference trajectory, is non-trivial. Without contingencies in the trajectory plan specifically accounting for the alternative goals implied by the human driver, the control inputs s/he provides, or the threat those inputs pose to the vehicle, path-based planning and control methods designed for autonomous implementation are not suitable for semi-autonomous operation.

1.2.3 SHARED CONTROL AND THE CASE AGAINST FULL AUTOMATION

Despite the frequent occurrence and significant socioeconomic costs of human error, and the promise of fully autonomous systems, humans will continue to control

dynamic systems for the foreseeable future. Their superior judgment and robust reasoning capabilities, together with high automation costs and powerful social pressures create a compelling need for humans to remain "in the loop". Complete automation often requires expensive and coordinated infrastructural changes such as the re-tooling of production facilities, the re-working of highways and traffic management systems, and redrafting of legislation to accommodate a fundamentally new definition of liability. While these changes are in some cases feasible, the social acceptance hurdles that must be surmounted to achieve complete automation may not be quite so tractable. Automating traditionally human-controlled tasks eliminates jobs. On this premise alone, labor unions such as the National Air Traffic Controllers Association have fiercely opposed the introduction of even low-level automation [49], with other groups such as transportation and manufacturing unions sure to follow suit in the advent of a credible automation threat.

Perhaps more compelling than practical arguments against full automation of traditionally human-controlled machines is the prospect of improved semi-autonomous performance through the exploitation of human-automation synergies. As originally published in 1951 [50] and widely discussed since [51], humans and automation are uniquely well suited to specific types of tasks [52], [53]. Whereas automation excels at responding quickly and precisely to well-defined or repetitive control objectives, humans tend to make more mistakes as the frequency and complexity of the control task increase. Conversely, humans have the unique ability to detect and contextualize patterns and new information, reason inductively, and adapt to new modes of operation, whereas automation typically struggles at these tasks. The goal of semi-autonomy or "shared-adaptive" control is to exploit these synergies in the abilities of both humans and automation to improve planning and control performance of the combined system and – where possible – the actors therein [54]. To be effective, shared-adaptive systems should provide intuitive, intention-preserving assistance without increasing the human's mental workload, over-restricting the human's control freedom, reducing vigilance, or inducing a false sense of security [52], [55], [56].

Figure 1.4 illustrates the autonomy chasm that exists between the minimal level of decision-making and control required by today's driver assistance systems (described in

Section 1.2.1) and the complete control required by fully autonomous systems (detailed in Section 1.2.2). Shared-adaptive, or "adaptively autonomous" control systems have the potential to bridge this gap, replacing active driver assistance, passive driver assistance, and autonomous vehicular control with a single assistance strategy that adapts its level of autonomy (and corresponding safety contribution) as circumstances require.

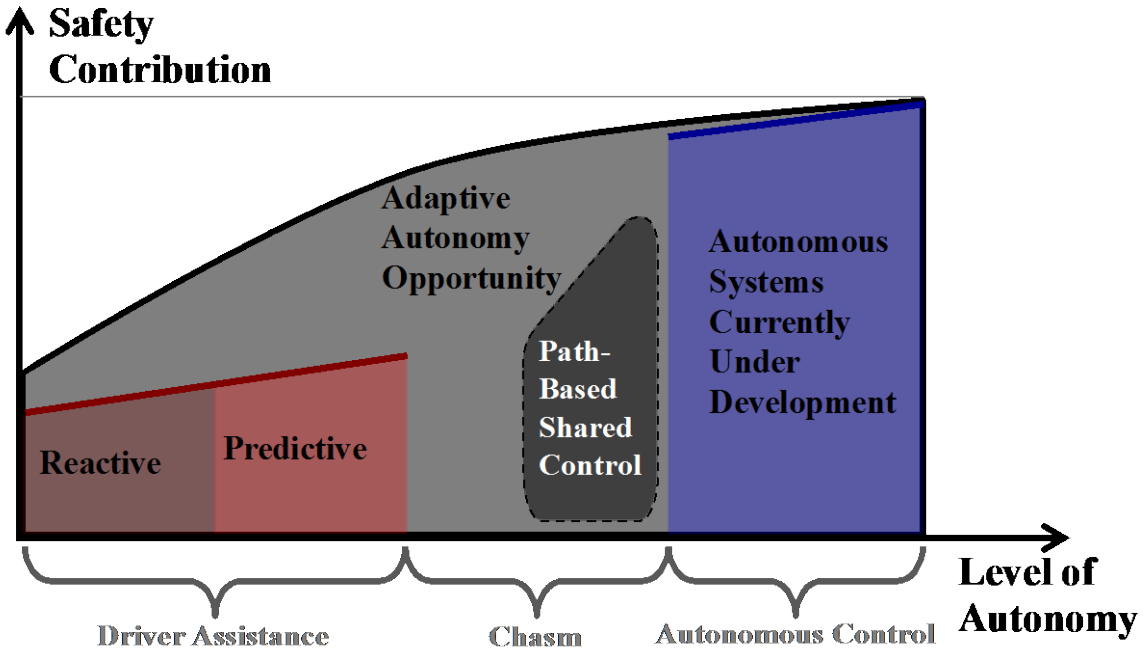


Figure 1.4. Illustration of the level of autonomy and safety contribution of today's driver assistance systems (red), proposed autonomous systems (blue), existing shared control systems (dark gray) and the unrealized potential of adaptively autonomous systems (light gray)

Similar to autonomous control systems, shared control methods proposed in the literature today also rely on specific paths and must therefore choose, infer, or accept (typically from a human operator) a specific goal or end point. In supervisory control, these waypoints are often explicitly designated prior to, or during the navigation task. Other methods infer them from the operator's control actions.

In [57], the authors introduce a hierarchical control strategy that considers human inputs at various levels of the wheelchair navigation task. At the "deliberative", or motion

planning layer, users designate a desired goal, which the controller sets as the target of a potential field. The controller then navigates toward the target via gradient descent on the potential field, modifying the user's command only when that command deviates by more than 90° from the field gradient or comes within a pre-configured proximity of an obstacle. When the latter occurs, a reactive controller modifies the direction of the human input to satisfy the constraint. Urdiales et al. [58] use a similar, potential-field-based approach to assist a human user while accommodating some human input at both the motion planning and execution levels. In their embodiment, control authority is allocated between the human and the PFA controller based on assessments of the "smoothness" (angle between robot's current direction and that of the provided motion vector), "directiveness" (angle between the current wheelchair heading and the vector to the goal), and "safety" (distance to the closest obstacle with respect to the wheelchair heading) of the human's input. Besides being subject to the local minima common to potential-fields-based approaches, the ad hoc assistance modulation employed by both of these strategies requires considerable tuning and does not guarantee 100% obstacle avoidance.

In contrast to supervisory or reactive control strategies that require a priori knowledge of the user's target, other control strategies have been proposed that infer the user intention from the operator's control actions. This approach is used in [59], for example, where the authors interpret the human user's desired goal with inverse reinforcement learning and adjust the level of autonomy based on the certainty of this goal estimation. When prediction certainty is high, the robot is allocated a greater degree of control. When certainty is low, the user retains more autonomy. Other approaches which rely similarly on a specific goal estimate to control the vehicle have also been proposed [60], [61]. The reliance of these systems on prediction accuracy in order to determine appropriate control allocation makes them ill-suited to assist the human in unplanned or emergency scenarios where the actual human intent is either 1) predictable but not safe (as in cases where the human fails to identify a hazard and continues to operate as before), or 2) difficult to ascertain or predict. In such scenarios, rather than take *more* control, systems whose intervention and control actions rely on the certainty of a path prediction can actually take *less*. Further, in low-threat scenarios, where direct

control by the human operator would be sufficient to keep the vehicle safe, these approaches effectively reduce the human's role to a supervisory one – not following his/her commands directly, but using the control inputs s/he provides to infer a desired goal.

1.3 PROPOSED APPROACH

The work presented in this thesis builds on the premise that adaptive autonomy based on *constraints*, rather than inferred goals and associated motion paths, is better suited to share control with humans whose planning and control strategy is better represented by operating constraints and fields of safe travel than by specific paths [62]. Such fields contain an infinite number of continuously deformable trajectory candidates, or path homotopies of similar perceived "goodness", as illustrated in Figure 1.5. Identifying and circumscribing these homotopies by constraints, then allocating control authority as necessary to satisfy them neither over restricts the human operator in safe scenarios nor fails in risky ones, but instead ensures that the driver retains as much direct control freedom as possible without allowing the vehicle to lose control or collide with obstacles.

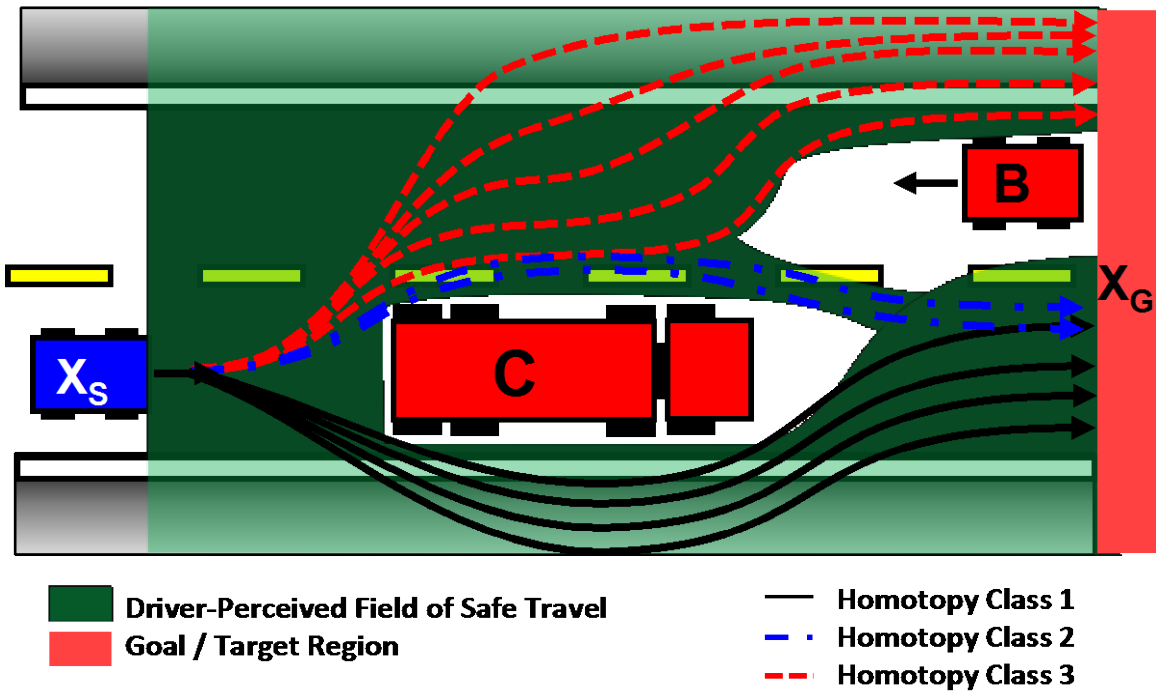


Figure 1.5. Illustration of prominent homotopies as they might be perceived by the human driver

Figure 1.6 illustrates how three prominent homotopies in a cluttered environment might be perceived by a human operator. In off-road environments, the desired homotopy may not be as clearly delineated, though vehicle dynamic constraints require that it exclude any region through which the vehicle cannot travel without colliding with obstacle(s).

Figure 1.6. Visualization of prominent homotopies available to a human operator (image best viewed in color)

In addition to homotopy-imposed position constraints, the approach proposed in this thesis enforces stability-imposed state constraints and actuator-imposed input constraints on the vehicle. Together, these constraints bound an n-dimensional tube through the vehicle state space. The threat-based intervention system focuses its planning and control strategy on disallowing dangerous maneuvers or departure from the safe or controllable state envelope (which can be objectively defined given knowledge of vehicle dynamics and estimates of the environmental topology and conditions) rather than imposing a non-unique and potentially unsafe avoidance maneuver. Instead of telling the system *what* to do, the constraint-based system determines *what not* to do, given the current state of the vehicle, driver, and environment. Rather than forcing the driver to track a specific path, which is in many cases arbitrary from the human perspective, the constraint-based approach allows the driver to take any of an infinite number of safe (collision-free and dynamically-stable) paths through the environment. Figure 1.7 shows the basic architecture of a constraint-based, shared control system.

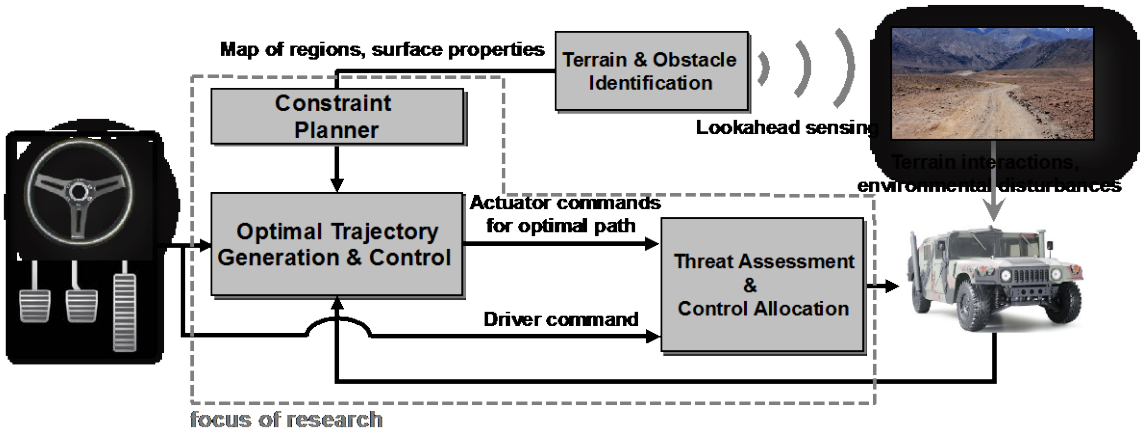


Figure 1.7. Block diagram of a constraint-based, shared control system

While potentially better suited for shared, human-in-the-loop control, constraint-based semi-autonomy also presents significant challenges. Similar to its path-based counterpart, planning in constraint or "homotopy space" requires the identification of homotopies and an evaluation of their goodness. However, because the constraints bounding a path homotopy admit an infinite number of paths, identifying these constraints and assessing their goodness requires a new set of evaluation criterion from those commonly used in path planning. For example, whereas the goodness or optimality of a specific path is well defined using metrics such as length, curvature, and dynamic feasibility, corresponding measures lose their traditional meaning when applied to a set of constraints and the many paths they admit. Further, planning methods typically used to design paths, such as grid-based search, potential fields, and sampling-based algorithms, will not necessarily work to plan constraints since the latter must be designed to circumscribe – rather than simply remain within – a safe operating region.

1.4 Outline and Contributions of this Thesis

This thesis develops, simulates, and experimentally evaluates a constraint-based approach to shared human-machine control. Chapter 2 describes two methods for identifying, evaluating, and circumscribing path homotopies with semi-autonomously enforceable constraints and illustrates these method in single- and multi-agent target tracking and obstacle avoidance. Chapter 3 then describes a method for assessing the

threat, or likelihood of violating position and stability constraints and using that threat assessment to allocate control authority between the human and an onboard controller. This chapter also illustrates a threat-based control allocation method in shared control of a simulated ground vehicle. Chapter 4 then presents the setup and results of a 1,200-trial study conducted to assesses the effect of constraint-based navigation and associated driver feedback on the teleoperation performance of twenty drivers navigating an unmanned ground vehicle through an outdoor obstacle course. Finally, Chapter 5 closes the thesis with general conclusions and a proposal for future work.

2

CHAPTER 2: HOMOTOPY-BASED CONSTRAINT PLANNING

In 1938, James Gibson, and Laurence Crooks postulated that rather than plan and track a specific path, human drivers identify and seek to remain within a field of safe travel comprised of "…all possible paths which the car may take unimpeded" [62]. As envisioned by Gibson and Crooks, this field extended "[like a] sort of tongue protruding forward along the road" (see Figure 2.1). This thesis extends their conception of human planning behavior to incorporate consideration of the many possible collision-free fields and their associated path homotopies existing within an environment with obstacles. If path homotopies can be identified, and their goodness evaluated, vehicle position constraints can be designed at the edges of a desired or optimal homotopy to circumscribe the set of paths it contains and ensure that the vehicle remains safely within it.

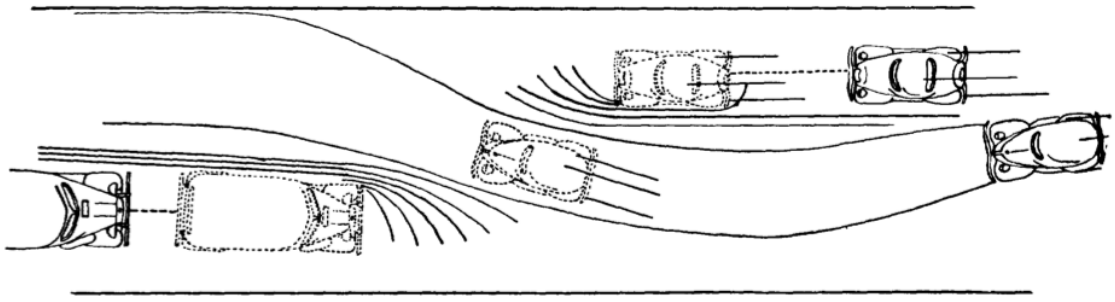


Figure 2.1. Illustration of a "field of safe travel" as envisioned by [62]

This chapter defines path homotopies, explains their significance to the vehicle navigation problem and notes how they have been used in traditional, path-based approaches to vehicle control. It then describes two new methods for identifying and evaluating the desirability of these homotopies, given the vehicle dynamics and control

constraints, and demonstrates each method in simulated studies of one or more ground vehicles.

The methods introduced in this chapter assume that the location, velocity, and size of obstacles in the environment are known over a finite preview horizon. In practice, obstacle information provided by onboard sensing is uncertain and range-limited. Improving the accuracy and coverage of these predominantly radar-, LIDAR-, infrared-, and camera-based sensing techniques is an active research area with significant implications for planning and control techniques that rely on these sensors [63], [64]. As demonstrated in several autonomous vehicle initiatives in recent years, these sensing systems have reached an acceptable level of accuracy for use in vehicular collision avoidance applications [11], [40], [65], [66]. As inter-vehicle communication techniques and protocols are implemented in the coming years, the accuracy, range, and robustness of environmental data is anticipated to improve further [67], [68].

The homotopic path planning techniques presented in this chapter also assume that the vehicle operates on a two-dimensional plane unless otherwise specified. This assumption is relaxed in Section 2.3.2.

2.1 PATH HOMOTOPIES

A path homotopy is a topological equivalence relation comprising multiple obstacle-avoiding paths that can be continuously deformed into one another without encroaching on obstacles [69]. More formally, a path homotopy in topological space X consists of a family of paths $h_t : I \rightarrow X$, indexed by t such that: 1) $h_t(0) = x_v$ and $h_t(1) = x_1$ are fixed, and 2) the map $H : I \times I \rightarrow X$ given by $H(s, t) = h_t(s)$ is continuous [70].

In the context of vehicle control, all paths spanning from the vehicle's current position X_s to a goal location X_g are said to be homotopic if they pass through the obstacles in the same manner. Visualized another way, a path homotopy spanning X_S to X_G is a subset of the field of safe travel that does not contain any holes. Thus, the on-road environment illustrated in Figure 1.5 contains at least four path homotopies; one passes to the right of both Obstacle C and Obstacle B, while another passes both obstacles to the

left. Yet another homotopy passes to the left of Obstacle C and to the right of Obstacle B, and the final passes to the right of Obstacle C and to the left of Obstacle B.

2.1.1 TRADITIONAL USES OF HOMOTOPY RELATIONS IN PATH PLANNING

In robotic applications, homotopies have traditionally been employed as a topological guide to the path planning step of hierarchal motion planners [71]–[73]. In [74], Jenkins uses homotopy classes to simplify the shortest path problem (which in the presence of obstacles does not lend itself to exhaustive search) by partitioning the workspace into a set of mutually exclusive and collectively exhaustive classes, within each of which a shortest path solution may be found. Hernandez [75] extends Jenkin's method by introducing a *Homotopic RRT* to search within each homotopy class for a desired path. This use of homotopy classes to partition the planning calculation or reduce the search space of probabilistic path planners is not uncommon; various methods have been proposed using homotopy classes to reduce the size of probabilistic roadmaps while ensuring that they capture the multiple-connectedness of a robot's configuration space [76]–[78].

Still other path planning approaches use visibility graphs, Voronoi diagrams, or Delaunay triangles to identify homotopies and thereby facilitate sample-based or optimal path planning [12], [15], [79]–[82]. In [83], the authors present a method for quickly assessing the homotopy to which any path belongs, independent of the method used to derive it, by defining a complex function that is analytical everywhere in the two-dimensional vehicle plane except for at distinct points placed at obstacle locations. This approach then allows one to verify the homotopy to which any path belongs by simply integrating this function along it and verifying the value of the result. This allows for relatively simple integration with existing sampling or graph search methods, since path plans derived from any of a variety of methods can be retained or discarded according to their homotopy equivalence.

2.1.2 PROPOSED USE OF PATH HOMOTOPIES IN CONSTRAINT-BASED VEHICLE CONTROL

In contrast to existing approaches, the constraint-based navigation framework proposed in this thesis plans and evaluates path homotopies as a proxy for the myriad of trajectory options each homotopy presents to the human driver. Rather than use topological equivalence to facilitate the planning of a single path, the proposed solution identifies and characterizes path homotopies in order to design and evaluate a set of position constraints bounding a heuristically-optimal, or driver-preferred set of collision-free paths. To this end, two approaches to homotopy-based constraint design are presented here. In the first, the environment is discretized into constrained Delaunay triangles, with path homotopies represented by sequences of adjacent triangles, across whose unconstrained edges every possible vehicle trajectory belonging to that homotopy must pass. Geometric and reachability heuristics are presented as a means of evaluating the "goodness" and dynamic feasibility of competing homotopies and establishing position constraints to bound the optimal class. The second uses a rectangular cell decomposition to discretize a reduced vehicle state space and facilitate calculation of a reachable and optimal homotopy by dynamic programming.

2.2 CONSTRAINT DESIGN WITH DELAUNAY TRIANGULATION

One particularly useful tool for revealing the topological structure of the workspace is the Voronoi Diagram [84]. As illustrated in Figure 2.2, the Voronoi diagram for a set of points, P, partitions configuration space C into a set of cells R, such that every point within cell R_i is closer to P_i than to any other point in P. The line segments at the boundaries of these cells trace out the topological skeleton, or medial axes of the free workspace; points along these axes are equidistant from the nearest obstacles [85].

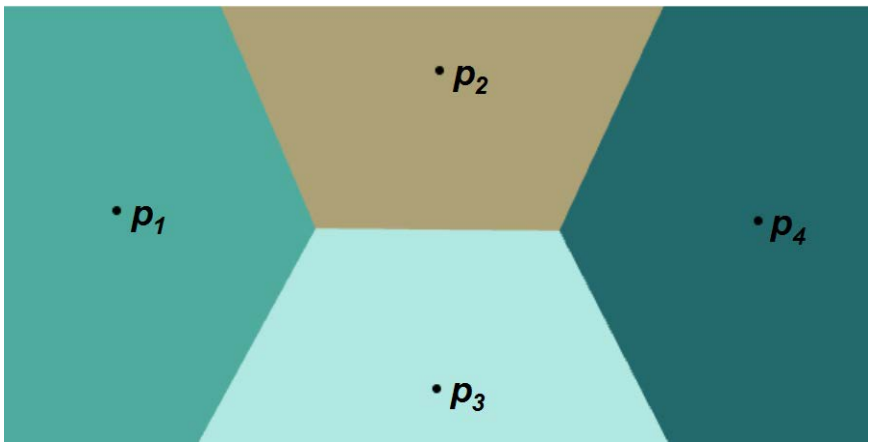


Figure 2.2. Voronoi diagram for a set of points, *P*, with cells R_i shaded

Besides providing an effective means of planning paths that meet specific clearance requirements [86], the Voronoi diagram also provides a provably complete method for capturing the connectedness of a workspace or designing a roadmap that traverses it [87], [88]. As a result, the lateral expansion of each medial axis in a Voronoi diagram describes a unique homotopy class with that axis's start and end points. For example, in the workspace illustrated in Figure 2.3, all paths connecting the start position, *S*, with a particular goal position *G*, will belong to one of two possible homotopies: the first passing to the left of Obstacle A, and the second passing to its right. The Voronoi segment existing within each represents the path of maximal clearance belonging to the homotopy.

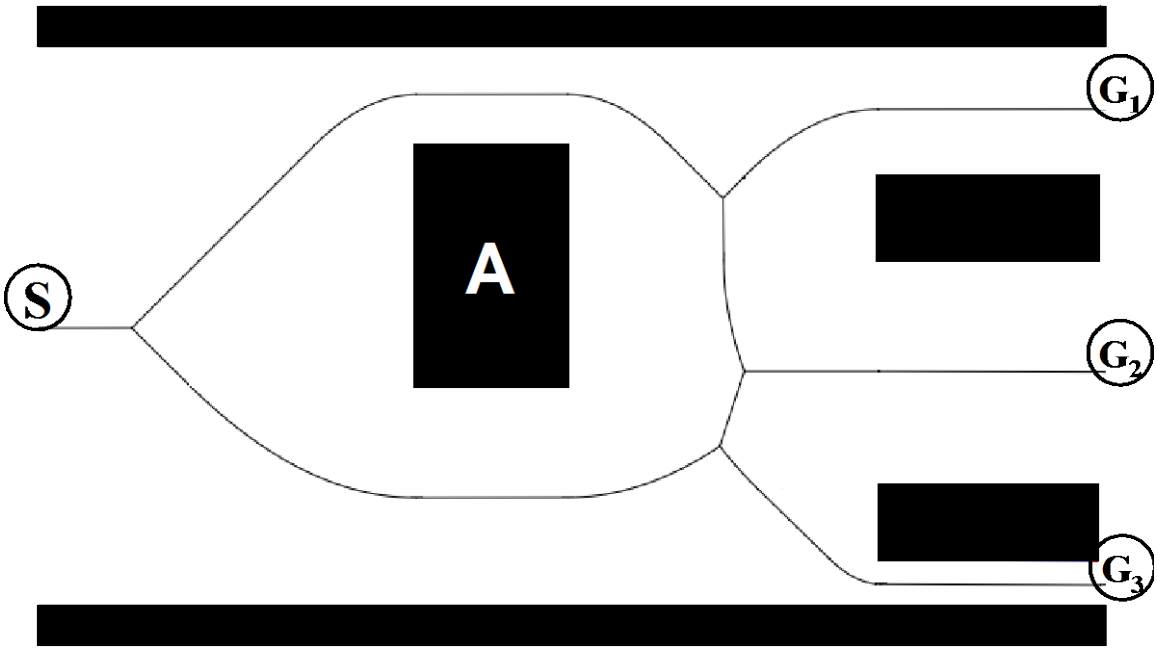


Figure 2.3. Generalized Voronoi diagram in an environment with obstacles

Various methods have been proposed for calculating Voronoi diagrams in worst-case O(*n*log*n*) time and with O(*n*) complexity [89], [90] and for using the result to plan collision-free paths [86], [91], [92]. While well suited for identifying homotopies and calculating maximal-clearance paths through obstacles, Voronoi diagrams do not encode the aggregate properties of the homotopies they delineate [80], [91]–[94]. In the paragraphs that follow, we introduce the Voronoi diagram's geometric dual – the Delaunay triangulation – as a more succinct representation of the paths existing within each homotopy that retains the connectedness information embedded in the Voronoi diagram.

The Delaunay triangulation for a set of points in a plane is a triangulation (or homogenous simplicial 2-complex) in which no point lies within the circumcircle of any triangle [95]. Its dual relationship with the Voronoi diagram captures the homotopy classes identified by the Voronoi; when connected by straight lines, the centers of the Delaunay triangulation's circumcircles form the medial axis of the Voronoi diagram.

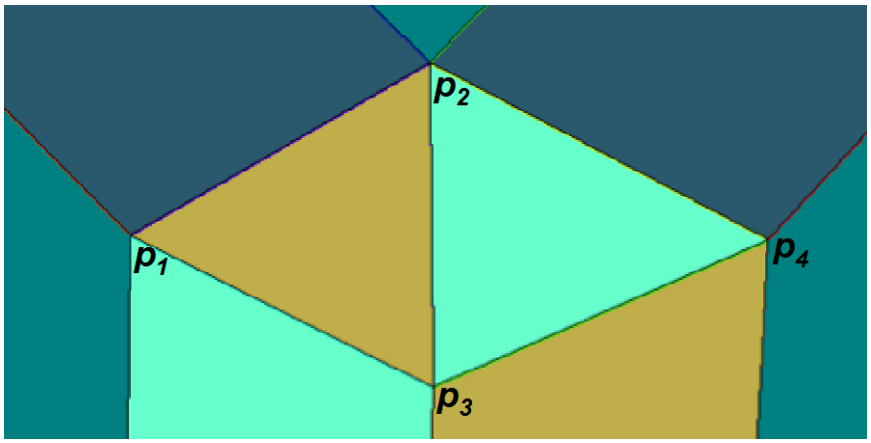


Figure 2.4. Delaunay triangulation for a set of points, *P*, with Delaunay triangles shaded

The *Constrained* Delaunay Triangulation (CDT) of a planar, straight-line graph G is a triangulation of the vertices, P, of G that includes the edges, ε, of G (such as those existing along obstacle boundaries) as part of the triangulation but otherwise remains as close as possible to the Delaunay triangulation [96] (see Figure 2.5). Introduced in 1934, the CDT has been used in various path planning algorithms [97]. In [81], the authors construct a channel from a sequence of triangles. Within this channel, a modified version of Hershberger and Snoeyink's funnel algorithm plans a minimum length path with requisite clearance from the channel's vertices [72]. Others have similarly used the constrained Delaunay triangulation to design a convex polygon or dual graph within which a minimum length or dynamically feasible path may be calculated using potential fields [98], model predictive control [79], or graph search algorithms [99]. These methods are similar in technique, though very different in purpose and execution from the usage proposed in this thesis.

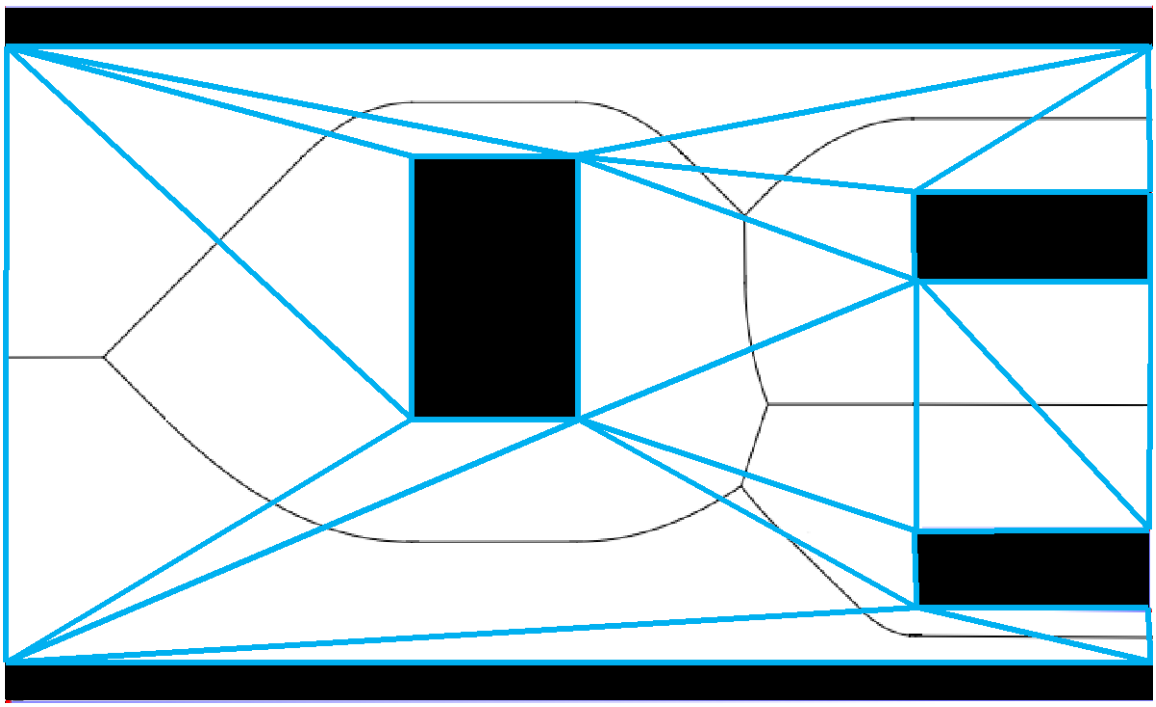


Figure 2.5. Generalized Voronoi diagram (thin black lines) and corresponding constrained Delaunay triangulation (blue) of an environment with obstacles (black rectangles)

The constraint-based planning and control approach presented here uses the exact discretization and topological information provided by the constrained Delaunay triangulation to: 1) identify the edges across which paths belonging to specific homotopies must pass, the constraints they must satisfy, and the "control freedom" they provide, 2) evaluate the dynamic feasibility of transitions between those edges, and 3) design vehicle position constraints to circumscribe a desired homotopy. Each of these considerations is discussed below.

2.2.1 HOMOTOPY IDENTIFICATION

In this work, we identify homotopies by decomposing a two-dimensional configuration space $\mathbf{C}_2 \in \mathbf{R}^2$ into a homogenous simplicial 2-complex of constrained Delaunay triangles. Because the CDT provides an exact decomposition of $\mathbf{C}_2$, each sequence of adjacent triangles existing within it uniquely describes a single path homotopy connecting any point lying within the first triangle of that sequence with any

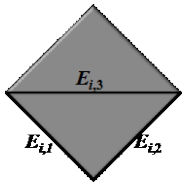
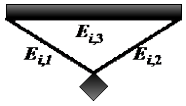
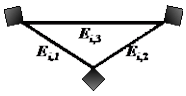
point lying within the last. Paths traversing each of these homotopies, while not necessarily remaining within the unconstrained boundaries of the triangulated channel, must cross each of the adjacent edges shared by its triangles. In this section, we present a method of simplifying this triangle connectedness to create a dual graph with nodes N_i corresponding to triangles T_i and edges ε_i representing adjacent and unconstrained triangle edges E_i.

The constrained Delaunay triangulation of an environment with polygonal obstacles O consists of three types of triangles, each of which can be classified by the number of vertices, P_i, it shares with obstacles in the workspace. Triangles with three vertices corresponding to the same obstacle, O_j, do not contain any free space and thus do no admit collision-free paths. For the present purpose, we label these triangles "Type 3" and exclude them from subsequent consideration in homotopy evaluation and constraint planning decisions.

Those with only two vertices per obstacle contain both free space and a constrained edge; simple paths that traverse them via their unconstrained edges are homotopic with respect to those edges. That is, any path through T_i that begins on edge $E_{i,1}$, ends on $E_{i,2}$, and does not violate the constrained edge $E_{i,3}$ or encroach on the vertex opposite this edge may be continuously deformed into any other such path. For the sake of convenience, this thesis labels these triangles "Type 2".

Finally, triangles whose vertices span three different obstacles contain free space but do not describe a single homotopy by their unconstrained edges. With respect to starting and ending edges, these "Type 1" cells describe a bifurcation in the two homotopies that begin at any of their three free edges. Table 2.1 summarizes and illustrates this classification.

Table 2.1. Triangle classification method and implications for planning

Type/Illustration	Characteristic	Utility/Implications
3 	Shares three vertices with a single obstacle	Prohibited region (excluded from search)
2 	Shares two vertices with a single polygonal obstacle (one constrained edge)	Simple paths that traverse it via its unconstrained edges are homotopic with respect to those edges
1 	Vertices lie on three different obstacles (zero constrained edges)	Describes a bifurcation in the homotopy search space

With triangle types defined, we construct the dual graph of the CDT to delineate each homotopy existing in the workspace. Triangles in the CDT are represented by nodes in the dual graph, while edges in the graph represent the connectivity of those triangles across their adjacent edges, E_i. Any feasible homotopy containing the vehicle's current position X_0, and the position of the goal location, X_G, may be defined as a sequence $\boldsymbol{H^n}$ of adjacent triangles $T_0 \ldots T_n$ extending from the triangle circumscribing the vehicle's current position (T_0 in Figure 2.6) to any of the triangles containing the goal location(s). This goal may be described by a single point or by a given region of $\mathbf{R^2}$, such as the distal edge of the local sensing window illustrated in red in Figure 2.6.

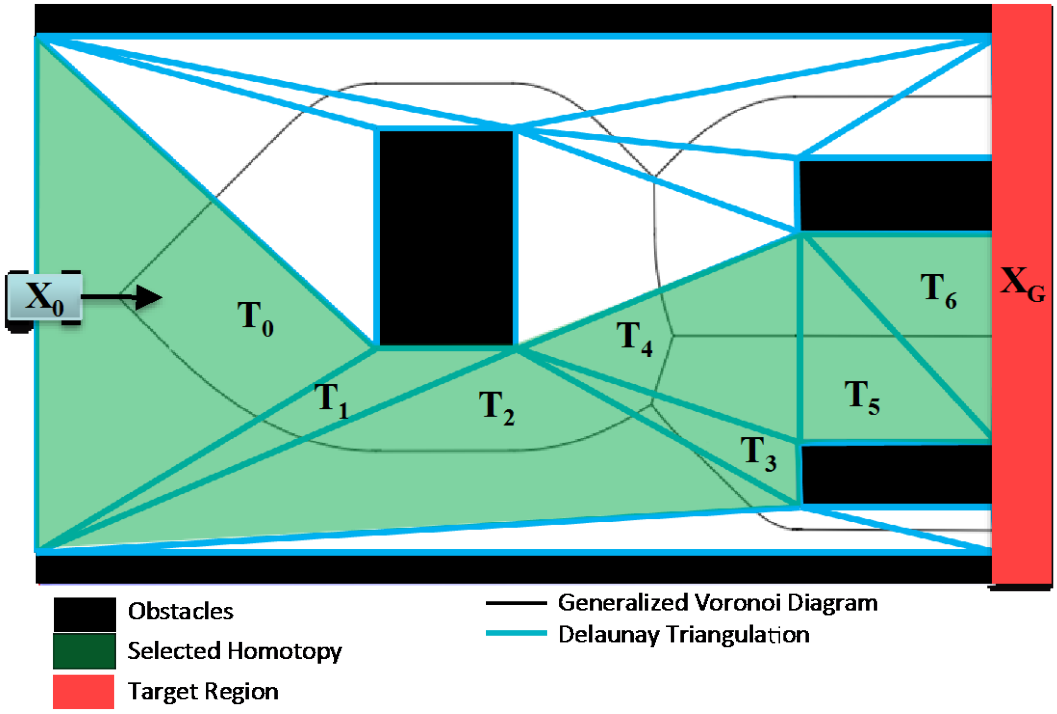


Figure 2.6. Illustration of triangulated environment highlighting one of its homotopies

2.2.2 HOMOTOPY EVALUATION

In order to plan a set of constraints circumscribing a desired homotopy, metrics describing homotopy goodness must be defined and ascribed to individual triangles (node costs) and transitions between them (edge costs) to enable graph search over the dual graph. This thesis proposes three distinct heuristics for evaluating homotopy goodness: an estimate of the average "distance" traveled by paths belonging to the homotopy, an estimate of the control freedom available to an operator while navigating within the homotopy, and the dynamic reachability of the homotopy's paths from the vehicle's current state. We note that because the human driver's eventual path through any given homotopy is not known, evaluations of that homotopy's goodness necessarily require some assumption of the general shape and direction of the driver's path, as well as generalizations about the control freedom it provides and the dynamic feasibility it allows.

I) LENGTH HEURISTIC

Assuming that all obstacles in the environment are or can be represented by convex regions*, any path starting at a point X_0 in triangle T_0, ending in a goal region $X_G \cap T_n$, and belonging to a particular homotopy $\mathbf{H}^n = T_0 \cup T_1 \cup \rightleftharpoons \cup T_n$, will pass through each triangle T_i in $\mathbf{H}^n$, entering T_i through the edge it shares with $T_{i\text{-}1}$ ($E_{i\text{-}1,\,i}$) and exiting through $E_{i,\,i+1}$ into T_{i+1}. The average "distance" $L_{i\text{-}1,i+1}$ traveled by all simple (non-self-intersecting) paths as they cross T_i may therefore be heuristically described as the distance from the midpoint of $E_{i\text{-}1,i}$ to that of $E_{i,i+1}$. Figure 2.7 illustrates the dual graph embodying this heuristic and the resemblance it shares with the generalized Voronoi diagram.

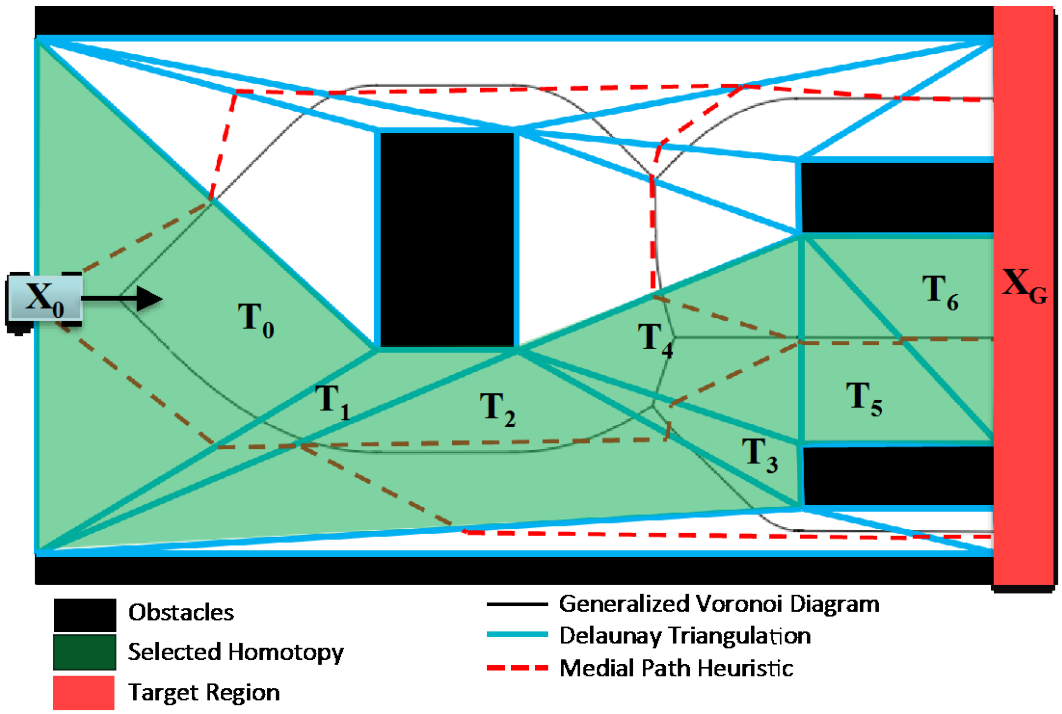


Figure 2.7. Triangulated environment illustrating the relationship between the medial path heuristic used to estimate homotopy length and the Voronoi Diagram

* Several methods have been proposed for convexifying non-convex polygons [100], [101]. One of the simplest and most conservative methods circumscribes obstacles by a convex buffer shell and regards this shell as the obstacle in subsequent computations.

II) CONSTRAINT FEASIBILITY HEURISTIC

Various measures have been devised for planning a robust path or describing its disturbance invariance and dynamic feasibility [102]–[104]. In the presence of an unbounded and equally likely (given the human's unknown future inputs) set of homotopic paths, these measures loose their traditional meaning and utility. To accommodate this ambiguity in the evaluation of a set of homotopy constraints, the feasibility metric heuristically describes the control freedom or range of motion these constraints allow and the dynamic demands they present by evaluating the pass-through clearance and required "curvature" of the triangles comprising the homotopy. We motivate this choice of heuristics with the following observations:

1. The dynamic feasibility of any path followed by a vehicle with Dubins constraints and friction-limited tires may be characterized by the lateral acceleration that path requires. This lateral acceleration directly relates to tire friction utilization, which is limited by tire-road interactions and vehicle weight.
2. This lateral acceleration is directly proportional to the square of vehicle velocity and inversely proportional to the radius of curvature of the path it traverses.
3. In any homotopy $\mathbf{H}^n$, the maximum radius of curvature of any of the constant-velocity paths belonging to $\mathbf{H}^n$ is limited by the "width" w_i, or minimum pass-through clearance of the Delaunay Triangles comprising the homotopy and the relative orientation, ϕ_i, of the constraint edges it must satisfy. The blue dashed line in Figure 2.8 illustrates the maximal radius path belonging to one particular homotopy.
4. In a constrained Delaunay triangulation, the medial "pass-through" line spanning from the midpoint of the entry edge to the midpoint of the exit edge is parallel to the constrained edge. Thus, changes in the orientation of this line – segments of which derive from dual graph for the length heuristic – reflect changing orientations of the constraint boundaries that must be satisfied by paths traversing the homotopy.

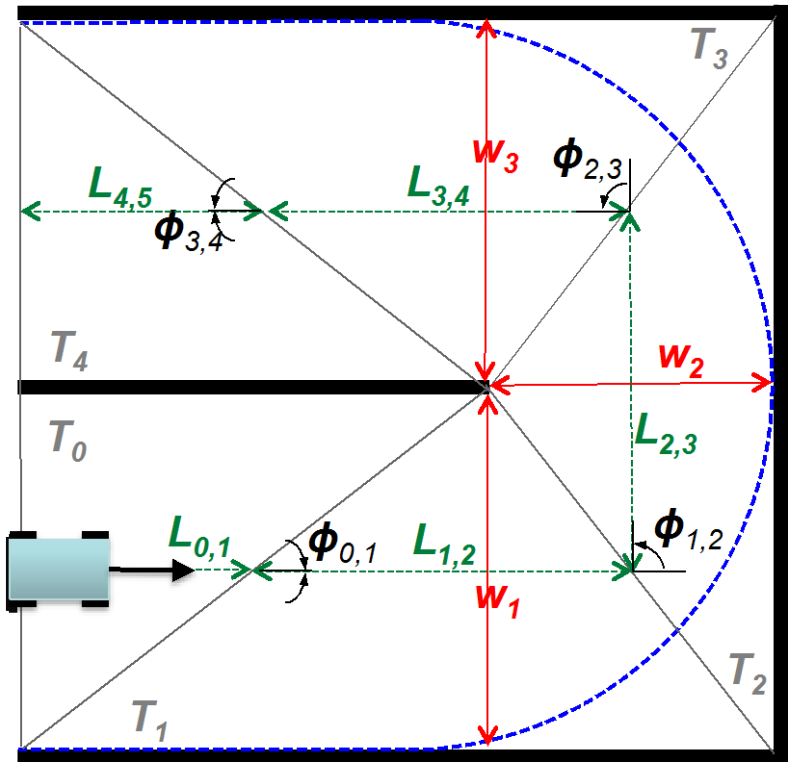


Figure 2.8. Illustration of a triangulated homotopy and the heuristics used to describe its constraint restrictiveness and dynamic feasibility

III) CONSTRAINT REACHABILITY HEURISTIC

In [105], Bertsekas and Rhodes introduce a backstepping approach to estimating the reachability of a target set by recursively estimating the size of the "effective target sets" that the system must traverse in order to reach it. Broadly stated, for the discrete dynamical system

$$x_{k+1} = f(x_k, u_k), \quad k \in \mathbf{N} \tag{2.1}$$

in configuration space $\mathbf{C}^n \in \mathbf{R}^n$, with input constraints $u_{\min} \le u \le u_{\max}$, initial state x_0, and target set $\mathbf{X}_N \subseteq \mathbf{C}^n$, this approach reduces the problem of driving the system from x_0 to $\mathbf{X}_N$ to one of driving it to intermediate, or "effective" target sets $\mathbf{X}_{k<N}$ that satisfy the constraint

$$\mathbf{X}_k := \left\{ x_k \in \mathbf{C}^n : f\left(x_k, u_k\right) \in \mathbf{X}_{k+1} \right\}. \tag{2.2}$$

Applying this concept to a constrained Delaunay triangulation and defining sets $\mathbf{X}_{k,k+1} \in \Re^{n-1}$ as (n-1)-dimensional slices of n-dimensional configuration space $\mathbf{C}^n$ coinciding with adjacent triangle edges $E_{k,k+1}$, we make a similar observation: dynamically feasible paths spanning from the vehicle's current position, x_0, to a goal region, $\mathbf{X}_N$, must also traverse only the reachable subset of $E_{k,k+1}$:

$$\mathbf{X}_{k-1,k} := \left\{ x_{k-1,k} \in E_{k-1,k} : f\left(x_{k-1,k}, u_{k-1,k}\right) \in \mathbf{X}_{k,k+1} \right\}. \tag{2.3}$$

Figure 2.9 illustrates how the effective target sets comprising one homotopy spanning from x_0 to $\mathbf{X}_N$ might appear.

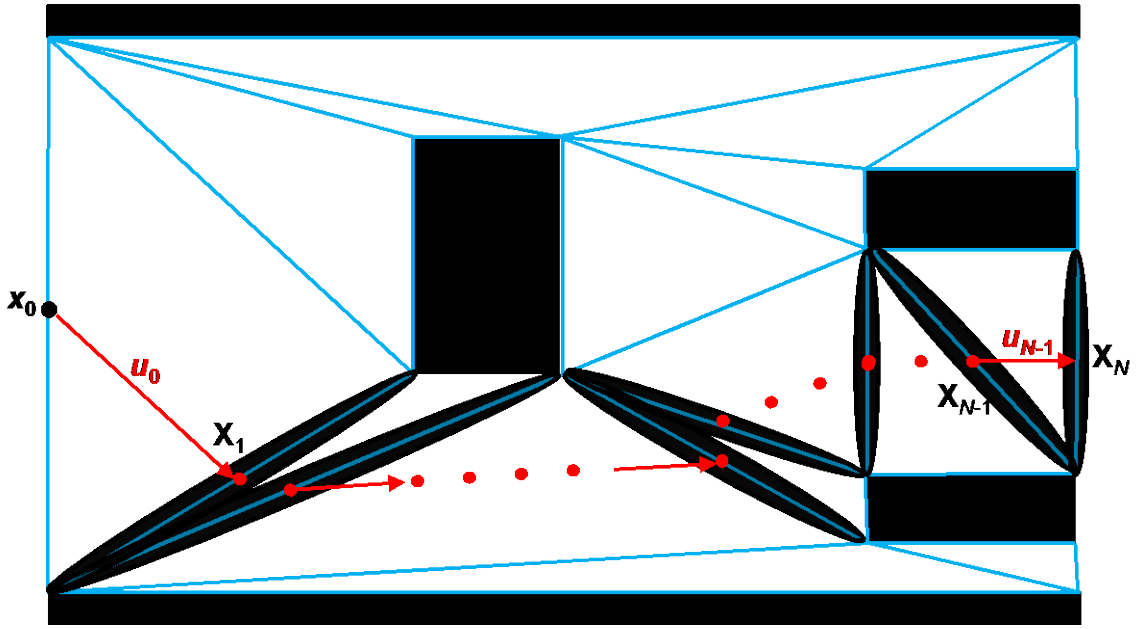
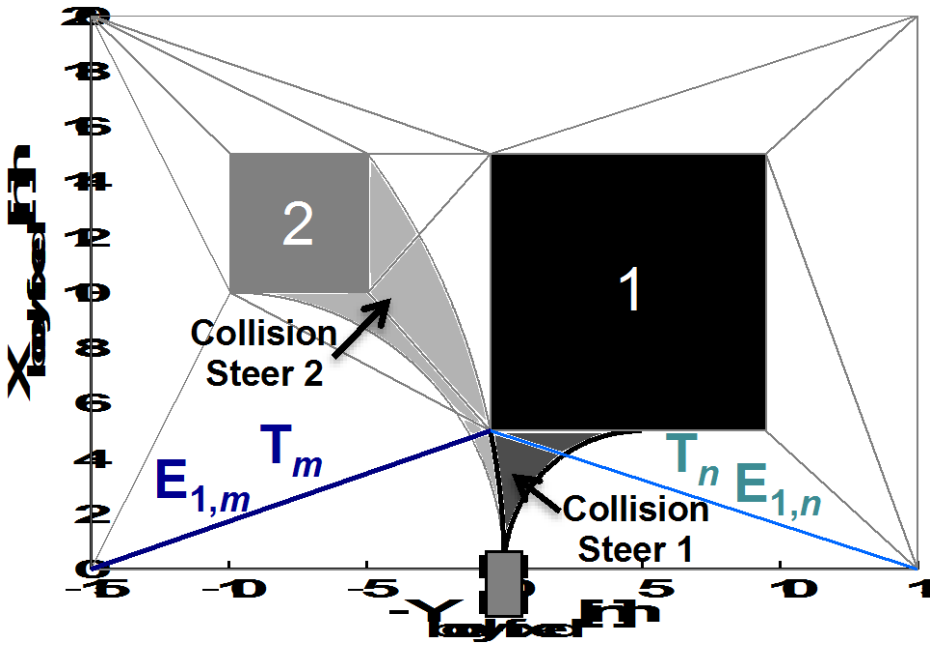


Figure 2.9. Illustration of effective target sets as edges of constrained Delaunay triangulation

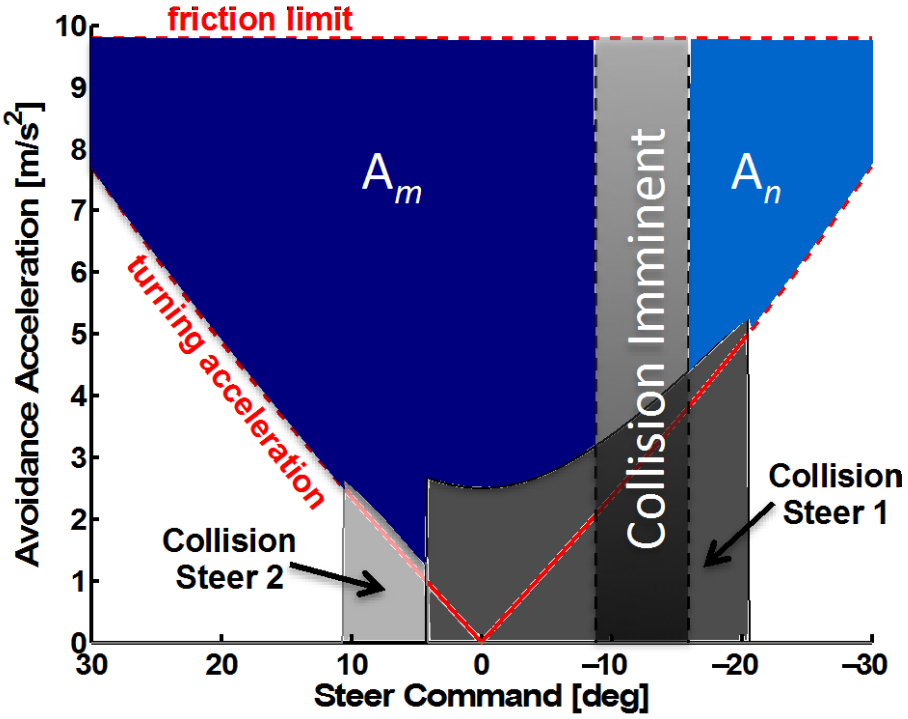
The "size" or volume of effective target sets or the effective target tube can be used to assess the dynamic reachability and current constraint "restrictiveness" of the homotopy itself from the vehicle's current state. In what follows, we develop this reachability heuristic using an adaptation of the dynamic window approach originally

presented in [106]. In their work, Fox, Burgard, and Thrun map the location of obstacles in the robot's environment to inadmissible translational and rotational robot velocities; those from which the synchro-drive robot's actuators could not prevent collision with an obstacle. The authors then search within a "dynamic window," or velocity space that can be reached within the next time interval, and calculate optimal translational and rotational velocities that allow the robot to progress toward a goal without colliding with obstacles.

The adaptation of the dynamic window approach presented here similarly considers the vehicle's actuator limitations, but uses those limitations, together with "avoidance accelerations" – the acceleration required to avoid collisions with obstacles – to approximate the margin of control freedom afforded by the available homotopies. Rather than map obstacles onto a 2-dimensional velocity space, this approach instead maps the total vehicle acceleration required to avoid obstacles onto the one-dimensional steering space of the vehicle. It then calculates the surplus tire friction available to the human driver if s/he were to steer into one homotopy or the other. Steering angles from which the vehicle cannot avoid a collision with an impending obstacle are excluded from this area calculation. Figure 2.10b illustrates one such region (labeled "Collision Imminent") corresponding to a range of steering angles from within which the vehicle cannot avoid the black obstacle at its current speed.


2
1
Collision
Steer 2
T_m
$E_{1,m}$
T_n $E_{1,n}$
Collision
Steer 1

(a)


friction limit
A_m
A_n
turning acceleration
Collision Imminent
Collision
Steer 2
Collision
Steer 1
Avoidance Acceleration [m/s^2]
Steer Command [deg]
10
9
8
7
6
5
4
3
2
1
0
30
20
10
0
–10
–20
–30

(b)

Figure 2.10. Illustration showing a) triangularized environment with obstacles (gray and black), and b) avoidance acceleration mapped to steering commands (with gray and black blocks corresponding to obstacles in (a).

Assuming constant velocity V, wheelbase length L, tire friction coefficient μ, gravity g, stationary obstacles (see 2.3.2 for treatment of moving obstacles), and no-slip conditions (turns of constant radius), the turning radius required to avoid an obstacle by passing it to the left (R_L) or to the right (R_R) by steering is given by Table 2.2 and Table 2.3.

Table 2.2. Avoidance radii calculation for polygonal obstacles

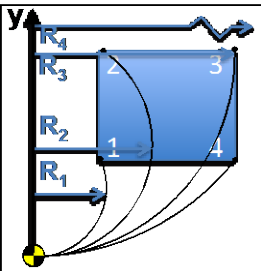
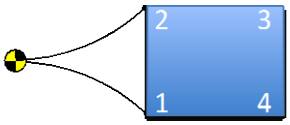
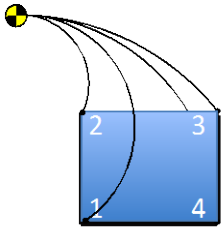
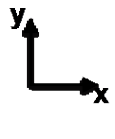
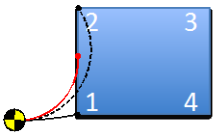
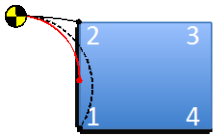
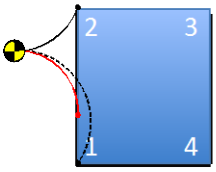
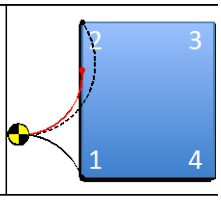
			
Condition	$y_2 < 0$ & $y_1 < 0$	$y_2 > 0$ & $y_1 < 0$	$y_2 > 0$ & $y_1 > 0$
Avoidance Radii	$R_L := \min(R_1, R_2, R_3, R_4)$ $R_R := \max(R_1, R_2, R_3, R_4)$	$R_L := \min(R_2, R_3)$ $R_R := \max(R_1, R_4)$	$R_L := \min(R_1, R_2, R_3, R_4)$ $R_R := \max(R_1, R_2, R_3, R_4)$

Table 2.3. Avoidance steer special cases

				
Condition	sign(y_2)=sign(y_1)		sign(y_2)≠sign(y_1)	
	$y_1<R_1$ & $y_2>R_2$		$y_1<R_1$	$y_2>R_2$
	$y_1>0$	$y_1<0$		
Avoidance Radius	$R_L := x_2$	$R_R := x_1$	$R_R := x_1$	$R_L := x_2$

The lateral acceleration required to avoid an obstacle to the left or to the right by steering alone is then given by

$$a_{lat,\bullet} = \frac{V^2}{R_\bullet}, \tag{2.4}$$

which for a front-wheel-steered vehicle with Dubins constraints, wheelbase L, and steering angle δ, can be approximated by

$$a_{lat,\bullet} = \frac{V^2 \tan(\delta_\bullet)}{L}. \tag{2.5}$$

As a function of vehicle steering angle, the longitudinal acceleration required to avoid the vehicle-facing edge of an obstacle at a distance x is then approximated by

$$a_{long} = \begin{cases} \dfrac{-V^2 \tan\delta}{2L \arctan\left(\dfrac{x \tan\delta}{\sqrt{L^2 - x^2 \tan^2\delta}}\right)} & \delta \neq 0 \\ \dfrac{V^2}{2x} & \delta = 0 \end{cases}. \tag{2.6}$$

while the longitudinal acceleration required to avoid the heading-tangent edge of an obstacle at a lateral distance y from the vehicle is approximated as

$$a_{long} = \frac{-V^2 \tan\delta}{2L \arctan\left(\frac{\sqrt{y\tan(\delta)\left(2L - y\tan(\delta)\right)}}{L - y\tan\delta}\right)}. \tag{2.7}$$

Thus, the total acceleration required to avoid collision with obstacle ●, with extremal steering commands $\delta_{\bullet,1}$ and $\delta_{\bullet,2}$ passing to its left and right, respectively, as a function of vehicle steer angle is given by

$$a_{total} = \sqrt{a_{lat}^2 + a_{long}^2}$$
$$= \begin{cases} 0 & \delta \le \delta_{\bullet,1} \\ \frac{1}{2}\sqrt{\frac{V^4 \tan\delta\left(4\phi^2 + 1\right)}{L^2\phi^2}} & \delta_{\bullet,1} \le \delta \le \delta_{\bullet,2} \\ 0 & \delta_{\bullet,2} \le \delta \end{cases}, \tag{2.8}$$

where

$$\phi = \arctan\left(\frac{x\tan\delta}{L - y\tan\delta}\right), \tag{2.9}$$

$$\delta = \arctan\left(\frac{2Ly}{x^2 + y^2}\right), \tag{2.10}$$

and

$$y = \frac{L - \sqrt{L^2 - \tan^2(\delta)x^2}}{\tan(\delta)}. \tag{2.11}$$

Summed over all steering angles corresponding to each homotopy choice, the surplus tire friction for homotopy i is then given by

$$a^i_{surplus} = \int_{\Delta^i_{\min}}^{\Delta^i_{\max}} \left[\mu g - \max\left(\{ a_j(\delta) : j = 1, \ldots, n \} \right) \right] d\delta \tag{2.12}$$

$$\begin{aligned} \Delta^i_{\max} &= \min\left(\delta^{\max}_{actuator}, \delta^{\max}_{kinematic}, \max\left(\delta_{E_{1,m}}(1), \delta_{E_{1,n}}(2) \right) \right) \\ \Delta^i_{\min} &= \max\left(-\delta^{\min}_{actuator}, -\delta^{\min}_{kinematic}, \min\left(\delta_{E_{1,m}}(1), \delta_{E_{1,n}}(2) \right) \right) \end{aligned} \tag{2.13}$$

where $\delta_{E1,\bullet}(1)$ and $\delta_{E1,\bullet}(2)$ are the extremal steering angles from which the two edges of set $E_{1,\bullet}$ can be reached. $\delta^{\bullet}_{actuator}$ refers to physical steering limits, and $\delta^{\bullet}_{kinematic}$ represents the maximum non-slip steering angle allowed by the tire friction and current vehicle velocity as given by

$$\delta^{\bullet}_{kinematic} = \pm \arctan\left(\frac{\mu m g L}{V^2} \right). \tag{2.14}$$

With heuristics L_k, w_k, ϕ_k, and $a_{surplus}$ thus calculated, a graph search (Dijkstra's algorithm is used here) may be performed to calculate the optimal path homotopy and its associated constraints. In the results shown below, the objective function is defined as

$$\min_{T_i \ldots T_n} \left\{ k_{DW} \frac{1}{a^i_{surplus}} + \sum_{i=0}^{n} \left(k_L\left(L_{i-1,i} \right) + k_W \frac{1}{\min\left(w_{i-1}, w_i \right)} + k_\phi \left| \phi_{i-1,i} \right| \right) \right\} \tag{2.15}$$

$$\text{s.t.} \quad \begin{aligned} X_G &\subseteq T_G \\ E_k &= \{ (T_k, T_{k+1}) \} \quad \text{where } E_k \text{ is unconstrained} \end{aligned} \tag{2.16}$$

This objective function incorporates an estimate of average homotopy "length" (from L) with an approximation of the control freedom it provides (from w), the dynamic stability it affords (from ϕ), and the present reachability it allows to the vehicle (from $a_{surplus}$).

2.2.3 CONSTRAINT-BASED NAVIGATION WITH MULTIPLE AGENTS

Once an optimal homotopy has been identified, the orientation of its centerline or the physical position of its edges can be enforced as constraints on the heading or

position of the vehicle by a lower level controller. While the constraint-based framework presented in this thesis is specifically designed for, and provides distinct advantages when combined with, shared-adaptive control strategies, the lower-level controller need not be semi-autonomous. In what follows, we demonstrate one example of a decentralized, autonomous controller taking advantage of the constraint-based framework's homotopy identification to improve efficiency of transport for multiple agents.

This demonstration adapts the control law presented by Chang et al. in [107] to navigate a group of vehicles through an obstacle course toward a goal. As presented by Chang et al., this control law uses scalar potentials to drive an agent toward the goal and gyroscopic forces and damping forces to avoid collisions with obstacles and other vehicles. In what follows, we compare this controller's performance and navigational efficiency with the performance and efficiency of a slightly modified version which makes use of CDT-derived homotopy constraints.

I) CONTROLLER SETUP

Given a point mass with mass m, state q, control input u, and second-order translational dynamics

$$\ddot{\vec{q}} = \frac{1}{m}\vec{u} \tag{2.17}$$

Chang et al. apply a control law $\mathbf{u}$ consisting of potential ($\mathbf{F}_p$), dissipative ($\mathbf{F}_d$), and gyroscopic ($\mathbf{F}_g$) forces described by equations (**2.18**) – (**2.21**)

$$\vec{u} = \mathbf{F}_p + \mathbf{F}_d + \mathbf{F}_g \tag{2.18}$$

$$\mathbf{F}_p = -\nabla V(\vec{q}) \tag{2.19}$$

$$\mathbf{F}_d = -\left(D_{\boldsymbol{k}} + D_b(\vec{n})\right)\dot{\vec{q}} \tag{2.20}$$

$$\mathbf{F}_g = -S(\vec{n}, \dot{\vec{q}})\dot{\vec{q}} \tag{2.21}$$

As given in [107], the potential force Fp, is calculated as

$$V(\vec{q}) = \frac{1}{2}\mathbf{K_p}\left\|\vec{q} - \vec{q}_G\right\|^2. \tag{2.22}$$

The damping force $\mathbf{F}_d$ consists of a positive definite dissipative damping constant D_{dc}, and a positive definite braking component D_b for avoiding frontal collisions with obstacles. With the vector $|\vec{n}| \le R_s$ representing the distance and direction to the nearest detected obstacle, D_b is given by

$$D_b(\vec{n}) = C_1 \exp\left(-\left\|\vec{n}\right\|\right) - C_2 . \tag{2.23}$$

Finally, the gyroscopic force $\mathbf{F}_g$ is calculated as

$$\mathbf{F}_g = \begin{bmatrix} 0 & -S_g \\ S_g & 0 \end{bmatrix} \dot{\vec{q}}, \tag{2.24}$$

where

$$S_g = C_3 \cdot \text{sign}\left(\vec{n} \times \dot{\vec{q}}\right) \cdot \exp\left(-\left\|\vec{n}\right\|\right) - C_4 , \tag{2.25}$$

Figure 2.11 illustrates the forces applied by this control law.

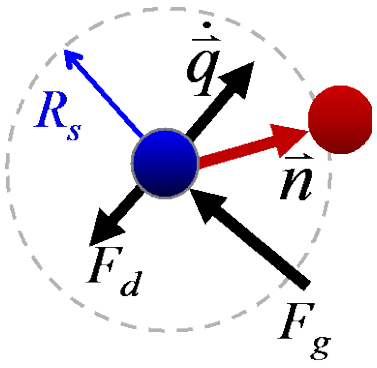


Figure 2.11. Diagram of a vehicle with detection shell R_s, normal vector to nearest obstacle (*n*) and corresponding control forces

II) SIMULATION SETUP

Two embodiments of this control strategy were simulated to demonstrate the utility of triangulation-derived constraints in a distributed, multi-agent target acquisition and obstacle avoidance scenario. In the first test configuration, the gyroscopic controller was implemented as described in [107] – with the attractive potential force $\mathbf{F}_g$ directed toward the target from each vehicle's current position.

The second configuration replaced the target attractive force with a proportional control on the vehicle's heading. At each time step in this approach, each vehicle triangulated the known environment, planned an optimal sequence of triangles through it, and proportionally controlled the vehicle heading to the orientation, ϕ_k of the resulting channel. In both cases, vehicle velocities were taken as constant ($\mathbf{F_d}$ = 0). Table 2.4 summarizes the control law used by each vehicle in the corridor-tracking case. Note that with the exception of lines 3 and 4, the control simulation (without corridors) was identical (replacing the reference heading in line 5 with the reference heading from the vehicle toward the goal).

Table 2.4. Pseudocode for control law used in corridor-based controller

1	**FOR each vehicle v**
2	**Identify** all obstacles within sensing radius R_s
3	**Decompose** known/static environment (from map) into a contiguous set of Delaunay Triangles (note that this decomposition does not account for unanticipated or previously-unmapped obstacles.
4	**Plan** an optimal sequence of adjacent triangles from triangle containing the current vehicle position to the goal using graph search. Dijkstra's algorithm was used here, though others can also be used.
5	**Compute** turning force F_p (align vehicle heading with reference heading of current triangle)
6	**Identify** nearest (frontal) hazard and construct normal vector to that hazard
7	**Compute** gyroscopic avoidance force F_g (avoid nearest frontal hazard)
8	**Execute** control action $\vec{u} = \mathbf{F}_p + \mathbf{F}_d + \mathbf{F}_g$

Thirty vehicles, each performing decentralized constraint planning and control, start at X=0, -6<Y<6 and travel toward a goal region at X=200, -10<Y<10. Rectangular obstacles (shown in gray in Figure 2.12 and Figure 2.13) are assumed to be known globally (as though from a map or road database), while the circular obstacles shown in red are only knowable locally via each vehicle's local sensing.

III) SIMULATION RESULTS

Figure 2.12 shows the vehicle paths resulting from navigation without a homotopy-guided controller. While all thirty vehicles eventually arrive collision-free at the target, their routes are inefficient and involve significant maneuvering to avoid

vehicle-vehicle and vehicle-obstacle collisions. In light of the reactive collision avoidance strategy being employed by each vehicle, these results are not surprising; traveling blindly toward the goal causes each vehicle to spend a significant amount of time maneuvering along the face of intermediate obstacles and directly into the path of oncoming vehicles.

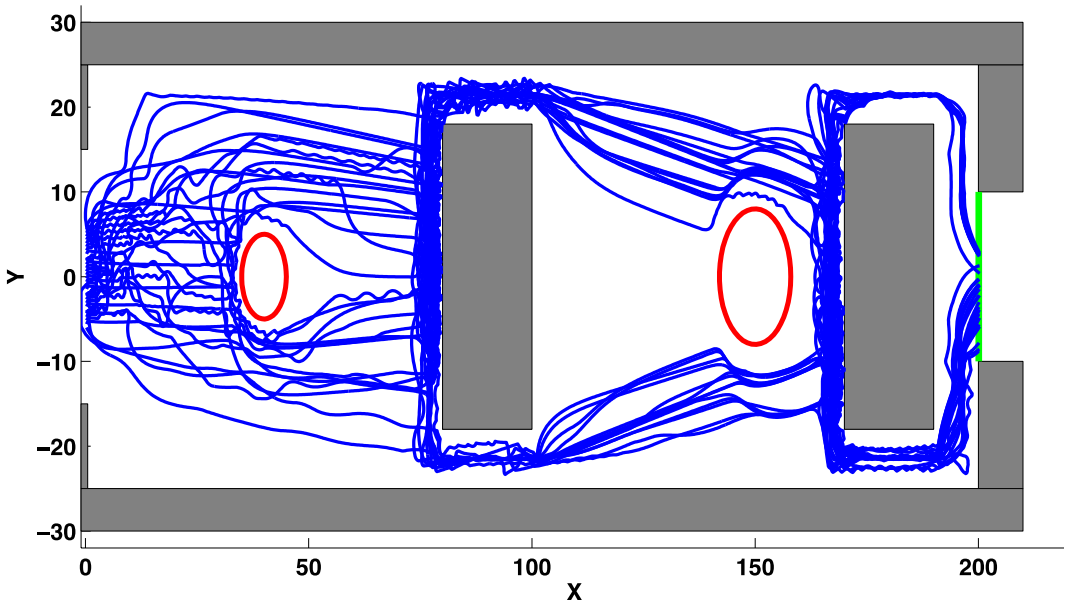


Figure 2.12. Simulation results showing traces of the paths taken by 30 vehicles without corridor planning capability

When homotopy constraints are incorporated into the goal seeking behavior of each vehicle, the common orientation shared by the triangles comprising these homotopies (as seen from each vehicle's homotopy planner) provides order to an otherwise decentralized control strategy. That is, within each *Type 2* triangle, vehicles share the same orientation reference. While navigating through *Type 1* triangles, these orientation references can differ by up to 180°, though while traversing these triangles, vehicles are free to violate any of the unconstrained edges if necessary to avoid a collision. This alignment of vehicle references does not require vehicle-to-vehicle communication and leads to more efficient vehicle paths that require less collision avoidance maneuvering compared to decentralized strategies that rely solely on potential fields and local collision avoidance.

As Figure 2.13 shows, when using this approach, all thirty vehicles again successfully navigate the course without collisions, though with significantly better efficiency. Note that when traversing narrow corridors with unconstrained edges, the reactive collision avoidance controller (**2.18**) causes the vehicles to fan out, with some crossing into unrestricted regions of the environment. This increases vehicle-to-vehicle clearance and eases congestion without significantly affecting completion time. Note that in this particular scenario, all thirty vehicles happened to choose the same homotopy. While constraint plans starting from different locations may and often will be unique, their construction from a common workspace and identical homotopy evaluation criterion ensures that all vehicles passing through a triangle T_i share a similar orientation reference and direction of travel.

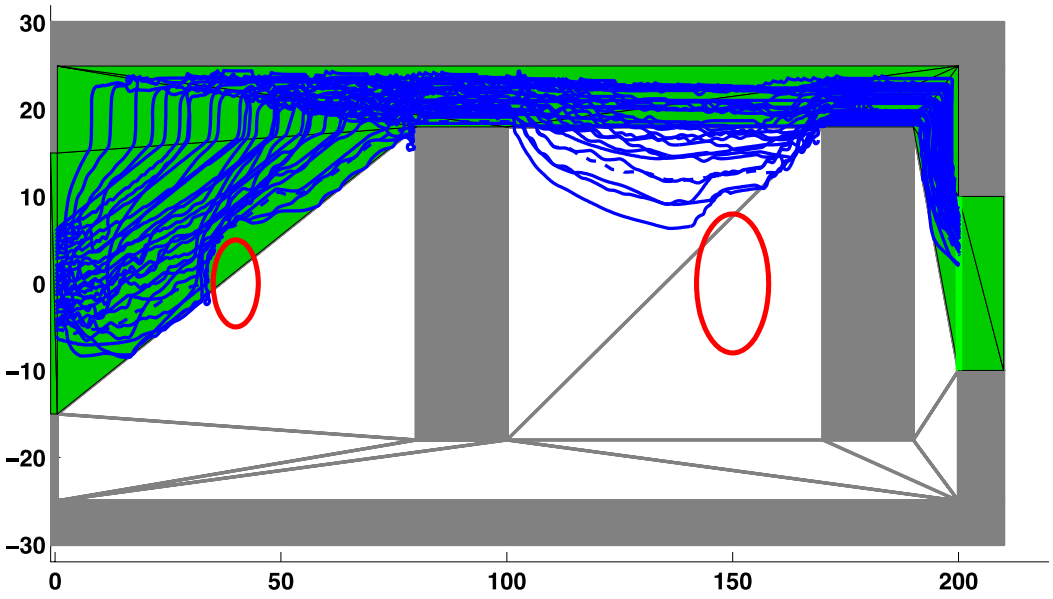


Figure 2.13. Simulation results showing traces of the paths taken by 30 vehicles with corridor planning capability

Figure 2.14 plots a distribution of the time that vehicles employing each controller configuration required to complete the course. Note that due to their shorter, more efficient paths, vehicles navigating with homotopy constraints reduced average course completion time by 25% relative to vehicles navigating without these constraints. this

decrease was statistically significant: without constraints, vehicles required an average 61 seconds to complete the course. With them, the average dropped to 46 seconds ($F(1,58) = 166.9$, $p < 1e\text{-}18$). Uniformity provided by a similar set of constraints also significantly decreased variation in course completion time, from a 5.8 second standard deviation for the controller that did not consider homotopies, to 1.6 seconds for the homotopy-following controller.

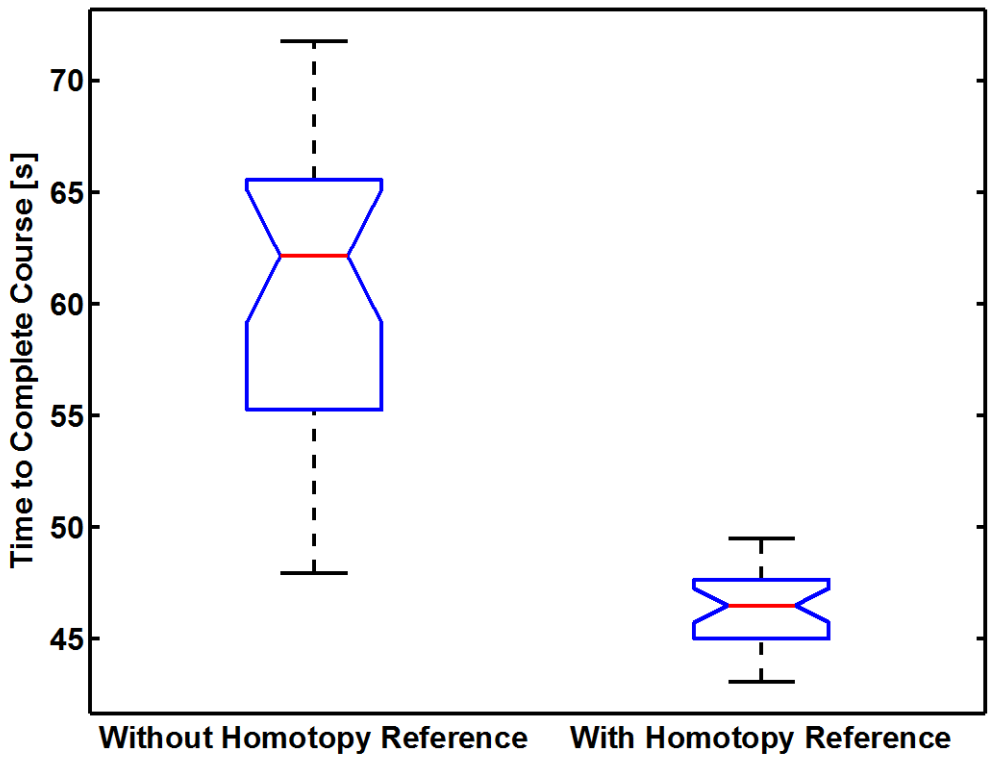


Figure 2.14. Course completion time for controllers with and without a homotopy reference

The time each vehicle spent actively avoiding imminent collisions also decreased significantly as a proportion of total travel time with the homotopy-following controller enabled. Since all vehicles shared a common velocity, we estimate collision risk by the time it spends avoiding all "head-on" collisions with the nearest obstacle within its detection radius. More specifically, we define a head-on collision incident for vehicle *i* as one in which $\dot{\vec{q}}_i \cdot \dot{\vec{q}}_j < 0$ when $\left\|\vec{q}_i - \vec{q}_j\right\| \le R_s$ where $\left\|\vec{q}_i - \vec{q}_j\right\| < \left\|\vec{q}_i - \vec{q}_k\right\| \quad \forall k \ne i, j$. Figure 2.15 illustrates one such incident.

As shown in Figure 2.16, the average percentage of course navigation time each vehicle spent avoiding head-on collisions with other vehicles was reduced by 83% when homotopy constraints are enforced (from M = 10%, SD = 3% to M = 1.7%, SD = 1%). This reduction was also statistically significant given a 95% confidence interval ($F(1,58)$ = 221, p < 1e-20). This result follows simply from the observation made above; homotopy constraints in a given obstacle field provide a common reference and direction to vehicles traveling within them. Enforcing those constraints allows vehicles to avoid much of the uncertainty and conflicting trajectories inherent to purely local navigation.

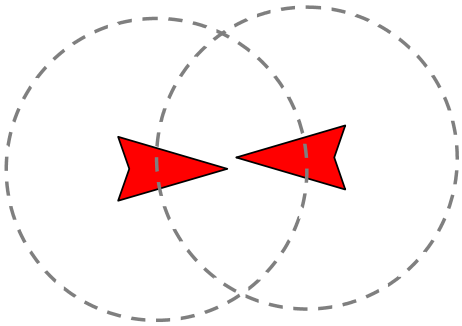


Figure 2.15. Illustration of a "head-on" collision incident

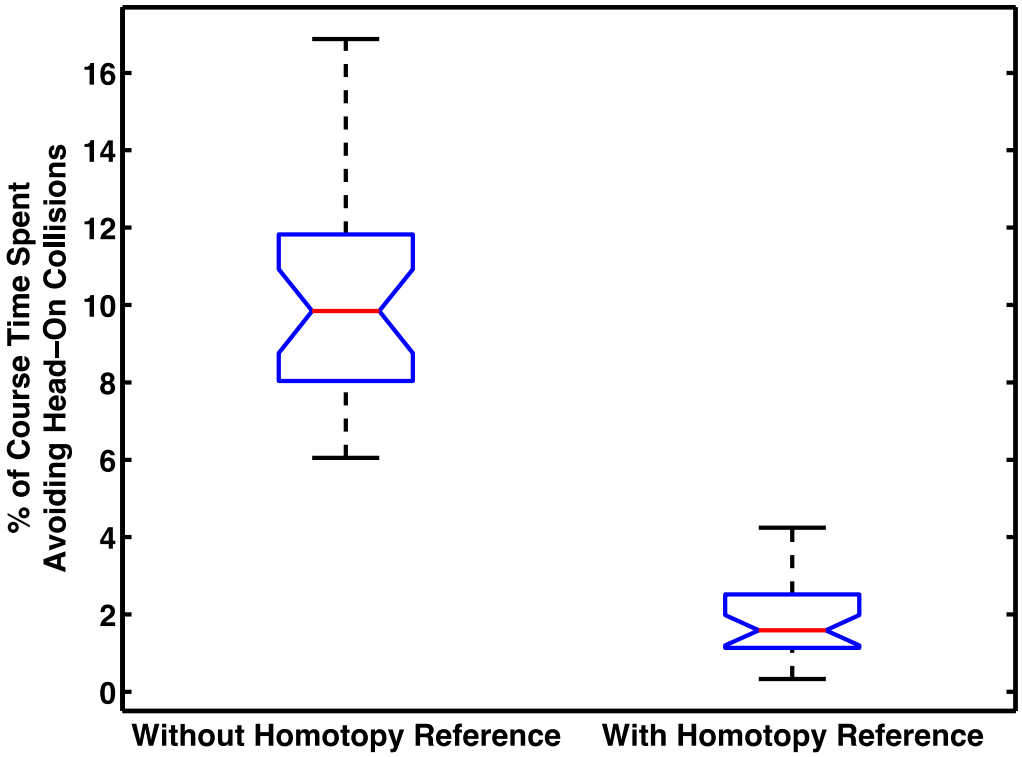


Figure 2.16. Percent of total course time spent avoiding head-on collision for controllers with and without a homotopy reference

2.3 CONSTRAINT DESIGN WITH PATHS

While the method described in Section 2.2 provides a particularly convenient means of identifying and evaluating constraints from the environment's homotopies and effective target sets, it is not the only approach to homotopy-based constraint design. Given that the function of the constraint planner is to identify an optimal homotopy and design constraints to keep the vehicle safely inside it (with or without a human in the loop), the method it uses need only be fast, dynamically aware (to ensure that constraints admit dynamically feasible trajectories), and predictive (in order to provide sufficient preview for predicting threat and engaging control intervention). This section demonstrates a, path-based alternative to the triangulation-based approach that uses dynamic programming to calculate a dynamically optimal, zero-width path and expands that path into a set of position constraints bounding the homotopy that contains it. Reachable set constraints, transition symmetry, and maneuver primitives are also introduced as a means of reducing the computational burden.

2.3.1 STATE REDUCTION VIA ACKERMANN POINT TRANSFORMATION

Representing its steering and acceleration inputs in the body-fixed frame, a vehicle with negligible roll dynamics can be modeled by

$$\begin{bmatrix} ma_x \\ ma_y \\ I_{zz}\ddot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & x_f & 0 & x_r \end{bmatrix} \begin{bmatrix} F_{xf} \\ F_{yf} \\ F_{xr} \\ F_{yr} \end{bmatrix} \quad \textbf{(}2.26\textbf{)}$$

where F_{xf}, F_{yf}, F_{xr}, and F_{yr} represent the longitudinal and lateral (in a body-fixed frame) tire forces at the front and rear wheels respectively, as illustrated in Figure 2.17.

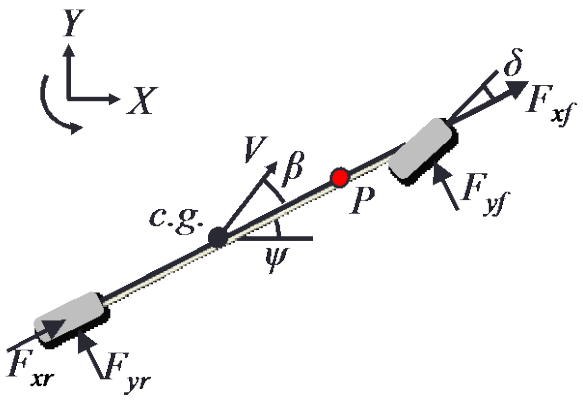


Figure 2.17. Illustration of vehicle model used in constraint planning

In order to reduce the computational complexity of the dynamic programming algorithm, we simplify these dynamics by assuming a constant forward velocity, *V*, and applying a state transformation originally described by Ackermann in [108]. This transformation effectively decouples the effect of front and rear lateral forces on the vehicle lateral dynamics and allows for the approximation of steering maneuvers as constant radius turns decoupling point, *P*. Transforming (2.26) using the method described by Ackermann yields

$$\begin{bmatrix} ma_x \\ ma_y + \frac{I_{zz}}{x_r}\ddot{\psi} \\ I_{zz}\ddot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & \left(1+\frac{x_f}{x_r}\right) & 0 & 0 \\ 0 & x_f & 0 & -x_r \end{bmatrix} \begin{bmatrix} F_{xf} \\ F_{yf} \\ F_{xr} \\ F_{yr} \end{bmatrix}, \tag{2.27}$$

which decouples the lateral acceleration at a point distanced $x_p = I_{zz}/mx_r$ ahead of the vehicle's center of gravity (labeled *P* in Figure 2.17) from the vehicle yaw dynamic. The lateral acceleration of this point may then be written as

$$a_{yp} = \frac{x_f + x_r}{mx_r} F_{yf}, \tag{2.28}$$

where the lateral force at the front wheels is a nonlinear function of the tire stiffness curve, C_f, and the lateral sideslip at the front wheels, α_f:

$$\begin{aligned} F_{yf} &= -C_f\left(\alpha_f\right)\alpha_f \\ &= -C_f\left(\alpha_f\right)\left[\tan^{-1}\left(\frac{v_y + x_f\psi}{v_x}\right) - \delta\right] \\ &\approx -\tilde{C}_f\left(\frac{v_y + x_f\psi}{v_x} - \delta\right) \end{aligned} \quad \textbf{(2.29)}$$

.

In a strict implementation of the path computed by the dynamic program, a low level controller might be used to manipulate v_y, v_x, and δ and thereby track the desired F_{yf}. In the constraint-planning implementation presented here, however, only the (physically-realizable) lateral acceleration a_{yp} required to avoid obstacles is of interest as it provides an objective (friction-constrained) gauge by which to penalize maneuver aggressiveness or nearness to instability. Ignoring friction circle effects (assuming constant velocity and no longitudinal acceleration), this friction limit on achievable lateral acceleration is described by

$$\left|a_{yp}\right| \le \mu g \quad \textbf{(2.30)}$$

Considering piecewise-constant lateral acceleration of the Ackerman point (constant-radius turns) as the input action, discretizing x, y, and ψ over a preview horizon, and considering continual forward progression between fixed Δx intervals, the state transitions are given by:

$$x_{k+1} = x_k + \Delta x \quad \textbf{(2.31)}$$

$$y_{k+1} = y_k + \frac{2V^2\cos\psi_k - \sqrt{4V^4\cos^2\psi_k - 4a_{yp}\Delta x\left(a_{yp}\Delta x + 2V^2\sin\psi_k\right)}}{2a_{yp}} \quad \textbf{(2.32)}$$

$$\psi_{k+1} = \psi_k + \sin^{-1}\left[\kappa_k\left(\Delta x\cos\psi_k + \Delta y_k\sin\psi_k\right)\right] \quad \textbf{(2.33)}$$

where curvature κ_k is given by

$$\kappa_k = 2\frac{\Delta y_k \cos\psi_k - \Delta x \sin\psi_k}{\Delta x^2 + \Delta y_k{}^2} \tag{2.34}$$

and

$$\Delta y_k = \frac{2V^2\cos\psi_k - \sqrt{4V^4\cos^2\psi_k - 4a_{yp}\Delta x\left(a_{yp}\Delta x + 2V^2\sin\psi_k\right)}}{2a_{yp}} \tag{2.35}$$

2.3.2 CONSIDERATION OF DRIVING OBJECTIVES AND ENVIRONMENTAL DISTURBANCES

In its simplest form, a cost function penalizing lateral acceleration and distance traveled is used in the finite horizon DP problem. This penalty on lateral acceleration is consistent with the objective of finding the most stable/controllable (and comfortable) path through the constraint space. Penalizing distance traveled provides the necessary incentive for the vehicle (assumed to be moving at a constant speed) to progress along the road. Equations (**2**.**36**)-(**2**.**38**) illustrate this cost function.

$$J_{base}(x_0) = h_{base}(\mathbf{x}[N]) + \sum_{n=0}^{N-1} g(\mathbf{x}[n], a_{lat}[n], n) \tag{2.36}$$

$$g(\mathbf{x}[n], a_{lat}[n], n) = g_{base} = K_a\left(\frac{a_{lat}[n]}{a_{\max}}\right)^2 + K_d\left(\Delta x^2 + \Delta y[n]^2\right) \tag{2.37}$$

$$h_{base}(\mathbf{x}[N]) = 0 \tag{2.38}$$

Consideration of rules of the road, moving obstacles, and terrain effects may be incorporated into the constraint planning problem via various modifications of this cost function. For example, assuming that lane directions are known, the constraint planner can be biased to drive in the appropriate lane (or, more specifically, to avoid the oncoming lane where possible) by augmenting (**2**.**37**) as

$$g = g_{base} + \begin{cases} 0 & y < y_{centerline} \\ K\left(y - y_{centerline}\right)^2 & y \geq y_{centerline} \end{cases} \quad \textbf{(2.39)}$$

Constraint planning objectives may be further extended to consider hazard motion by predicting the anticipated intersection of the host vehicle with dynamic hazards and shifting the regions of high cost corresponding to predicted collision states accordingly. Many methods for deterministic and probabilistic collision state prediction have been proposed in the literature[37], [109]–[111]. In this work, perfect sensing or vehicle-to-vehicle communication is assumed, and collision states are predicted by estimating time to collision as follows.

Given a (constant) host velocity $\dot{x}_{host}$ and obtaining the current velocity of roadway hazards $\dot{x}_{haz}$ from tracking sensors or vehicle-to-vehicle communication, where $x_{host}(t)$ and $x_{haz}(t)$ represent the current position of the host and hazard, respectively at time t, the estimated time to collision Δt_c evaluated at time t_0 was evaluated by

$$\Delta t_c\big|_{t_0} = t_c - t_0 = \begin{cases} -\dfrac{\tilde{x}(t_0)}{\dot{\tilde{x}}(t_0)} & \begin{cases} \text{for } \dot{\tilde{x}}(t_0) < 0, \;\; \tilde{x}(t_0) > 0 \\ \text{or } \;\; \dot{\tilde{x}}(t_0) > 0, \;\; \tilde{x}(t_0) < 0 \end{cases} \\ \pm\infty & \begin{cases} \text{for } \dot{\tilde{x}}(t_0) \leq 0, \;\; \tilde{x}(t_0) < 0 \\ \text{or } \;\; \dot{\tilde{x}}(t_0) \geq 0, \;\; \tilde{x}(t_0) > 0 \end{cases} \end{cases} \quad \textbf{(2.40)}$$

to first order where

$$\dot{\tilde{x}}(t) = \dot{x}_{haz}(t) - \dot{x}_{host}, \quad \textbf{(2.41)}$$

and

$$\tilde{x}(t) = x_{haz}(t) - x_{host}(t), \quad \textbf{(2.42)}$$

or

$$\Delta t_c\big|_{t_0} = \begin{cases} \dfrac{-\dot{\tilde{x}}(t_0) - \sqrt{\dot{\tilde{x}}^2(t_0) - 2\ddot{x}_{haz}(t_0)\cdot\tilde{x}(t_0)}}{\ddot{x}_{haz}} & \begin{cases}\text{for } \ddot{x}_{haz}(t_0)\neq 0,\ \dot{\tilde{x}}(t_0)<0\\ \text{or } \ddot{x}_{haz}(t_0)<0,\ \dot{\tilde{x}}(t_0)\geq 0\end{cases} \\ -\dfrac{\tilde{x}(t_0)}{\dot{\tilde{x}}(t_0)} & \text{for } \ddot{x}_{haz}(t_0)=0,\ \dot{\tilde{x}}(t_0)<0 \\ \pm\infty & \begin{cases}\text{for } \ddot{x}_{haz}(t_0)>0,\ \dot{\tilde{x}}(t_0)\geq 0\\ \text{or } \ddot{x}_{haz}(t_0)=0,\ \dot{\tilde{x}}(t_0)\geq 0\end{cases} \end{cases} \quad \textbf{(2.43)}$$

to second-order (requiring that $\tilde{x}(t_0)\geq 0$ in (2.43)).

Given $\Delta t_c\big|_{t_0}$, the x position of each road hazard at t_c is then estimated as

$$x_{haz}\left(t_c\big|_{t_0}\right) = x_{haz}(t_0) + \dot{x}_{haz}(t_0)\cdot\Delta t_c\big|_{t_0} \quad \textbf{(2.44)}$$

or

$$x_{haz}\left(t_c\big|_{t_0}\right) = x_{haz}(t_0) + \dot{x}_{haz}(t_0)\cdot\Delta t_c\big|_{t_0} + \frac{1}{2}\ddot{x}_{haz}(t_0)\cdot\Delta t_c\big|_{t_0}^{\,2} \quad \textbf{(2.45)}$$

to first- and second-order, respectively. Assuming some knowledge of true hazard depth Δx_{haz} (which may in practice be difficult to gain from onboard sensors alone), the depth of the predicted collision state from the host vehicle's perspective may be estimated by

$$\Delta x_{haz}\left(t_c\big|_{t_0}\right) \approx \Delta x_{haz}(t_0) - \dot{x}_{haz}(t_0)\cdot\left(\Delta x_{haz}(t_0)/\dot{\tilde{x}}(t_0)\right). \quad \textbf{(2.46)}$$

For a vehicle with negligible roll dynamics, terrain effects (such as sloped shoulders) change the magnitude and direction of the accelerations it experiences during turning maneuvers. More specifically, a sloped roadway decreases the magnitude of friction-critical normal forces at the tires while simultaneously applying an additional lateral acceleration on the vehicle's center of gravity. Sloped road shoulders were considered in the constraint planner by modifying (**2.37**) to read

$$g(\mathbf{x}[n], a_{lat}[n], n) = K_a\left(\frac{a_{lat}[n]-a_{base}}{a_{\max}}\right)^2 + K_d\left(\Delta x^2 + \Delta y[n]^2\right), \quad \textbf{(2.47)}$$

where in this formulation,

$$a_{base} = g \sin\theta \cos\varphi \tag{2.48}$$

represents the lateral acceleration caused by gravity (independent of steering or F_{yf} command). Acceleration constraints imposed on the tires in this formulation are reduced to

$$\left|a_{\max}\right| \le \mu g \cos\theta \,. \tag{2.49}$$

Figure 2.18 illustrates θ and φ on a common road shoulder.

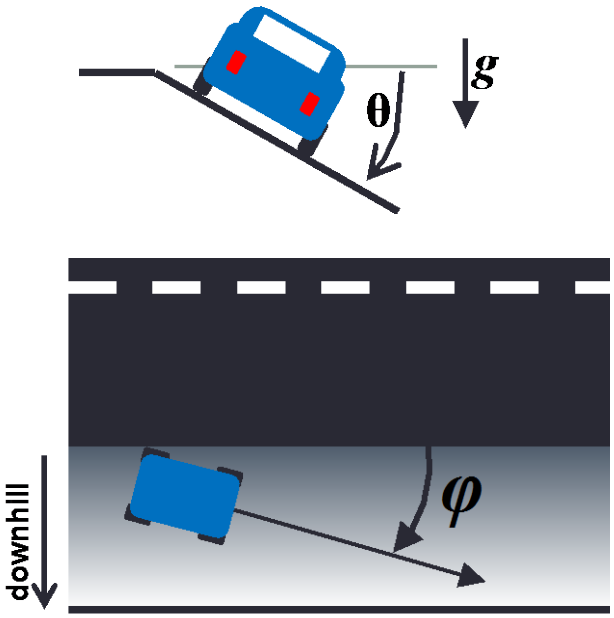


Figure 2.18. Illustration of sloped road shoulder and associated parameters

2.3.3 SIMULATED TRAVERSAL OF AN OPTIMAL HOMOTOPY

The simulations that follow demonstrate constraint-based navigation using dynamic programming to guide homotopy constraint planning.

I) SETUP

These simulations assume that road lane data is available and that the instantaneous position, velocity, and acceleration of road hazards have been measured or

estimated by on-board sensors or vehicle-to-vehicle communication devices. At each timestep, a dynamic programming problem using one of the cost functions described above and the state discretization described in Appendix D, is solved to obtain an optimal vehicle path from the vehicle's current position to the limits of onboard sensing (~80 m). A *y*-convex corridor bounding this path's homotopy is then calculated and enforced by a Model Predictive (MPC) Controller as a constraint on vehicle's lateral position. Subject to these constraints, this MPC controller plans and tracks a stability-optimal trajectory that keeps the vehicle within the desired homotopy through a 40-sample (~40m at 0.5 sec/sample and V = 20 m/s) prediction horizon. Section 3.2 describes this controller in greater detail. Note that the simulations presented below implemented the MPC control actions autonomously for the sake of demonstration. The dynamics of the vehicle were simulated using a nonlinear MSC Adams model of a generic light truck (described in Appendix A). Tire forces were approximated from a Pacejka tire model [112].

II) RESULTS

Figure 2.19 illustrates a DP-generated cost function (sampled on the ψ=0 plane), together with its path plan and the resulting constraint corridor (red wall) enforced by the MPC controller.

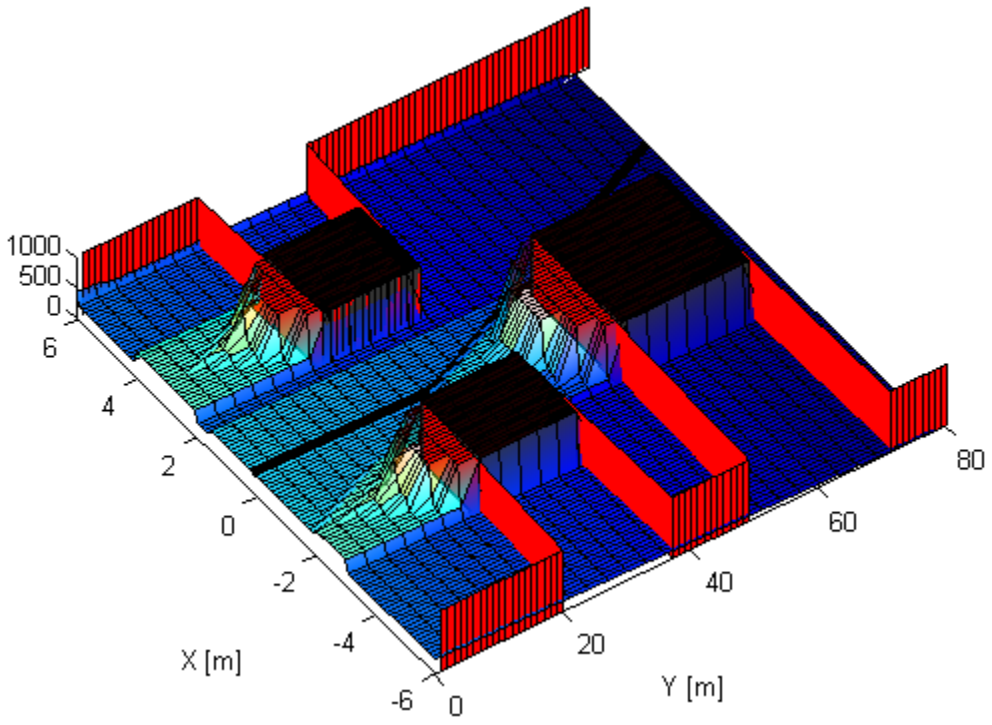


Figure 2.19. Illustration of DP-generated cost surface, path, and constraint plan

Figure 2.20 illustrates how this cost varies within the three-dimensional state cube x, y, and ψ. Note that the long and relatively narrow road/shoulder surface causes yaw angles larger than ~30° to be heavily penalized as these require large (accumulated) accelerations to recover and remain inside the region of interest. Also note that regions of low to moderate cost existing on any given slice roughly correspond to "effective target sets" similar to those described by [105] and exploited in Section 2.2.2.

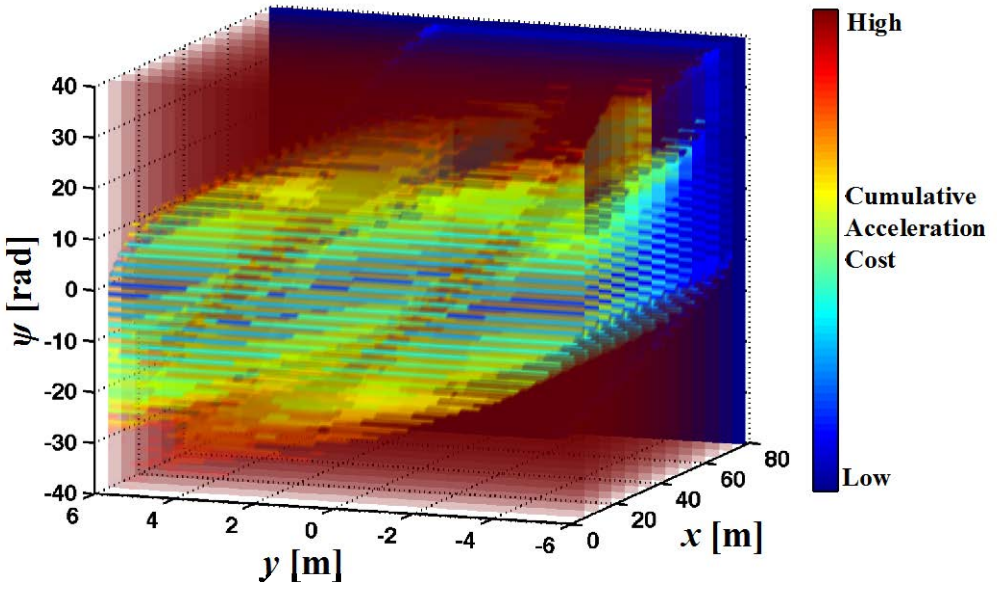


Figure 2.20. Illustration of objective function cost in the state cube

Figure 2.21 shows a snapshot of an obstacle avoidance scenario in which the DP planner calculates a minimum-acceleration, minimum-time homotopy through a non-convex obstacle field. Note that in this simulation, the symmetric hazard setup (translucent gray rectangles), together with the initial vehicle state at $x=y=\psi=0$, makes the choice of homotopies passing to the left or right of the first obstacle rather arbitrary. In a semi-autonomous implementation of the MPC control law (described in Chapter 3), this directional neutrality in homotopy selection would allow the human driver to bias the constraint plan by changing the initial conditions it uses in its goodness/feasibility estimate (where the control authority available to do so would be much greater in low threat scenarios such as before the vehicle reaches the first obstacle).

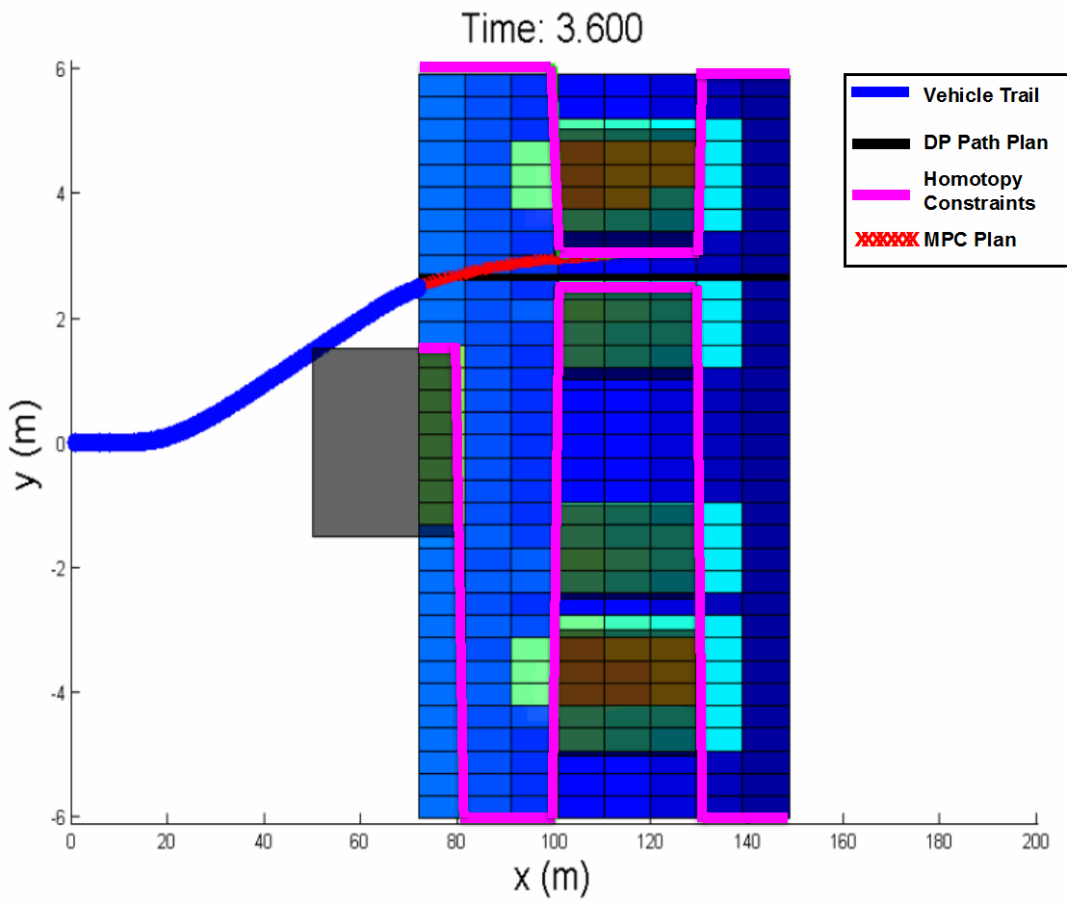


Figure 2.21. Snapshot of an obstacle avoidance simulation (obstacles stationary)

Just as directional neutrality in the corridor plan can be beneficial in some scenarios (like the semi-autonomous application discussed above), not considering preferred regions of the roadway in the constraint plan may present a liability in others. One such scenario includes that experienced by a vehicle traveling on a bi-directional highway. In this scenario, a careless or inattentive driver input may cause the corridor planner to draw a path that deviates into an oncoming lane in spite of the fact that another corridor of similar expected "goodness" exists in the host lane. Consideration of lane directionality and overall "goodness" of different road regions motivates the use of cost incentives in the corridor plan that bias it toward selecting "desirable" regions of the road surface. The relative magnitude of these cost incentives (with respect to penalties on friction utilization, distance, etc.), however, requires careful tuning to avoid causing the planner to choose an overly-aggressive corridor in the host lane when a safer (and perhaps even obstacle-free) option exists in the opposite lane. Figure 2.22 shows how the

DP cost function defined by (2.36) and (2.39) affects the corridor plan and subsequent vehicle trajectory through an obstacle field. Notice that in this simulation, the opposing lane was defined by $y \geq y_{centerline} = 0$.

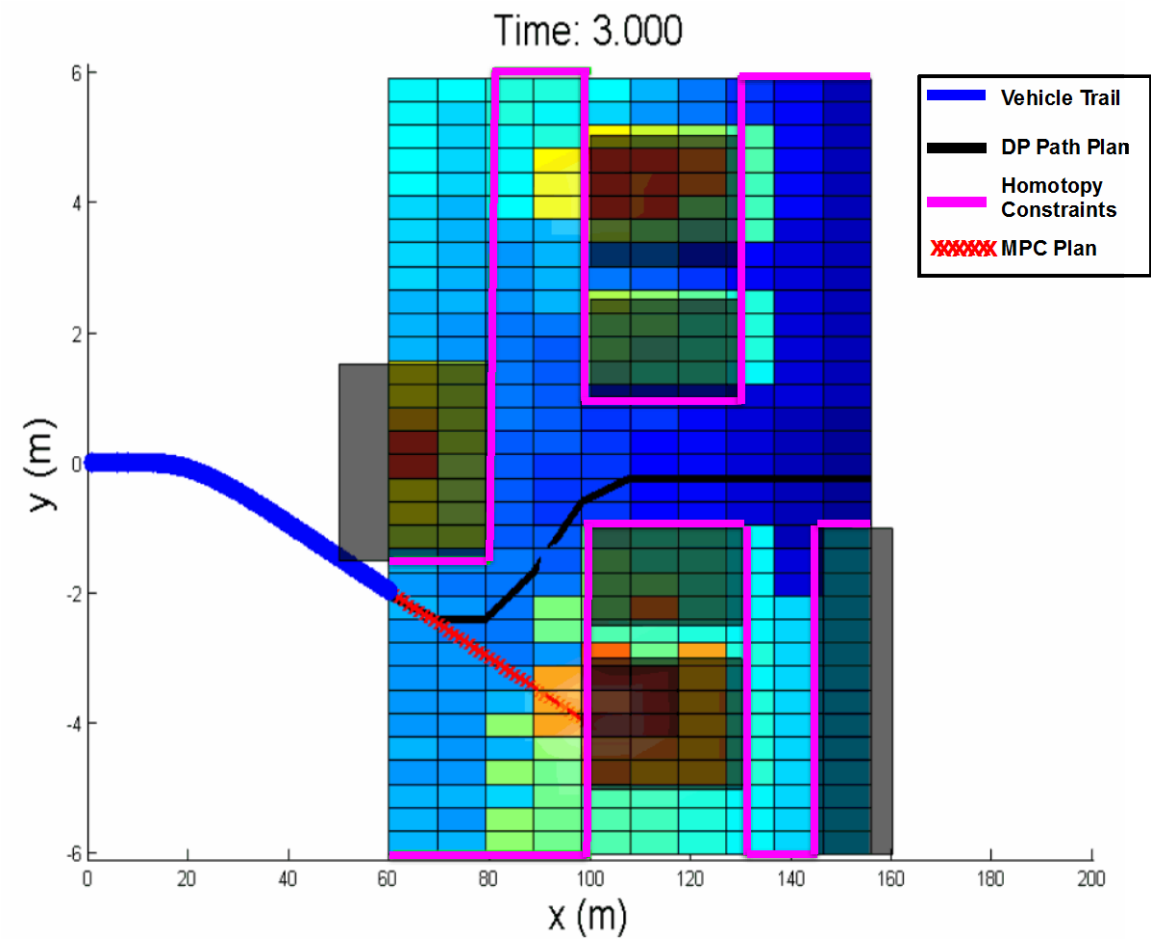


Figure 2.22. Snapshot of a simulation with penalties applied for deviation into an oncoming lane

Beyond highlighting the utility of lane/region penalties on the corridor plan, this simulation demonstrates how the DP algorithm's additional lookahead (relative to that used by the MPC controller) can improve the locally-optimality MPC solution by providing an optimal corridor plan through a larger and non-convex configuration space. Note that at t = 3s, the MPC solution is as yet unaware of the large obstacle looming at x = 145m. Were a corridor to be planned around the MPC's predicted solution, the vehicle might pass through the second column of obstacles at y=-5 or $-3 \leq y \leq -2.5$, resulting in a

"dead-end" scenario that would require a dangerous avoidance maneuver. With its added preview distance, the DP planner appropriately calculates a path and designs corridor constraints that ensure the MPC solution remains within the more dynamically feasible homotopy. Figure 2.23 shows the vehicle, corridor, and path plan(s) a few seconds later.

Notice that although the DP solution seeks to avoid traveling in the opposing lane, the MPC solution applies no such penalty, causing it to travel freely in either lane when no obstacles are present. Were another hazard to appear (at, e.g., 210m), the MPC solution would be constrained to pass it on the DP-preferred side. This partial decoupling of corridor plan and MPC objectives is seen as a desirable characteristic where semi-autonomous operation is concerned. That is, the goal of corridor constraints is to provide the driver with as much freedom as possible in the absence of road hazards. On a two-lane highway, for example, the driver should be free to choose a preferred trajectory and lane of travel if the roadway is clear and lane markings allow. Only in the presence of obstacles (and corresponding homotopy bifurcations) does the lane convention restrict the driver's freedom.

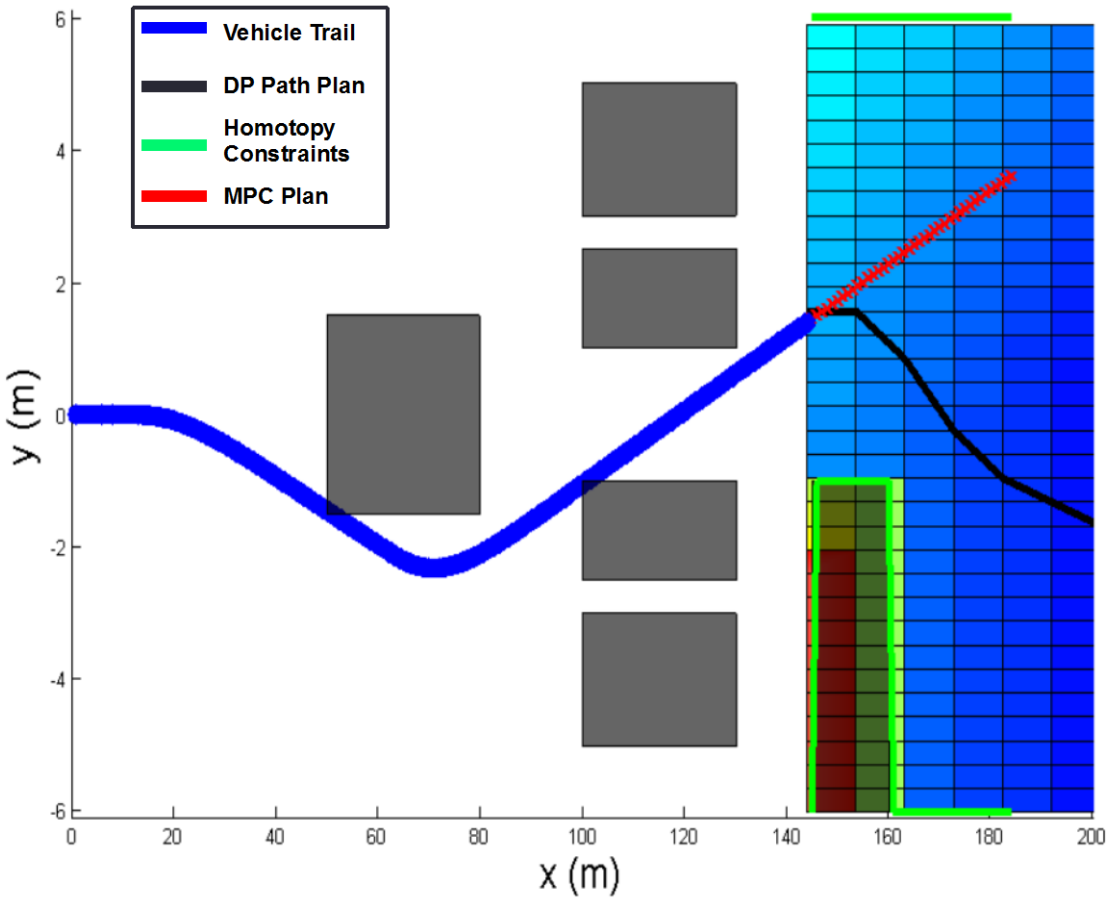
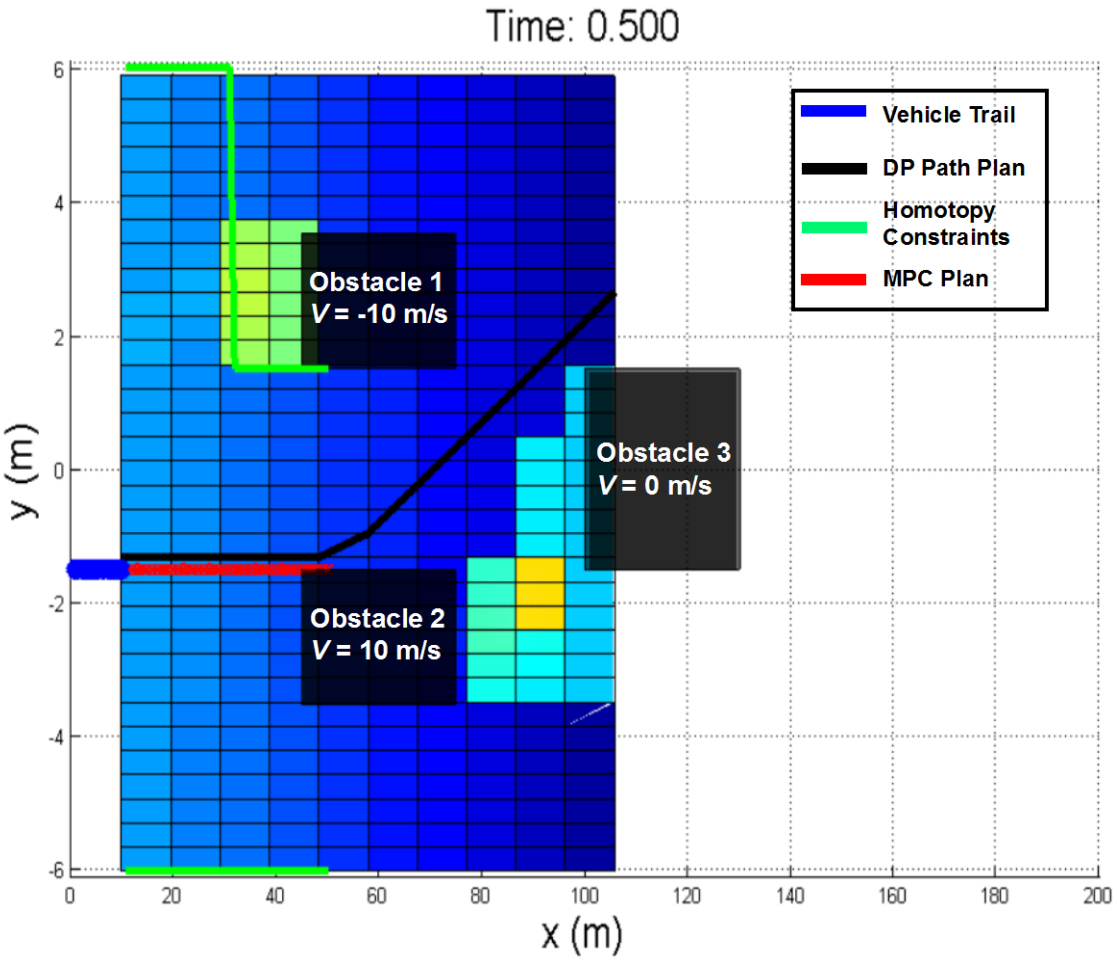
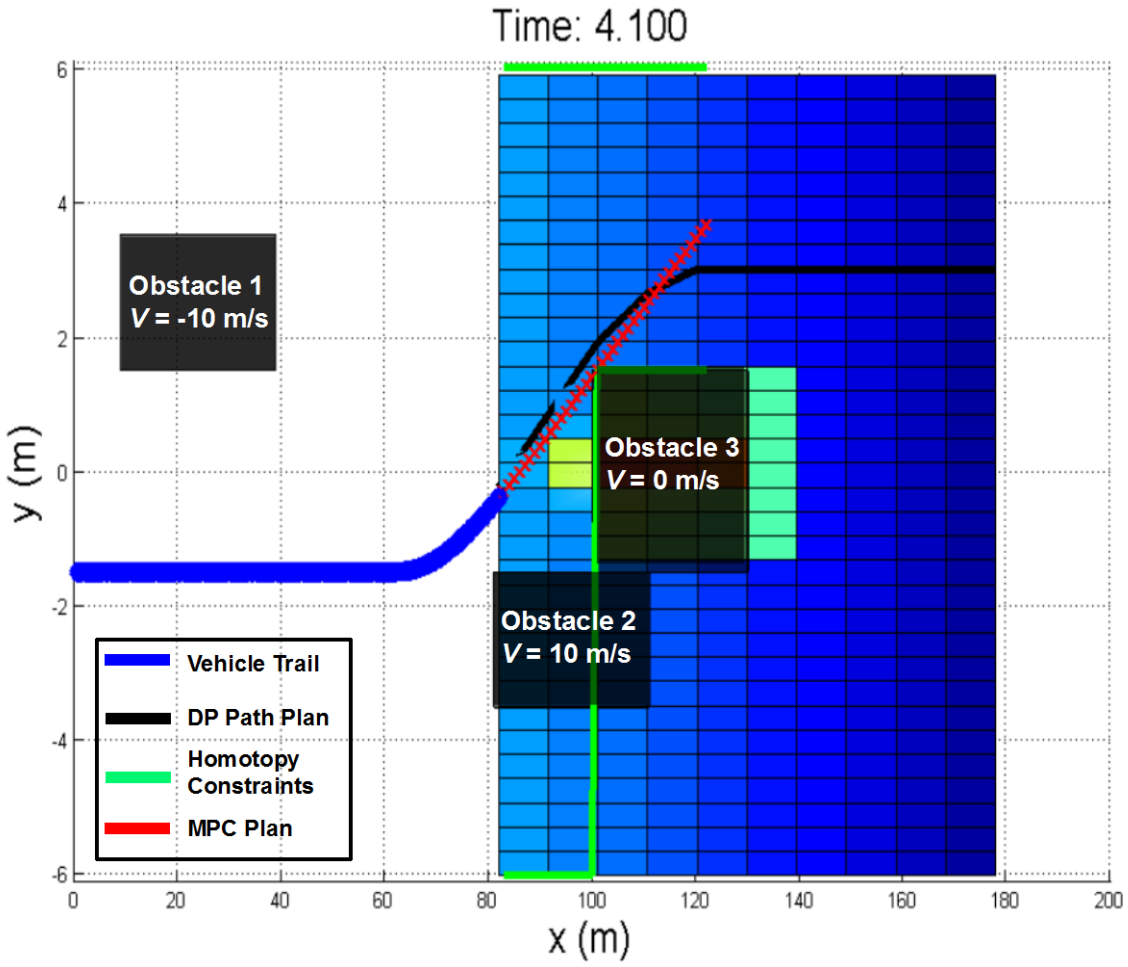


Figure 2.23. Snapshot of a simulation with penalties applied for deviation into an oncoming lane

Figure 2.24 shows two clips of a simulation with moving obstacles. In this simulation, obstacle motion was predicted using a first-order hold (equations (2.40), (2.41), and (2.44)) on obstacle velocities (which were constant in this case). Notice that at t = 0.5s, obstacles 1 and 2 share the same x-position and the obstacle field is symmetric. Were the corridor planner to neglect obstacle velocity, this setup would lead to a corridor plan that causes the vehicle to pass under the third (stationary) obstacle. Accounting for obstacle velocities leads to a shifting in the high-cost (predicted collision state) regions and leads to a corridor plan that more appropriately passes above the third obstacle once obstacle 1 has passed.


Time: 0.500
Vehicle Trail
DP Path Plan
Homotopy Constraints
MPC Plan
Obstacle 1
V = -10 m/s
Obstacle 3
V = 0 m/s
Obstacle 2
V = 10 m/s
y (m)
x (m)
6
4
2
0
-2
-4
-6
0
20
40
60
80
100
120
140
160
180
200

(a)



(b)

Figure 2.24. Snapshots of a simulation with moving obstacles

As mentioned above, when load transfer is neglected, the effect of gravity on sloped terrain both scales and shifts the reachable set. This causes the DP solution described by (2.36) and (2.47)-(2.49)) to penalize slopes that push the vehicle toward unsafe regions (either on or off the road surface). Figure 2.25 illustrates a simulation in which the corridor planner constrains the MPC solution to pass above an obstacle rather than travel on a road shoulder sloped at a relatively high $\theta = 50°$. This scenario highlights one additional effect of using distinct cost functions and vehicle models in the corridor planner and MPC controller. As mentioned above, in many semi-autonomous driving scenarios, the distinct corridor/path goals caused by this disparity can provide a desirable freedom to the human driver. This freedom, however, comes at the cost of the increased

lookahead that would be available to the MPC were the DP costmap used to extend the MPC preview via terminal penalties interpolated from the cost gradient of the DP map.

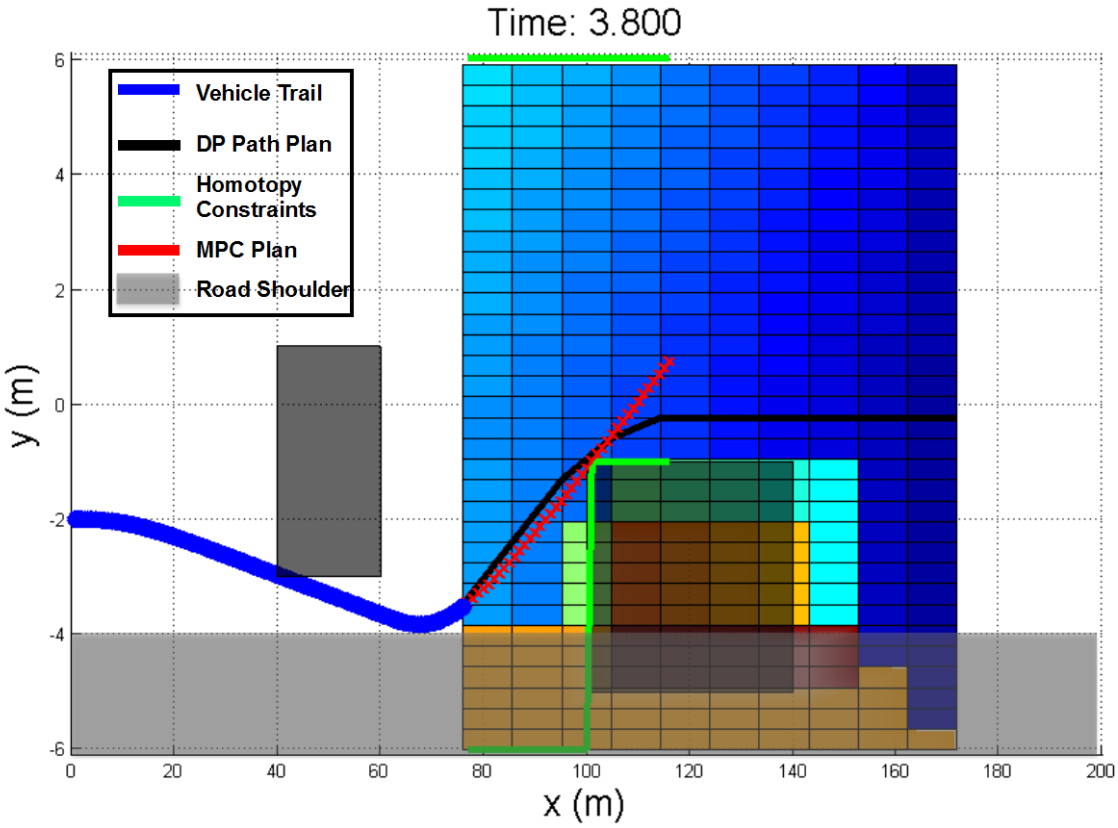


Figure 2.25. Simulation of a vehicle traveling near a (50°) sloped shoulder

2.4 CONCLUSIONS

This chapter introduced two general methods for identifying, evaluating, and planning constraints to bound a path homotopy. While other evaluation heuristics and cost functions might also be used to plan constraints, the methods described in this chapter result in a set of minimally restrictive position constraints that is particularly well suited for semi-autonomous enforcement. Chapter 3 describes the means by which these homotopy constraints are combined with vehicle dynamic and control restrictions and enforced via a shared adaptive control policy.

3

CHAPTER 3: THREAT-BASED CONSTRAINT ENFORCEMENT

3.1 INTRODUCTION

Once a desired homotopy has been identified, the position, state, and input constraints required to keep the vehicle within it must be converted into a semi-autonomously enforceable control policy. This chapter describes a model-based threat assessment method and a shared-adaptive control law that are particularly well suited to this task. This threat assessment method and control law use constrained model predictive control to predict the feasibility of satisfying position constraints, stability limits, and control actuator restrictions within the desired homotopy and calculate an optimal control policy that satisfies them. This control policy is then enforced according to the threat, here defined as the nearness of the optimal predicted trajectory to the limits of vehicle stability. When threat is low, the human retains significant autonomy. As threat increases, control authority is increasingly allocated to an MPC controller to ensure that safety constraints are satisfied.

The paragraphs that follow briefly describe the model and controller used to predict threat and the threat-based control policy employed to enforce constraints.

3.2 MODEL-BASED THREAT ASSESSMENT

This thesis uses Model Predictive (alt. “receding horizon optimal”) Control to predict the vehicle’s future state evolution and calculate an associated optimal control input. Model predictive control is a family of finite-horizon optimal control schemes that iteratively minimizes a performance objective defined for a forward-simulated plant model subject to state and input constraints [18], [113], [114]. As illustrated in Figure

3.1, MPC is particularly well suited for predicting a future state trajectory through the position constraints imposed by a particular environment and path homotopy. Within the convex constraints bounding this homotopy, the MPC calculation predicts a time-parameterized state and control trajectory that is optimal with respect to a set of performance criteria outlined in an objective function. Given an objective function that emphasizes vehicle stability, this prediction then captures an objective measure of the need for intervention and a stability-optimal control policy that satisfies constraints if allocated sufficient control authority.

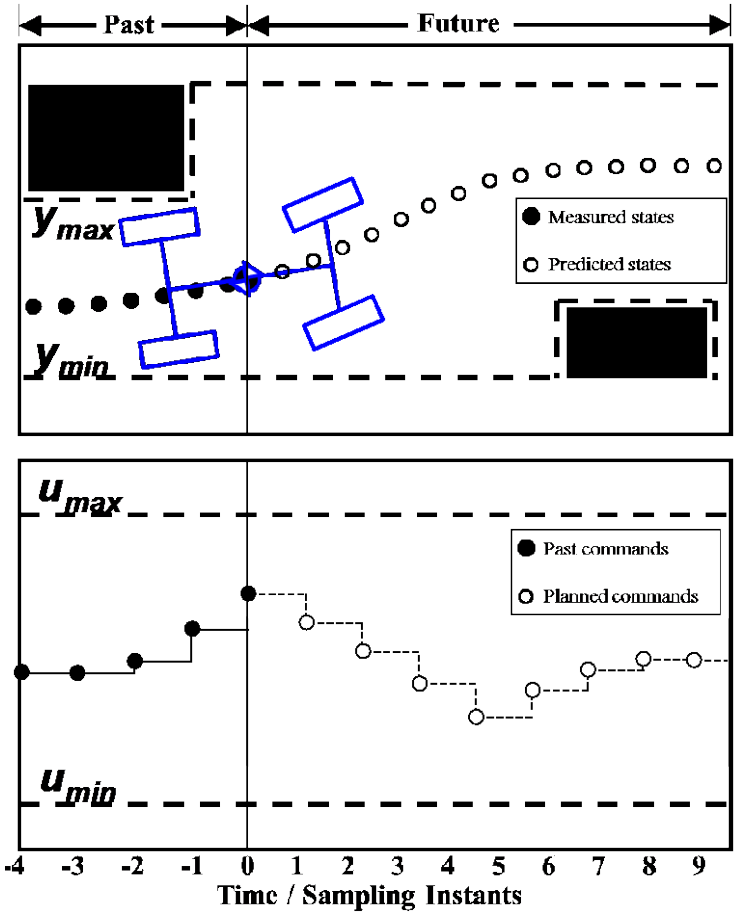


Figure 3.1. Model predictive control illustration

3.2.1 MODEL PREDICTIVE CONTROL

When it was originally developed in the process control industry in the late 1970's, MPC's intensive computational requirements restricted its application to

processes with low control update rates. Subsequent improvements in both computational hardware and algorithm efficiency have significantly expanded its range of opportunity to include diverse applications ranging from robot manipulators [115], [116] to vehicle navigation systems [47], [117]–[119], inventory management [120], and clinical anesthesia [121].

The MPC algorithm progresses as follows. At each time step, *t*, the current plant state is sampled and a cost-minimizing control sequence spanning from time *t* to the end of a control horizon of *n* sampling intervals, $t+n\Delta t$, is computed subject to inequality constraints. The first element in this input sequence is implemented at the current time and the process is repeated at subsequent time steps. The following development describes the specific MPC implementation employed in this work and is included for completeness.

For a four-wheeled, front-steer vehicle with discrete-time state equations given by

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}_u\mathbf{u}_k + \mathbf{B}_v\mathbf{v}_k \tag{3.1}$$

$$y_k = \mathbf{C}\mathbf{x}_k + \mathbf{D}_v\mathbf{v}_k \tag{3.2}$$

with **x**, **y**, **u**, and **v** representing states, outputs, inputs, and disturbances of the system respectively, a quadratic objective function over a prediction horizon of *p* sampling intervals is defined as

$$J_k = \sum_{i=k+1}^{k+p}\frac{1}{2}\left(\mathbf{y}_i - \mathbf{r}_i\right)^{\mathrm{T}}\mathbf{R}_y\left(\mathbf{y}_i - \mathbf{r}_i\right) + \sum_{i=k}^{k+p-1}\frac{1}{2}\mathbf{u}_i{}^{\mathrm{T}}\mathbf{R}_u\mathbf{u}_i + \sum_{i=k}^{k+p-1}\frac{1}{2}\Delta\mathbf{u}_i{}^{\mathrm{T}}\mathbf{R}_{\Delta u}\Delta\mathbf{u}_i + \frac{1}{2}\rho_\varepsilon\varepsilon^2 \tag{3.3}$$

where $\mathbf{R_y}$, $\mathbf{R_u}$, and $\mathbf{R}_{\Delta\mathbf{u}}$ represent diagonal weighting matrices penalizing deviations from $\mathbf{y}_i = \mathbf{r}_i$, and $\mathbf{u}_i = \mathbf{0}$, ρ_ε represents the penalty on constraint violations and ε represents the maximum constraint violation over the prediction horizon *p*. In this work, the elements of $\mathbf{R_y}$ corresponding to vehicle position states are zeroed to reflect the absence of a vehicle path reference. Inequality constraints on the states and inputs of the vehicle are defined as:

$$
\begin{aligned}
&\mathbf{y}^{j}{}_{\min}(i) - \varepsilon \mathbf{V}^{j}{}_{\min}(i) \le \mathbf{y}^{j}(k+i+1|k) \le \mathbf{y}^{j}{}_{\max}(i) + \varepsilon \mathbf{V}^{j}{}_{\max}(i) \\
&\mathbf{u}^{j}{}_{\min}(i) \le \mathbf{u}^{j}(k+i+1|k) \le \mathbf{u}^{j}{}_{\max}(i) \\
&\Delta\mathbf{u}^{j}{}_{\min}(i) \le \Delta\mathbf{u}^{j}(k+i+1|k) \le \Delta\mathbf{u}^{j}{}_{\max}(i) \\
&i = 0,\ldots, p-1 \\
&\varepsilon \ge 0
\end{aligned}
\tag{3.4}
$$

where the vector $\Delta\mathbf{u}$ represents the change in input from one sampling instant to the next, the superscript "$(\bullet)^{j}$" represents the j^{th} component of a vector, k represents the current time, and the notation $(\bullet)^{j}(k+i|k)$ denotes the value predicted for time $k+i$ based on the information available at time k. The vector $\mathbf{V}$ is used to variably soften constraints over the prediction horizon, p, by including ε in the objective function.

Position constraints bounding the desired homotopy are sampled over the predicted vehicle trajectory as upper and lower limits on the vehicle lateral position as

$$
\mathbf{y}^{y}{}_{\max}(k) = \begin{bmatrix} y^{y}{}_{\max}(k+1) \\ \vdots \\ y^{y}{}_{\max}(k+p) \end{bmatrix}, \quad \mathbf{y}^{y}{}_{\min}(k) = \begin{bmatrix} y^{y}{}_{\min}(k+1) \\ \vdots \\ y^{y}{}_{\min}(k+p) \end{bmatrix} \tag{3.5}
$$

With n representing the number of free control moves, ΔU_k and U_k are calculated by choosing a blocking vector $\mathbf{J_m}$ such that

$$
\Delta\mathbf{U}_k = \begin{bmatrix} \Delta u(k) \\ \vdots \\ \Delta u(k+p-1) \end{bmatrix} = \mathbf{J}_m \begin{bmatrix} z_0 \\ \vdots \\ z_{n-1} \end{bmatrix} \tag{3.6}
$$

and

$$
\mathbf{U}_k = \begin{bmatrix} \mathbf{I} & 0 & \cdots & 0 \\ \mathbf{I} & \mathbf{I} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{I} & \mathbf{I} & \mathbf{I} & \mathbf{I} \end{bmatrix} \Delta\mathbf{U}_k + \begin{bmatrix} \mathbf{I} \\ \mathbf{I} \\ \vdots \\ \mathbf{I} \end{bmatrix} \mathbf{u}_{k-1} \tag{3.7}
$$

where $\mathbf{z} = [z_0, \ldots, z_{n-1}]^T$ represents the free optimization variables of the optimization problem. By augmenting the vectors **y**, **U**, $\Delta\mathbf{U}$, and **V** over the prediction horizon as

$$\mathbf{Y}_{k+1} = \begin{bmatrix} y_{k+1} \\ y_{k+2} \\ \vdots \\ y_{k+p} \end{bmatrix}, \mathbf{U}_k = \begin{bmatrix} u_k \\ u_{k+1} \\ \vdots \\ u_{k+p-1} \end{bmatrix}, \Delta\mathbf{U}_k = \begin{bmatrix} \Delta u_k \\ \Delta u_{k+1} \\ \vdots \\ \Delta u_{k+p-1} \end{bmatrix}, \mathbf{V}_k = \begin{bmatrix} v_k \\ v_{k+1} \\ \vdots \\ v_{k+p-1} \end{bmatrix} \tag{3.8}$$

and calculating weighting matrices $\mathbf{S_u}$, $\mathbf{S_{u1}}$, $\mathbf{S_x}$, and $\mathbf{H_x}$, the augmented plant outputs over *p* are given by

$$\begin{bmatrix} y_{k+1} \\ \vdots \\ y_{k+p} \end{bmatrix} = \mathbf{S}_x\mathbf{x}_k + \mathbf{S}_{u1}\mathbf{u}_{k-1} + \mathbf{S}_u \begin{bmatrix} \Delta\mathbf{u}_k \\ \vdots \\ \Delta\mathbf{u}_{k+p-1} \end{bmatrix} + \mathbf{H}_v \begin{bmatrix} \mathbf{v}_k \\ \vdots \\ \mathbf{v}_{k+p-1} \end{bmatrix} \tag{3.9}$$

Finally, representing the optimal control problem as a quadratic program in its augmented outputs, inputs, and disturbances by calculating $\mathbf{K_x}$, $\mathbf{K_u}$, $\mathbf{K_{ut}}$, $\mathbf{K_v}$, and $\mathbf{K_{\Delta u}}$ gives

$$J = \left(x_k{}^{\mathrm{T}}\mathbf{K}_x + u_{k-1}{}^{\mathrm{T}}\mathbf{K}_u + \mathbf{U}_{k-1}{}^{\mathrm{T}}\mathbf{K}_{ut} + \mathbf{V}_k{}^{\mathrm{T}}\mathbf{K}_v\right)\Delta\mathbf{U}_k + \frac{1}{2}\Delta\mathbf{U}_k{}^{\mathrm{T}}\mathbf{H}\Delta\mathbf{U}_k + \rho_\varepsilon\varepsilon^2, \tag{3.10}$$

with constraints represented by

$$\mathbf{A}_c\Delta\mathbf{U}_k \le \mathbf{b}_c\,. \tag{3.11}$$

Expressed as a quadratic program, the control problem is then solved using conventional optimization routines [18], [113].

3.2.2 VEHICLE MODEL

The MPC prediction model used in this thesis considers the kinematics of a four-wheeled vehicle, along with its lateral and yaw dynamics. Vehicle suspension dynamics and roll states were excluded from present consideration to simplify the computational

problem given the low center of gravity and high suspension stiffness of the experimental platforms.

Figure 3.2 illustrates the vehicle model and its associated parameters. States considered in this model include the position of the vehicle's center of gravity [x, y], its yaw angle ψ, yaw rate $\dot{\psi}$, and sideslip angle β, and velocity V (with direction defined by $\psi+\beta$). Steering input at the front wheels is denoted by δ.

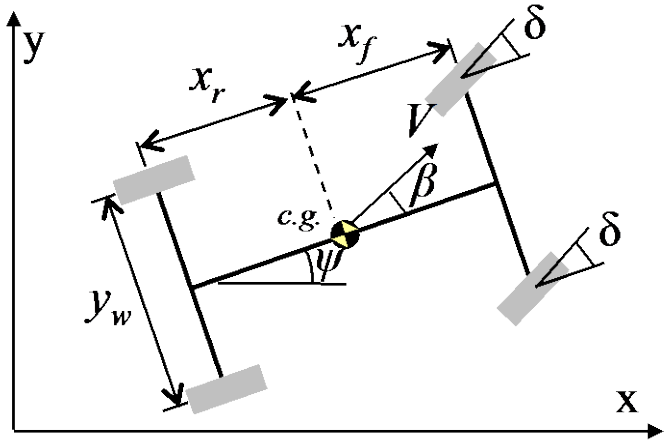


Figure 3.2. Illustration of vehicle model used in MPC controller

Equations of motion describing this model's state evolution include

$$\dot{x} = V\cos(\psi+\beta) \tag{3.12}$$

$$\dot{y} = V\sin(\psi+\beta) \tag{3.13}$$

$$\dot{V} = \frac{V(\dot{\psi}+\dot{\beta})\sin(\psi+\beta) - C_f(\beta-\delta)\cos(\psi+\delta) - C_r\beta\cos\psi}{\cos(\psi+\beta)} \tag{3.14}$$

$$\dot{\beta} = \frac{-\frac{1}{m}\left[C_f\alpha_f\cos(\psi+\delta) + C_r\alpha_r\cos(\psi)\right] + \dot{V}\sin(\psi+\beta)}{V\cos(\psi+\beta)} - \dot{\psi} \tag{3.15}$$

$$\ddot{\psi} = \frac{1}{I_{zz}}\left[-C_f\alpha_f x_f\cos(\delta) + C_r\alpha_r x_r\right] \tag{3.16}$$

where the sideslip angles at the front (α_f) and rear (α_r) are given by

$$\alpha_f = \tan^{-1}\left(\frac{V\sin(\psi+\beta)+x_f\dot{\psi}\cos(\psi)}{V\cos(\psi+\beta)-x_f\dot{\psi}\sin(\psi)}\right)-\delta \tag{3.17}$$

$$\alpha_r = \tan^{-1}\left(\frac{V\sin(\psi+\beta)-x_r\dot{\psi}\cos(\psi)}{V\cos(\psi+\beta)+x_r\dot{\psi}\sin(\psi)}\right) \tag{3.18}$$

and the lateral force at the tires is approximated from the tire stiffness characteristics at the front (*f*) and rear (*r*) tires $F_{yf} = \alpha_f C_f(\alpha_f)$ and $F_{yr} = \alpha_r C_r(\alpha_r)$ as described in [112] and illustrated in Figure 3.3.

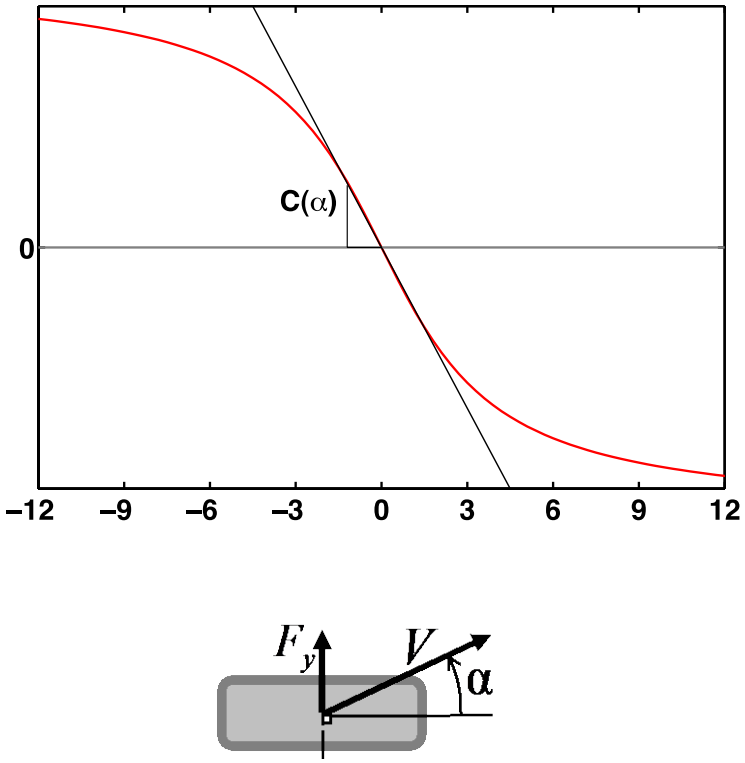


Figure 3.3. Tire cornering stiffness definition and approximation

Linearized about a constant speed and small slip angles, equations of motion for this model become

$$\dot{x} = V \tag{3.19}$$

$$\dot{y} = V(\psi + \beta) \tag{3.20}$$

$$\dot{\beta} = \frac{-(C_r + C_f)}{mV}\beta + \left(\frac{(C_r x_r - C_f x_f)}{mV^2} - 1\right)\dot{\psi} + \frac{C_f}{mV}\delta \tag{3.21}$$

$$\ddot{\psi} = \frac{(C_r x_r - C_f x_f)}{I_{zz}}\beta - \frac{(C_r x_r{}^2 + C_f x_f{}^2)}{I_{zz} V}\dot{\psi} + \frac{C_f x_f}{I_{zz}}\delta \tag{3.22}$$

with the tire stiffness characteristic linearized to $F_{yf} \approx C_f \bullet \alpha_f$ and $F_{yr} \approx C_r \bullet \alpha_r$.

Table A.1 (of Appendix A) defines and quantifies this model's parameters for the model used in these simulations.

3.2.3 THREAT ASSESSMENT

The controller's projected path along a predefined trajectory or through a constraint-imposed tube is shaped by the performance objectives established in the MPC objective function. We configure the MPC objective function to satisfy homotopy-required position constraints and actuator-imposed input constraints while quadratically penalizing stability-critical states. As demonstrated in [122], several vehicle states are coupled with its stability and can be used to gauge the vehicle's nearness to lateral, yaw, or roll instability. In this thesis, we use front wheel sideslip ($\alpha = (x_f / V)\dot{\psi} + \beta - \delta$) as both the trajectory characteristic to minimize in the MPC objective function and the state prediction to consider in the threat calculation. This choice is motivated by a number of observations. Minimizing front wheel slip in the MPC objective function tends to lead to trajectories that also minimize the lateral acceleration and roll angle required by those trajectories. Additionally, front wheel sideslip strongly influences the controllability of front-wheel-steered vehicles, since cornering friction begins to decrease above critical slip angles (see Figure 3.3). These critical angles are well known and provide a direct mapping from environmental conditions such as weather or terrain properties, to vehicle

handling limitations; when predicted threat approaches this known limit, loss of stability is imminent. The linearized tire compliance model's failure to account for this decrease further motivates the suppression of front wheel slip angles to reduce controller-plant model mismatch. Finally, trajectories that minimize wheel slip also tend to minimize lateral acceleration and yaw rates, leading to a safer and more comfortable ride.

The hierarchy of objectives created by this position-constrained cost function causes the controller to predict (and calculate the control commands required to track) the trajectory of maximum stability existing within the safe homotopy. As such, the nearness of this prediction's stability-critical states to their physical limits provides a useful indication of "threat" and an objective assessment of the need for intervention to arrest collisions or instability before they happen.

Ref. [122] presents an analysis of various norms for reducing the time-sampled vector of predicted vehicle states $\vec{x}$ to a scalar threat metric $\Phi(t_k)$, the instantaneous threat assessment at time *k*). In the simulations and experiments presented in this thesis, we set $\Phi(t_k)$ to the maximum value of front wheel sideslip occurring within the MPC prediction horizon:

$$\Phi(t_k) = \max|\alpha_i| \quad : \ 1 \le i \le n. \tag{3.23}$$

This conservative estimate of threat is then used to determine the level of control authority required to prevent the most dangerous predicted vehicle state from leaving the stable performance envelope, as described below.

3.3 THREAT-BASED SHARED AUTONOMY

In the absence of plant-model mismatch and unanticipated constraint motion (e.g. due to the sudden appearance of unanticipated obstacles in near proximity to the vehicle), the MPC controller whose predicted control and state evolution do not exceed actuator or stability limits is capable of performing the avoidance maneuver that constraint satisfaction requires and the MPC controller (and therefore the threat assessment) predicts. Given this observation, we design a control strategy that allocates control authority between human and automation in accordance with predicted threat: when

threat is low and the predicted state trajectory relatively safe, the driver retains full control. As threat increases and the control inputs required to safely satisfy homotopy constraints become increasingly more challenging, control authority is shifted to the controller. In the limiting case – when the threat assessment reaches a critical value above which stable constraint satisfaction by the human operator is unlikely – the MPC controller is allocated full control authority and navigates the vehicle autonomously until threat has been reduced to an acceptable level. Figure 3.4 illustrates extremes in control allocation given different MPC state predictions and their corresponding threat assessments.

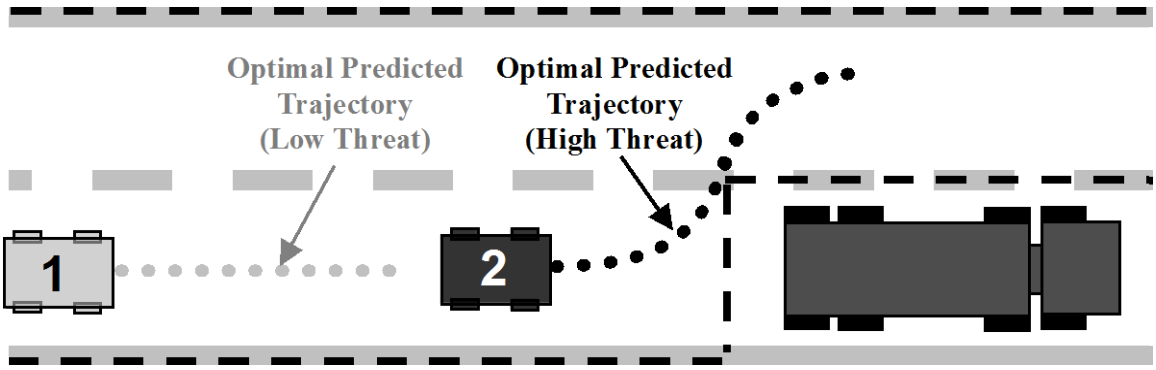


Figure 3.4. Obstacle avoidance scenario illustrating low- (1) and high- (2) threat predictions required to remain within the safe homotopy (outlined by dashed lines)

Denoting the current driver input by u_{dr} and the current controller input by u_{MPC}, the input seen by the vehicle, u_v, as a function of the intervention metric $K(\Phi) \in \left[\begin{array}{cc} 0 & 1 \end{array}\right]$ is given by

$$u_v = K(\Phi)u_{MPC} + (1 - K(\Phi))u_{dr} \quad \textbf{(3.24)}$$

Various control authority allocation schemes, $K(\Phi)$ satisfy the objectives outlined above without overburdening (with too much intervention) or startling (with intervention too abrupt) the human driver [123], [124]. This thesis uses intervention laws of the general form

$$K(\Phi)=\begin{cases} 0 & \Phi \le \left((1-s)\Phi^{A}_{eng}+s\Phi^{B}_{eng}\right) \\ \dfrac{\Phi-\Phi^{A}_{eng}}{\Phi^{A}_{aut}-\Phi^{A}_{eng}} & \Phi^{A}_{eng} \le \Phi \le \Phi^{A}_{aut},\ \ s=0 \\ \dfrac{\Phi-\Phi^{B}_{eng}}{\Phi^{B}_{aut}-\Phi^{B}_{eng}} & \Phi^{B}_{eng} \le \Phi \le \Phi^{B}_{aut},\ \ s=1 \\ 1 & \Phi \ge \left((1-s)\Phi^{A}_{aut}+s\Phi^{B}_{aut}\right) \end{cases} \quad \textbf{(3.25)}$$

where the switch $s(t_k)$ is used to allow for hysteric control authority allocation as

$$s(t_k)=\begin{cases} 0 & K(t_{k-1})=0 \\ 1 & K(t_{k-1})=1 \end{cases}. \quad \textbf{(3.26)}$$

Note that in (3.25),

$$\Phi^{\bullet}_{aut} \ge \Phi^{\bullet}_{eng}, \quad \textbf{(3.27)}$$

and generally,

$$\Phi^{A}_{\bullet} \ge \Phi^{B}_{\bullet}. \quad \textbf{(3.28)}$$

Note that (3.28) is typically enforced to ensure that the vehicle retains sufficient control authority following full intervention (K=1) to reduce the predicted threat to a safer level before transitioning control back to the human driver.

As illustrated in Figure 3.5, the intervention function (3.25) is parameterized by the threat level at which the MPC controller begins to intervene ($\Phi^{\bullet}_{eng}$), and the level at which it is given full control authority and effectively acts as an autonomous controller ($\Phi^{\bullet}_{aut}$). When predicted threat Φ is less than the low-threat threshold $\Phi^{\bullet}_{eng}$, K is set to zero, effectively passing all of the driver's control input (and none of the controller's) to the vehicle. Above the high-threat threshold Φ^{A}_{aut}, K is set to unity. This allows the MPC controller full control authority to autonomously track the high-threat path. Once the predicted threat has been reduced to a safer level (below Φ^{B}_{aut}), the driver is again

given more control authority, eventually controlling the vehicle autonomously once threat has dropped below Φ^{B}_{eng}. Note that, as described in [125], the parameters of this intervention law allow considerable adaptation of the controller's intervention characteristic to the performance and/or preference of the human driver, and to the type or urgency of the navigation task.

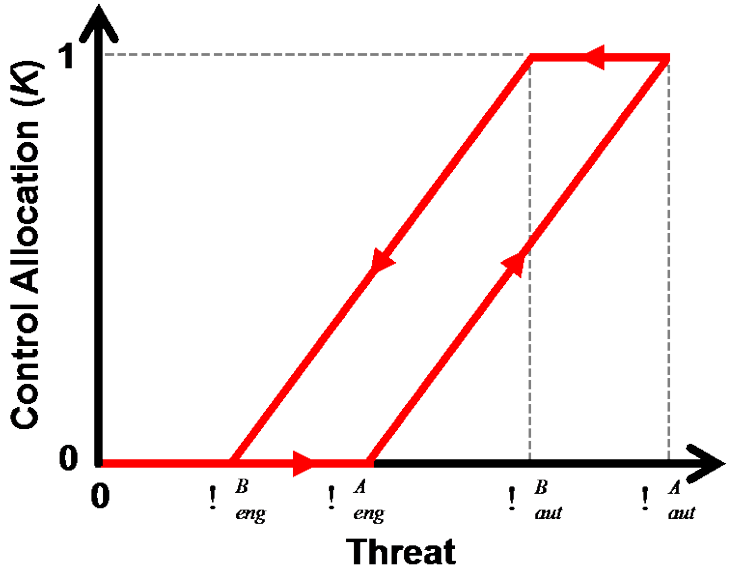


Figure 3.5. General form of control allocation schemes

3.4 DRIVER FEEDBACK

As mentioned in Chapter 1, remotely teleoperating an unmanned vehicle reduces a human operator's situational awareness by limiting his or her perception of visible, inertial, and tactical cues. If not supplemented with appropriate visual and/or haptic feedback, we hypothesize that shared control can aggravate these effects by degrading the driver's mental model when the vehicle response deviates from the driver's expectations. In [126], the authors show a positive correlation between human performance at a remote "pick-and-place" task and his or her telepresence, or sense of being present at the remote site. In this work, we present two modes of feedback intended to similarly improve

telepresence in the remote vehicle operator, inform his or her high-level planning decisions, and communicate the controller's planning intentions and control actions.

3.4.1 VISUAL

The first driver feedback modality communicates the controller's intentions to the human operator via a visual overlay of its desired homotopy on the driver's screen. Indicators at the bottom of the screen show the driver where the vehicle is currently steering (red line) and where the driver's steering command lies with respect to the vehicle's current input. In high threat scenarios where safe control inputs deviate from the driver's current command, these indicators diverge. Figure 5 illustrates these overlays in a typical scenario.

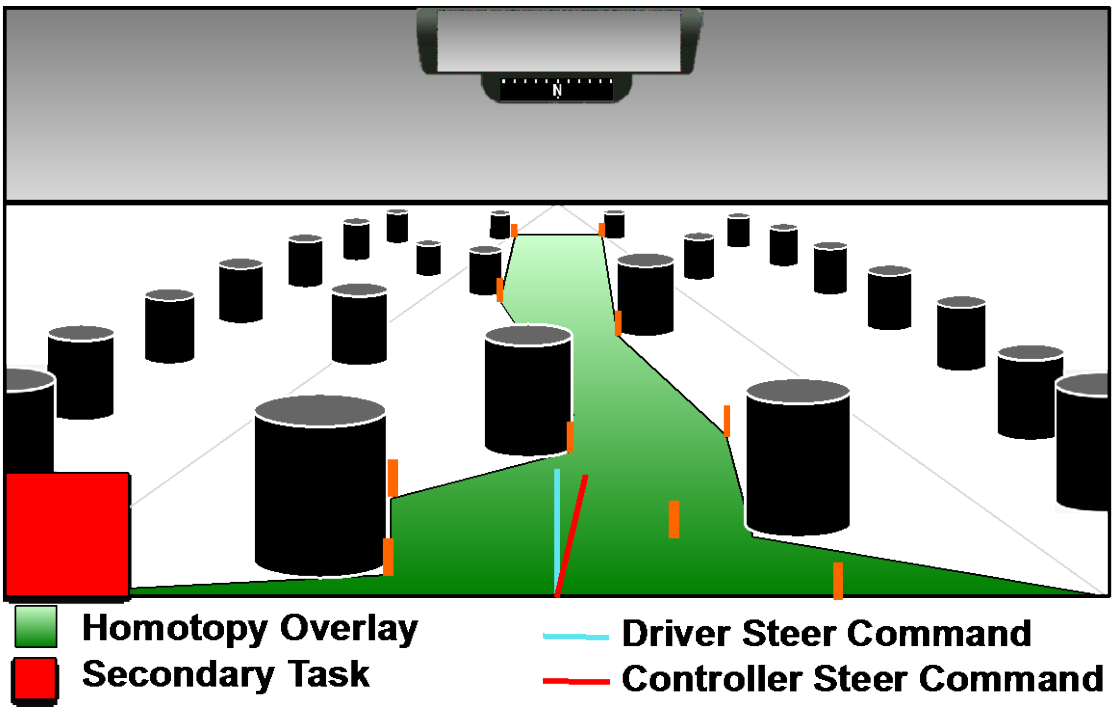


Figure 3.6. Illustration of the operator control interface showing the homotopy overlay (green) and steering indicators (red and cyan lines at bottom center)

3.4.2 HAPTIC

The second driver feedback modality applies a torque overlay to the steering wheel to indicate to the driver where the control constraints lie and how urgently they

must be satisfied to avoid a collision. This overlay is provided not as a means of *pushing* the operator in the right direction, per se, but of *communicating* to him/her what the vehicle is currently doing and thereby preserve his/her mental model and telepresence when control authority shifts. This sharply differs from the haptic feedback approaches used in traditional driver assistance systems. In traditional usage, torque feedback is often provided to *motivate* corrective action on the part of the driver [23], [24], [127]. In the approach presented here, feedback is provided not because safe vehicle operation *depends on it* (the controller will avoid collisions and losses of stability regardless of what the driver does), but because the driver's situational awareness (and telepresence in teleoperation applications) may improve *because of it*. Stated another way, the torque feedback described here is designed to communicate predicted threat to the driver (with the magnitude of that torque proportional to the predicted threat), and inform him/her of action already being taken by the vehicle to reduce it (via this torque's direction and magnitude).

The resistance torque applied to the operator's steering wheel is calculated as

$$\mathrm{T} = k_{\max} K \left| \delta_{driver} - \delta_{MPC} \right| \tag{3.29}$$

where $k_{\max}$ represents the maximum available steering wheel torque, and is used to re-dimensionalize the $K \in \left[\begin{array}{cc} 0 & 1 \end{array} \right]$ given by (3.25).

Figure 6 illustrates the (hypothetical) response of the torque restoring function to increasingly threatening MPC predictions (assuming the driver fails to steer). Notice that as time progresses (denoted by t_i labels on the host vehicle), the threat posed by the optimal escape maneuver increases. Additionally, the immediate steering command required to track this optimal trajectory (u^*) begins to drift leftward. The combined effect of an increasingly-urgent, and progressively-leftward u^* recommendation increases K and shifts the torque resistance "trough". In the limiting case, for which only the optimal steering command can reasonably be expected to avoid both the hazard and loss of control (sometime shortly after t_4), the controller exerts the maximum available torque on the operator's steering wheel, ensuring that the steering wheel tracks that of the vehicle

(preserving the operator's mental model) and communicating predicted threat and related controller actions to the driver.

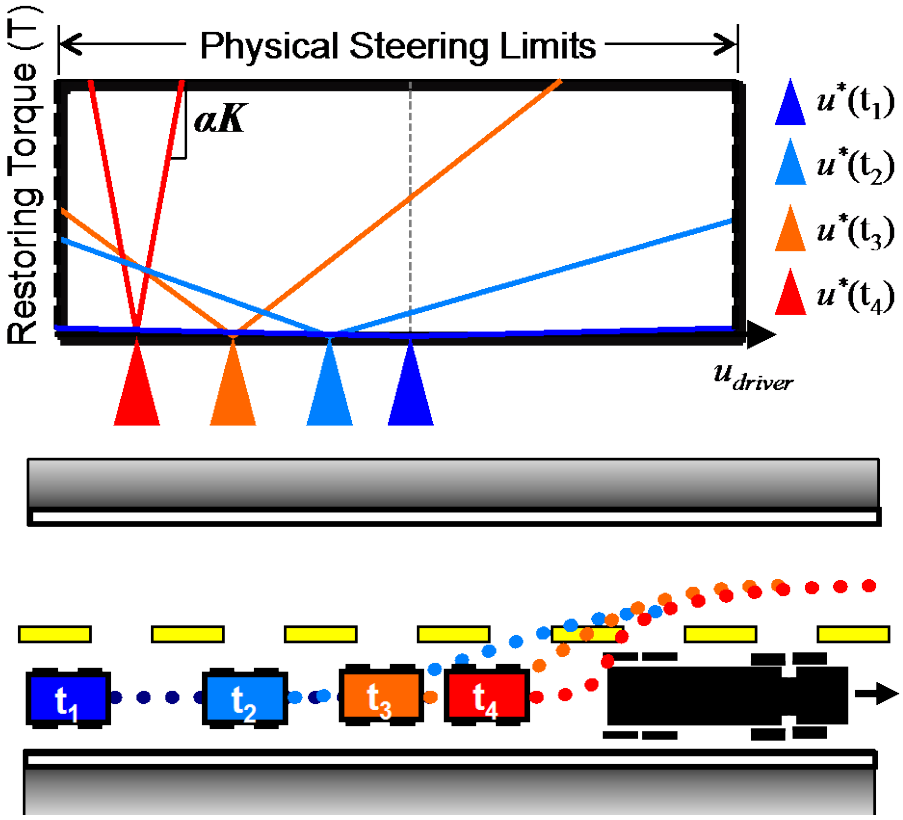


Figure 3.7. Scenario illustration showing the response of the restoring torque function as a vehicle successively approaches a hazard from behind

3.5 SIMULATION STUDY

This section illustrates 1) the effect of threat-based constraint enforcement on the performance of a simulated teleoperator operating under a 100-200 ms time delay, and 2) the effect of homotopy-based constraint planning and threat-based constraint enforcement on the simulated traversal of a static obstacle field by a semi-autonomous ground vehicle.

3.5.1 SETUP

These tests simulated the vehicle dynamics as those of a generic light truck with a double wishbone suspension, passive roll stabilizers, and rack and pinion steering. This vehicle was modeled in MSC Adams and approximated tire forces via a Pacejka tire

model [112]. Appendix A describes this model's parameters. The MPC controller ran at 20 Hz, with its prediction and control horizons calculated over 60 and 40 timesteps, respectively. Parameters in the MPC model (equations (3.19)-(3.22)) were linearized about the 20 m/s simulation velocity and configured to closely match those of the Adams plant.

In delay simulations, a pure pursuit driver model was used to simulate the inputs of an operator seeking to track a predefined route through the center of a safe corridor. This model implements proportional feedback on the path tracking error. Its main tuning parameter is the lookahead distance, L. Driver steering inputs δ are calculated as

$$\delta = \left(y_{des}(t) - y(t)\right)\frac{2\left(x_f + x_r\right)}{L^2}\sin(\Theta), \tag{3.30}$$

where Θ is illustrated in Figure 3.8.

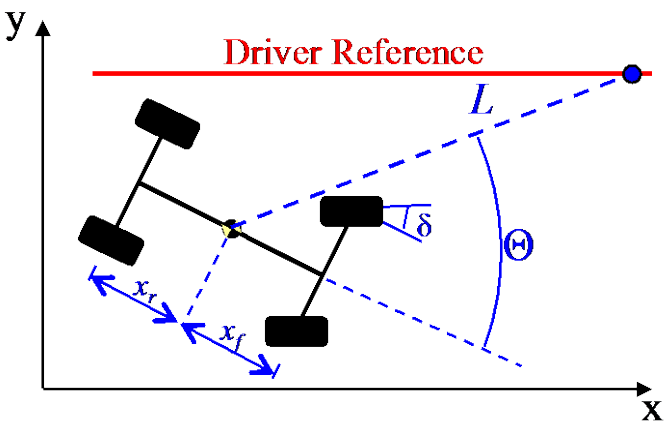


Figure 3.8. Illustration of pure pursuit driver model

Two different driver inputs were tested. In the first, the driver failed to steer around a hazard. Such a scenario can occur when a drowsy, inattentive, or otherwise unresponsive driver fails to notice an impending hazard or when communications and controls are impaired. [128]. The second type of driver input was derived from a well-tuned (L=30), pure-pursuit controller (3.30). In the first set of simulations, time delays of 100 – 200 ms were introduced to simulate the effect of feedback and control delays on a

remote human operator. Note that because the semi-autonomous controller runs on the vehicle, its state feedback and control inputs are not directly affected by wireless communication delays.

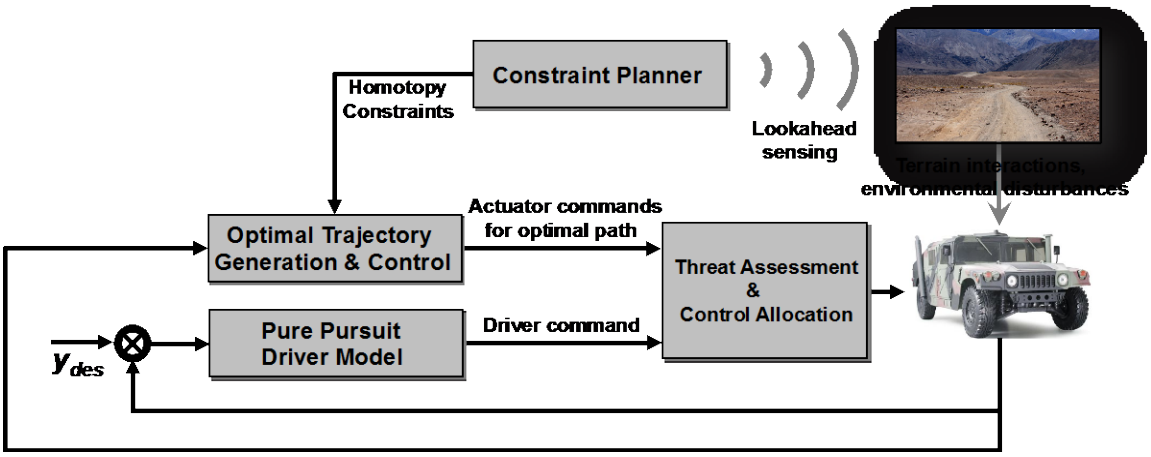


Figure 3.9. Block diagram of shared control system with a pure-pursuit driver model tracking a predefined reference, y_{des}

3.5.2 RESULTS

I) HOMOTOPY-BASED CONSTRAINT PLANNING

Figure 3.11 shows the path homotopy and associated position constraints designed by the homotopy planner (green channel) as well as the degree of control (K) allocated to the MPC controller.

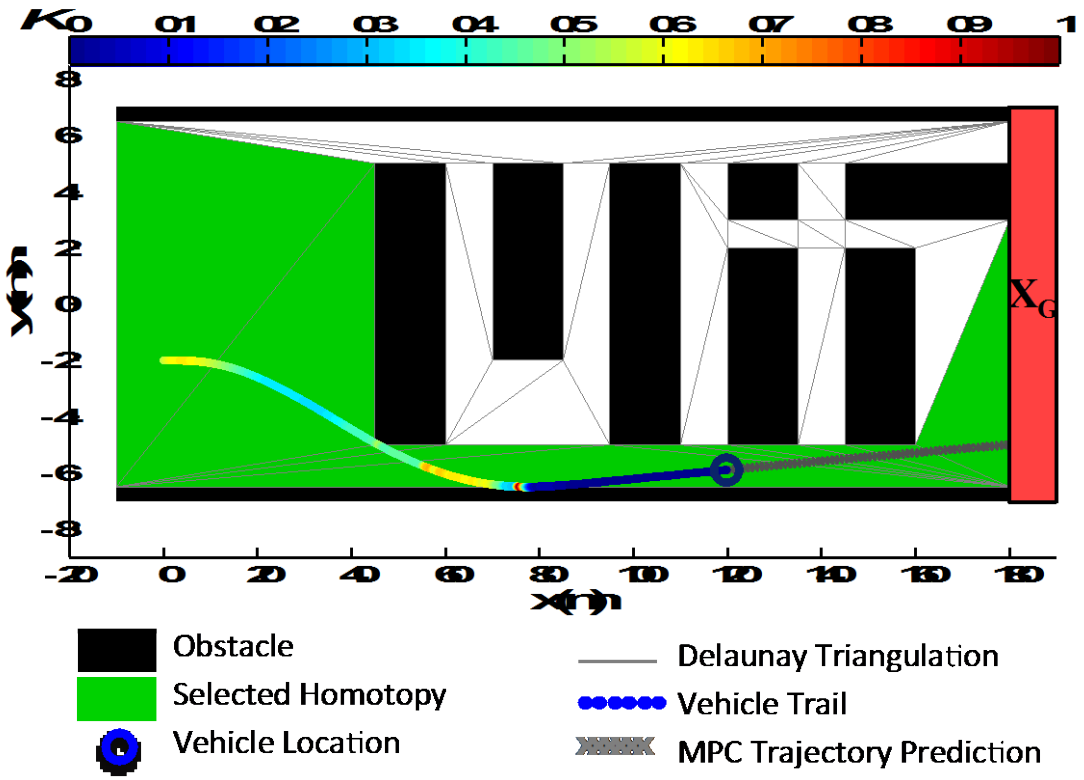


Figure 3.10. Simulation results demonstrating constraint-based shared autonomy through an obstacle field

Note that the "ricochet" off the lower bound of the workspace occurs because the simulated input remains at zero for the entire maneuver. In practice, the significant control freedom offered by the relaxed constraints between x=40 and 80 meters allows the human operator to straighten out or steer clear of the walls if desired. Also note that given the vehicle's initial position at (0,-2) [m], the optimal (shortest and most reachable) path homotopy passes under the obstacles.

When an additional obstacle is placed under the second obstacle, the homotopy plan changes. Because the homotopy passing below the obstacles is more tortuous and offers less control freedom to the human operator, the objective function given by (2.15) instead chooses the wider and less dynamically-challenging homotopy passing above the obstacles. In this case, the controller must initially intervene strongly in order to avoid the impending hazard, but quickly relaxes intervention as the vehicle enters a less restricted

region of the homotopy above the obstacles, where the driver's (δ=0) steering command incurs less risk.

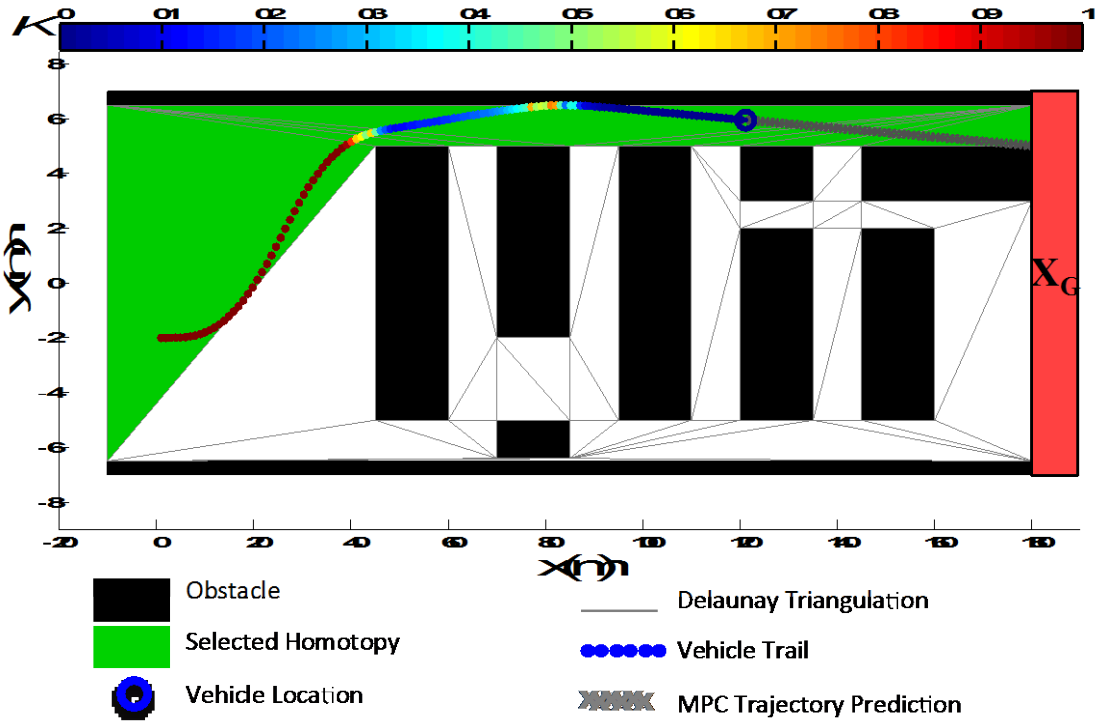


Figure 3.11. Simulation results demonstrating constraint-based shared autonomy through an obstacle field

II) THREAT-BASED CONSTRAINT ENFORCEMENT WITH COMMUNICATION DELAYS

Figure 3.12 compares the performance of 1) the driver model without time delays, 2) the same driver model in the presence of a 200 ms communication (sensing and control) delay, and 3) the semi-autonomously-assisted driver model in the presence of a 200 ms delay. Note that introducing this delay into the control and feedback loops of an otherwise-well-tuned (ie. stable in the absence of time delays) driver model (L = 14m) renders the unassisted driver unable to maintain control of the vehicle while negotiating a curve. This instability observed in the presence of time delays as short as a few hundred milliseconds is consistent with experimental observations [128].

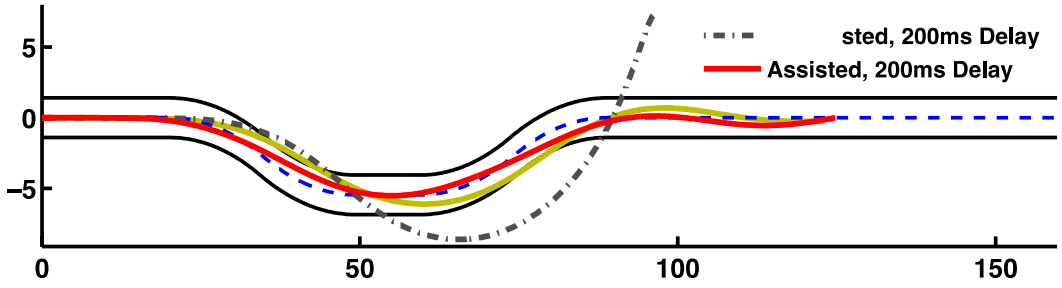


Figure 3.12. Simulation results comparing the performance of an unassisted driver (simulated by a pure-pursuit controller) with the performance of that same driver in the presence of communication latency and with the assistance of a constraint-based controller

With the semi-autonomous controller in the loop (operating on the vehicle itself), the vehicle successfully negotiates the curve and prompts more moderate steering commands from the human driver. This in turn, allows the human to retain significant vehicle control and better track the driver-desired trajectory. Note that even though its assistance helps the driver model more closely track the reference, the constraint-based

controller in this simulation sought only to satisfy position and stability constraints and was not aware of this trajectory. These results are consistent with experimental observations from real human drivers discussed in detail in Chapter 4.

That the shared controller exhibits similar performance improvements in the presence of time-delays as it does with inexperienced drivers or poorly-tuned driver models [124] is unsurprising considering the driver-agnostic nature of its threat-based intervention characteristic. Because the constraint-based control strategy and threat-based intervention method seek only to keep the vehicle within a constraint-bounded envelope of operation, this framework's control actions treat any human error, regardless of its source, the same way; if driver actions put the vehicle at imminent risk of leaving the safe homotopy, it intervenes to prevent constraint violation and loss of stability.

3.5.3 CONCLUSIONS

This chapter has described a threat-based method for semi-autonomously enforcing the position, input, and state constraints designed and imposed by the constraint planners of Chapter 2. In simulated traversals of an obstacle field and a curved roadway, and in the presence of time delay, this control method has been shown to effectively avoid accidents and losses of stability while providing the human with significant control when his or her control inputs are safe. Note that this capability assumes knowledge of vehicle parameters, an accurate estimate of road friction coefficients, and perfect knowledge of the state, position, and size of obstacles in the environment. In the presence of uncertain sensing data, unknown surface friction coefficients, or model parameter mismatch, safe semi-autonomous operation requires the application of safety margins bounding the uncertain location of obstacles, conservatively estimating road friction coefficients, and adaptively modifying the parameters of the MPC model consistent with true vehicle parameters. The experimental testing performed in Chapter 4 implements many of these techniques in order to safely avoid obstacles in spite of imperfect sensing.

This chapter has also introduced two new methods for communicating the controller's chosen homotopy, predicted threat, and intervention actions to the human over visual and haptic channels. These feedback mechanisms, while not employed in the simulations presented in this chapter, are extensively studied in Chapter 4.

4

CHAPTER 4: EXPERIMENTAL USER STUDIES

4.1 INTRODUCTION

This chapter describes a large-scale, repeated measures study of the constraint-based control architecture and driver feedback mechanisms developed in this thesis and its effect on the performance of twenty drivers remotely teleoperating a modified utility vehicle through an obstacle field. Specifically, this study was designed to investigate the architecture's effect on three key aspects of the navigation and control task:

1. The navigation and control performance of the combined, human-vehicle system, as indicated by objective measures of control performance
2. The driver's cognitive workload as indicated by that driver's performance on a secondary task, and
3. The driver's confidence in the system and sense of control over its performance.

Four configurations of the control framework were evaluated to independently examine the effects of its control sharing capabilities and the effect of its driver feedback mechanisms. The experiment was designed as a 2 (Feedback: off vs. on) x 2 (Control: unshared vs. shared) full factorial, between-subjects study and analyzed with a mixed measures analysis of variance (ANOVA) and a significance threshold $p = 0.05$. The four test configurations resulting from this design are named in Table 1. Note that because feedback was provided as a means of communicating controller intent and actions to the operator (see 3.4.2), feedback in the unshared control ("Feedback Only") configuration was limited to visual overlays. For the shared, "Semi-Autonomy with Feedback"

configuration, torque was also provided to communicate the result of controller intervention. Finally, in "Transparent Semi-Autonomy," control was shared but no visual or haptic feedback was provided to the human operator.

Table 4.1. Experimental factors and levels

		Control	
		Unshared	**Shared**
Feedback	**Off**	**"Unassisted"**	**"Transparent Semi-Autonomy"**
	On	**"Feedback Only"**	**"Semi-Autonomy with Feedback"**

The following sections describe the setup and results of this study.

4.2 SETUP

In this section, we briefly describe the experimental platform, testing course, drivers, test schedule, and hypotheses. Section 4.3 then describes the results of these tests.

4.2.1 VEHICLE

Experimental testing was performed with a Kawasaki 4010 Mule – a four-wheeled utility vehicle with 23x11-10 tires, an independent, MacPherson strut front suspension, a semi-independent rear suspension, Electric Power assisted rack-and-pinion Steering (EPS), four-wheel hydraulic drum brakes, a continuously-variable transmission, and a top speed of 25 miles per hour.

Several modifications were made and components added to this vehicle to enable semi-autonomous teleoperation. Among other things, these modifications included the addition of a Roboteq AX3500 motor control board and a relay to transfer control of the EPS motor between the stock Electronic Control Unit (ECU) and the Roboteq controller.

This configuration was capable of a maximum 31°/s steering rate and actuation over the vehicle's entire ±33° steering range. Braking and acceleration commands were remotely applied via servo-gearmotor-driven winches. Potentiometers, limit switches, and relative quadrature encoders were installed to provide real-time feedback of the steering angle and other control commands.

A Velodyne LIDAR, NavCom GPS, triaxial Inertial Measurement Unit (IMU), and 640x480-pixel, 84° (horizontal) x 64° (vertical) Field of View (FOV), progressive area scan color CCD camera were also installed to provide obstacle sensing, positioning, motion feedback, and camera feedback, respectively. Figure 4.1 shows the completed experimental platform. Appendix B lists its parameters as estimated and implemented in the MPC prediction model.

Figure 4.1. Experimental platform

An on-board Linux PC running at 2.66 GHz processed sensing and positioning data, planned constraints, predicted threat, calculated control inputs, generated an operator feedback signal, and allocated control authority at 10 Hz. Obstacles detected in LIDAR scans were circumscribed by bounding boxes, dilated by the lateral and longitudinal half-widths of the vehicle to account for the vehicle body (given a c.g.-centered position reference), and spatially shifted along the longitudinal vehicle axis, *x*, by $\Delta x = -(\Delta t_C + \Delta t_L)V_x$, and along the lateral vehicle axis, *y*, by $\Delta y = -(\Delta t_C + \Delta t_L)V_y$, where Δt_C represents the ~300 ms control delay (measured empirically), Δt_L represents the 100 ms sensing delay, and V_x and V_y give the longitudinal and lateral velocities of the vehicle, respectively. Without filtering, the GPS position estimate had an accuracy of approximately 2 m. Processed through a Kalman filter, this signal combined with IMU data to give approximately 0.5 m positioning accuracy. Sideslip feedback, β, not available from the IMU, was estimated as $\beta = \beta_{rear} + \frac{x_r}{V}\dot{\phi}$ assuming zero slip at the rear wheels ($\beta_{rear} = 0$).

The location of obstacles in the vehicle-fixed frame were obtained from LIDAR scans, and were known to within approximately 0.1 m. The MPC control algorithm and its state prediction model ran at 20 Hz, with 40-step control and prediction horizons. Figure 4.2 illustrates the general architecture of the combined system. Note that all components except for the remote operator interface were physically located on the vehicle (and were thus not subject to communication delays or dropout).

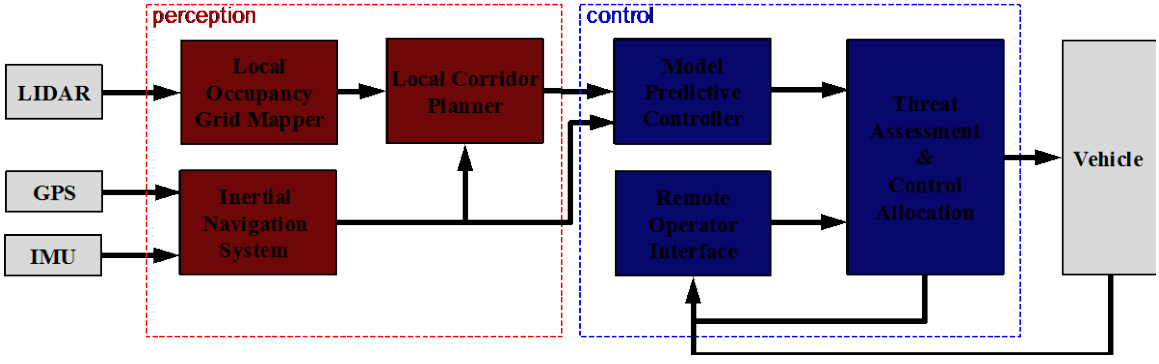


Figure 4.2. Sensing and control architecture

Constraint locations, current threat, and other driver feedback signals were transmitted to the remote operator control station over an 802.11g wireless link via an omnidirectional, high-gain antenna with a 2 Watt amplifier on both the vehicle and at the remote control station. In ideal conditions and operating line-of-sight, this setup provided 4 Mb/s bandwidth at a maximum range of approximately 1 km. In practice, range was limited to approximately 100 m, and line-of-sight was not maintained. Figure 4.3 shows the LIDAR, GPS receiver, and antennae.

Figure 4.3. LIDAR, GPS, and communication antennae

4.2.2 OPERATOR CONTROL UNIT AND USER INTERFACE

At the remote operator control station, a teleoperator received video and state feedback data on a computer monitor and issued steering commands through a Logitech G27 steering wheel. Torque feedback was applied to the operator's steering wheel via a dual-motor force feedback mechanism. In order to simulate communication delays and periodic loss of vision caused by random occurrences such as camera obfuscation, sensor outages, and loss of communication, the camera feed seen by the teleoperator was

delayed by 500ms and frozen at random intervals for up to 2 seconds at a time. Figure 4.4 shows the operator control unit setup. Note that in order to prevent drivers from looking directly at the vehicle during testing, this control unit was placed inside a nearby building.

Figure 4.4. Operator control unit

The Operator Control Unit (OCU) interface consisted of a (delayed) video feed from the onboard CCD camera and various overlays. A compass, speedometer, and secondary task button (described in Section 4.2.3) were overlaid on the screen regardless of the control configuration in use. In addition to these indicators, feedback-enabled configurations overlaid a wireframe representation of the selected homotopy, along with a steering indicator showing the current position of the driver's current steering command and the actual command being implemented by the shared controller. This indicator was provided to give the driver a visual indication of the magnitude and difference between his/her current input and the vehicle's response. Note that torque feedback on the steering wheel was designed to communicate the same information over a different sensory channel.

a) **b)**

Figure 4.5. Operator control unit interface with (a) and without (b) driver feedback

4.2.3 COURSE SETUP & OPERATOR TASKS

Testing was performed on a large (50 m x 30 m) and relatively flat grassy field. Thirty-five barrels were spaced throughout the field in a sequence of five rows, each of which contained one opening that was slightly larger than the others. As described in the test schedule below, this arrangement and the location of preferred openings was changed between rounds to prevent drivers from relying on worn paths or memorized patterns in navigating it. Drivers were instructed to cross this course as quickly as possible without hitting barrels. Each time the vehicle collided with (knocked over) or brushed (touched but did not knock over) a barrel, a referee logged the event in a synchronized log of vehicle state, threat assessment, and control inputs.

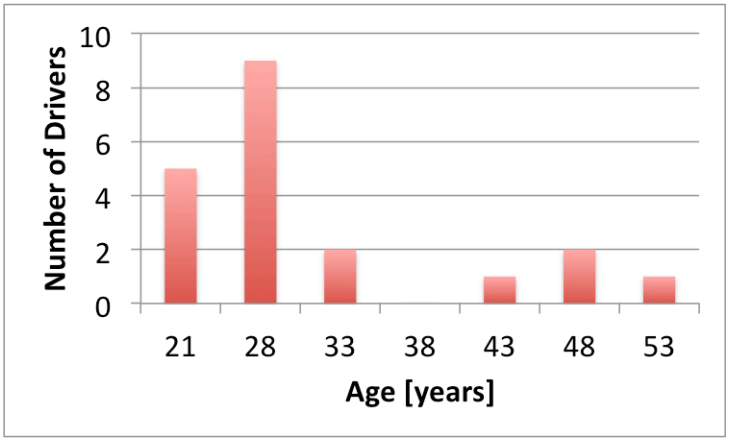
Figure 4.6. Picture of test environment

In addition to the primary control task, operators were given a secondary monitoring and response task. This secondary task was administered during each trial to estimate the cognitive workload imposed by the various control configurations on the operator. The secondary task required the operators to press a button on the steering wheel every time a warning indicator box in the lower left side of their screen indicated the need. To make this task more challenging, the warning light assumed three states at random, approximately 2-second, intervals during the trial. These states used redundant text and color modalities and included: “Resting…” (white), “Don’t Act!” (blue), and “Press Headlights!” (red) as shown in Figure 4.7. Operators were instructed to press the button only when this indicator assumed its red, “Press Headlights!” state. True positives, false positives, and missed responses were logged along with the time that elapsed between the start of each state and the operator’s response.

Figure 4.7. Secondary monitoring task button states

4.2.4 TEST DRIVERS AND INCENTIVE STRUCTURE

Twenty operators ranging in age from 20 to 51 years with 0-35 years of driving experience, and 0-20+ years of video game experience (see Figure 4.8) were tasked with remotely (non-line-of-sight) teleoperating the vehicle across the obstacle course shown in Figure 4.6. These operators came from mostly technical backgrounds; programmers, 3d artists, technical writers, and mechanical, industrial, quality assurance engineers were all represented.



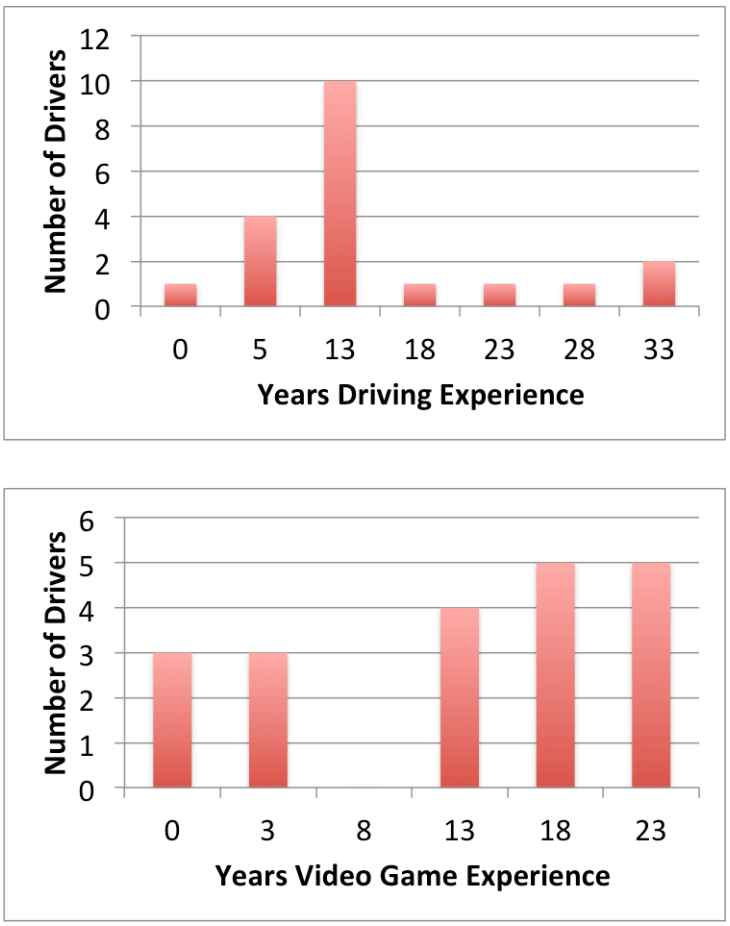


Figure 4.8. Test driver composition

Operators were instructed to minimize the performance score, measured as

Score = ***Time*** + 10****Collisions*** + 5****Brushes*** – ***Hits*** + ***Misses***. **(4.1)**

This score represents the total time it takes to navigate the course (in seconds), plus 10-second penalties for each collision, plus 5-second penalties for each brush, 2-second penalties for incorrect responses to the secondary task and 2-second rewards for correct responses. As an incentive for good performance, $150, $100, and $50 gift certificates were promised to the top three finishers.

4.2.5 QUALITATIVE SURVEYS AND USER CONFIDENCE

Each operator's evolving comfort with, trust in, and preference for the various feedback and control configurations was assessed via an 18-question survey administered at the end of each day's testing (three surveys total). Sixteen of this survey's questions (four for each configuration) were configuration-specific, while the final two questions gauged the perceived helpfulness of the torque and visual feedback mechanisms for each driver. All questions were posed on a 5-point Likert scale to provide some room for subjective assessment without eliciting an unnecessarily granular level of detail. Response sets were aligned with positive or numerically-large values consistently on the right and negative or small values consistently on the left. Figure 4.9 shows the four questions posed for each control configuration (the "Unassisted" configuration in the example shown).

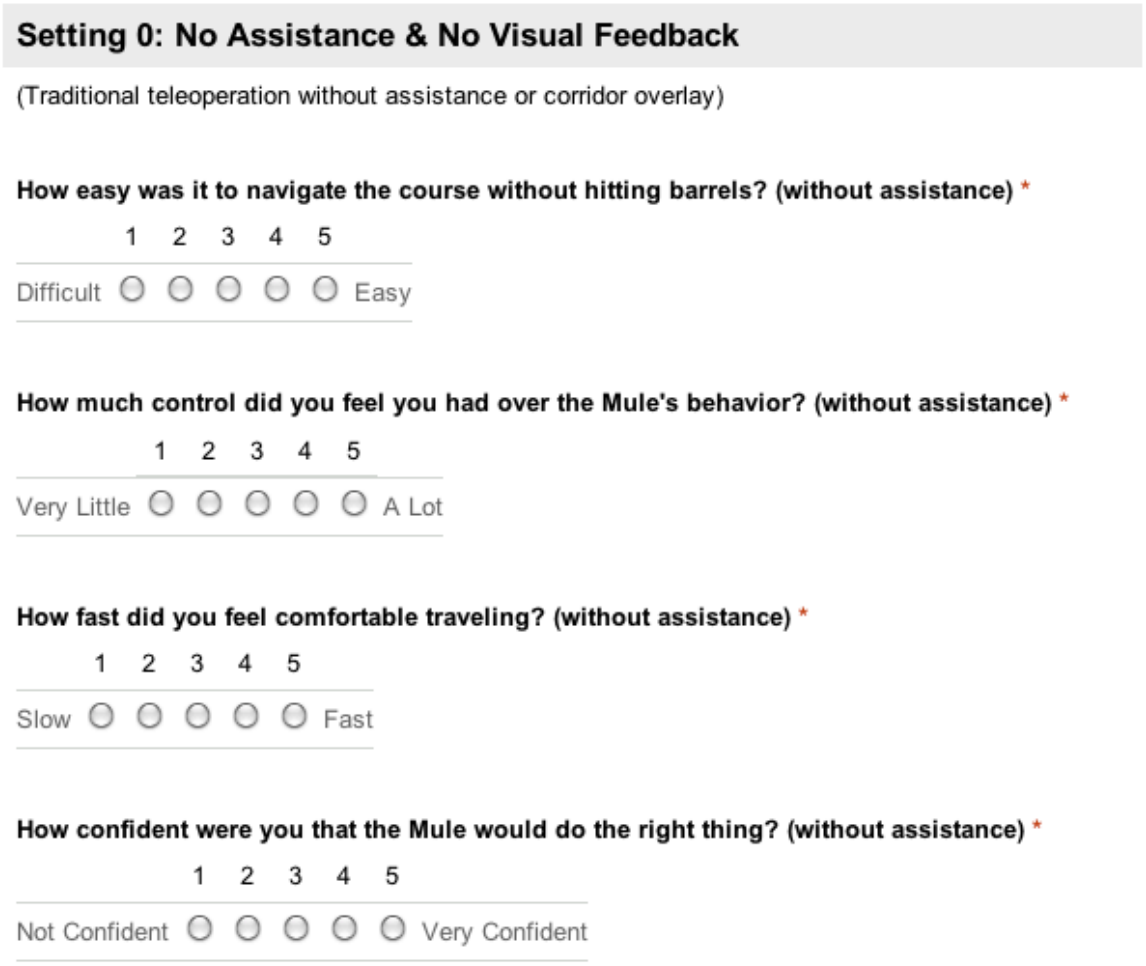


Figure 4.9. Subset of the post-trial questionnaire pertaining to "Unassisted" control configuration

Each survey ended with a question about the visual and haptic (torque) feedback modalities specifically, and provided an opportunity for operators to comment on the day's testing.

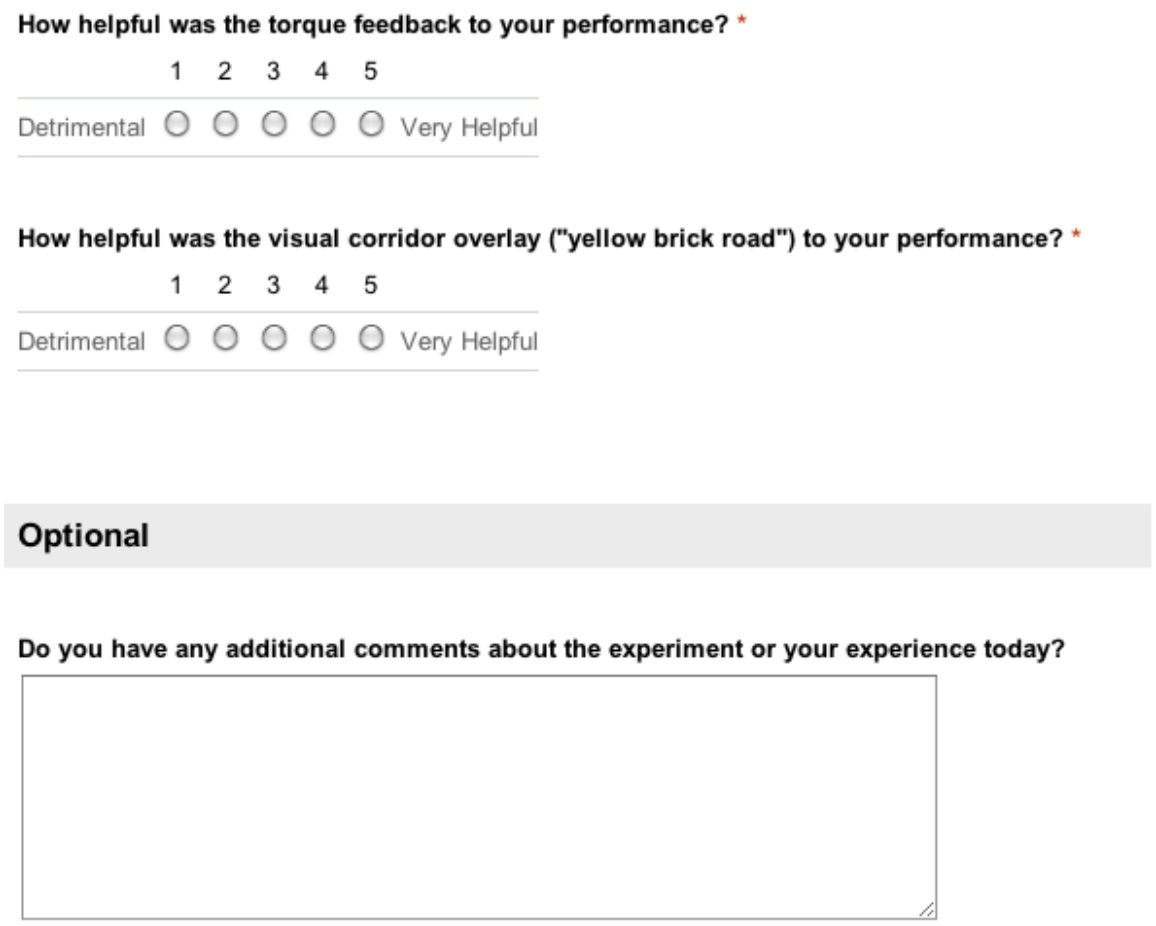


Figure 4.10. Feedback modality questions and optional, free-form feedback questions provided at the end of each day's survey

4.2.6 TRAINING AND TEST SCHEDULE

Prior to the experiments, all operators were briefed regarding the test setup, control interface, and shared control details and provided with a detailed presentation of the control algorithm and test configurations. In the weeks prior to the start of testing, each operator manually drove the vehicle through a similar course several times to accustom them to the teleoperation environment.

Each round of testing consisted of four unscored warm-up runs (one per configuration), followed by 16 scored rounds (four per configuration), with the

configuration order randomized to avoid ordering effects. Immediately following each round's vehicle trials, drivers were administered the trust/preference survey. Operators were staggered over the course of each week (approximately five per day), and staggered such that each operator was provided approximately one week between rounds. The course setup and barrel spacing was slightly altered each week to prevent users from relying on worn paths or memorized control sequences to get through them.

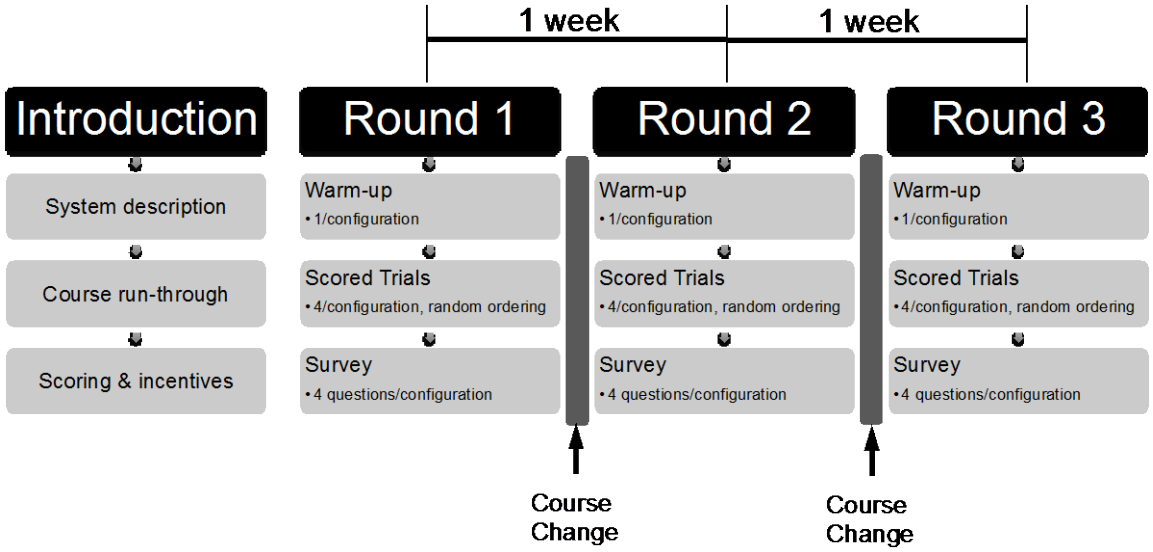


Figure 4.11. Experimental sequence and ordering

Altogether, 1,200 trials were conducted, with 960 of those trials scored (240 scored trials per test configuration), and 1,080 qualitative survey responses were collected. Vehicle states, sensor data, constraint calculations, controller predictions, threat assessments, driver inputs, collision events, and several other signals were recorded for each run.

4.2.7 HYPOTHESES

This study tests the following hypotheses:

1. Main effects: Control type and driver feedback both have a significant effect on operator performance, cognitive workload, and confidence in the teleoperation task.
2. Interaction effects: None – feedback improves performance and confidence both with and without control assistance.

4.3 RESULTS

Figure 4.12 plots the results of a typical run with shared control and operator feedback enabled. Note that obstacle were sensed in the local (vehicle-fixed) frame. Corresponding bounding boxes were therefore subject to some positioning error when plotted in the global frame (seen by the apparent obstacle "motion" of Figure 4.12).

Main and interaction effects were assessed using ten objective measures and four subjective measures. Objective measures of teleoperation performance were assessed from run data logged at 10 Hz. These include collision frequency (collisions/run), brush frequency (brushes/run), average velocity (m/s), course completion time (seconds), driver steer volatility (standard deviation of driver steer command, degrees), vehicle steer volatility (standard deviation of vehicle steer response, degrees), secondary task reaction accuracy and reaction time (s), average predicted threat, and overall performance score (seconds). Subjective measures of operator confidence include driver-reported ease of collision free navigation, perceived control over vehicle behavior, comfortable speed, and confidence that the vehicle would "do the right thing". Appendix C tabulates key performance metrics and survey results by factor. The discussion below elucidates these and other results.

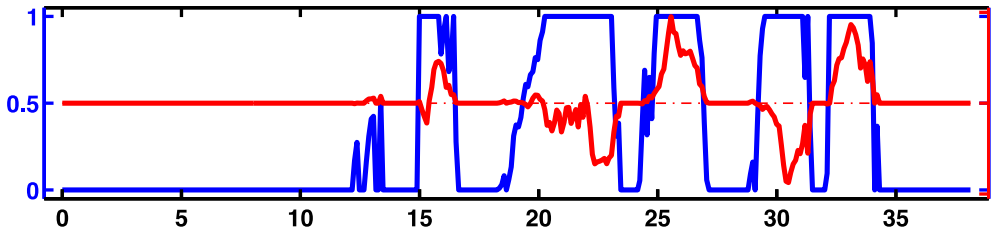


Figure 4.12. Plot of a typical run showing level of intervention and its effect on the steering angle seen by the vehicle

4.3.1 NAVIGATION PERFORMANCE

Assessed over all drivers, courses, and dependent measures, the main effect of constraint-based semi-autonomy was improved performance.

I) SHARED CONTROL

Shared control reduced the occurrence of collisions from 0.41 collisions per run to 0.11 collisions per run – an improvement of over 72% ($F(1,956) = 65.54$, $p < 0.0001$). Brush frequency also decreased, though by a smaller, 44% margin ($F(1,956) = 12.39$, $p < 0.001$). We attribute this disparity between collision avoidance and brush prevention to sensor deficiencies and an insufficient dilation of the obstacle bounding boxes to account for them. When obstacles entered the LIDAR's ~3m blind spot near the vehicle, their predicted location was estimated using a combination of vehicle position estimates (accurate to ~0.5 m) and visual odometry performed on obstacles that remain within the LIDAR's view. Obstacle positioning errors arising from this tracking slightly reduced the shared controller's obstacle avoidance efficacy. We hypothesize that with the addition of SICK LIDAR or other short-range sensing, most of the remaining brushes and many of the collisions can be avoided.

Presumably emboldened by the vehicle's obstacle avoidance capabilities when shared control was enabled, operators drove faster – increasing average speed by 26%, from 1.36 to 1.72 m/s ($F(1,956) = 176.3$, $p < 0.0001$) and reducing course completion time by 25% – from 47.0 to 35.4 seconds ($F(1,956) = 172.1$, $p < 0.0001$). Note that the distance traveled by the operators in their chosen routes was similar with and without control assistance. This is principally due to the alternating "gate" structure of each row of barrels and the slalom route users typically identified during their "warm-up" (unscored) runs prior to the start of each day's trials. Figure 4.13 shows each of these performance metrics and its standard error by configuration.

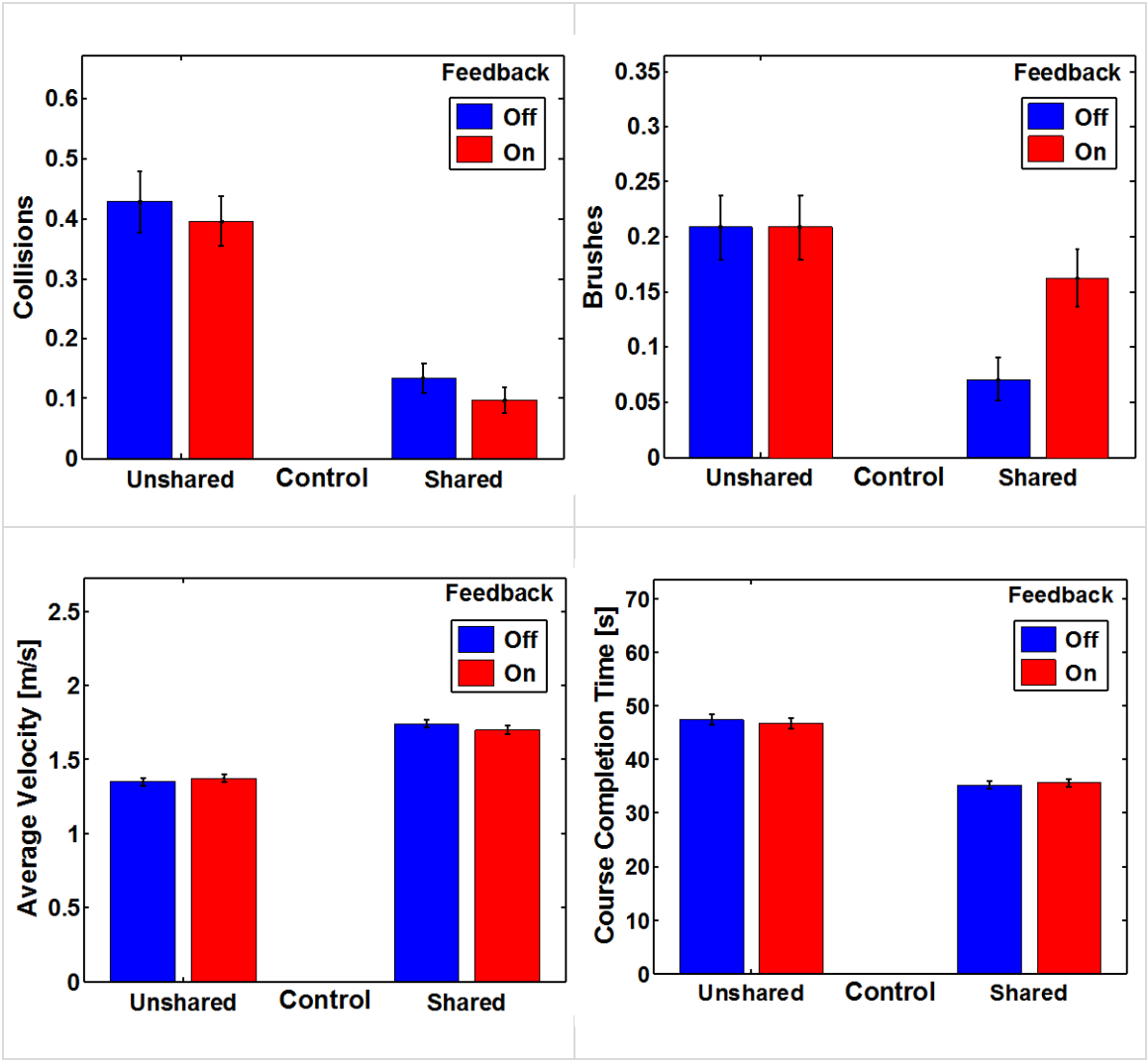


Figure 4.13. Effect of shared control and driver feedback on the collision frequency, brush frequency, average velocity, and course completion time of study participants

With the semi-autonomous control system in the loop, drivers not only drove faster and with fewer collisions, but were also significantly more moderate in their control inputs. With shared control enabled, driver steer volatility decreased by 41% ($F(1,956) = 409.93$, $p < 0.0001$). This reduction in driver steer volatility led to a corresponding reduction in the steering volatility experienced by the vehicle: with shared control, the standard deviation of the vehicle's steering commands decreased from 15 to 11 degrees ($F(1,956) = 152.23$, $p < 0.0001$).

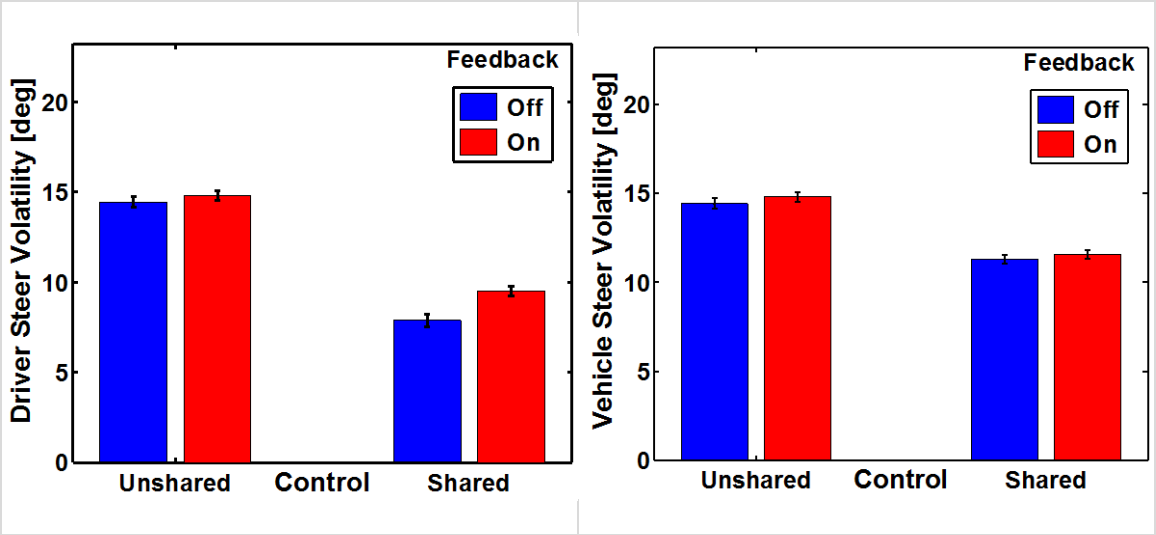


Figure 4.14. Effect of shared control and driver feedback on the driver and vehicle steer volatility

Figure 4.15 shows the Receiver Operating Characteristic (ROC) curve for the secondary task response. While control sharing did not significantly change the rate of True Positives (TP), it did increase the rate of False Positives (FP) from 0.03 FP/trial to 0.05 FP/trial ($F(1,956) = 6.15$, $p = 0.01$). At the same time, reaction times to the secondary task significantly improved, from 0.78 seconds per response without shared control to 0.69 seconds with it ($F(1,956) = 6.51$, $p = 0.01$). Figure 4.16 shows the mean and standard errors for reaction times to the secondary task.

We hypothesize that this willingness to respond more quickly – albeit at the cost of accuracy – was the result of operators paying more attention and allocating greater mental resources to the secondary task when shared control was enabled. While tenuous, this hypothesis may be supported by the steering volatility results, which show that with control enabled, operators made fewer and smaller adjustments to their steering commands.

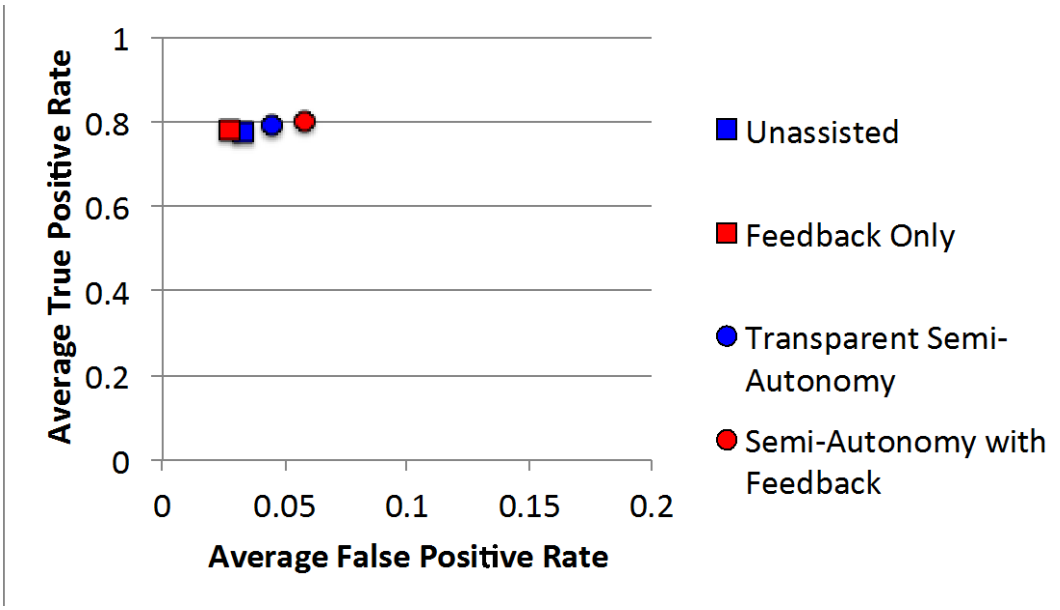


Figure 4.15. Receiver operating characteristic for each system configuration

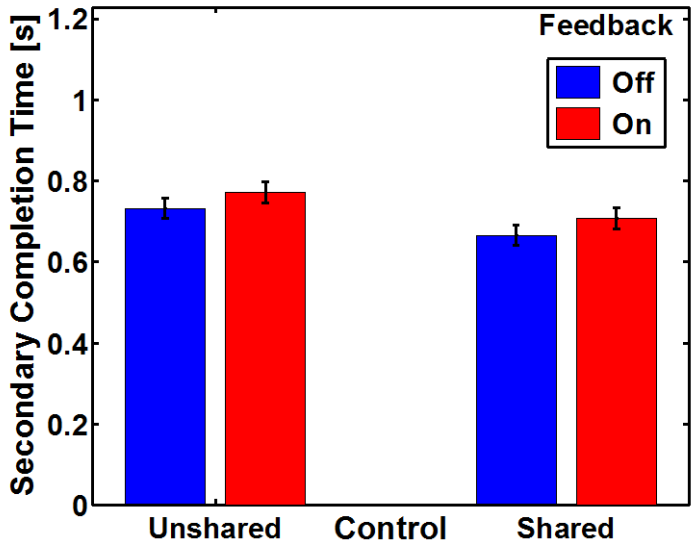


Figure 4.16. Secondary task reaction time (mean and standard error) by configuration type

In addition to improving collision and brush avoidance, shared control also reduced the average threat experienced by the vehicle by 62% ($F(1,956) = 81.4$, $p < 0.0001$). Given the objectives of this shared control formulation – that of maintaining predicted vehicle threat below a specified maximum (Φ_{aut}), this result is not surprising. It

suggests that, on average, the shared controller provided safer trajectory options to the vehicle by maintaining its current state and orientation better aligned with the safe homotopy.

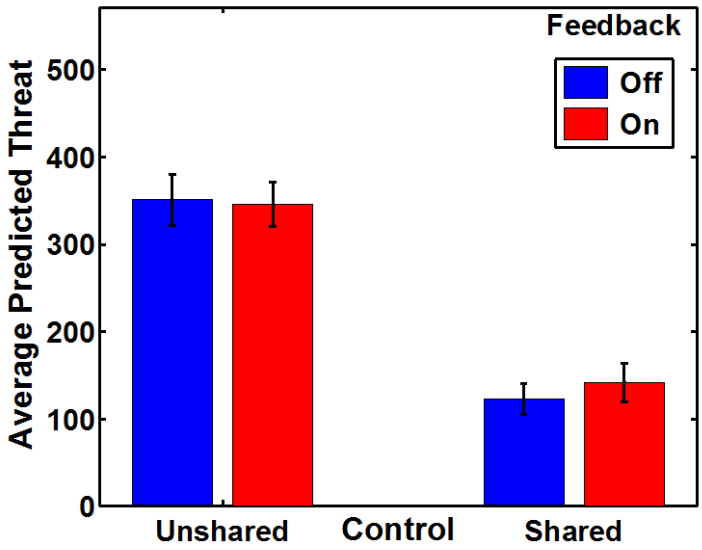


Figure 4.17. Average predicted threat (mean and standard error) by configuration type

Finally, constraint-based semi-autonomy improved driver performance scores by 30% – from an average score of 47.2 seconds without assistance to an average of 33.3 seconds with it ($F(1,956) = 186.87$, $p < 0.0001$). Figure 9 summarizes the effect of shared control and feedback on teleoperation performance.

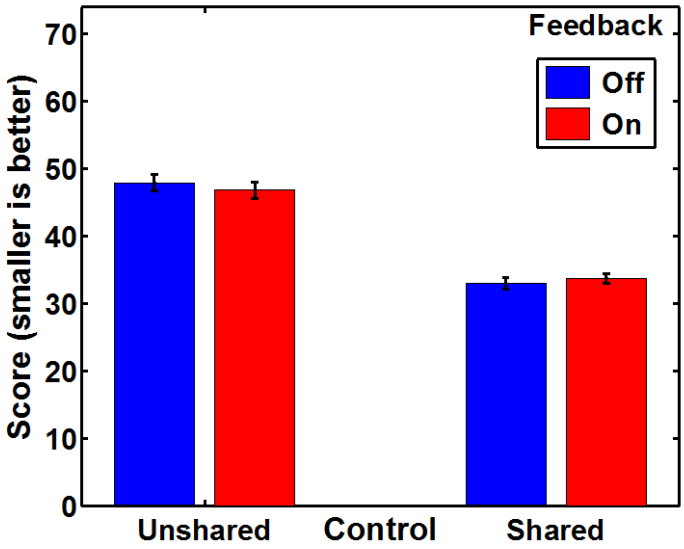


Figure 4.18. Overall driver performance score (mean and standard error) by configuration type

With enough control intervention, similar improvements in collision avoidance and average speed might be expected of any controller. What makes the constraint-based framework unique is the minimal degree of adjustments it requires to achieve the above results. Averaged across all drivers with shared control and feedback both enabled, the controller took only 43% of the available control authority (mean(K) = 0.43, SD = 0.13) to effect the above performance improvements. This minimal restriction on human commands afforded the operators significant freedom to maneuver as desired while simultaneously reducing the cognitive burden of high-speed, high-precision obstacle avoidance calculations (as evidenced by improved reaction times on the secondary task).

II) Operator Feedback

The main effect of feedback was less significant than that of control strategy, largely owing to greater variability in user response to it. The presence of feedback decreased collision rates by an insignificant 13% ($F(1,956) = 0.93$, $p = 0.34$), and made no significant difference for brush rates, average velocity, course completion time (see Figure 4.13).

The presence of visual and torque feedback did, however, significantly *increase* driver steer volatility by 8.9% ($F(1,956) = 11.93$, $p < 0.001$) (see Figure 4.14). This increase in driver steer volatility when presented with visual and/or torque feedback suggests that one effect of such feedback on a teleoperator navigating with delayed and intermittent visual cues is to prompt larger and more frequent control actions. We also note that when presented with torque feedback, many operators naturally cede some control (and corresponding placement) of the steering wheel to the torque feedback controller, whose commands are typically larger and more rapid than those of the driver when intervention is required. In most cases, including the run shown in Figure 8, this additional input was necessary to compensate for an otherwise insufficient operator command. As Figure 4.14 shows, the average steer volatility seen by the vehicle does not significantly change with feedback ($F(1,956) = 1.58$, $p = 0.21$), owing largely to the moderating effect of the shared controller.

Neither the accuracy, nor the response times to the secondary task changed significantly when driver feedback was provided. Average performance scores were also largely unchanged, decreasing by a statistically insignificant 0.002% when driver feedback was provided ($F(1,956) = 0.04$, $p = 0.85$).

As discussed in Section 3.4, the intention of feedback in the context of constraint-based semi-autonomy is not to directly improve vehicle performance, but to inform the operator of actions the controller is taking (with shared control enabled) or would like to take (shared control disabled) to improve vehicle performance. Thus, we anticipated its effect on the comfort and confidence of the human operator to be more pronounced than it was on objective performance metrics. This impact on operator confidence is explored through qualitative user surveys (discussed in Section 4.3.2 below).

III) INTERACTION EFFECTS

As expected, interactions between control and feedback settings were minimal. The only exception to this trend was driver steer volatility, which increased 63% more ($F(2,956) = 4.56$, $p = 0.03$) when visual and haptic feedback was added to the shared control configuration than it did when visual feedback was provided for an otherwise-unassisted operator. This result follows quite closely from the observation above

regarding operator propensity to yield to torque feedback. Since torque feedback was not present in the unshared control configurations, the effect was less pronounced.

4.3.2 USER CONFIDENCE

Post-trial survey responses were also analyzed using mixed-measures ANOVA with a significance threshold of $p = 0.05$ to evaluate the effect of control and feedback strategy on operator confidence. Main and interaction effects discovered in this analysis are discussed below.

I) SHARED CONTROL

Across all dependent measures, operators felt more confident in the system and comfortable in their performance when the shared controller was enabled. With assistance, operators reported a 43% increase in ease of navigation ($F(1,236) = 85.8$, $p < 0.0001$), a 38% increase in comfortable speed ($F(1,236) = 72.4$, $p < 0.0001$), a 12% increase in perceived control ($F(1,236) = 9.4$, $p < 0.01$), and a 44% increase in confidence that the vehicle would do the right thing ($F(1,236) = 79.38$, $p < 0.0001$). Given the performance improvements noted in Section 4.3.1, many of these subjective measures of user confidence are not particularly surprising; operator sentiment may have simply been reflecting the performance improvements they observed.

What *is* surprising, and bodes well for the merits of constraint-based semi-autonomy, is that even with an autonomous agent acting in concert to limit or adjust their steering commands, operators still reported feeling like they had *more* control over the Mule's behavior – not less. This result speaks to the minimally-restrictive nature of constraint-based intervention; rather than assume, and force the operator to follow a desired path, controllers based on constraints need only disallow unsafe regions that, presumably, the operator would not have intended to traverse anyway. This parity of objectives, together with the controller's ability to forgo intervention whenever possible, instilled a greater sense of control in operators who, technically, had 43% less.

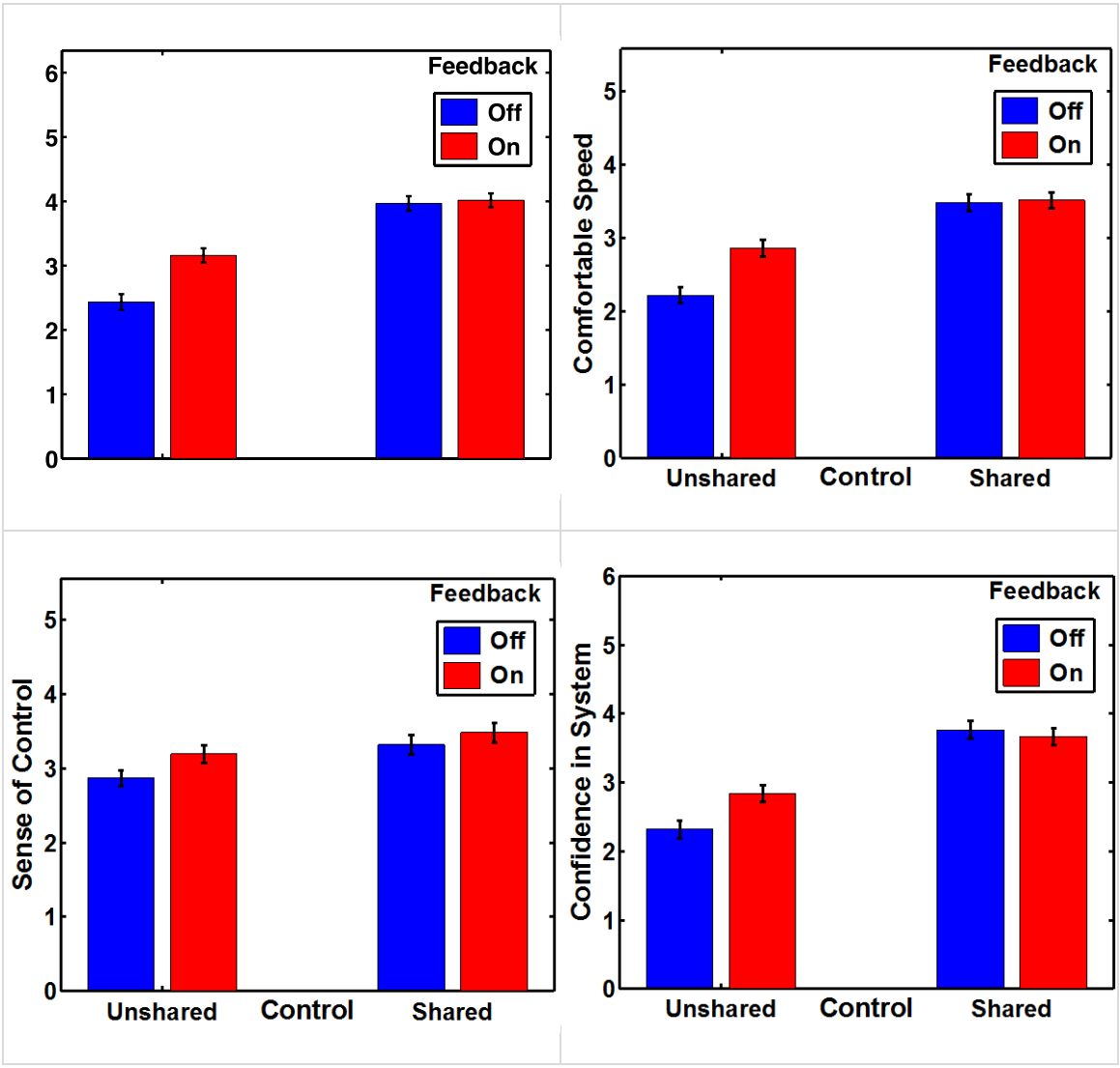


Figure 4.19. Mean and standard error of subjective measures of operator confidence

II) OPERATOR FEEDBACK

We also note that, as expected, communicating controller intentions and control actions to the operator via visual and haptic feedback improved most measures of driver confidence. Main effects of this feedback included a 12% increase in ease of navigation ($F(1,236) = 11.59$, $p < 0.001$), a 12% increase in comfortable speed ($F(1,236) = 8.9$, $p < 0.01$), an 8% increase in perceived control ($F(1,236) = 4.1$, $p = 0.04$), and an insignificant

7% increase in confidence that the vehicle would do the right thing ($F(1,236) = 2.77$, $p = 0.10$). We note that, in long-form survey responses, many operators expressed that the more familiar they became with the shared control framework and its capabilities, the less they depended on visual or haptic feedback to understand or trust what it was doing. This sentiment is borne out in survey responses – shared control strongly affected driver confidence that the Mule would "do the right thing", while the effect of feedback was statistically insignificant.

III) INTERACTION EFFECTS

Interaction effects were more significant for subjective measures of performance than for the objective measures. Across nearly all subjective measures of operator confidence, operators felt more comfortable, confident, and in control of the unshared control system when provided with visual feedback. With shared control enabled, operator confidence remained largely unchanged by feedback, the notable exception being overall confidence in the system, which actually *decreased* by 3% when feedback was added to shared control ($F(2,236) = 5.99$, $p = 0.02$). Taken together with the main effects of shared control and feedback on collision rates and driver steer volatility, we posit that this decreased confidence comes more as a result of operators misunderstanding torque feedback (as a *mandate* rather than as a *signal*) rather than as the result of actual performance degradation. In post-trial surveys, operators rated both feedback modalities as helpful, but also consistently rated visual overlay as more so ($M = 3.7$, $SD = 1.0$ vs. $M = 3.2$, $SD = 1.0$) than torque feedback ($t[56] = 6.5$, $p = 0.01$). Figure 11 shows average responses and standard error. This result does not, however, necessarily require that user perception of relative helpfulness wasn't shaped by configuration setup; where feedback was more needed (and impactful) with unshared control, only visual overlay was present. For shared control configurations, both visual and haptic modalities were used, and had less significant an effect on overall driver performance.

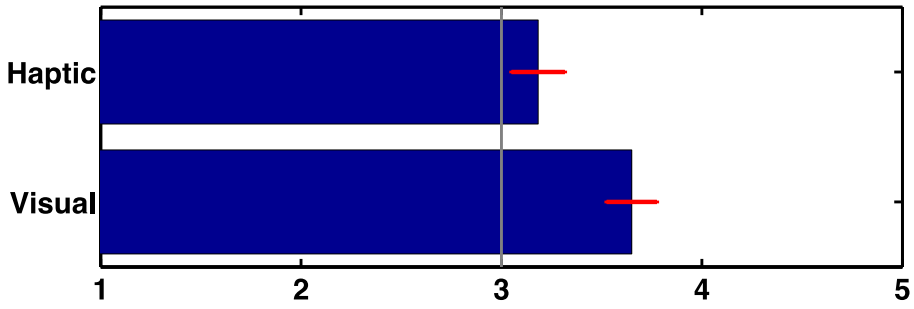


Figure 4.20. User impressions of the helpfulness of visual and haptic feedback modalities as reported in post-trial surveys

4.3.3 LEARNING AND TRUST EFFECTS

Sections 4.3.1 and 4.3.2 describe the main and interaction effects of this thesis's shared control framework and teleoperator feedback mechanisms on objective measures of operator performance and subjective assessments of operator confidence. What this analysis does not capture is the effect of these control configurations on operator performance and preference over time, and the relative effect of perceived performance – regardless of the configuration used – on subjective configuration rankings. This section explores these elements of learning and trust.

I) EFFECT OF EXPERIENCE ON PERFORMANCE AND OPERATOR CONFIDENCE

Increasing operator experience tended to provide a similar improvement to most objective performance measures for each control configuration studied. Figure 4.21 shows one such case in which operator performance increased as they became more accustomed to the vehicle, the course, and the control interface. Note that "Run Number" in this figure and those following refers to each run's sequence within that control configuration's tests. That is, Run Number 3 for the "Feedback Only" configuration could have happened before or after similarly numbered runs from other configurations in Round One due to randomization of the run order. Hard breaks at runs one, five, and nine represent the start of the first, second, and third rounds, respectively.

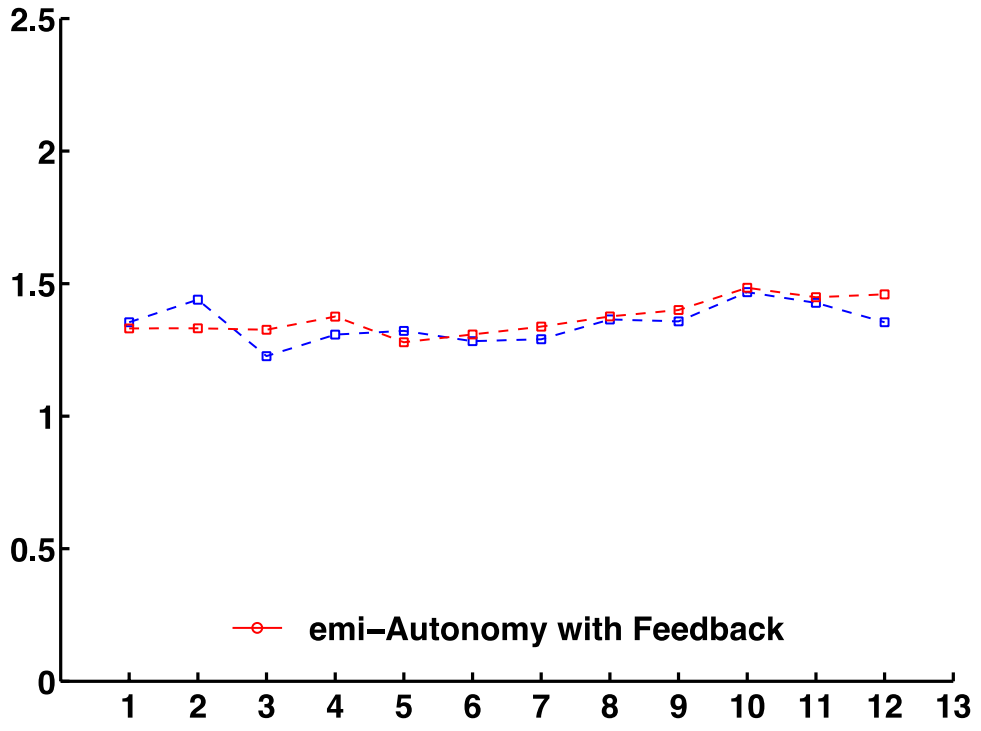


Figure 4.21. Effect of experience on average velocity for the four control/feedback configurations

Other objective measures showed a greater influence of learning on performance. Operator steering volatility was particularly affected by experience; on average, operators controlling the "Transparent Semi-Autonomy" configuration (feedback off, shared control on) became more and more measured in their steering inputs as the experiment progressed. As shown in Figure 4.22, this trend persisted across all three rounds. Both configurations without shared control exhibited the opposite trend (though weakly so). As the number of runs progressed, operators of the "unassisted" or "feedback only" configurations actually became slightly more volatile in their steering commands. We note that, given the limited dataset, these trends are merely suggestive. We also observe that due to the random ordering of configurations, some of this increase in driver steer volatility without assistance may be caused by changing expectations as the operator switches mindsets from shared control configurations to unshared, and back again. A

more rigorous study of learning would sample a larger user base consistently controlling one configuration or another.

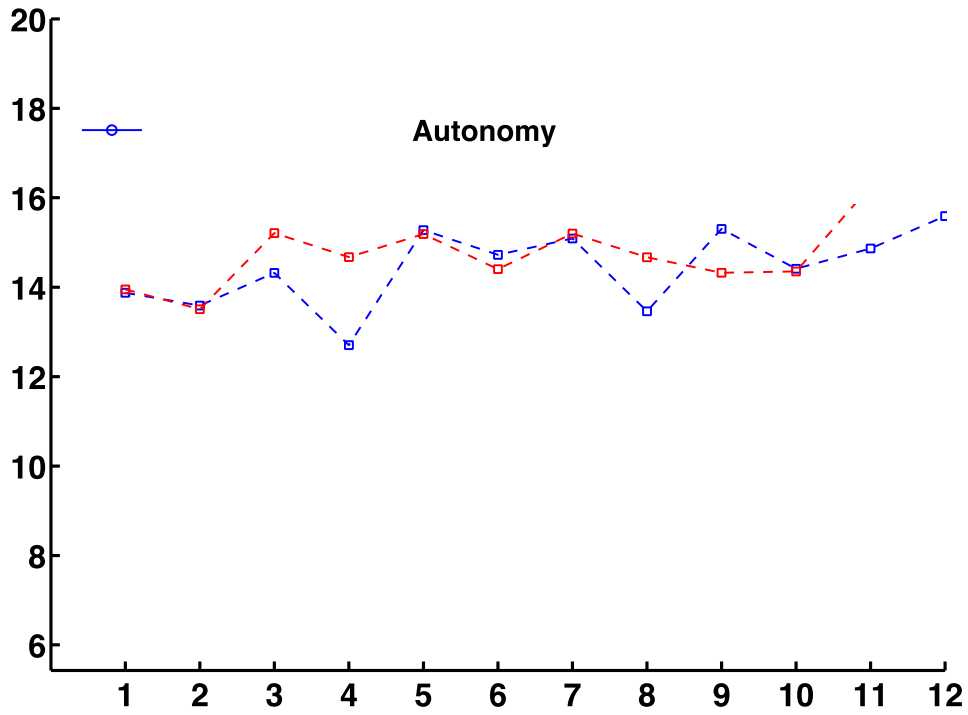


Figure 4.22. Effect of experience on average driver steer volatility for the four control/feedback configurations

As expected, course completion time roughly tracked average velocity, exhibiting a slight downward trend for all four control configurations as the runs progressed and operators became more accustomed to the vehicle, the control interface, and the course. Figure 4.23 illustrates this result.

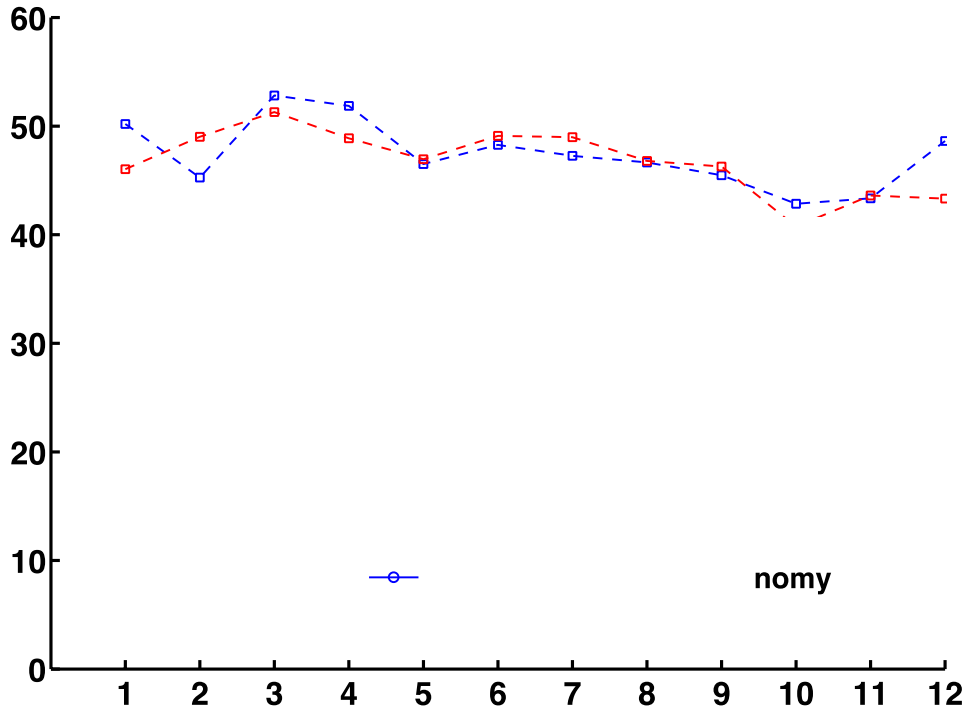


Figure 4.23. Effect of experience on average course completion time for the four control/feedback configurations

Collisions per run were significantly less correlated with run order. As Figure 4.24 shows, collisions happened more often and with greater irregularity for the unshared control configurations than for the shared, but showed little trend, experiment-wide. Where weak trends were observed was within each round's results, as operators appear to have improved their control strategies with shared control enabled as the round progressed. Round 2 showed a particularly strong trend toward fewer collisions per run when operators were assisted by the shared controller. The final round, however, proved an exception to this trend – particularly for the shared control configurations. While the underlying factors leading to this uptick in collisions are not completely understood, we note that these final few rounds do correlate with a similar uptick in driver steer volatility and a decrease in average velocity for all configurations, suggesting that either 1) the course setup for this round was particularly challenging, or 2) some operators assumed that they had not won the competition anyway, and were less careful in their control.

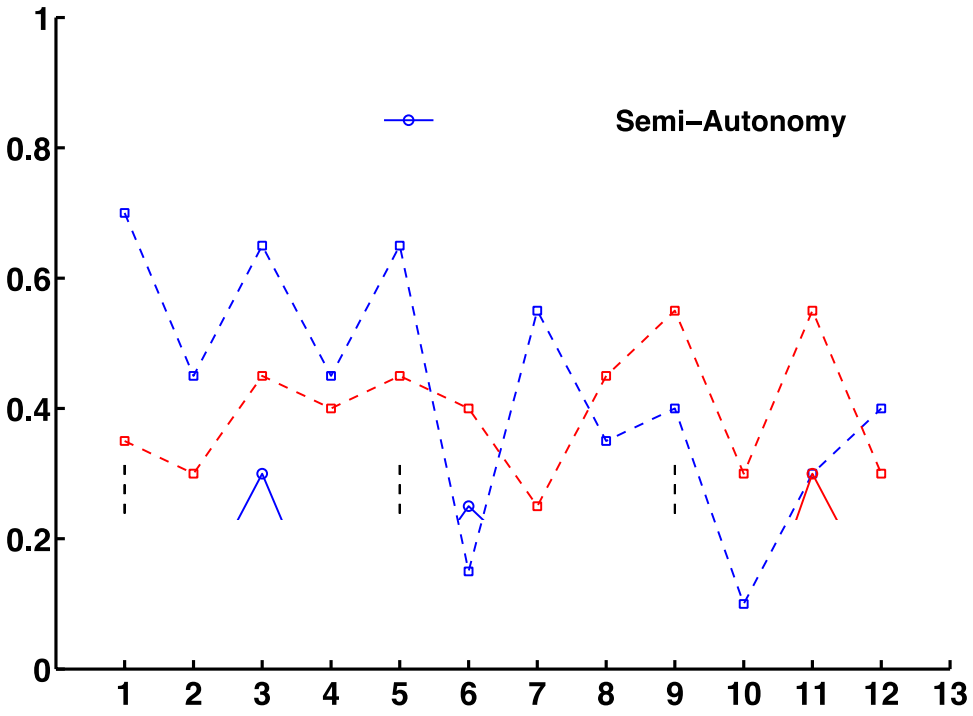


Figure 4.24. Effect of experience on average collision rate for the four control/feedback configurations

We also note that in all three rounds, and for all but one configuration (Feedback-Assisted Semi-Autonomy," operators generally appear to have improved their collision avoidance on the second run of the round. Taken by itself, this result might suggest that one unscored warm-up round per configuration provided insufficient familiarity with each new course configuration. But considered in light of later increases in collision rate as the run progressed, as well as general variation that makes trends in the unshared runs statistically insignificant, we posit that the number of warm-up rounds was not insufficient. Finally, Figure 4.25 shows the general evolution in performance scores over time.

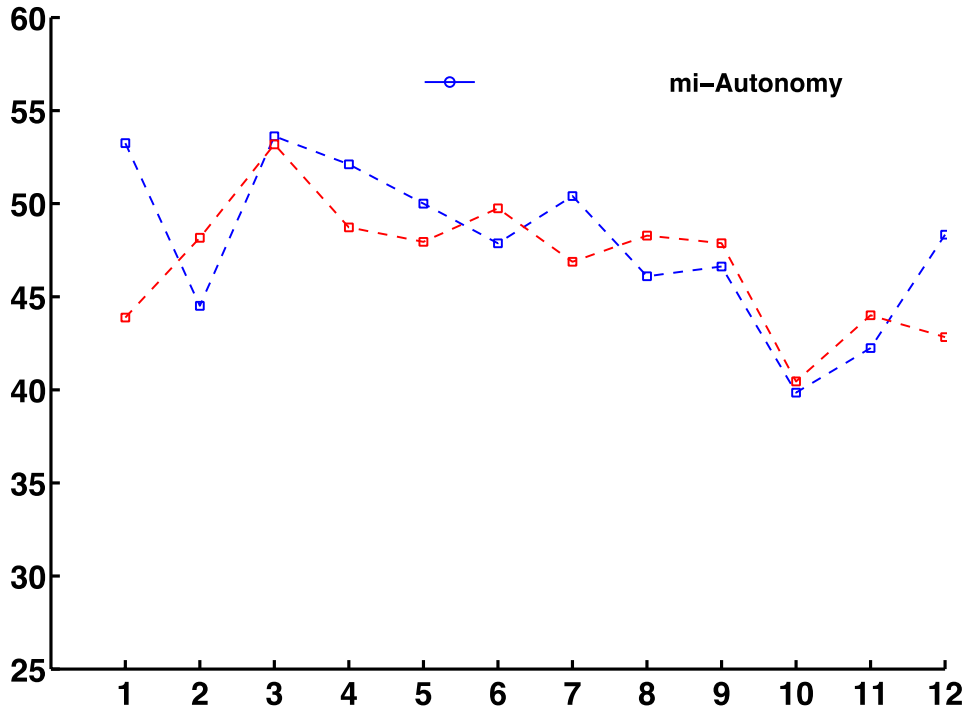


Figure 4.25. Effect of experience on average driver score for the four control/feedback configurations

Figure 4.26 – Figure 4.29 illustrate how operator confidence measures evolved over time. Averaged across all drivers at the conclusion of each round, driver responses to the "[ease of collision-free navigation]" question was relatively static, but trends in the other three confidence measures were instructive and promising. In each post-trial survey, operators consistently indicated both shared control configurations as having made the collision-free navigation problem easier for them than the unshared control configurations. While this reported ease of navigation generally trended upward as the testing rounds progressed, its results were consistent with (and anticipated by) those shown in Figure 4.19.

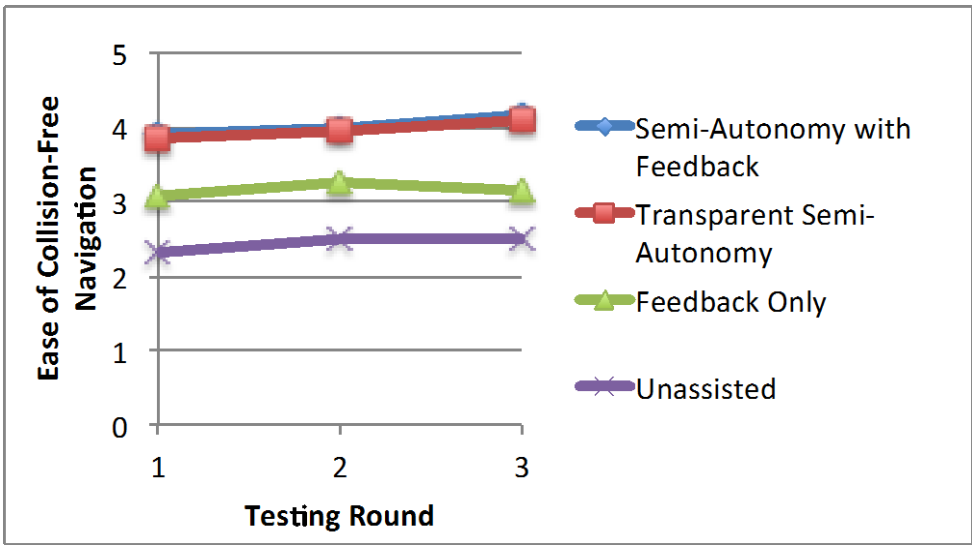


Figure 4.26. Trends in operator sentiment about each control configuration as it affected the ease of the navigation task

Figure 4.27 shows trends in the reported sense of control operators felt over the vehicle's behavior as the testing rounds progressed. Here, we begin to see some of the underlying trends leading to the result discussed in Section 4.3.2 II). Specifically, while both shared control configurations gave users a greater sense of control on the whole, feedback became less and less significant as the rounds progressed and users became more familiar with the shared control system. In survey responses at the end of the first round, users reported feeling a greater sense of control for configurations with feedback, and less when that feedback was not present. By the end of the second round, most users reported feeling roughly the same amount of control for both shared control configurations regardless of feedback, and less with the unshared configurations.

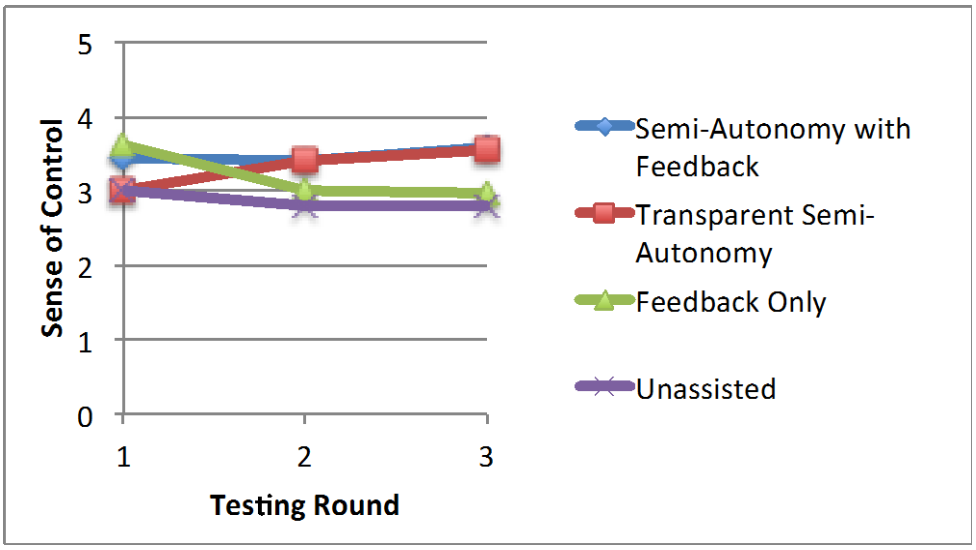


Figure 4.27. Trends in operator sentiment about each control configuration as it affected their sense of control over the vehicle's behavior

Figure 4.28 shows how the speed at which the operator felt comfortable driving changed with experience. This sentiment correlated with the reported ease of collision-free navigation (Figure 4.26); on average, users reported feeling increasingly comfortable traveling faster as they became more experienced. Perhaps not surprisingly when viewed in light of Figure 4.27, this reporting initially gave the edge to semi-autonomy with feedback before gradually shifting to favor transparent semi-autonomy.

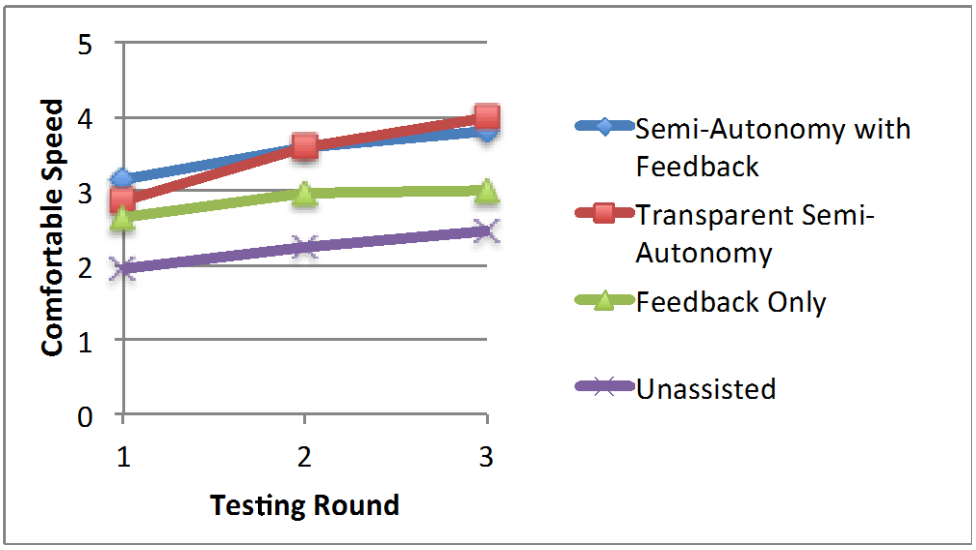


Figure 4.28. Trends in operator sentiment about each control configuration as it affected the speed at which the operator felt comfortable traveling

Finally, overall user confidence that the combined system would "do the right thing" followed a similar trend to users' sense of control. As operator experience increased, feedback became less significant to user confidence and shared control became more so. We note that the apparent flat lining of this confidence around a Likert scale response of four may be the result of central tendency bias.

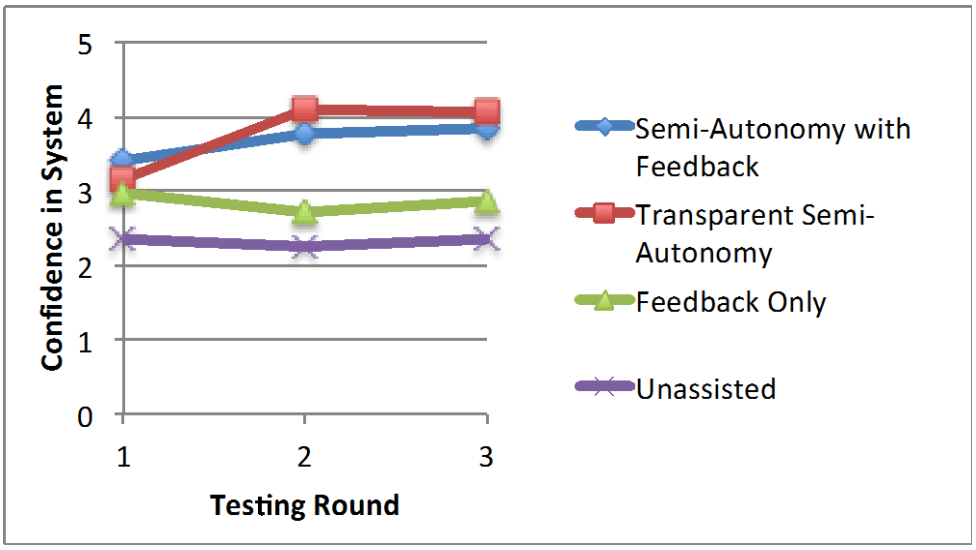


Figure 4.29. Trends in operator confidence that the combined system would "do the right thing" broken down by control configurations

II) The Relative Importance of Performance on Operator Sentiment

Section 4.3.2 describes the relative importance of control configurations on operator confidence. As noted previously, this analysis does not explain whether and how much of these confidence measures could be explained by the performance improvements that operators directly observed (ie. causality between improved performance and operator confidence). Here, we seek to identify and explain some of these correlations.

Correlations between actual operator performance and reported operator confidence were generally less significant than correlations comparing perception to control configuration. Nevertheless, a few weak ties did emerge. For instance, as operator confidence that the vehicle would "do the right thing" increased, objective performance metrics generally improved. In the second and third rounds, this correlation was particularly significant; once operators had become accustomed to the test procedure, user confidence that the system would do the right thing explained much of the decrease in operator steering volatility (R^2 = 0.89), increase in average velocity (R^2 = 0.94), decrease in average collisions per run (R^2 = 0.71), and improvement in score (R^2 = 0.86).

Figure 4.30 plots various performance metrics against operator responses to the survey question, "How confident were you that the [vehicle] would do the right thing?"

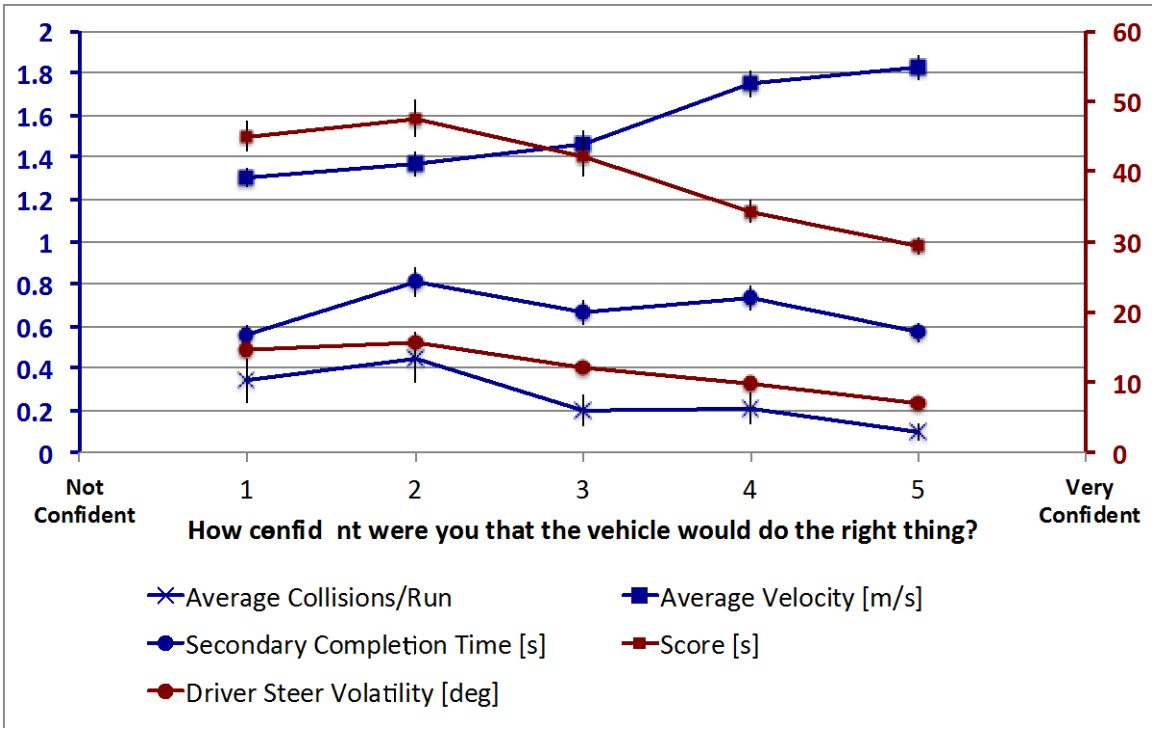


Figure 4.30. Correlations between the operator's confidence that the vehicle would "do the right thing" and measured performance (mean and standard error for rounds 2 & 3). Metrics plotted in red correspond to the secondary y-axis.

Other survey responses explained less of the variation in driver performance. We hypothesize that some of this correlation error may be due to central tendency bias. As Figure 4.31 – Figure 4.33 illustrate, this bias appears to have been present in most survey responses as users were hesitant to use either extreme in the 5-point Likert scales provided.

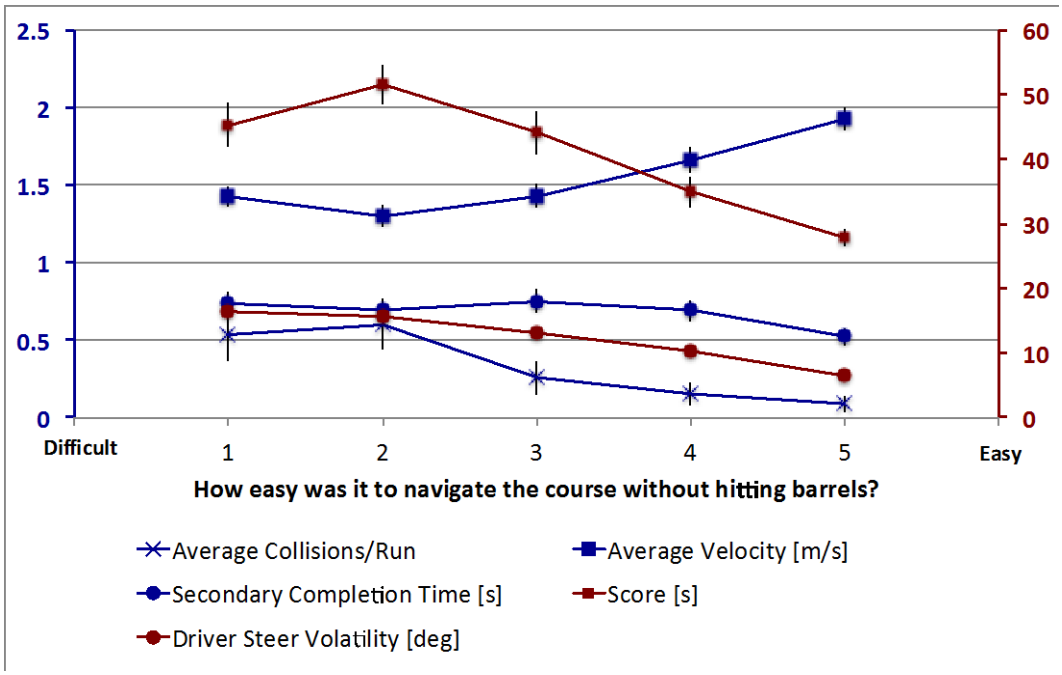


Figure 4.31. Operator-reported ease of navigation vs. measured performance (means and standard errors for rounds 2 & 3). Metrics plotted in red correspond to the secondary y-axis

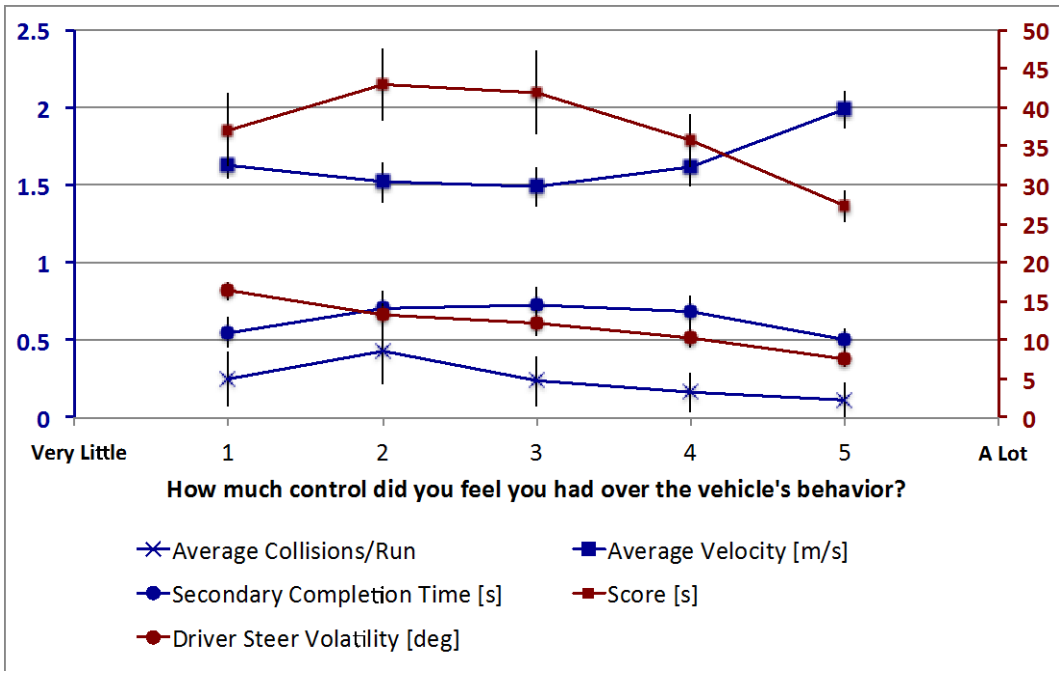


Figure 4.32. Operator-perceived sense of control vs. measured performance (means and standard errors for rounds 2 & 3). Metrics plotted in red correspond to the secondary y-axis

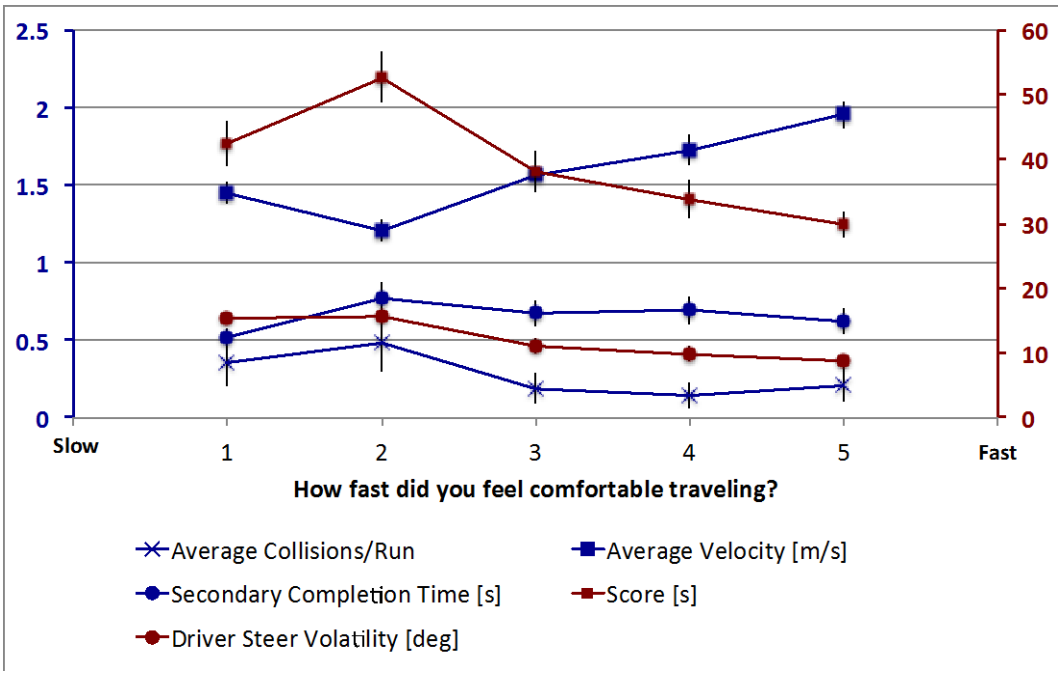


Figure 4.33. Operator-reported comfortable speed of travel vs. measured performance (means and standard errors for rounds 2 & 3). Metrics plotted in red correspond to the secondary y-axis

Averaged across all three rounds, and plotted against the average sentiment expressed in the four survey questions, operator control, confidence, and comfort measures were only weakly correlated with performance metrics of interest. As Figure 4.34 illustrates, user sentiment explains only some of the confidence that users placed in the system.

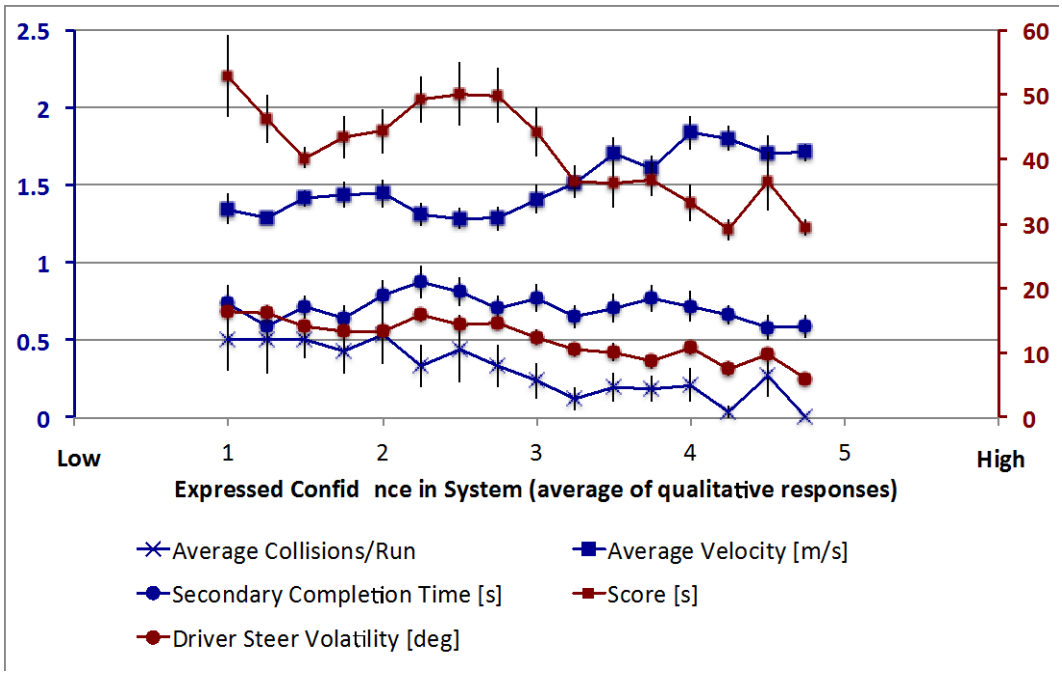


Figure 4.34. Average sentiment of operator survey responses vs. measured performance (means and standard errors for all rounds). Metrics plotted in red correspond to the secondary y-axis

4.4 CONCLUSION

This user study suggests that human performance and confidence in the vehicle teleoperation task can be significantly improved via constraint-based planning and threat-based semi-autonomy. While this shared control approach is effective both with and without driver feedback, results indicate that providing visual and haptic feedback may only marginally improve some objective measures of performance while significantly improving subjective measures of driver confidence. In its complete configuration (full autonomy with feedback), and compared to unassisted teleoperation under the conditions of this study, this semi-autonomy framework eliminates 78% of all collisions while simultaneously increasing average speed by 26%. The 0.096 collisions that continued to occur per trial resulted largely from three sources. First, as mentioned previously, the large LIDAR blind spot required a method for tracking obstacles when they dropped out of view in close proximity to the vehicle. Given poor (~0.5 m) global positioning estimates for the vehicle, this tracking sometimes resulted in misplaced obstacles and

subsequently incorrect placement of MPC constraints. Improved positioning hardware, as well as advances in the accuracy of onboard sensing techniques will reduce the occurrence of collisions caused by sensing uncertainty. Secondly, the row spacing and gate offsets in this course were configured such that beyond a certain (operator-commanded) trajectory and speed, it became impossible for the controller to turn the wheels fast enough to avoid collisions (giving steering rate constraints). The user study shown here did not incorporate the velocity constraints or speed intervention necessary to prevent this. Finally, while the MPC controller was configured to match the estimated parameters of the vehicle and its interactions with the environment (tire friction, etc.), it represents a linearized approximation of the true vehicle dynamics and is thus subject to some model mismatch. This mismatch may also have played a role in failing to avoid the occasional collision.

5

CHAPTER 5: CONCLUSIONS

5.1 CONTRIBUTIONS OF THIS THESIS

This thesis has investigated methods for improved performance of human-controlled systems through the homotopy-based design, and threat-based enforcement of constraints. Shared-adaptive control presents a unique opportunity to reduce or eliminate the effects of human error without unduly sacrificing human autonomy. By exploiting the human's perceptual, judgment, and context-based reasoning in low-threat scenarios and automation's facility for fast, accurate, and repeatable control in high-threat scenarios, this approach ensures that the human operator retains as much control of the system as s/he can safely wield without causing a collision or loss of stability.

Chapter 2 presented two methods for identifying, evaluating the goodness of, and planning constraints to bound a desired path homotopy through a field of obstacles. In the first, constrained Delaunay triangles are used to decompose the environment into a complete set of contiguous cells, through any sequence of which a different path homotopy may pass. A fundamentally new definition of "goodness" was derived to evaluate the desirability of a constraint plan based on the aggregate properties of the topologically equivalent paths it contains. The second constraint-planning method used a more conventional, rectangular grid, dynamic programming, and reachability constraints to identify homotopies and evaluate them based on the length and dynamic feasibility of the optimal path each contained. Both constraint design methods presented in this chapter were shown to improve various measures of the planning and control performance of one or multiple vehicles navigating in an environment with obstacles.

Chapter 3 described a model-based means of assessing the "threat" posed to a vehicle given the position, input, and stability constraints it must satisfy. This chapter also described a threat-based method for allocating control between a human operator and automation to ensure that the vehicle does not violate these constraints. Two new feedback mechanisms were designed specifically for this constraint-based framework to communicate the actions and assessments of the autonomous assistant to the human driver and thereby facilitate "co-situational awareness" through visual and haptic channels. This chapter also presented various simulations investigating the effect of this constraint enforcement technique on a simulated human driver in various environments and in the presence of communication delays. These simulations showed that, when assisted by the controller, the simulated driver is both more measured in its commands, and better able to traverse the obstacle course without hitting obstacles or leaving the safe road surface.

Finally, Chapter 4 presented the results of an extensive, 1,200-trial study of the performance and control preference of 20 drivers teleoperating an unmanned ground vehicle through an outdoor obstacle course. These experiments showed that, across nearly all measures of performance, vehicle control performance and operator confidence improved with the assistance of the shared-adaptive controller and visual and haptic feedback. Additional improvement is expected with the addition of improved sensors and shared-adaptive velocity control.

5.2 FUTURE WORK

Although simulated and experimental results have shown the shared control framework presented in this thesis to be stable even in the presence of time delays and non-binary control allocation between the human and the MPC controller, no formal proof of stability has been presented. Developing such a proof will provide a useful set of conditions for the constraint planner, threat assessor, and intervention method.

Anecdotally, we observe that allocating control authority based on the threat, or objective function tradeoffs made by the MPC controller in order to satisfy constraints, is qualitatively similar to including an input reference (placed at the current driver steering angle and assumed constant through the predicted time horizon) in the objective function

alongside the stability-maximizing term and implementing the control algorithm as a standard, provably-stable, model predictive control algorithm. The evolution of the driver reference assumed by this approach can be predicted using any of a number of approaches. Its likely deviation from the true future driver inputs over time can be considered by hyperbolically discounting penalties on deviations of the optimal predicted control input through time.

REFERENCES

[1] National Highway Traffic Safety Administration, "2010 Motor Vehicle Crashes: Overview," US Department of Transportation, Washington, D.C., Research Note DOT HS 811 552, Feb. 2012.

[2] Defense Manpower Data Center, Statistical Information Analysis Division, "Military Casualty Information: Global War on Terrorism," Data, Analysis, and Programs Division, Jan. 2012.

[3] United States Department of Labor Occupational Safety & Health Administration, "Section 10 - X. Summary of the Final Economic Analysis, including the Regulatory Flexibility Analysis," 1998. [Online]. Available: http://www.osha.gov/pls/oshaweb/owadisp.show_document?p_table=PREAMBLES&p_id=1069. [Accessed: 02-Sep-2010].

[4] C. E. Lathan and M. Tracey, "The effects of operator spatial perception and sensory feedback on human-robot teleoperation performance," *Presence*, vol. 11, no. 4, pp. 368–77, 2002.

[5] J. Carlson and R. R. Murphy, "How UGVs physically fail in the field," *Robotics, IEEE Transactions on*, vol. 21, no. 3, pp. 423–437, 2005.

[6] "Human Error Cited in Most UAV Crashes." [Online]. Available: http://www.military.com/news/article/human-error-cited-in-most-uav-crashes.html. [Accessed: 03-Jan-2011].

[7] Office of the Secretary of Defense, USA, "Unmanned Aerial Vehicles Roadmap 2002 - 2027," Office of the under secretary of defense for acquisition technology and logistics, Washington, DC, USA, Dec. 2002.

[8] T. Dingus, *Human factors design issues for crash avoidance systems*. Mahwah, NJ: Lawrence Erlbaum Associates Inc, US, 1998.

[9] National Highway Traffic Safety Administration, "National Motor Vehicle Crash Causation Survey," US Department of Transportation, Washington, D.C., Report to Congress DOT HS 811 059, Jul. 2008.

[10] J. R. Treat, N. S. Tumbas, S. T. McDonald, D. Shinar, R. D. Hume, R. E. Mayer, R. L. Stansifer, and N. J. Castellan, "Tri-level study of the causes of traffic accidents: final report.," US Department of Transportation - National Highway Traffic Safety Administration, Washington, D.C., DOT-HS-034-3-535-77, 1977.

[11] J. Leonard, J. How, S. Teller, M. Berger, S. Campbell, G. Fiore, L. Fletcher, E. Frazzoli, A. Huang, S. Karaman, O. Koch, Y. Kuwata, D. Moore, E. Olson, S. Peters, J. Teo, R. Truax, M. Walter, D. Barrett, A. Epstein, K. Maheloni, K. Moyer, T. Jones, R. Buckley, M. Antone, R. Galejs, S. Krishnamurthy, and J. Williams, "A perception-driven autonomous urban vehicle," *Journal of Field Robotics*, vol. 25, no. 10, pp. 727–774, 2008.

[12] T. Sattel and T. Brandt, “From robotics to automotive: Lane-keeping and collision avoidance based on elastic bands,” *Vehicle System Dynamics*, vol. 46, no. 7, pp. 597–619, Jul. 2008.

[13] T. B. Sheridan, “Adaptive Automation, Level of Automation, Allocation Authority, Supervisory Control, and Adaptive Control: Distinctions and Modes of Adaptation,” *IEEE Transactions on Systems, Man and Cybernetics, Part A (Systems and Humans)*, vol. 41, no. 4, pp. 662–7, Jul. 2011.

[14] P. Cheng, Z. Shen, and S. M. Lavalle, “RRT-based trajectory design for autonomous automobiles and spacecraft,” in *7th National Conference on Robotics*, Poland, 2001, vol. 11(47), pp. 167–94.

[15] R. Wein, J. P. van den Berg, and D. Halperin, “The visibility-Voronoi complex and its applications,” *Computational Geometry*, vol. 36, no. 1, pp. 66–87, Jan. 2007.

[16] S. Shimoda, Y. Kuroda, and K. Iagnemma, “High-speed navigation of unmanned ground vehicles on uneven terrain using potential fields,” *Robotica*, vol. 25, no. 4, pp. 409–424, 2007.

[17] L. B. Cremean, T. B. Foote, J. H. Gillula, G. H. Hines, D. Kogan, K. L. Kriechbaum, J. C. Lamb, J. Leibs, L. Lindzey, C. E. Rasmussen, A. D. Stewart, J. W. Burdick, and R. M. Murray, “Alice: An information-rich autonomous vehicle for high-speed desert navigation,” *Journal of Field Robotics*, vol. 23, no. 9, pp. 777–810, 2006.

[18] A. Bemporad, M. Morari, V. Dua, and E. N. Pistikopoulos, “The explicit linear quadratic regulator for constrained systems,” *Automatica*, vol. 38, no. 1, pp. 3–20, 2002.

[19] T. Tsukagoshi and H. Wakaumi, “Highly-reliable semi-autonomous vehicle control on lattice lane,” in *Proceedings of the IEEE International Workshop on Intelligent Robots and Systems*, New York, NY, USA, 1990, pp. 731–8.

[20] P. Falcone, H. E. Tseng, F. Borrelli, J. Asgari, and D. Hrovat, “MPC-based yaw and lateral stabilisation via active front steering and braking,” *Vehicle System Dynamics*, vol. 46, no. SUPPL1, pp. 611–628, 2008.

[21] L. Evans, “The dominant role of driver behavior in traffic safety.,” *Am J Public Health*, vol. 86, no. 6, pp. 784–786, Jun. 1996.

[22] J. Jansson, “Collision avoidance theory with application to automotive collision mitigation,” Doctoral Dissertation, Linkoping University, SE-581 83 LINKÖPING Sweden, 2005.

[23] J. Pohl, W. Birk, and L. Westervall, “A driver-distraction-based lane-keeping assistance system,” *Proceedings of the Institution of Mechanical Engineers. Part I: Journal of Systems and Control Engineering*, vol. 221, no. 4, pp. 541–552, 2007.

[24] P. Griffiths and R. B. Gillespie, “Shared control between human and machine: haptic display of automation during manual control of vehicle heading,” in *Proceedings of the 12th international conference on Haptic interfaces for virtual environment and teleoperator systems*, Washington, DC, USA, 2004, pp. 358–366.

[25] J.-F. Liu, J.-H. Wu, and Y.-F. Su, “Development of an interactive lane keeping control system for vehicle,” in *VPPC 2007 - 2007 IEEE Vehicle Power and Propulsion Conference, Sep 9-12 2007*, Piscataway, NJ 08855-1331, United States, 2007, pp. 702–706.

[26] S. Zeng, J. A. Salinger, and P. V. V. Ganesan, “Method and apparatus for rear cross traffic collision avoidance,” U.S. Patent United States Patent Application 2008030666611-Dec-2008.

[27] T. Pilutti, G. Ulsoy, and D. Hrovat, “Vehicle steering intervention through differential braking,” in *Proceedings of the 1995 American Control Conference*, Seattle, WA, USA, 1995, vol. 3, pp. 1667–1671.

[28] A. Widyotriatmo and K.-S. Hong, “Decision making framework for autonomous vehicle navigation,” in *Proceedings of the International Conference on Instrumentation, Control and Information Technology*, Tokyo, Japan, 2008, pp. 1002–1007.

[29] A. Alleyne, “A comparison of alternative obstacle avoidance strategies for vehicle control,” *Vehicle System Dynamics*, vol. 27, no. 5–6, pp. 371–92, Jun. 1997.

[30] M. Netto, J.-M. Blosseville, B. Lusetti, and S. Mammar, “A new robust control system with optimized use of the lane detection data for vehicle full lateral control under strong curvatures,” in *ITSC 2006: 2006 IEEE Intelligent Transportation Systems Conference*, Piscataway, NJ, 2006, pp. 1382–1387.

[31] G. Engelman, J. Ekmark, L. Tellis, M. N. Tarabishy, G. M. Joh, R. A. Trombley, and R. E. Williams, “Threat level identification and quantifying system,” U.S. Patent US 7034668 B225-Apr-2006.

[32] A. van der Horst, “A time-based analysis of road user behavior in normal and critical encounters,” Ph.D. Thesis, Delft University, Delft, The Netherlands, 1990.

[33] R. Kiefer, J. Salinger, and J. Ference, “The Status of NHTSA’s Rear-End Crash Prevention Research Program,” NHTSA, Washington, DC, Final Report 05-0282, Apr. 2005.

[34] R. G. C. Fuller, “Determinants of time headway adopted by truck drivers,” *Ergonomics*, vol. 24, no. 6, pp. 463–474, 1981.

[35] R. Kiefer, D. LeBlanc, M. Palmer, J. Salinger, R. K. Deering, and M. A. Shulman, “Development and validation of functional definitions and evaluation procedures for collision warning/avoidance system,” Crash Avoidance Metrics Partnership (CAMP), Washington, DC, Final Research Report DOT HS 808-964, Aug. 1999.

[36] S. J. Brunson, E. M. Kyle, N. C. Phamdo, and G. R. Preziotti, “Alert Algorithm Development Program NHTSA Rear-End Collision Alert Algorithm,” The Johns Hopkins University Applied Physics Laboratory, Washington, DC, Final Report DOT HS 809 526, Sep. 2002.

[37] A. Polychronopoulos, M. Tsogas, A. Amditis, U. Scheunert, L. Andreone, and F. Tango, “Dynamic situation and threat assessment for collision warning systems:

The EUCLIDE approach," in *Proceedings of the 2004 IEEE Intelligent Vehicles Symposium*, Parma, Italy, 2004, pp. 636–641.

[38] K. Lee and H. Peng, "Evaluation of automotive forward collision warning and collision avoidance algorithms," *Vehicle System Dynamics: International Journal of Vehicle Mechanics and Mobility*, vol. 43, no. 10, p. 735, 2005.

[39] R. Siegwart and I. R. Nourbakhsh, *Introduction to Autonomous Mobile Robots*. The MIT Press, 2004.

[40] M. Buehler, K. Iagnemma, and S. Singh, *The DARPA Urban Challenge*, vol. 56. 2009.

[41] P. Falcone, M. Tufo, F. Borrelli, J. Asgari, and H. E. Tseng, "A linear time varying model predictive control approach to the integrated vehicle dynamics control problem in autonomous systems," in *Proceedings of the 46th IEEE Conference on Decision and Control*, Piscataway, NJ 08855-1331, United States, 2007, pp. 2980–2985.

[42] Y. Kuwata, S. Karaman, J. Teo, E. Frazzoli, J. P. How, and G. Fiore, "Real-time motion planning with applications to autonomous urban driving," *IEEE Transactions on Control Systems Technology*, vol. 17, no. 5, pp. 1105–18, Sep. 2009.

[43] N. C. Tsourveloudis, K. P. Valavanis, and T. Hebert, "Autonomous vehicle navigation utilizing electrostatic potential fields and fuzzy logic," *IEEE Transactions on Robotics and Automation*, vol. 17, no. 4, pp. 490–7, 2001.

[44] P. Falcone, F. Borrelli, H. E. Tseng, J. Asgari, and D. Hrovat, "A hierarchical model predictive control framework for autonomous ground vehicles," in *Proceedings of the 2008 American Control Conference*, Piscataway, NJ 08855-1331, United States, 2008, pp. 3719–3724.

[45] P. Falcone, F. Borrelli, J. Asgari, H. E. Tseng, and D. Hrovat, "Predictive active steering control for autonomous vehicle systems," *IEEE Transactions on Control Systems Technology*, vol. 15, no. 3, pp. 566–580, 2007.

[46] P. Falcone, F. Borrelli, J. Asgari, H. E. Tsengy, and D. Hrovat, "A model predictive control approach for combined braking and steering in autonomous vehicles," in *Proceedings of the 2007 Mediterranean Conference on Control and Automation*, Piscataway, NJ 08855-1331, United States, 2007.

[47] F. Borrelli, P. Falcone, T. Keviczky, J. Asgari, and D. Hrovat, "MPC-based approach to active steering for autonomous vehicle systems," *International Journal of Vehicle Autonomous Systems*, vol. 3, no. 2–4, pp. 265–291, 2005.

[48] R. Mobus and Z. Zomotor, "Constrained optimal control for lateral vehicle guidance," in *Proceedings of the 2005 IEEE Intelligent Vehicles Symposium, 6-8 June 2005*, Piscataway, NJ, USA, 2005, pp. 429–34.

[49] Christopher D. Wickens, Anne S. Mavor, Raja Parasuraman, and James P. McGee, Editors; Panel on Human Factors in Air Traffic Control Automation, National

Research Council, *The Future of Air Traffic Control:Human Operators and Automation*. Washington, D.C.: The National Academies Press, 1998.

[50] P. M. Fitts, M. S. Viteles, N. L. Barr, D. R. Brimhall, G. Finch, E. Gardner, W. F. Grether, W. E. Kellum, and S. S. Stevens, "Human engineering for an effective air-navigation and traffic-control system, and appendixes 1 thru 3," Mar. 1951.

[51] J. C. F. Winter and D. Dodou, "Why the Fitts list has persisted throughout the history of function allocation," *Cognition, Technology & Work*, Aug. 2011.

[52] T. B. Sheridan and R. Parasuraman, "Human-Automation Interaction," *Reviews of Human Factors and Ergonomics*, vol. 1, no. 1, pp. 89 –129, Jun. 2005.

[53] L. Bainbridge, "Ironies of automation," *Automatica*, vol. 19, no. 6, pp. 775–9, Nov. 1983.

[54] R. Oppermann, Ed., *Adaptive user support: ergonomic design of manually and automatically adaptable software*. Hillsdale, NJ, USA: L. Erlbaum Associates Inc., 1994.

[55] C. J. Kahane and J. N. Dang, "The Long-Term Effect of ABS in Passenger Cars and LTVs," National Highway Traffic Safety Administration, NHTSA Technical Report DOT HS 811 182, 2009.

[56] R. Parasuraman and V. Riley, "Humans and automation: use, misuse, disuse, abuse," *Human Factors*, vol. 39, no. 2, pp. 230–53, Jun. 1997.

[57] S. P. Parikh, J. Grassi, V., V. Kumar, and J. Okamoto, J., "Incorporating user inputs in motion planning for a smart wheelchair," in *Proceedings of the 2004 IEEE International Conference on Robotics and Automation*, 2004, vol. 2, pp. 2043 – 2048 Vol.2.

[58] C. Urdiales, M. Fernandez-Carmona, J. M. Peula, R. Annicchiaricco, F. Sandoval, and C. Caltagirone, "Efficiency based modulation for wheelchair driving collaborative control," in *2010 IEEE International Conference on Robotics and Automation (ICRA)*, 2010, pp. 199 –204.

[59] A. D. Dragan and S. S. Srinivasa, "Formalizing Assistive Teleoperation," presented at the Robotics: Science and Systems (RSS), 2012, Sydney, NSW, Australia, 2012.

[60] R. Chipalkatty and M. Egerstedt, "Human-in-the-Loop: Terminal constraint receding horizon control with human inputs," in *Proceedings of the 2010 IEEE International Conference on Robotics and Automation*, Piscataway, NJ, USA, 2010, pp. 2712–17.

[61] D. Vanhooydonck, E. Demeester, A. Hüntemann, J. Philips, G. Vanacker, H. Van Brussel, and M. Nuttin, "Adaptable navigational assistance for intelligent wheelchairs by means of an implicit personalized user model," *Robotics and Autonomous Systems*, vol. 58, no. 8, pp. 963–977, Aug. 2010.

[62] J. J. Gibson and L. E. Crooks, "A Theoretical Field-Analysis of Automobile-Driving," *The American Journal of Psychology*, vol. 51, no. 3, pp. 453–471, Jul. 1938.

[63] R. R. Murphy, “Sensor and information fusion improved vision-based vehicle guidance,” *IEEE Intelligent Systems and their Applications*, vol. 13, no. 6, pp. 49 – 56, Dec. 1998.

[64] T. Y. Shin, S. Y. Kim, J. Y. Choi, K. S. Yoon, and M. H. Lee, “Modified lateral control of an autonomous vehicle by look-ahead and look-down sensing,” *Int.J Automot. Technol.*, vol. 12, no. 1, pp. 103–110, Feb. 2011.

[65] A. Lambert and D. Gruyer, “Safe path planning in an uncertain-configuration space,” in *Proceedings of the IEEE International Conference on Robotics and Automation*, 2003, vol. 3, pp. 4185 – 4190 vol.3.

[66] D. L. Marruedo, J. M. Bravo, T. Alamo, and E. F. Camacho, “Robust MPC of constrained discrete-time nonlinear systems based on uncertain evolution sets: application to a CSTR model,” in *Proceedings of the 2002 International Conference on Control Applications*, 2002, vol. 2, pp. 657– 662 vol.2.

[67] I. Jawhar, N. Mohamed, and Liren Zhang, “Inter-vehicular communication systems, protocols and middleware,” in *Proceedings of the 2010 IEEE International Conference on Networking, Architecture, and Storage*, Los Alamitos, CA, USA, 2010, pp. 282–7.

[68] S. Tsugawa, “Inter-vehicle communications and their applications to intelligent vehicles: an overview,” in *IEEE Intelligent Vehicle Symposium, 2002*, 2002, vol. 2, pp. 564 – 569 vol.2.

[69] R. Brown, *Topology and Groupoids*. BookSurge Publishing, 2006.

[70] J. Munkres, *Topology*, 2nd ed. Prentice Hall, 2000.

[71] O. Brock and O. Khatib, “Real-time replanning in high-dimensional configuration spaces using sets of homotopic paths,” in *Proceedings of the IEEE International Conference on Robotics and Automation*, San Francisco, CA, USA, 2000, vol. 1, pp. 550–555.

[72] J. Hershberger and J. Snoeyink, “Computing minimum length paths of a given homotopy class,” *Computational Geometry: Theory and Applications*, vol. 4, no. 2, pp. 63–97, Jun. 1994.

[73] R. V. Eynde, “Historical evolution of the concept of homotopic paths,” *Arch. Hist. Exact Sci.*, vol. 45, no. 2, pp. 127–188, 1992.

[74] K. D. Jenkins, “Shortest Path Problem in the Plane with Obstacles: A Graph Modeling Approach to Producing Finite Search Lists of Homotopy Classes,” Master’s Thesis, Naval Postgraduate School, Monterey, CA, United States, 1991.

[75] E. Hernandez, M. Carreras, J. Antich, P. Ridao, and A. Ortiz, “A topologically guided path planner for an AUV using homotopy classes,” in *Proceedings of the 2011 IEEE International Conference on Robotics and Automation*, Piscataway, NJ, USA, 2011, pp. 2337–43.

[76] L. Jaillet and T. Simeon, “Path Deformation Roadmaps: Compact Graphs with Useful Cycles for Motion Planning,” *The International Journal of Robotics Research*, vol. 27, no. 11–12, pp. 1175–1188, Nov. 2008.

[77] E. Schmitzberger, J. L. Bouchet, M. Dufaut, D. Wolf, and R. Husson, "Capture of homotopy classes with probabilistic road map," in *2002 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Lausanne, Switzerland, 2002, vol. 3, pp. 2317–2322.

[78] C. Eldershaw, M. Yim, Y. Zhang, K. Roufas, and D. Duff, "Motion planning with narrow C-space passages," in *International Conference on Intelligent Robots and Systems*, Las Vegas, NV, United States, 2003, vol. 2, pp. 1608–1613.

[79] S. H. Kim and R. Bhattacharya, "Motion Planning in Obstacle Rich Environments," *Journal of Aerospace Computing Information and Communication*, vol. 6, no. 7, pp. 433–450, 2009.

[80] D. Leven and M. Sharir, "Planning a purely translational motion for a convex object in two-dimensional space using generalized Voronoi diagrams," *Discrete and Computational Geometry*, vol. 2, no. 1, pp. 9–31, Dec. 1987.

[81] D. Demyen and M. Buro, "Efficient triangulation-based pathfinding," in *Eighteenth Innovative Applications of Artificial Intelligence Conference*, Menlo Park, CA, USA, 2006, pp. 942–947.

[82] R. Geraerts and M. H. Overmars, "The corridor map method: A general framework for real-time high-quality path planning," *Computer Animation and Virtual Worlds*, vol. 18, no. 2, pp. 107–119, 2007.

[83] S. Bhattacharya, V. Kumar, and M. Likhachev, "Search-based path planning with homotopy class constraints," in *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence*, Atlanta, GA, United states, 2010, vol. 2, pp. 1230–1237.

[84] F. Aurenhammer, "Voronoi diagrams--a survey of a fundamental geometric data structure," *ACM Comput. Surv.*, vol. 23, no. 3, pp. 345–405, 1991.

[85] H. Blum, "A Transformation for Extracting New Descriptors of Shape," *Models for the Perception of Speech and Visual Form*, pp. 362–380, 1967.

[86] P. Bhattacharya and M. L. Gavrilova, "Road map-based path planning. Using the Voronoi diagram for a clearance-based shortest path," *IEEE Robotics & Automation Magazine*, vol. 15, no. 2, pp. 58–66, Jun. 2008.

[87] J. Kim, F. Zhang, and M. Egerstedt, "A provably complete exploration strategy by constructing Voronoi diagrams," *Autonomous Robots*, vol. 29, no. 3–4, pp. 367–380, Aug. 2010.

[88] O. Takahashi and R. J. Schilling, "Motion planning in a plane using generalized Voronoi diagrams," *IEEE Transactions on Robotics and Automation*, vol. 5, no. 2, pp. 143–150, 1989.

[89] S. Fortune, "Sweepline Algorithm for Voronoi Diagrams," *Algorithmica (New York)*, vol. 2, no. 2, pp. 153–174, 1987.

[90] O. Setter, M. Sharir, and D. Halperin, "Constructing two-dimensional Voronoi diagrams via divide-and-conquer of envelopes in space," in *2009 Sixth International Symposium on Voronoi Diagrams*, 2009, pp. 43–52.

[91] H. Choset, I. Konukseven, and J. Burdick, “Mobile robot navigation: Issues in implementing the generalized Voronoi graph in the plane,” in *Proceedings of the 1996 IEEE/SICE/RSJ International Conference on Multisensor Fusion and Integration for Intelligent Systems*, 1996, pp. 241–248.

[92] S. Garrido, M. Abderrahim, and L. Moreno, “Path Planning and navigation using Voronoi diagram and Fast Marching,” in *8th International IFAC Symposium on Robot Control*, 2006, vol. 8.

[93] A. Sud, *Efficient Computation of Discrete Voronoi Diagram and Homotopy-preserving Simplified Medial Axis of a Three-dimensional Polyhedron*. University of North Carolina at Chapel Hill, 2006.

[94] C. O’Dunlaing and C. K. Yap, “A `retraction’ method for planning the motion of a disc,” *Journal of Algorithms*, vol. 6, no. 1, pp. 104–111, Mar. 1985.

[95] F. P. Preparata and M. I. Shamos, *Computational Geometry: An Introduction*. Springer-Verlag, 1985.

[96] L. P. Chew, “Constrained Delaunay triangulations,” in *Proceedings of the third annual symposium on Computational geometry*, New York, NY, USA, 1987, pp. 215–222.

[97] B. Delaunay, “Sur la sphere vide,” *Izvestia Akademii Nauk SSSR, Otdelenie Matematicheskikh i Estestvennykh Nauk*, vol. 7, pp. 793–800, 1934.

[98] Yao-hong Qu, Quan Pan, and Jian-guo Yan, “Flight path planning of UAV based on heuristically search and genetic algorithms,” in *IECON 2005. Thirty-First Annual Conference of the IEEE Industrial Electronics Society*, 2005, p. 5 pp.

[99] H. Yan, H. Wang, Y. Chen, and G. Dai, “Path planning based on constrained delaunay triangulation,” in *Proceedings of the 7th World Congress on Intelligent Control and Automation*, Chongqing, China, 2008, pp. 5162–5167.

[100] T. C. Biedl, E. D. Demaine, S. Lazard, S. M. Robbins, M. A. Soss, A. Aggarwal, and C. P. Rangan, “Convexifying Monotone Polygons,” *Proceedings of the 10th International Symposium on Algorithms and Computation*, pp. 415–424, 1999.

[101] W. Lenhart and S. Whitesides, “Reconfiguring closed polygonal chains in Euclidean <i>d</i> -space,” *Discrete & Computational Geometry*, vol. 13, no. 1, pp. 123–140, 1995.

[102] P. Reist and R. Tedrake, “Simulation-based LQR-trees with input and state constraints,” in *2010 IEEE International Conference on Robotics and Automation*, Piscataway, NJ, USA, 2010, pp. 5504–10.

[103] J. Ackermann, J. Guldner, and V. I. Utkin, “Robust nonlinear control approach to automatic path tracking of a car,” in *Proceedings of the International Conference on Control. Part 1 (of 2)*, Stevenage, Engl, 1994, vol. 1, pp. 196–201.

[104] R. Tedrake, I. R. Manchester, M. Tobeekie, and J. W. Roberts, “LQR-trees: Feedback motion planning via sums-of-squares verification,” *International Journal of Robotics Research*, vol. 29, no. 8, pp. 1038–52, Jul. 2010.

[105] D. P. Bertsekas and I. B. Rhodes, "On the minimax reachability of target sets and target tubes," *Automatica*, vol. 7, no. 2, pp. 233–47, Mar. 1971.

[106] D. Fox, W. Burgard, and S. Thrun, "The dynamic window approach to collision avoidance," *IEEE Robotics & Automation Magazine*, vol. 4, no. 1, pp. 23–33, Mar. 1997.

[107] D. E. Chang, S. C. Shadden, J. E. Marsden, and R. Olfati-Saber, "Collision avoidance for multiple agent systems," in *42nd IEEE International Conference on Decision and Control*, Piscataway, NJ, USA, 2003, vol. Vol.1, pp. 539–43.

[108] J. Ackermann, "Robust decoupling, ideal steering dynamics and yaw stabilization of 4WS cars," *Automatica*, vol. 30, no. 11, pp. 1761–8, Nov. 1994.

[109] Y. Zhang, E. K. Antonsson, and K. Grote, "A new threat assessment measure for collision avoidance systems," in *Proceedings of the 2006 IEEE Intelligent Transportation Systems Conference*, Toronto, ON, Canada, 2006, pp. 968–975.

[110] I. M. Mitchell, A. M. Bayen, and C. J. Tomlin, "A time-dependent Hamilton-Jacobi formulation of reachable sets for continuous dynamic games," *IEEE Transactions on Automatic Control*, vol. 50, no. 7, pp. 947–57, Jul. 2005.

[111] C. S. Tan, R. Sutton, and J. Chudley, "An integrated collision avoidance system for autonomous underwater vehicles," *International Journal of Control*, vol. 80, no. 7, pp. 1027–49, Jul. 2007.

[112] H. B. Pacejka, *Tire and Vehicle Dynamics*, 2nd ed. Warrendale, PA: Society of Automotive Engineers, 2006.

[113] C. E. Garcia, D. M. Prett, and M. Morari, "Model predictive control: theory and practice-a survey," *Automatica*, vol. 25, no. 3, pp. 335–48, May 1989.

[114] E. F. Camacho and C. Bordons, *Model Predictive Control*, 2nd ed. New York, NY, USA: Springer, 2004.

[115] M. A. Lelic and M. B. Zarrop, "Generalized pole-placement self-tuning controller: Part 1. Basic algorithm," *International Journal of Control*, vol. 46, no. 2, pp. 547–568, 1987.

[116] M. A. Lelic and P. E. Wellstead, "Generalized pole-placement self-tuning controller: Part 2. Application to robot manipulator control," *International Journal of Control*, vol. 46, no. 2, pp. 569–601, 1987.

[117] T. Besselmann and M. Morari, "Hybrid Parameter-Varying Model Predictive Control for Autonomous Vehicle Steering," *European Journal of Control*, vol. 14, no. 5, pp. 418–431, May 2008.

[118] B. Kim, D. Necsulescu, and J. Sasiadek, "Model predictive control of an autonomous vehicle," in *Proceedings of the 2001 IEEE/ASME International Conference on Advanced Intelligent Mechatronics Proceedings*, 2001, vol. 2, pp. 1279–1284.

[119] L. D. Baskar, B. De Schutter, and H. Hellendoorn, "Model predictive control for intelligent speed adaptation in intelligent vehicle highway systems," in *Proceedings*

of the 17th IEEE International Conference on Control Applications, New York, NY 10016-5997, United States, 2008, pp. 468–473.

[120] P. Doganis, E. Aggelogiannaki, and H. Sarimveis, "A combined model predictive control and time series forecasting framework for production-inventory systems," *International Journal of Production Research*, vol. 46, no. 24, pp. 6841–6853, 2008.

[121] M. Mahfouf and D. A. Linkens, "Non-linear generalized predictive control (NLGPC) applied to muscle relaxant anaesthesia," *International Journal of Control*, vol. 71, no. 2, pp. 239–257, 1998.

[122] S. J. Anderson, "A Unified Framework for Trajectory Planning, Threat Assessment, and Semi-Autonomous Control of Passenger Vehicles," Master's Thesis, Massachusetts Institute of Technology, Cambridge, MA, 2009.

[123] S. J. Anderson, S. C. Peters, T. E. Pilutti, and K. Iagnemma, "An Optimal-Control-Based Framework for Trajectory Planning, Threat Assessment, and Semi-Autonomous Control of Passenger Vehicles in Hazard Avoidance Scenarios," *IJVAS*, vol. 8, no. 2–4, pp. 190–216, 2010.

[124] S. J. Anderson, S. C. Peters, T. E. Pilutti, and K. D. Iagnemma, "Design and Development of an Optimal-Control-Based Framework for Trajectory Planning, Threat Assessment, and Semi-Autonomous Control of Passenger Vehicles in Hazard Avoidance Scenarios," in *Proceedings of the 14th International Symposium on Robotics Research*, Lucerne, Switzerland, 2009.

[125] S. J. Anderson, S. C. Peters, K. D. Iagnemma, and T. E. Pilutti, "A unified approach to semi-autonomous control of passenger vehicles in hazard avoidance scenarios," in *IEEE International Conference on Systems, Man and Cybernetics*, 2009, pp. 2032–2037.

[126] D. B. Kaber, J. M. Riley, R. Zhou, and J. Draper, "Effects of Visual Interface Design, and Control Mode and Latency on Performance, Telepresence and Workload in a Teleoperation Task," *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, vol. 44, no. 5, pp. 503–506, Jul. 2000.

[127] T. Brandt, T. Sattel, and M. Bohm, "Combining haptic human-machine interaction with predictive path planning for lane-keeping and collision avoidance systems," in *2007 IEEE Intelligent Vehicles Symposium*, Piscataway, NJ 08855-1331, United States, 2007, pp. 582–587.

[128] J. Y. C. Chen, E. C. Haas, and M. J. Barnes, "Human Performance Issues and User Interface Design for Teleoperated Robots," *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, vol. 37, no. 6, pp. 1231–1245, 2007.

[129] D. Bertsekas, "Convergence of discretization procedures in dynamic programming," *IEEE Transactions on Automatic Control*, vol. 20, no. 3, pp. 415 – 419, Jun. 1975.

[130] R. Munos and A. Moore, "Variable resolution discretization in optimal control," *Machine Learning*, vol. 49, no. 2–3, pp. 291–323, Nov. 2002.

A

APPENDIX A: ADAMS MODEL PARAMETERS

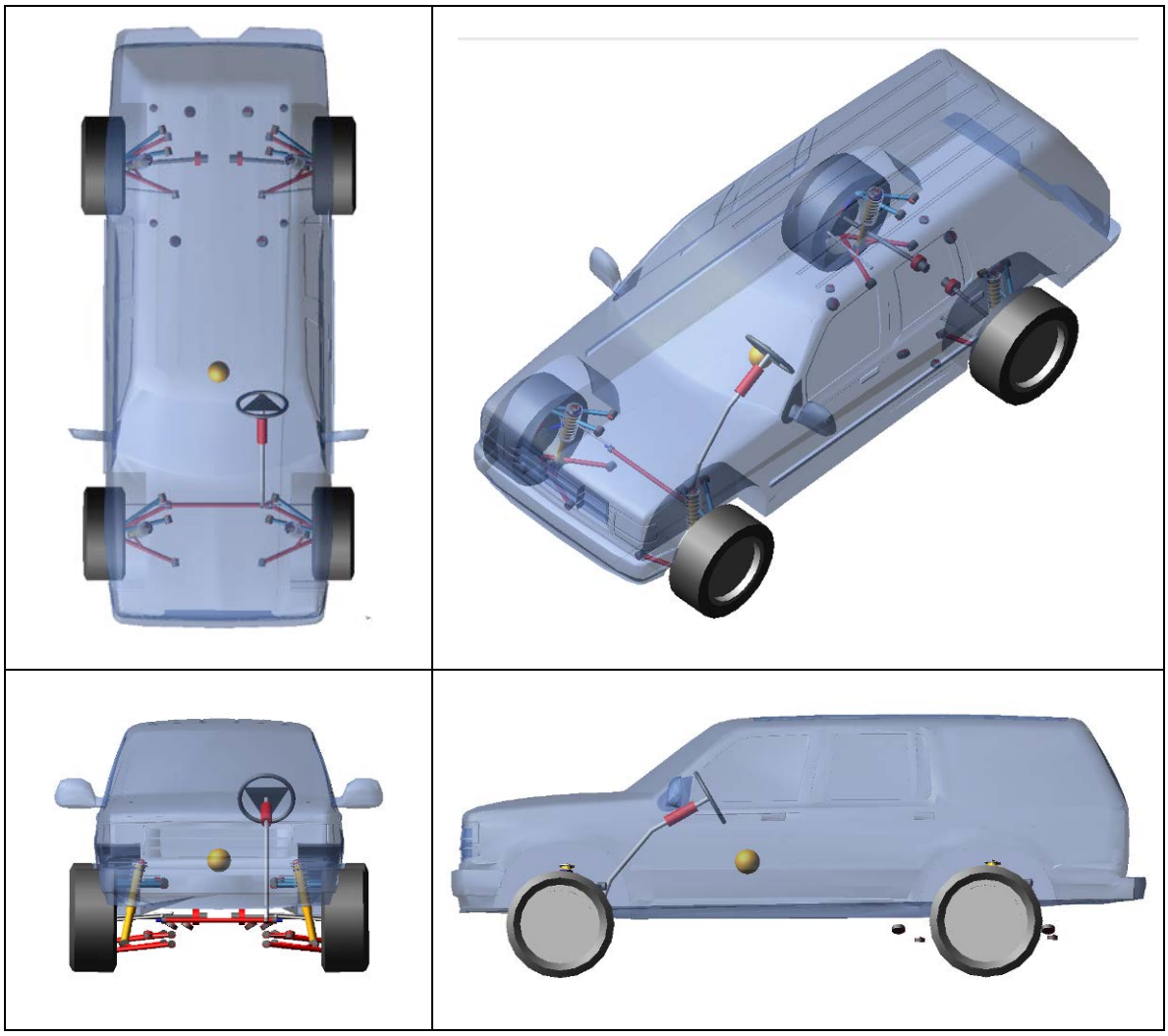


Figure A.1. Model of vehicle used in MSC Adams simulations

Table A.1. Plant model parameters for Adams vehicle model

Parameter	Value [units]
Total mass	**2450 [kg]**
Body mass	2210 [kg]
Unsprung mass	240 [kg]
Wheel mass	60 [kg]
Body roll inertia	1240 [kg·m^2]
Body gyroscopic inertia	0 [kg·m^2]
Wheel gyroscopic inertia	0.2 [kg·m^2]
Measurements	
Wheelbase	2.85 [m]
Track width	1.62 [m]
C.G. height	0.76 [m]
C.G. longitudinal distance from front wheels	1.07 [m]
Wheel diameter	0.79 [m]
Tire full width	0.24 [m]
Suspension and tire stiffness	
Suspension spring stiffness	40,000 [N/m]
Suspension roll stiffness	3700 [N·m/deg]
Suspension damping	5,300 [N·s/m]
Tire vertical stiffness	250,000 N/m
Tire cornering stiffness	1200 N/deg (F_z = 6000 N)
Steering wheel ratio	**35 deg/deg**
Max steer rate	30 [deg/s]
Steering range	±30 [deg]

B

APPENDIX B: UTILITY VEHICLE PARAMETERS

Figure B.1. Kawasaki 4010 Mule used in experimental user studies

Table B.1. Inertial, stiffness, and size parameters of the experimental platform

Parameter	Value [units]
Total mass	**2450 [kg]**
Body mass	842 [kg]
Unsprung mass	160 [kg]
Wheel mass	60 [kg]
Body roll inertia	253 [kg·m^2]
Measurements	
Wheelbase	1.87 [m]
Track width F/R	1.16/1.18 [m]
C.G. height	0.74 [m]
C.G. longitudinal distance from front wheels	1.01 [m]
C.G. longitudinal distance from rear wheels	0.86 [m]
Wheel diameter	0.58 [m]
Tire full width	0.28 [m]
Suspension and tire stiffness	
Suspension spring stiffness	40,000 [N/m]
Suspension roll stiffness	3700 [N·m/deg]
Suspension damping	5,300 [N·s/m]
Tire vertical stiffness	92,000 [N/m]
Tire cornering stiffness (estimated)	200 [N/deg]
Steering wheel ratio	**14 deg/deg**
Max steer rate	31 [deg/s]
Steering range	±33 [deg]

C

APPENDIX C: DETAILED USER STUDY RESULTS

Table C.1. Main effects of control (unshared vs. shared) and feedback (off vs. on) on qualitative survey responses

	Control				Feedback			
	Unshared	**Shared**	**% Change**	***p* Value**	**Off**	**On**	**% Change**	***p* Value**
How easy was it to navigate the course without hitting barrels?	2.8	4.0	43	<0.0001	3.2	3.6	12	<0.001
How fast did you feel comfortable traveling?	2.5	3.5	38	<0.0001	2.9	3.2	12	<0.01
How much control did you feel you had over the vehicle's behavior?	3.0	3.4	12	<0.01	3.1	3.3	8.0	0.04
How confident were you that the system would do the right thing?	2.6	3.7	44	<0.0001	3.0	3.3	7.0	0.10

Table C.2. Main effects of control (unshared vs. shared) and feedback (off vs. on) on quantitative performance metrics

	Control				Feedback			
	Unshared	**Shared**	**% Change**	***p* Value**	**Off**	**On**	**% Change**	***p* Value**
Collisions/Run	0.41	0.11	-72	<0.0001	0.28	0.25	-13	0.34
Brushes/Run	0.21	0.12	-44	<0.001	0.14	0.19	33	0.08
Average Velocity [m/s]	1.36	1.72	26	<0.0001	1.54	1.54	-0.6	0.72
Completion Time [s]	46.9	35.4	-25	<0.0001	41.1	41.2	0.2	0.89
Driver Steer Volatility [°]	14.6	8.69	-41	<0.0001	11.2	12.2	8.9	<0.001
Vehicle Steer Volatility [°]	14.6	11.4	-22	<0.0001	12.9	13.2	2.6	0.21
True Positive Rate	0.78	0.80	2.6	0.38	0.78	0.79	1.0	0.67
False Positive Rate	0.03	0.05	70	0.01	0.04	0.04	11	0.72
Secondary Task Reaction Time [s]	0.75	0.69	-8.7	0.01	0.70	0.74	5.8	0.12
Performance Score [s]	47.2	33.3	-30	<0.0001	40.3	40.3	0.0	0.85

D

APPENDIX D: REACHABLE SETS AND DISCRETIZATION

Many discretization and function approximation techniques have been presented in the literature to improve the computational efficiency of DP implementations [129], [130]. In this work, a discrete state, continuous action implementation was used to provide geometrically-uniform coverage of the road and shoulder surfaces and facilitate fast, closed-form solutions of the lateral acceleration inputs required to transition between states (via the constant-radius turns described above). Nonholonomic constraints require interpolation of the yaw state, which was rounded to the nearest $\Delta\psi = 1°$.

To further improve computational efficiency, state transitions are constrained in this work to remain within the friction-bounded reachable sets described by (2.30) and (2.35) on flat surfaces and (2.49) and (2.35) on sloped surfaces. Note that on sloped surfaces, acceleration due to gravity both scales (by $\cos\theta$) and shifts (by $g\sin\theta\cos\varphi$) the reachable set. For a given discretization in Δx, the reachable set in Δy and ψ are given by (2.33) and (2.35), respectively. At lower velocities ($V <$ ~7.5 m/s for the vehicle length, vehicle mass, and surface friction considered here), steering angle constraints ($|\delta| \leq 30°$) dictate available state transitions, since the turns they allow require less tire friction than what the road provides. At higher speeds like those considered here, tire friction becomes the acting constraint in reachable set calculations.

Figure D.1 illustrates the set of states at $x = 6$m reachable from an initial condition $x = y = 0$ and $\psi = \pi/6$.

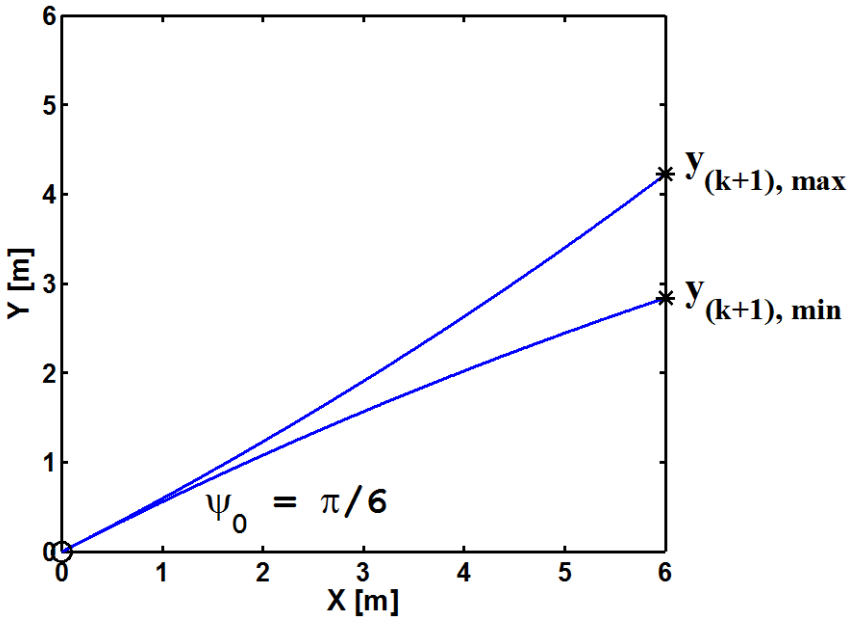
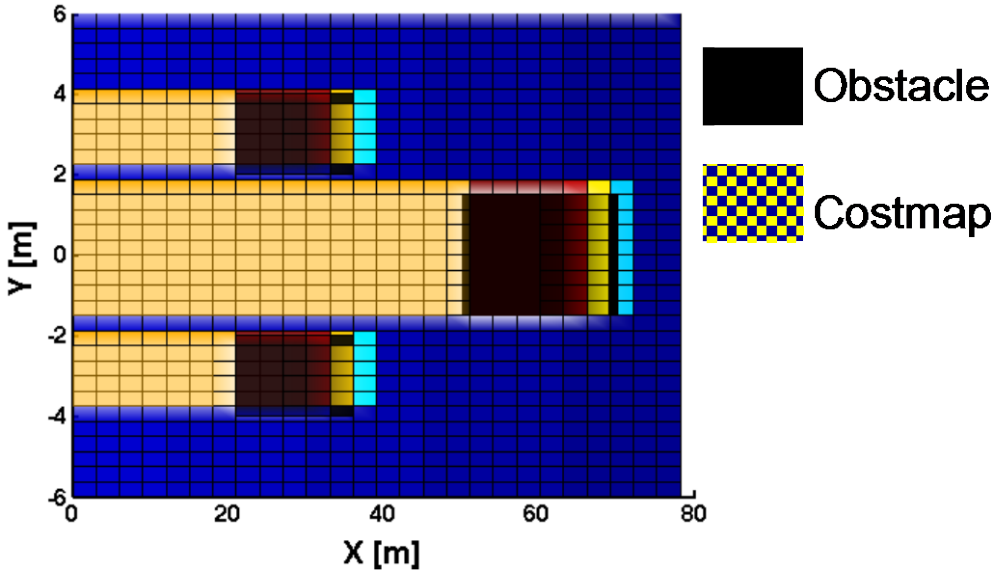


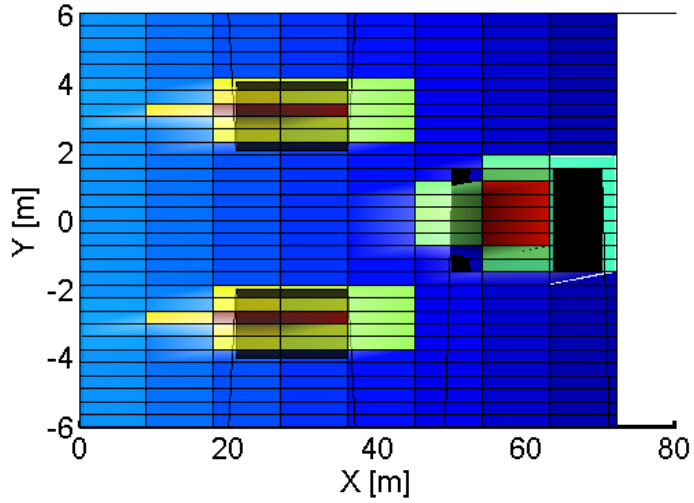
Figure D.1. Illustration of an acceleration-bounded reachable set for V=20 m/s and $\psi_0 = \pi/6$

As illustrated in Figure D.1, nonholonomic vehicle constraints, together with the limitations on feasible state transitions, limit the granularity with which the state space may be discretized in x for a given ψ_0 before $\Delta y_{(k+1),max} = y_{(k+1),max} - y_{(k+1),min}$ is less than the Δy discretization (or $\Delta y/2$ if the reachable state is rounded to the nearest y_{k+1}). This range of reachable y states becomes important in the continuous state implementation used here since acceleration inputs exceeding their friction-limited bounds are not considered. Overly-fine Δx discretization for a given Δy grid may thereby lead to states (such as those with a low initial yaw angle ψ_k) from whose y position the vehicle may not escape.

Figure D.2 illustrates this effect on a (ψ=0) slice of the costmap. Notice that both simulations use the same discretization in y and different discretizations in x (one three times sparser than the other).



(a)



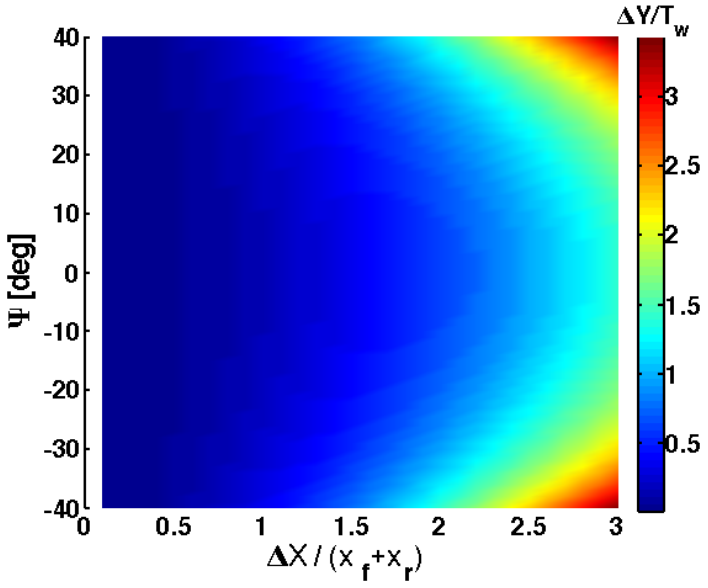
(b)

Figure D.2. Illustration of a ψ=0 slice of the DP costmap for a) small Δx / Δy and b) large Δx / Δy.

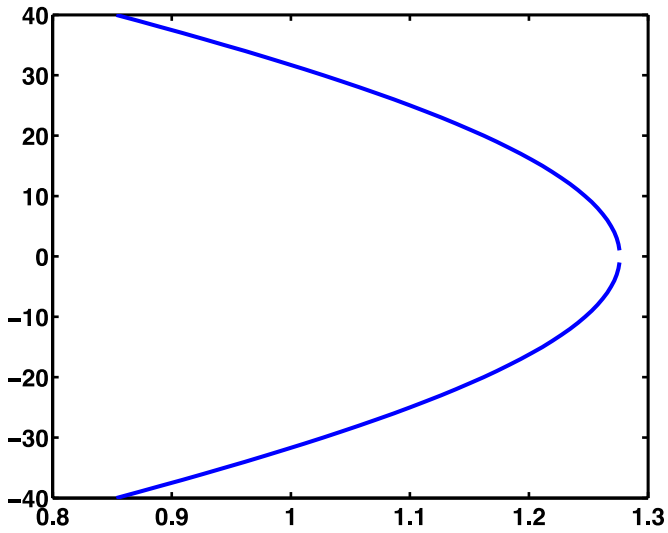
Notice that when reachability constraints are imposed, small $\Delta x/\Delta y$ causes the cost "shadow" cast by obstacles to be much longer, as paths satisfying vehicle dynamic constraints cannot complete a full step in y. The minimum discretization in x necessary to allow a vertical spread of at least Δy given an initial yaw angle ψ and acceleration constraint a_{max} can be described by

$$\Delta x \geq \frac{\left[V^2 \sin(2\psi) - a_{\max}\Delta y \sin\psi\right]\sqrt{\dfrac{a_{\max}\Delta y\left(4V^2\cos\psi - a_{\max}\Delta y\right)}{4V^4 - 4a_{\max}\Delta y V^2\cos\psi + {a_{\max}}^2\Delta y^2}}}{2a_{\max}\sin\psi} \qquad \textbf{(D.1)}$$

Figure D.3 a) shows the range of reachable Δy (normalized by vehicle width T_w) given a discretized Δx (normalized by wheelbase length x_f and x_r) and initial ψ. Figure D.3 b) shows the minimum value of Δx for which the reachability set will allow lateral transitions of ¼ vehicle width. As these figures show, small initial yaw angles set the lower limit on Δx.



(a)



(b)

Figure D.3. Effect of rectangular discretization granularity on reachable states

This result allows one to choose a desired grid resolution in y (based on vehicle width, obstacle density, etc.) and calculate (for a given velocity and acceleration constraint) a suitable grid resolution in x. In the simulation results shown in Section 2.3, Δx was set at or above the critical value.