# Hazard Avoidance for High-Speed Rough-Terrain Unmanned Ground Vehicles

by

Matthew J. Spenko

Bachelor of Science, Mechanical Engineering
Northwestern University, 1999

Master of Science, Mechanical Engineering
Massachusetts Institute of Technology, 2001

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

June 2005





Signature of Author .....

Department of Mechanical Engineering
April 29, 2005

Certified by .....

Steven Dubowsky
Professor of Mechanical Engineering
Thesis Supervisor and Committee Chairman

Accepted by .....

Professor Lallit Anand
Chairman, Committee on Graduate Studies

# Hazard Avoidance for High-Speed Rough-Terrain Unmanned Ground Vehicles

by

Matthew J. Spenko

Submitted to the Department of Mechanical Engineering
on April 29, 2005, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy in Mechanical Engineering

ABSTRACT

High-speed unmanned ground vehicles have important applications in rough-terrain. In these applications unexpected and dangerous situations can occur that require rapid hazard avoidance maneuvers. At high speeds, there is limited time to perform navigation and hazard avoidance calculations based on detailed vehicle and terrain models. Furthermore, detailed models often do not accurately predict the robot's performance due to model parameter and sensor uncertainty.

This thesis presents the development and analysis of a novel method for high speed navigation and hazard avoidance. The method is based on the two dimensional "trajectory space," which is a compact model-based representation of a robot's dynamic performance limits on natural terrain. This method allows a vehicle to perform dynamically feasible hazard avoidance maneuvers in a computationally efficient manner. This thesis also presents a novel method for trajectory replanning, based on a "curvature matching" technique. This method quickly generates a path connects the end of the path generated by a hazard avoidance maneuver to the nominal desired path.

Simulation and experimental results with a small gasoline-powered high-speed unmanned ground vehicle verify the effectiveness of these algorithms. The experimental results demonstrate the ability of the algorithm to account for multiple hazards, varying terrain inclination, and terrain roughness. The experimental vehicle attained speeds of 8 m/s (18 mph) on flat and sloped terrain and 7 m/s (16 mph) on rough terrain.

Thesis Supervisor: Steven Dubowsky, Professor of Mechanical Engineering

# ACKNOWLEDGEMENTS

I would like to express my gratitude first and foremost to my advisor, Professor Steven Dubowsky, for his support and guidance throughout my career at the Massachusetts Institute of Technology. His ability to always "see the big picture" and to constantly challenge my work has proven invaluable. I also thank Dr. Karl Iagnemma. He has helped me immensely with both the formulation of my work and my writing. I thank my two committee members, Professor Eric Feron and Professor John Leonard. Professor Feron's enthusiasm and Professor Leonard's tough questioning have both been instrumental to the development of this thesis.

I have been fortunate to work with a group of exceptional individuals in the Field and Space Robotics Laboratory over the past few years. I extend my deepest gratitude to Professor Yoji Kuroda. Without his help I would have never finished the experimental work presented in this thesis. I also thank Professor Guillaume Morel and Dariusz Golda for their help in fabrication and programming of the experimental system. Lastly I thank Matt Lichter for his friendship over the past five and a half years.

So much of my time at MIT has dealt with my research, but as time passes, my memories of life here will most likely be less of my work and more of the friendships that I developed. I am sure that coaching little league with Ben Paxton, snowboarding with Max Berniker, and driving cars that we could never afford with Bryan Crane will be some of the memories that last.

I also thank my parents for their ever present support in my career. Their emphasis on education has been one of the driving forces in my decision to pursue a Ph.D. Last I thank my wife, Elizabeth, for her continual love, encouragement, and support. Many students have someone who supports them when they return home; few have someone who spends her Sunday afternoons hauling test equipment outside in below freezing weather.

The research in this thesis was supported by the U.S. Army Tank-automotive and Armaments Command and the Defense Advanced Research Projects Agency.

# CONTENTS

# FIGURES

# TABLES

# SYMBOLS

$a$ ..... Acceleration
$A$ ..... Sideslip space
$a_b$ ..... Deceleration due to braking
$a_{lateral}$ ..... Vehicle lateral acceleration
$a_p$ ..... Acceleration due to the powertrain
$A_r$ ..... Reference area of vehicle
$B$ ..... Rollover space
$C$ ..... Steering space
$C_k$ ..... Tire cornering stiffness
$d$ ..... One-half vehicle track width
$D$ ..... Drive train space
$d_{control}$ ..... Maximum lateral path tracking error
$d_{hazard}$ ..... Euclidian distance from vehicle to hazard
$d_l$ ..... Lateral distance from center of mass to left wheels
$d_r$ ..... Lateral distance from center of mass to right wheels
$d_{sensor}$ ..... Vehicle position sensor error
$d_{susp}$ ..... Total deflection of suspension member
$d_t$ ..... Fractal grid spacing
$d_w$ ..... Vehicle wheel diameter
$e_l$ ..... Distance from a vehicle to point on the path
$e_{lat}$ ..... Lateral error for curvature matching method
$e_{lon}$ ..... Longitudinal error for curvature matching method
$e_{total}$ ..... Acceptable threshold for position error in feedback control method
$e_\theta$ ..... Heading error
$F_{drag}$ ..... Force due to aerodynamic drag
$F_{f,r}$ ..... Lateral force acting on front (rear) tires
$F_{powertrain}$ ..... Force generated by vehicle power train
$F_{rolling}$ ..... Force due to rolling resistance
$f_{x,y}$ ..... Spatial terrain frequency in x (y) direction
$g$ ..... Acceleration due to gravity
$G$ ..... Power train transmission gear ratio
$h$ ..... Height of vehicle center of mass
$h_s$ ..... Distance from center of mass to suspension instantaneous center of rotation
$h_t$ ..... Fractal terrain height scale
$h_u$ ..... Distance from ground to suspension instantaneous center of rotation
$K_{1,2}$ ..... Trajectory space weighting gains
$K_{cable}$ ..... Rotational spring constant used in ARTEmiS inertial measurements
$K_{d\_\theta}$ ..... Heading error derivative gain
$K_{p\_l}$ ..... Lateral error proportional gain
$K_{p\_\theta}$ ..... Heading error proportional gain
$K_s$ ..... Vehicle suspension roll stiffness
$L$ ..... Vehicle wheelbase

$l_{f,r}$ ..... Distance from center of mass to front (rear) axle
$l_{susp}$ ..... Distance between suspension members
$m$ ..... Vehicle mass
$m_s$ ..... Vehicle sprung mass
$m_u$ ..... Vehicle unsprung mass
$N_{f,r}$ ..... Normal force acting on the front (rear) tires
$P$ ..... Power output of the powertrain
$r$ ..... Vehicle wheel radius
s ..... Path length
$S_v$ ..... Power spectral density
$T$ ..... Engine torque
$t$ ..... Time
$t_{comp}$ ..... Time to process sensor information and compute a new path
$v$ ..... Velocity
$v_{1,2}$ ..... Velocity of right (left) track or wheel of skid-steered vehicle
$\Gamma$ ..... Dynamic trajectory space
$\Lambda$ ..... Reachable trajectory space
$\Theta$ ..... Admissible trajectory space
$\Omega$ ..... Hazard trajectory space
Z ..... Total admissible trajectory space
$\alpha_{f,r}$ ..... Sideslip angle of front, rear tire
$\beta$ ..... Deflection angle caused by suspension compliance
$\chi$ ..... Maneuver
$\chi_h$ ..... Hazard Avoidance Maneuver
$\chi_r$ ..... Path resumption maneuver
$\delta$ ..... Steering angle of a front-steered vehicle
$\varepsilon$ ..... Acceptable threshold for position error for curvature matching method
$\varepsilon_\kappa$ ..... Heading tolerance in curvature matching method
$\phi$ ..... Vehicle or terrain roll angle
$\gamma$ ..... Deflection angle cause by tire compliance
$\varphi$ ..... Vehicle or terrain pitch angle
$\kappa$ ..... Path curvature
$\mu$ ..... Tire/terrain traction coefficient
$\theta$ ..... Vehicle or terrain yaw angle
$\rho$ ..... Density of air
$\tau$ ..... Trajectory
$\tau_{nominal}$ ..... Nominal trajectory
$\tau_r$ ..... Reachable trajectory
$\varpi$ ..... Terrain roughness (fractal dimension)
$\omega$ ..... Engine angular velocity

# CHAPTER 1

# INTRODUCTION

Unmanned ground vehicles (UGVs) have important military, reconnaissance, surveillance, and material transportation applications [31,86]. Currently the majority of UGVs operate at slow speeds over relatively benign terrain. Many potential applications would require a UGV capable of moving safely at high speeds through uneven, rough terrain with various surface compositions.

There are many advantages of using UGVs instead of human operated vehicles. UGVs are capable of traversing hazardous environments without placing humans in danger [89]. In a military setting UGVs reduce soldier presence on the battlefield, which can result in fewer casualties [13]. UGVs can be useful in civilian search and rescue and disaster relief situations. They can enhance productivity by reducing human workload and providing increased sensing capabilities. They can augment remotely located human operators by delivering additional sensors, computational power, and physical manipulation. Additionally, UGVs have the potential to perform more repeatable field operations than human driven vehicles.

High-speed UGVs are capable of reaching their target location in less time than their low-speed counterparts. This may be useful for equipment delivery. Higher speeds may also allow UGVs to increase their reconnaissance effectiveness by traversing more terrain in a fixed amount of time. UGVs could also benefit from increased operating speeds by spending less time in unsafe situations.

Current outdoor UGVs are limited to traversing benign terrain or following roads. This limits the UGV's effectiveness. The ability to traverse natural, rough, outdoor terrain expands the potential applications of UGVs by allowing them to operate in a broader range of environments and terrain conditions. One major drive in this area has been the recent introduction of the DARPA Grand Challenge, which was designed to encourage innovation in high-speed outdoor UGV traversal [17]. At the time of writing no UGV entered in the DARPA Grand Challenge has completed the event.

Operating at high speeds in unstructured terrain introduces difficult challenges. Often a UGV is directed to follow a pre-planned path (or navigate through pre-defined waypoints) designated by an off-line mission-level planning algorithm. However, in natural terrain at high speeds, it is likely that dangerous and unexpected situations will occur that were not foreseen by the high-level planning methods. These may be the result of outdated topographical map data, unidentified hazards due to sensor limitations or errors, or unanticipated physical terrain conditions. In these situations a UGV must quickly execute an emergency maneuver that allows it to safely avoid an impending hazard. Despite increasing computing power, at high speeds there is little time to

perform navigation based on detailed dynamic vehicle and terrain models. Furthermore, it is difficult to accurately model complex vehicle/terrain interactions due to uncertainty in the profile and nature of the terrain surface.

This thesis investigates some of the challenges that arise from traversing natural terrain at high speeds. Specifically this thesis addresses hazard avoidance in emergency situations. It must be noted that "high speed" is a function of vehicle geometry and terrain. What is considered fast for a small vehicle might be considered slow for a large vehicle. Similarly, terrain that appears to be rough and challenging for a small vehicle may be considered gentle and benign for a large vehicle. For these reasons, "high-speed" is loosely defined here as speeds that excite vehicle dynamics such as rollover, ballistic motion, sideslip, and wheel slip.

## 1.1 Problem Formulation and Key Assumptions

Here the problem of high speed UGV hazard avoidance in natural terrain is defined, and key assumptions are presented.

The challenge is to devise an algorithm that rapidly plans a dynamically feasible path that allows a UGV to avoid unexpected hazards. A dynamically feasible path is one that does not induce rollover or excessive sideslip and considers the UGV's initial steering angle and velocity. The algorithm must consider vehicle properties such as steering kinematics, steering velocity, acceleration, and braking limits, and terrain properties such as traction coefficient, roughness, and inclination (see Figure 1.1). The terrain may be either flat or undulating and either smooth or rough.

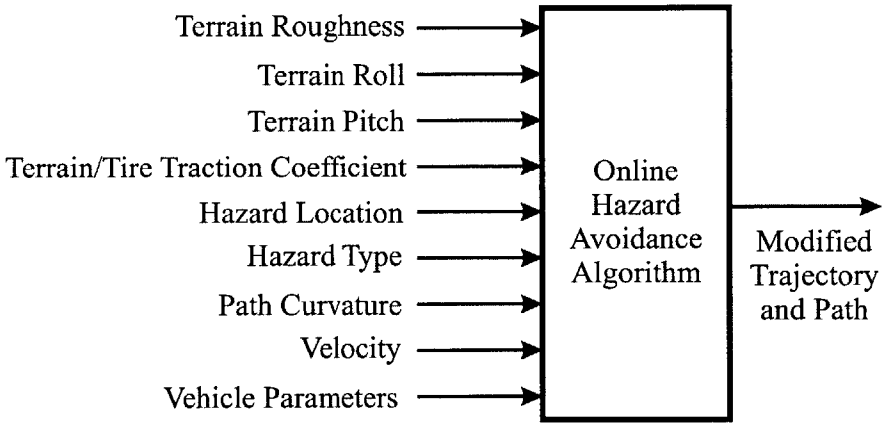


**Figure 1.1: Algorithm outline**

In this thesis a UGV is assumed to be following a nominal pre-planned path, $\mathbf{x}_{nominal}(s)=(x(s),y(s))$, $s\in[s_0,s_f]$. Associated with that path is a trajectory, which is defined here as comprising of a vehicle's velocity and path curvature as a function of path length, $\tau_{nominal}(s)=(v(s),\kappa(s))$. There exists a unique mapping from a vehicle's trajectory to its path given the vehicle's initial curvature, heading, and position.

After a hazard avoidance algorithm generates a dynamically feasible path to avoid a hazard, the algorithm must create a path from the final point of the hazard avoidance path back to the nominal desired path. This path must be created before the vehicle completes its traverse of the hazard avoidance path.

Hazards are defined as discrete objects or terrain features that significantly impede or halt UGV motion, such as trees, boulders, ditches, hillocks, knolls, and areas of poorly traversable terrain (i.e. hazards such as water or very soft soil). Hazards are assumed to be detected by on-board range sensors. It is recognized that hazard detection and sensing are important aspects of UGV mobility and an active research topic; however, they are not a focus of this work [22,74,75]. A terrain patch is described by its average roll $(\phi)$, pitch $(\psi)$, roughness $(\varpi)$, and traction coefficient $(\mu)$. It is assumed that coarse estimates of the tire/ground traction coefficient and ground roughness are known or can be determined online. Techniques for measuring or estimating the above parameters are available [2,38,39,54]. It is also assumed that vehicle inertial and kinematic properties are known with reasonable uncertainty.

The vehicle is assumed to be equipped with the following sensors:

- A forward-looking range sensor that can measure terrain elevation and locate hazards up to 30 vehicle lengths ahead;
- An inertial navigation sensor that can measure the vehicle's roll, pitch, yaw, roll rates, pitch rates, yaw rates, and translational accelerations with reasonable uncertainty;
- A global positioning system that can measure the vehicle's position and velocity in inertial space with reasonable uncertainty.

## 1.2 Background Literature

This section provides an overview of literature describing hazard avoidance techniques and path planning algorithms that could be useful in solving various challenges associated with high-speed rough-terrain hazard avoidance.

Hazard avoidance for UGVs has been traditionally performed by selecting from a set of predetermined paths (i.e. search techniques over small spaces), reactive (reflexive) behaviors, which evoke a predetermined action in response to specific sensor signals, or by completely re-planning the path. The former two techniques are grouped here as local path planning techniques since they only modify a portion of the path. The latter method is termed a global path planning technique since the entire path is re-planned using new information.

Many of the techniques presented in this section have been designed for use indoors or on gentle outdoor terrain at speeds that do not excite vehicle dynamics. For a method to be successful at high speeds in rough terrain the following points must be considered:

- The solution must be dynamically feasible. An example of a dynamically unfeasible maneuver is a sharp turn that causes a UGV to rollover. This is not a concern at low speeds since the centripetal force caused by turning is usually not high enough to initiate rollover. However, at high speeds the centripetal force cannot be ignored.

- The technique must be computationally efficient. Detection of a newly discovered hazard requires a vehicle to rapidly modify its preplanned path and begin to execute a new path in less than (at most) $\left(d_{hazard}/v - t_{comp}\right)$ seconds where $d_{hazard}$ is the distance to the hazard, $v$ is the forward velocity of the UGV, and $t$ is the time to process the information and compute a new path. To ensure safe traversal at high speeds, simple analysis shows that $t$ should be on the order of milliseconds.
- The technique must consider vehicle/terrain interaction effects. Terrain inclination, terrain roughness, and the tire/terrain traction coefficient affect a vehicle's performance capabilities. An example is a turn that is feasible on a paved surface, but results in excessive vehicle sideslip on gravel or other low friction surfaces. Another example is a turn that can be safely executed on flat terrain but causes a UGV to rollover when turning uphill because of the effect of gravity.
- The solution must account for uncertainty in the terrain profile. The fact that it is impossible to know the exact terrain profile makes it difficult to accurately forward simulate potential vehicle paths. Forward simulations based on an incomplete or inaccurate terrain profile may appear to be dynamically feasible, but may not be safe in reality. Consequently, hazard avoidance and path planning methods that do not consider terrain uncertainty may fail to predict the correct behavior of a UGV.
- The method needs to be applicable in highly unstructured environments such as natural outdoor terrain. This precludes most rule based solutions, which tend to be successful in semi-structured environments such as an indoor facility with obstacles placed in unknown locations. As the terrain becomes more unstructured, more rules are required, and the number of rules quickly becomes unwieldy.
- The method must not consider hazards in a solely binary manner. Hazard interaction is function of geometry, speed, and dynamics. Natural terrain and high speeds introduce trajectory dependent hazards, which are hazards that can be traversed at limited sets of velocities and path curvatures. One example is a deep ditch, which can only be traversed at high speeds when a UGV can "jump" the ditch.
- The technique must account for different vehicle designs. A method that does consider the differences among vehicles will fail to capture important factors such as suspension and inertial properties.

### 1.2.1 Local Search Techniques

In general, search techniques first generate a set of candidate paths through the local terrain. Second, the set is pruned, eliminating paths that are not feasible (i.e. ones that intersect with obstacles). Finally, a path is chosen from the remaining set according to a set of rules. Examples utilizing this technique are given in this section.

Previous researchers have addressed the problem of hazard avoidance with a search-based technique to navigate a High Mobility Multipurpose Wheeled Vehicle (HMMWV) at speeds up to 10 m/s while avoiding large hazards [15]. The method relies on a pre-computed database of approximately 15 million 20 to 30 meter long clothoid trajectories. Since the vehicle is assumed to travel on relatively flat terrain at fairly low speeds, the model used in the calculations does not consider vehicle dynamics. An online algorithm

eliminates candidate clothoids that intersect with hazards or are not feasible given the initial steering conditions. From the remaining paths, the algorithm chooses one that follows the most benign terrain. Computation time is ~250 ms, which suggests that with advances in computation power this technique might be applied to higher-speed situations.

A similar approach has been investigated [47]. To overcome the long computation time required to predict and simulate all potential maneuvers in real time, the authors propose pre-computing potential paths from a set of initial velocities and steering angles. The algorithm does not consider vehicle dynamics or vehicle/terrain interactions. After a hazard has been identified online, the set of potential maneuvers is searched for paths that coincide with the current steering angle, velocity, and desired goal position. Once this truncated set of potential maneuvers has been formed, the paths that coincide with the obstacle are removed. A single path is chosen from the set of remaining paths by minimizing a cost function. Similar to the work described above, this technique has been shown to be successful in reducing computation time.

Another local search technique first generates a nominal path from the start position to a goal position [25,26,27]. Next, parallel adjacent paths to the nominal path are created. As obstacles are detected as the vehicle traverses the nominal path, paths that intersect with obstacles or require excessive vehicle curvature are removed. Maneuvers between the adjacent paths are generated. This allows the vehicle to change paths throughout its trajectory if it encounters unforeseen hazards. This technique has been shown to be successful when dealing with moving obstacles; however, it does not consider vehicle dynamics or vehicle/terrain interaction. Furthermore, the method by which the vehicle moves between the paths is not fully developed, and the approach does not consider the vehicle speed or initial steering angle.

The first two techniques described above attempt to bypass the computational burden of predicting the vehicle's path by computing a large number of paths *a priori*. This reduces the on-board computation to searching through a library of potential maneuvers. This can work well on flat terrain with uniform traction. However, the models used to predict the paths do not consider terrain inclination, roughness, and the tire/terrain traction coefficient. Without incorporating these effects, the resulting path may be dynamically infeasible for the UGV to follow. Moreover, if these effects were considered when generating the potential paths, the set of all possible combinations of traction coefficients, terrain roughness, terrain inclinations, vehicle speeds, and vehicle steering angles could quickly increase to an intractable number.

### 1.2.2 Local Reactive Behavior Techniques

Reactive behaviors, also known as reflexive behaviors, generate a specific action in response to sensor signals. The original work in reactive behaviors involves creating "task-achieving behaviors" for the robot to follow [6]. The author defines several tasks, or behaviors, that the robot can perform. They include: avoid obstacles, wander aimlessly, locate and move toward an interesting object, build a map, monitor changes in the environment, and identify objects. The tasks are hierarchically layered such that the robot must first avoid obstacles, then wander aimlessly, and so on. This work provided

the basis for a large body of research, much of which is related to the questions addressed in this thesis. In the context of UGVs, the behaviors commonly include goal seeking and hazard avoidance. The resulting "actions" usually consist of candidate steering angles. Most of the research focuses on deciding which action to choose based on the available information.

Two successfully implemented reactive behavior techniques for outdoor hazard avoidance utilize a simple "winner takes all" algorithm to decide which steering angle to enact. The first approach arbitrates between hazard avoidance and goal seeking and allows for UGV navigation at speeds of up to 1 m/s [16,57]. This represents early work in the field and does not consider high speeds, vehicle dynamics, or vehicle/terrain interactions. The second approach defines five behaviors: avoid obstacles, follow the road, seek the goal, maintain heading, and track the path [48]. For each approach the "behavior" votes for or against every possible steering angle. The steering angle with the most votes is executed. These approaches have proven successful at low speeds on benign terrain.

Another reactive behavior method for choosing steering angles is termed blending [61]. In contrast to the previous approach, the steering angle with the most votes is not necessary chosen. Instead, the final steering angle is the average of the steering angles chosen by the reactive behaviors. As an example, consider a situation where steering angle $\delta_1$ receives two votes and steering angle $\delta_2$ receives one vote. In the "winner takes all" approach the resulting steering angle would be $\delta_1$. Using a blending method would result in a steering angle of $(2\delta_1 + \delta_2)/3$. This approach has been experimentally validated in an indoor environment. Similar to the "winner takes all" approach this work is not well-suited for an outdoor, natural terrain environment since actions are chosen independent of speed, dynamics, and terrain conditions.

Although these reactive behavior techniques have been successful at reaching a goal location while avoiding hazards at low to moderate speeds, none explicitly consider vehicle dynamics and changing terrain characteristics. Failure to consider these effects can result in trajectories that are impossible for a high speed UGV to safely execute.

### 1.2.3 Global Techniques

The methods reviewed in this section have been designed for generating paths on a global scale. However, it is feasible that they could be applied to hazard avoidance on a local scale if they could be executed quickly enough. Methods introduced in this section include probabilistic road maps, graph search techniques such as the A* and D* algorithms, rapidly exploring random trees, and potential fields (see [49] for an overview).

Probabilistic roadmaps work in two phases [42]. In the first phase, a roadmap of safe vehicle locations and headings, as well as paths between these configurations, is created and stored. In the second phase, a vehicle is given a start and finish node and then must connect the two with configurations and paths found during the first phase. Probabilistic roadmap techniques have proven to work well in static well-known environments and are considered computationally efficient for global path planning techniques [77].

The A* and D* algorithms are cell decomposition techniques that have proven to be an effective path planning method for UGVs [40,56,87]. Researchers have used the D* algorithm in conjunction with a local navigator to yield an algorithm that is capable of navigating a HMMWV-class UGV from an initial position to a goal position in outdoor terrain [79,80]. The algorithm maintains a map of the terrain segmented into cells. Each cell is labeled impassable, high cost, or traversable. As the UGV traverses the terrain, local sensors update the map after detecting new obstacles, and the D* algorithm plans a new optimal path to the goal. The method has been shown to effectively plan paths in unknown, partially known, and changing environments, in an efficient, optimal, and complete manner on a UGV traveling at 2 m/s.

A method to find a path on hilly three dimensional terrain subject to dynamic and physical interaction constraints has been presented [11,12]. The work uses detailed equations of motion of the vehicle and knowledge of the terrain traction coefficient to prevent vehicle sideslip. The work also uses a detailed and fully accurate map of smooth terrain. There is no mention of computational efficiency, so the possibility of real time application is not evident.

Rapidly-exploring random tree (RRT) algorithms start with initial conditions and build a tree of feasible trajectories from those conditions [28, 51]. As the tree is being built, the algorithm attempts to efficiently explore the reachable set of locations. Although this method is considered efficient for a global path planner, it requires solving the equations of motion in real time and thus is not suitable for high-speed emergency hazard avoidance.

Potential field methods were first introduced as a method for real time obstacle avoidance for manipulators and mobile ground robots [45]. Since then, researchers have applied the technique to a variety of situations, including mobile ground robots in natural terrain [9,36,49,68,83]. The method usually works by defining a potential field in Cartesian space. The goal location is defined by a "sink" function, and hazards are defined as "source" functions. The vehicle then navigates towards points of low potential from points of high potential. This technique can be used for global or local path planning since it is generally computationally efficient enough to be implemented in real time. One drawback to the approach is the existence of local minima, which can trap the robot in a particular location. Methods have been introduced to generate potential fields without local minima, but they tend to be computationally expensive [14]. To date these techniques have proven successful for slow moving UGVs on flat terrain. A potential field method has recently been developed for high speed vehicles on rough terrain. This method is based on the trajectory space introduced in this thesis [73].

An approach to motion planning on three dimensional terrain finds the optimal speed along a given path while considering vehicle dynamics such as slip, rollover, and ballistic motion [70]. The main contribution of this work is finding the fastest dynamically admissible speed along a preplanned path. However, it does not consider the problem of emergency hazard avoidance. Extensions of this work plan time optimal paths along a given terrain [71] and include consideration of static and moving obstacles [21]. Although advances in computation power could reduce the algorithm's computation time

significantly, the methods are most likely not computationally efficient enough for real time application.

The strength of the path planning methods presented in this section lie in global path planning. The advantage of these techniques is that a complete path can be planned off-line. Using these techniques for local hazard avoidance is generally difficult due to the lack of computational efficiency. Even for global path planning techniques that are computationally efficient, like some potential field methods, none of the methods consider the effects of vehicle dynamics, terrain inclination, tire/terrain interactions, or terrain roughness.

### 1.2.4 Other Techniques

This section contains approaches to the hazard avoidance problem that have either been applied to robots other than UGVs or differ from the three categories mentioned above.

A method for finding dynamically safe maneuvers has been investigated and applied to autonomous helicopters [28,29]. The work is based on building a library of dynamically feasible maneuvers that bring a vehicle from one stable trajectory to another. It is shown that stability is maintained when executing these maneuvers. A local mixed integer linear programming path planning technique that builds upon this work has been proposed [66,67]. In this work, an autonomous aircraft always maintains a dynamically feasible maneuver that ends in a safe trajectory. This guarantees that there will always be a "escape" maneuver that can be executed if an unforeseen hazard appears. This method works well for aircraft that can maintain a safe "holding pattern." Unfortunately, this is not an option for ground vehicles.

An arbitration approach that has been successfully demonstrated in outdoor environments has been presented [43,44]. In this work, a UGV forward simulates several candidate paths consisting of circular arcs corresponding to specific steering angles. Each path is evaluated using a voting scheme that considers vehicle roll, vehicle pitch, collision between the terrain and vehicle body, and collision between the terrain and wheels. Another "tactical" vote is the combination of the four hazard avoidance votes. A final "strategic vote" is a goal-seeking vote. An arbitrator then chooses the path closest to the strategic vote that satisfies the hazard avoidance tactical vote. This method has been experimentally demonstrated on a HMMWV class vehicle at speeds up to 4.5 m/s on natural terrain. However, this work uses simplified equations of motion to predict possible paths for the vehicle to follow. This is not appropriate for high speed rough terrain UGVs due to the inherent lack of knowledge about the exact terrain profile and the computational complexity of the vehicle models required for dynamic simulations. An extension of this work limits the set of initial candidate paths by placing constraints on the space of a vehicle's velocity and curvature [64]. This is similar to the method presented in this thesis; however, after the set of candidate paths is reduced, each path is forward simulated and evaluated using the same arbitration scheme introduced in [43].

A local hazard avoidance technique for on-road situations has been introduced and simulation results have demonstrated its effectiveness [72]. The method calculates a lane change maneuver that minimizes the longitudinal distance traveled. It considers vehicle

dynamics including cornering properties. The method is presented as being computationally efficient enough for real time implementation. However, the technique is designed for on-road situations where the potential paths are well defined, and it is not clearly applicable to off-road rough-terrain situations with multiple hazards.

Two very similar approaches are the curvature-velocity method [76] and the dynamic window technique [5,24,60]. Both were developed to consider velocity and acceleration constraints for synchro-drive robots (robots that can move in all directions but cannot change their orientation) in indoor environments. A two-dimensional space consisting of a vehicle's translational and rotational velocities is created. Velocity combinations that cannot be reached because of acceleration constraints are removed. Additional velocity combinations that would result in impact with an obstacle are also removed. A translational and rotational velocity is then chosen from that space by maximizing an objective function. The interesting aspect of this approach is that it places limits on the search space to prohibit velocities that are not attainable for a robot's current configuration. Both the curvature-velocity method and dynamic window approach have similar elements to the trajectory space method described in this thesis. However, they do not consider important vehicle/terrain interactions, vehicle dynamics other than acceleration constraints, or terrain inclination.

The methods presented in this section are for high speed avoidance applied to unmanned aerial vehicles, methods for high-speed on-road hazard avoidance, arbitration schemes, or novel approaches for indoor local hazard avoidance. Each method addresses some of the key requirements for high-speed rough terrain hazard avoidance. None address all of the requirements.

### 1.2.5 Literature Review Summary

A summary of the methods addressed in Section 1.2 is presented in Table 1.1.

Table 1.1: Summary of hazard avoidance methods

| Method | Dynamically feasible solution? | Computationally efficient? | Considers vehicle/terrain interaction? | Accounts for uncertainty in terrain profile? | Applicable to rough, outdoor terrain? | Considers non-binary hazards? | Considers vehicle characteristics? |
|---|---|---|---|---|---|---|---|
| Local Path Search | - | ✓ | - | - | ✓ | - | - |
| Reactive Behaviors | - | ✓ | - | - | - | - | - |
| Probabilistic Road Maps/ RRT | - | - | - | - | ✓ | - | - |
| A*, D*/Cell Decomposition | ✓ | - | - | - | ✓ | - | - |
| Classic Potential Field | - | ✓ | - | - | ✓ | - | - |
| Arbitrators | ✓ | ✓ | - | - | ✓ | - | - |
| MILP | ✓ | ✓ | - | - | - | - | - |
| Curvature-Velocity | - | ✓ | - | - | ✓ | - | - |
| Dynamic Window | - | ✓ | - | - | ✓ | - | - |

## 1.3 Statement of Contributions

The first and main contribution of this thesis is the development and analysis of the hazard avoidance algorithm, an online physics-based planning algorithm for high-speed robotic vehicles in rough, natural terrain. This algorithm allows a vehicle to perform dynamically feasible hazard avoidance maneuvers in a computationally efficient manner. The algorithm is based on the trajectory space, a two-dimensional space of a vehicle's instantaneous path curvature, $\kappa$, and longitudinal velocity, $v$. The trajectory space approach has the following advantages:

- Constraints can be imposed on the space to yield a compact representation of the vehicle's performance limits. This representation considers vehicle dynamic effects such as rollover, sideslip, and over/understeer, and actuation effects such as braking limits, acceleration limits, steering kinematic limits, and steering rate limits.
- There is a straightforward mapping from the trajectory space to the UGV actuation space, which generally consists of throttle and steering angle inputs.
- Path and direction dependent hazards are easily represented on the trajectory space. It is possible to represent both trajectory dependent and trajectory independent hazards.
- The trajectory space can consider the effects of terrain roughness and inclination.
- The algorithm is computationally efficient. Emergency maneuvers are generated on the order of milliseconds on the onboard PC104 computer of the experimental system described in the next section.

A second contribution of this thesis is the curvature matching algorithm, a method that quickly generates a path connecting the hazard avoidance path with the nominal desired path. This is a novel, computationally efficient method for accomplishing this task.

## 1.4 Experimental System and Simulation Environment

Experimental trials validating the trajectory space algorithm presented in this thesis were conducted on the Autonomous Rough Terrain Experimental System (ARTEmiS) shown in Figure 1.2. ARTEmiS is a small (0.56 m wheelbase) gasoline powered UGV. Details of its physical parameters and sensor capabilities are given in Chapter 4.

**Figure 1.2: Autonomous Rough Terrain Experimental System (ARTEmiS)**

Simulations were performed using vehicle and terrain models developed for a commercial dynamic analysis software package, MSC.ADAMS. MSC.ADAMS has the ability to model vehicle systems, including the tire/terrain interaction effects. For this work, the tires were modeled using the Magic Tire Model [58]. The terrain is considered to be rigid and the tires deformable. More detailed information concerning this software package and terrain, tire, and vehicle modeling can be found in Appendix B and elsewhere [1,34]. The vehicle model using the MSC.ADAMS software package is referred to as the "reference model" in this thesis.

## 1.5 Outline

The organization of this thesis is as follows. Chapter 2 contains a description of the trajectory space along with its related spaces: the dynamic trajectory space, the admissible trajectory space, the reachable trajectory space, and the hazard trajectory space. Each of the aforementioned spaces is defined by constraints that are placed on the trajectory space. For example, the dynamic trajectory space is defined by rollover constraints, sideslip constraints, power train constraints, and steering mechanism constraints. Computation of these constraints is introduced. A description of how terrain roughness is measured and its effect on the trajectory space constraints is also included.

Chapter 3 describes a method for generation of dynamically feasible hazard avoidance maneuvers using the trajectory space. The maneuver is based on the trajectory space of the UGV at the time a hazard is identified. A method for choosing a maneuver using the trajectory space is introduced. The chapter also presents the curvature matching method for generating a path connecting the hazard avoidance maneuver back to the nominal desired path. A comparison between the curvature matching method and a more conventional technique is presented.

Chapter 4 contains simulation and experimental results validating the trajectory space algorithm and the curvature matching algorithm. The chapter also contains a more detailed description of the experimental system. Chapter 4 concludes with a comparison between the trajectory space algorithm and other methods.

Conclusions of the work are presented in Chapter 5 along with ideas for future research in the area.

# CHAPTER 2

# THE TRAJECTORY SPACE

This chapter describes the trajectory space, the basis of the hazard avoidance algorithm developed in this thesis. First, the trajectory space is introduced. Four major subspaces of the trajectory space are presented next: the dynamic trajectory space, the reachable trajectory space, the admissible trajectory space, and the hazard trajectory space. The dynamic trajectory space consists of locations in the trajectory space that represent dynamically feasible configurations of a vehicle on a given terrain. The reachable trajectory space consists of locations in the trajectory space that a UGV can transition to in a given time considering steering, acceleration, and braking properties. The admissible trajectory space is the intersection of the reachable and dynamic trajectory spaces. The hazard trajectory space consists of locations inside the trajectory space that would result in a UGV impacting or failing to traverse a hazard. Each of the four spaces described above is defined by constraints placed upon the trajectory space. The calculation of those constraints from simple models is detailed in this chapter. Last, this chapter describes the effect of terrain roughness on the trajectory space constraints.

## 2.1 The Trajectory Space

**Definition 2.1 (Trajectory)** *A trajectory, $\tau(s)$, is defined as the pair $(v(s),\kappa(s))$, $s \in [s_0, s_f]$, where v is the vehicle's longitudinal velocity and $\kappa$ is the vehicle's path curvature.*

**Definition 2.2 (Trajectory Space)** *The trajectory space is the space of a vehicle's instantaneous longitudinal velocity, v, and instantaneous path curvature, $\kappa$.*

Figure 2.1 shows an illustration of the trajectory space with icons depicting a vehicle's actions corresponding to various coordinates in the space. A point in the trajectory space is thus a subset of the system's dynamics and configurations. It serves as a useful description of the current vehicle status.

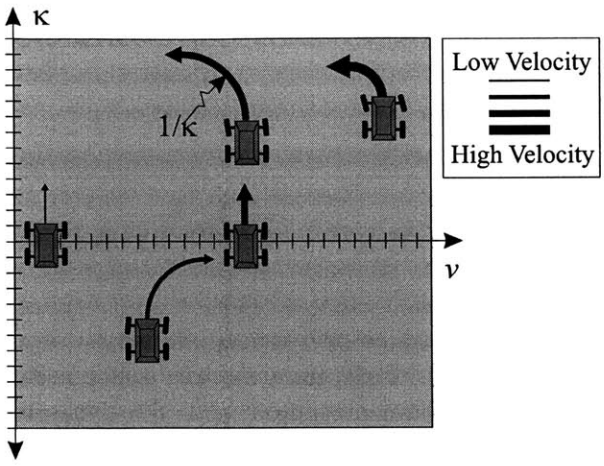


**Figure 2.1: Representation of vehicle action as described by its coordinates in the trajectory space.**

The icon on the left shows a vehicle traveling straight (zero curvature) at a low speed. The icon at the upper right represents a vehicle turning sharply at high speed. Velocities are limited to positive values in this work. Using the space of a vehicle's velocity and curvature for navigation has been proposed elsewhere [76] as discussed in Section 1.2.4. However, previous implementations of the technique do not consider vehicle dynamic effects such as sideslip, rollover, and cornering as well as terrain properties such as inclination and roughness.

Constraints are placed on the trajectory space to define safe trajectories that a vehicle can achieve on a given terrain. A first set of constraints limits a UGV to trajectories that are dynamically feasible. These constraints capture the effects of UGV rollover, sideslip, cornering properties, and power train properties. A second set of constraints bound the trajectories a vehicle can transition to in a given period of time. A third set of constraints includes velocity and curvature pairs that would result in impact with a local hazard.

The next four sections in this chapter describe computation of these constraints using simple models. These simple models do not capture all of the complex vehicle dynamics and vehicle/terrain interactions, but have been shown to be reasonably accurate. Section 2.6 discusses the use of dynamic simulations using high-order nonlinear vehicle and terrain models to capture complex vehicle dynamics and tire/terrain interactions.

## 2.2 The Dynamic Trajectory Space

The dynamic trajectory space consists of velocity and curvature pairs that do not cause excessive vehicle sideslip or rollover and are attainable considering vehicle cornering properties and steering geometry. This space is bounded by a set of constraints. Methods for computing these constraints are described below.

### 2.2.1 Sideslip Constraint Computation

A simple relation describing vehicle sideslip can be derived from the free body diagram shown in Figure 2.2 assuming that $\dot{v} = 0$ and $\dot{\kappa} = 0$. Here $\{\mathbf{xyz}\}$ represents a body-fixed coordinate frame and $\{\mathbf{XYZ}\}$ represents an inertial frame. Note that the

sideslip constraint is a function of the tire/terrain traction coefficient and terrain inclination. Although some sideslip is expected and unavoidable, substantial slip that causes large heading or path following errors is detrimental to vehicle performance.

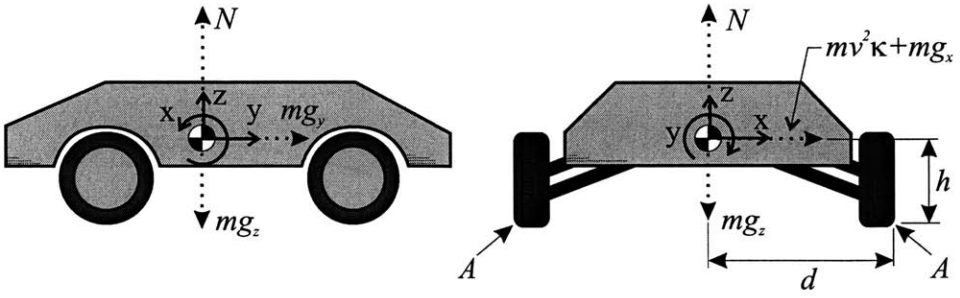


**Figure 2.2: Sideslip vehicle model**

For simplicity vehicle roll and pitch are assumed to be equal to the roll and pitch of the terrain beneath the vehicle. The vehicle roll $(\phi)$, pitch $(\psi)$, and yaw $(\theta)$ associated transformation matrices are:

$$G_\phi = \begin{bmatrix} \cos\phi & 0 & -\sin\phi \\ 0 & 1 & 0 \\ \sin\phi & 0 & \cos\phi \end{bmatrix} G_\psi = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\psi & \sin\psi \\ 0 & -\sin\psi & \cos\psi \end{bmatrix} G_\theta = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

The acceleration due to gravity in the body-fixed frame following a roll, pitch, yaw transformation is thus:

$$\mathbf{g}_{\mathbf{xyz}} = G_\phi G_\psi G_\theta \mathbf{g}_{\mathbf{XYZ}} \tag{2.1}$$

where $\mathbf{g}_{\mathbf{xyz}} = \begin{bmatrix} g_x & g_y & g_z \end{bmatrix}^T$. A vehicle begins to slip when the traction force is equal to the sum of the centripetal and gravitational force components:

$$\left| mv^2\kappa + mg_x \right| = \mu m g_z \tag{2.2}$$

Note that accelerations other than gravity in the z-direction are ignored. The maximum curvature before slip occurs is given as:

$$\kappa_{slip}^{\min,\max} = \frac{-g_x \pm \mu g_z}{v^2} \tag{2.3}$$

The two solutions correspond to downhill/uphill travel.

**Definition 2.3 (Sideslip Space)** *Let the sideslip space, A, be defined as the set of velocity and curvature pairs that do not lead to vehicle sideslip:*

$$A(\phi,\psi,\varpi,v,\mu) \equiv \left\{ \forall \kappa \mid \kappa_{slip}^{\min} < \kappa < \kappa_{slip}^{\max} \right\} \tag{2.4}$$

Figure 2.3 demonstrates how sideslip constraints change as terrain inclination changes. The sloped terrain constraint corresponds to a UGV traversing a side slope of 30° with the fall line perpendicular to the vehicle's heading. Negative curvatures represent a downhill turn.

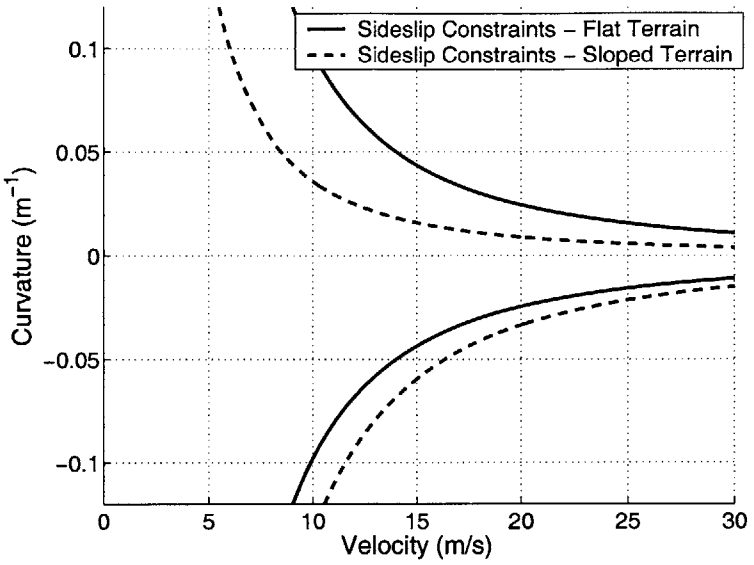


**Figure 2.3: Comparison of sideslip constraints for flat and sloped terrain**

The UGV in this example is able to turn downhill with higher velocities than it can turn uphill before sideslip is initiated. This is a result of the interaction of centripetal acceleration and gravity. Figure 2.4 shows how sideslip constraints vary for different traction coefficient values.

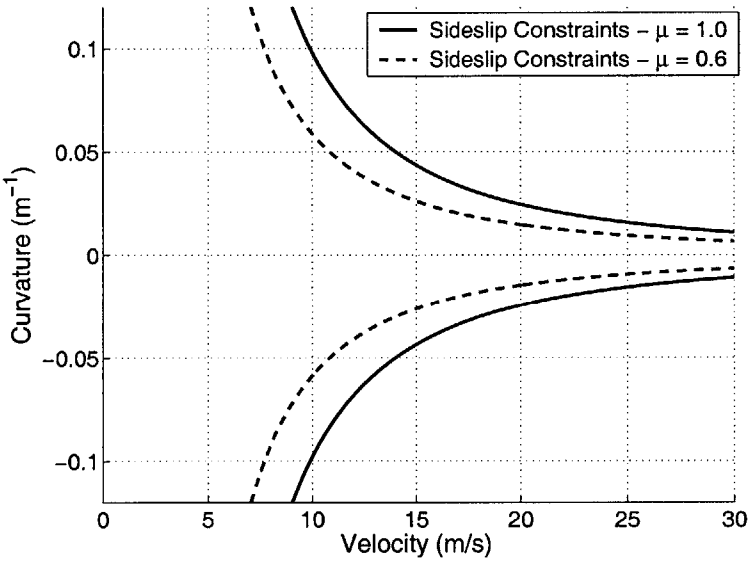


**Figure 2.4: Comparison of sideslip constraints for varying traction coefficient on flat terrain**

As expected, as the traction coefficient increases for a fixed speed, a vehicle can execute a sharper turn.

Clearly the sideslip model presented here does not include potentially significant effects such as those induced by excessive wheel slip; however, since it is undesirable to operate outside of the range of curvatures specified by Equation 2.3, higher order effects such as these can be ignored.

### 2.2.2 Rollover Constraint Computation

Rollover is generally undesirable despite the fact that some UGVs are designed to be invertible [86]. A trajectory space constraint related to vehicle rollover can be computed using simple but accurate analytical models. The constraint is a function of vehicle tire and suspension characteristics, center of mass location, terrain roughness, and terrain inclination. Three models with increasing complexity are introduced here. The predicted rollover constraints from each model are compared to the rollover constraint generated in simulations using the reference model.

Potential for rollover is initiated when the moment about either of the points A in Figure 2.2 is equal to zero, or when the force vector acting on the vehicle center of mass is directed outside of the convex polygon created by the tire/terrain contact points. These two conditions are the same when considering only lateral forces in the planar cases examined in this section. Rollover can be initiated in directions other than lateral direction, but it is not common. Thus it is appropriate to ignore accelerations in the longitudinal direction. The examined models include a rigid body model, a model that includes the effect of tire stiffness, and a model that includes both tire stiffness and suspension characteristics (see Figure 2.5, Figure 2.6, and Figure 2.7) [10, 35].

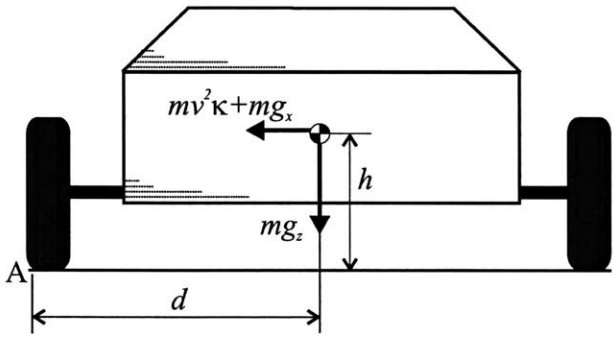


**Figure 2.5: Rigid body vehicle rollover model**

Summing the moments about point A for the rigid model results in:

$$\left(mv^2\kappa \mp mg_x\right)h = \left(mg_z\right)d$$

Solving for curvature yields:

$$\kappa_{rollover}^{\max,\min} = \frac{dg_z \pm hg_x}{hv^2} \tag{2.5}$$

where $g_x$ and $g_z$ are taken from Equation 2.1, $d$ is one half the axle length, $h$ is the center of mass height, and $v$ is the vehicle's longitudinal velocity. The two solutions correspond to uphill/downhill travel.

Figure 2.6 shows a tire stiffness model with $\gamma$ as the roll angle of the body due to tire deformation caused by cornering forces.

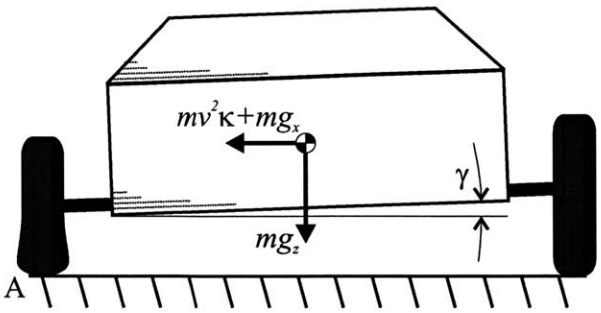


**Figure 2.6: Tire stiffness vehicle rollover model**

The deflection angle due to tire deformation is:

$$\gamma = \frac{mg}{2K_t d} \tag{2.6}$$

where $K_t$ is the vertical tire stiffness. Summing the moments about point A yields:

$$mv^2\kappa\left(\frac{h}{\cos\gamma}+(d-h\tan\gamma)\sin\gamma\right)\mp mg_x\left(\frac{h}{\cos\gamma}+(d-h\tan\gamma)\sin\gamma\right) = mg_z(d-h\sin\gamma) \tag{2.7}$$

It is assumed that $\gamma$ is small and thus $\sin\gamma \approx \gamma$ and $\cos\gamma \approx 1$. Linearizing Equation 2.7 about $\gamma = 0$ and solving for curvature yields:

$$\kappa_{rollover}^{\max,\min} = \frac{(d-h\gamma)g_z \pm (h+d\gamma)g_x}{(h+d\gamma)v^2} \tag{2.8}$$

Again, the two solutions correspond to uphill/downhill travel.

Figure 2.7 shows a vehicle model that considers the effects of tire compliance and suspension compliance. The designations $s$ and $u$ denote sprung and unsprung, respectively. The sprung mass is composed of the mass supported by the suspension, usually consisting of the vehicle body, powertrain, and steering mechanism. The unsprung mass is comprised of the tires and axles.

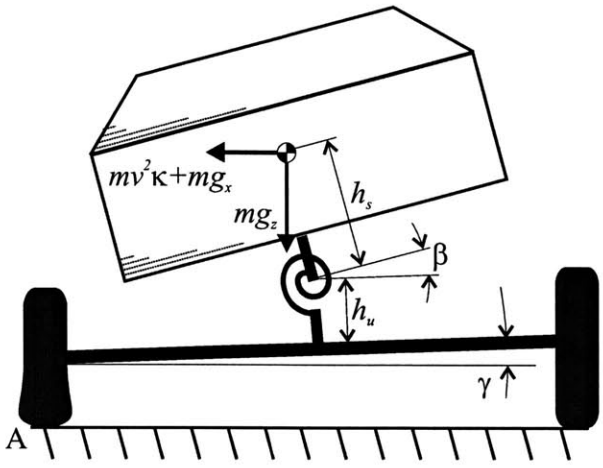


**Figure 2.7: Suspension vehicle rollover model**

Summing the moments about point A gives:

$$\left(mv^2\kappa \mp g_x m\right)\left(\sin\gamma\left(d-\left(\left(h_s\cos\beta+h_u\right)\tan\gamma+\left(h_s+h_u\right)\sin\beta\right)\right)+\frac{h_s\cos\beta+h_u}{\cos\gamma}\right) \quad (2.9)$$
$$= mg_z\left(d-\left(\left(h_s\cos\beta+h_u\right)\tan\gamma+h_s\sin\beta\right)\cos\gamma\right)$$

where $\beta$ is the roll angle between the sprung and unsprung masses. It is assumed that both $\gamma$ and $\beta$ are small and thus $\sin\gamma\approx\gamma$, $\cos\alpha\approx 1$, $\sin\beta\approx\beta$, and $\cos\beta\approx 1$. Linearizing Equation 2.9 about $\gamma=0$ and $\beta=0$ and solving for curvature yields:

$$\kappa_{rollover}^{\max,\min}=\frac{(d-(h_s+h_u)\gamma-h_s\beta)g_z\pm((h_s+h_u)+d\gamma)g_x}{((h_s+h_u)+d\gamma)v^2} \quad (2.10)$$

where the suspension deflection angle is given as:

$$\beta=\frac{\left(mv^2\kappa+mg_x\right)h_s}{K_s} \quad (2.11)$$

where $K_s$ is the vehicle suspension roll stiffness. For vehicles with compliant suspensions rollover usually occurs at the kinematic suspension limits. For example, the maximum suspension deflection angle for ARTEmiS is given as:

$$\beta_{\max}=\tan^{-1}\left(\frac{d_{susp}}{l_{susp}}\right) \quad (2.12)$$

where $d_{susp}$ is the maximum deflection of each suspension member, and $l_{susp}$ is the distance between the two suspension linkages (see Figure 2.8).

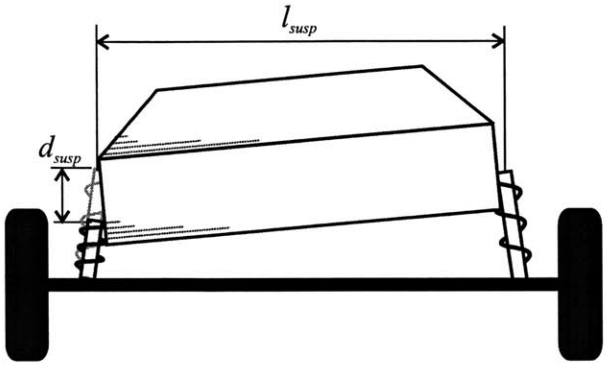


**Figure 2.8: Suspension parameter diagram**

The accuracy of the three models given above in predicting rollover was compared to results from simulations using the ARTEmiS reference model on flat ground. The reference model was commanded to follow a nominal desired path consisting of a straight line segment followed by a clothoid. A clothoid is a curve whose curvature is linearly related to its arc length:

$$\kappa = ps \tag{2.13}$$

where $s$ is the arc length and $p$ is the rate of change of curvature. An example of a clothoid curve is shown in Figure 2.9 beginning 20 m along the path.

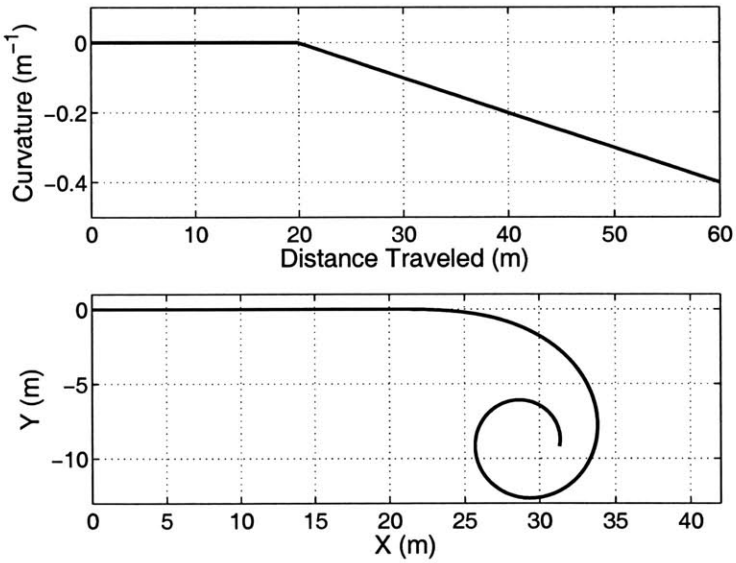


**Figure 2.9: Example of clothoid curvature (above) and path (below)**

This path allows the reference model to reach a desired longitudinal velocity and then slowly increase its curvature. Rollover was determined as initiating when $a_{lateral} = gd/h$. The roll angle and path of the vehicle were inspected to confirm that rollover initiated when this condition was satisfied. Figure 2.10 compares the rollover limits generated using the reference model to those calculated using the three rollover models.

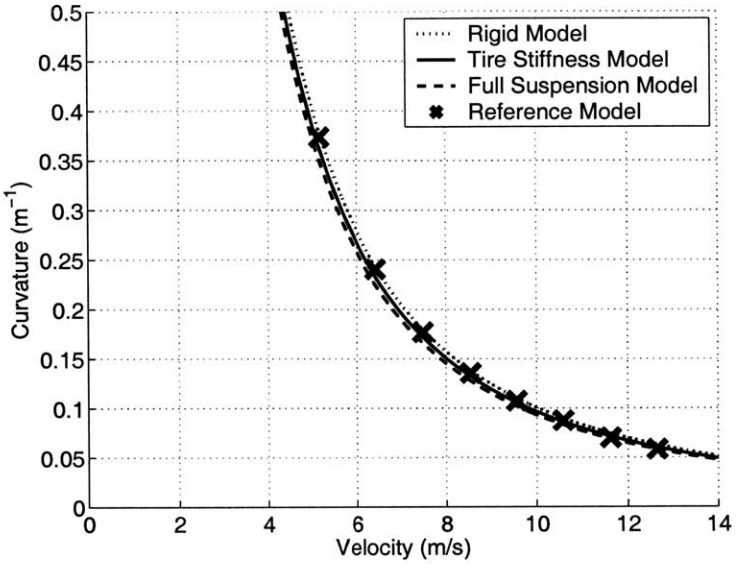


**Figure 2.10: Comparison of theoretical rollover models with reference model simulations**

The difference between the three models is small for this particular maneuver on flat terrain. In general, the three models have been found to accurately predict rollover for a large range of maneuvers on flat smooth terrain. For maneuvers over rough terrain these simple models may not be sufficient to capture the complex terrain/vehicle dynamic effects. This is investigated in Section 2.6.

**Definition 2.4 (Rollover Space)** *Let the rollover space, B, be defined as the set of velocity and curvature pairs that do not lead to vehicle rollover:*

$$B(\phi,\psi,\varpi,v) \equiv \left\{\forall \kappa \mid \kappa_{rollover}^{\min} < \kappa < \kappa_{rollover}^{\max}\right\} \tag{2.14}$$

Figure 2.11 shows a comparison of rollover constraints generated using the rigid model on flat and sloped terrain. The sloped terrain example corresponds to a UGV traversing a side slope of 30° with the fall line perpendicular to the vehicle's heading. As expected a UGV can safely execute downhill turns (negative curvatures in this example) with greater velocity than it can execute uphill turns, since gravity counters the centripetal acceleration.

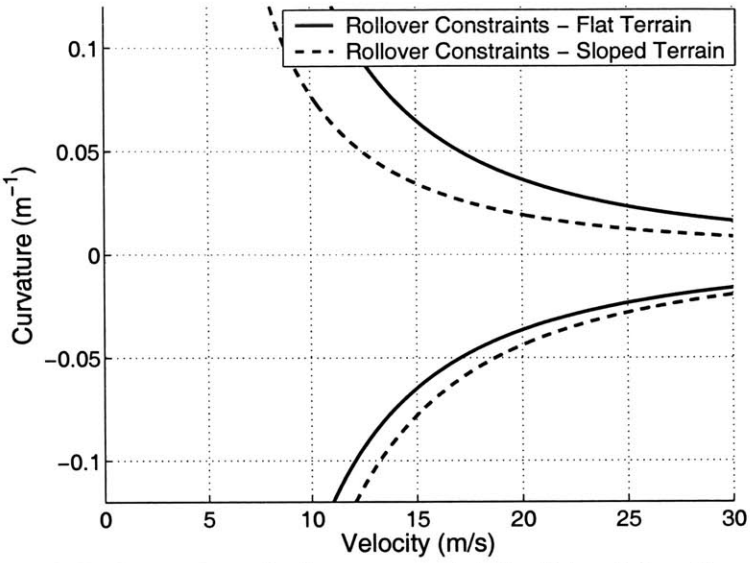


**Figure 2.11: Comparison of rollover constraints for flat and sloped terrain**

### 2.2.3 Steering Constraint Computation

The steering constraints are a function of a vehicle's maximum steering angle, center of mass location, and tire properties. The analysis presented in this section is based on the classic bicycle, or single-track, vehicle model (see Figure 2.12) [82]. In this model, the properties of the front and rear wheels are simplified into one wheel each on the centerline of the vehicle. Using the single-track model the maximum attainable curvature is:

$$\kappa_{\max} = \frac{\tan \delta_{\max}}{L} \tag{2.15}$$

where $L$ is the wheelbase length and $\delta_{max}$ is the maximum steering angle. This is also the maximum attainable curvature for a neutrally steered vehicle. A neutrally steered vehicle does not require a change in steering angle in order to maintain a constant radius curve while increasing forward velocity.

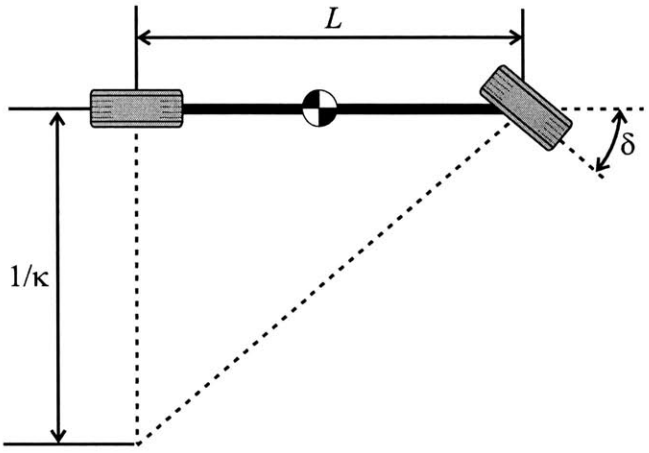


**Figure 2.12: Kinematic single-track vehicle model**

Non-neutral steered vehicles exhibit understeer or oversteer [33]. As forward speed increases for an oversteered vehicle, the steering angle must decrease in order to maintain a constant radius curvature. An understeered vehicle requires an increase in the steering angle to maintain the same curvature for increasing forward velocity.

Similarly, for a fixed steering angle a neutral steered vehicle follows the curvature given by Equation 2.15. An understeered vehicle "undershoots" that curve. An oversteered vehicle "overshoots" the curve (see Figure 2.13).

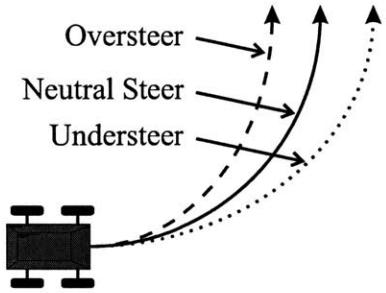


**Figure 2.13: Oversteer, understeer, and neutral steer comparison**

There are many factors that determine if a vehicle will exhibit oversteer or understeer. These include tire cornering stiffness, location of the center of mass along the wheelbase, suspension geometry and properties, tire camber thrust, aligning torque, and the steering system geometry [33]. Tire cornering properties and the location of the center of mass are major influences, and their effects are investigated here.

During cornering, the tires develop a lateral force, commonly referred to as "cornering force." The angle between the heading direction of the tire and the direction of travel is known as the slip angle, $\alpha$ (see Figure 2.14).

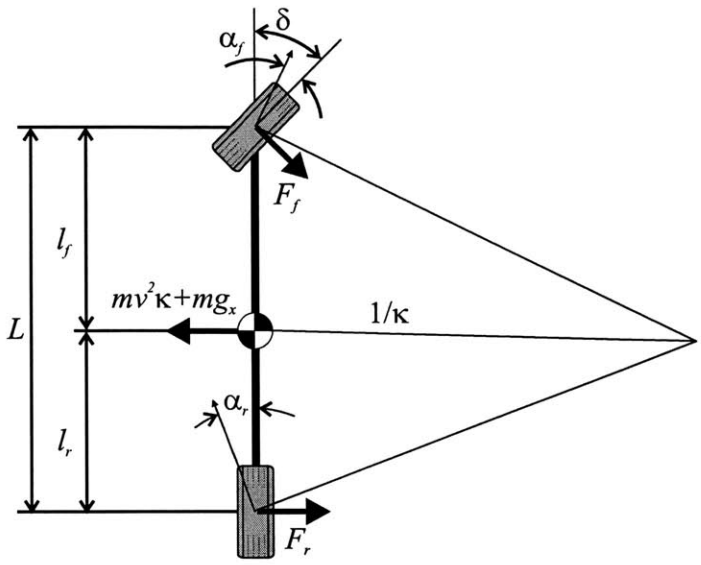


**Figure 2.14: Single-track vehicle model for steering mechanism limit computation (adapted from [33])**

At low slip angles (typically less than 5 deg), the relationship between the cornering force and slip angles is linear [33].

$$F_{f,r} = C_k \alpha_{f,r} \tag{2.16}$$

where $C_k$ is the cornering stiffness of the tire and $\alpha_{f,r}$ are the slip angles of the front and rear tires, respectively. Summing the lateral forces acting on the vehicle results in:

$$F_f \cos\delta + F_r = mv^2\kappa \pm mg_x \tag{2.17}$$

Summing the moments about the center of mass yields:

$$F_f l_f \cos\delta = F_r l_r \tag{2.18}$$

Combining equations 2.16, 2.17, and 2.18 and linearizing about $\delta = 0$ yields:

$$\alpha_{f,r} = \frac{m\left(v^2\kappa \pm g_x\right)}{C_k\left(1 + \frac{l_{f,r}}{l_{r,f}}\right)} \tag{2.19}$$

By examining Figure 2.14 it is evident that

$$\tan\delta = L\kappa + \alpha_f - \alpha_r \tag{2.20}$$

Combining equations 2.19, and 2.20 and yields:

$$\kappa_{steering}^{\max,\min} = \frac{C_k L \tan\delta_{\max} \pm mg_x\left(l_f - l_r\right)}{\left(C_k L^2 + mv^2\left(l_r - l_f\right)\right)} \tag{2.21}$$

**Definition 2.5 (Steering Space)** *Let the steering space, C, be defined as the set of velocity and curvature pairs that are attainable given a vehicle's kinematic configuration and tire parameters:*

$$C(\phi,\psi,\mu,v,m,C_k,l_{f,r}) \equiv \left\{\forall \kappa \mid \kappa_{steering}^{\min} < \kappa < \kappa_{steering}^{\max}\right\} \tag{2.22}$$

Note that a skid-steered vehicle does not have a steering space since it has no steering mechanism.

Figure 2.15 shows an example of the difference in steering limits between an understeered and oversteered vehicle. A neutral steered car's steering limits would be a line with zero slope.

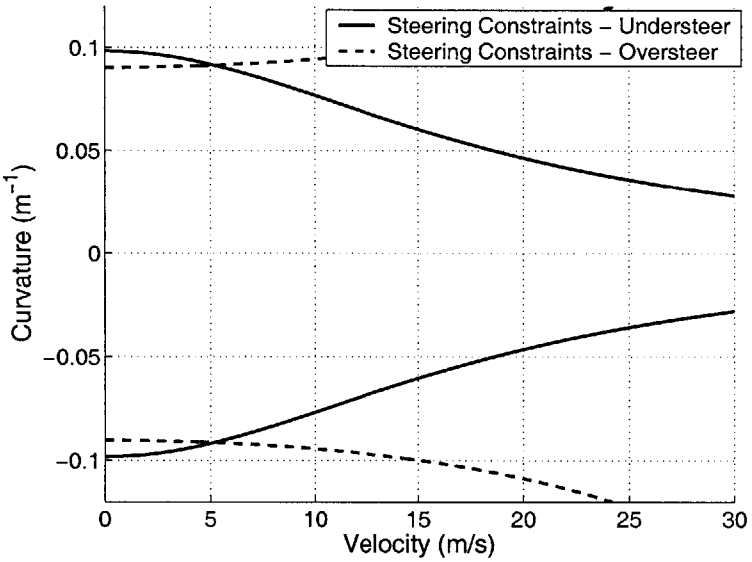


**Figure 2.15: Comparison of steering limits for over/understeered vehicles**

As the terrain inclination changes, the steering limits change for over and understeered vehicles. Figure 2.16 demonstrates the difference in steering angle limits for an understeered vehicle on flat and sloped terrain. The sloped terrain example corresponds to a UGV traversing a side slope of 30° with the fall line perpendicular to the vehicle's heading. Negative curvatures correspond to downhill travel.

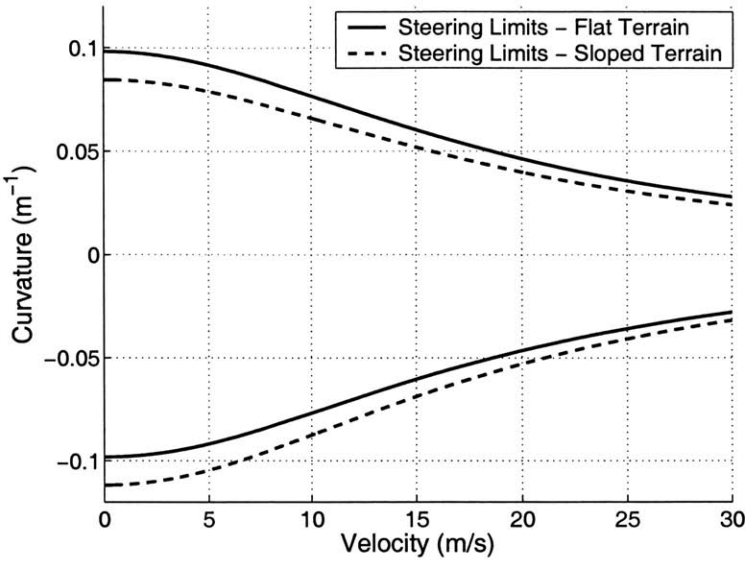


**Figure 2.16: Comparison of steering limits for an understeered vehicle on flat and sloped terrain**

### 2.2.4 Drive Train Constraint Computation

A rear-drive front-steered UGV's maximum forward velocity is a function of the power train (consisting of the engine and transmission) characteristics, rolling resistance, aerodynamic drag, and terrain inclination (see Figure 2.17).

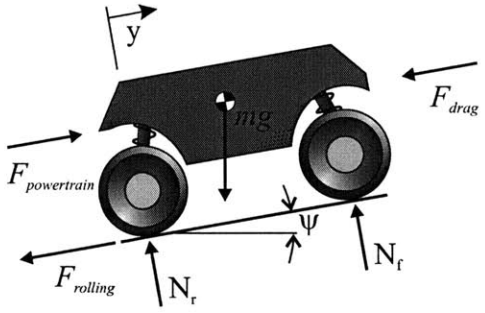


**Figure 2.17: Drive train vehicle model**

Summing the forces in the y-direction yields:

$$m\ddot{y} = F_{powertrain} - F_{drag} - F_{rolling} - mg\sin\psi \tag{2.23}$$

A simple model of the force generated by a power train is:

$$F_{powertrain} = \frac{P}{v} \tag{2.24}$$

where $P$ is the power output of the power train. The power is given as:

$$P = T(\omega)\omega \tag{2.25}$$

where $T(\omega)$ is the engine torque as a function of speed and $\omega$ is the engine angular velocity. A vehicle's longitudinal velocity is related to the engine speed for a fixed gear ratio as:

$$\omega = \frac{Gv}{r} \tag{2.26}$$

where $G$ is the gear ratio, and $r$ is the wheel radius. Combining Equations 2.24, 2.25, and 2.26 yields:

$$F_{powertrain} = \frac{T(v)G}{r} \tag{2.27}$$

The drag force on a vehicle can be modeled as [88]:

$$F_{drag} = \frac{1}{2}\rho A_r C_d (v)^2 \tag{2.28}$$

where $\rho$ is the density of air, $A_r$ is the reference area of the vehicle, and $C_d$ is the drag coefficient.

The rolling resistance acting on a wheeled vehicle can be modeled as [3]:

$$F_{rolling} = C_{rr} mg \cos\psi \tag{2.29}$$

where $C_{rr}$ is the coefficient of rolling resistance. Combining Equations 2.27, 2.28, and 2.29 and setting $\ddot{y} = 0$ yields the maximum vehicle forward velocity:

$$v_{\max} = \sqrt{\frac{2(T(v)G - rC_{rr}mg\cos\psi - rmg\sin\psi)}{rA_r\rho C_d}} \tag{2.30}$$

A skid-steered vehicle's maximum velocity is also a function of the power train, rolling resistance, aerodynamic drag, and terrain inclination. However, in contrast to a front-steer vehicle, a skid-steered UGV's curvature is a function of the maximum velocity. The following analysis calculates the curvature for a skid-steer vehicle as a function of vehicle velocity (see Figure 2.18).

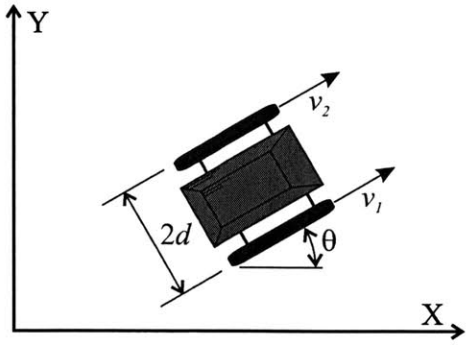


**Figure 2.18: Skid-steer vehicle parameters**

The rate of change of the heading angle of a skid-steered vehicle is given as:

$$\frac{d\theta}{dt} = \frac{v_1 - v_2}{4d} \tag{2.31}$$

The curvature of a skid-steered vehicle is defined as:

$$\kappa = \frac{d\theta}{ds} \tag{2.32}$$

where $s$ is path length. The rate of change in vehicle position is given as:

$$v = \frac{ds}{dt} = \frac{v_1 + v_2}{2} \tag{2.33}$$

Combining equations 2.31, 2.33, and 2.32 yields

$$\kappa = \frac{v_1 - v_2}{2d(v_1 + v_2)} \tag{2.34}$$

An example of the steering constraints for a skid-steered vehicle is shown in Figure 2.19.

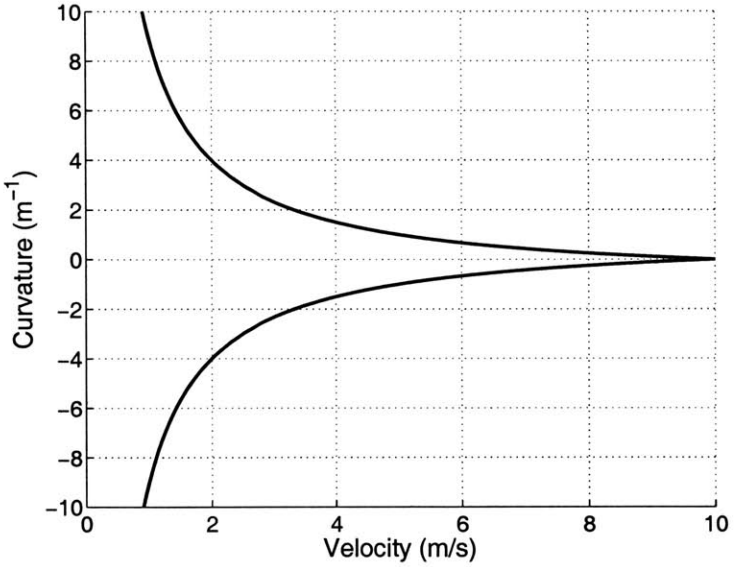


**Figure 2.19: Drive train limits for a skid-steered vehicle (track width = 1 m; $v_{max}$ = 10 m/s)**

As expected the skid-steered vehicle can obtain infinite curvature at zero forward velocity due to its ability to turn in place. However, at its maximum forward speed the attainable curvature is zero. Note that as the maximum UGV velocity changes, the curvature limits also change.

**Definition 2.6 (Drive Train Space)** *Let the drive train space, D, be the set of velocity and curvature pairs that are attainable give a vehicle's power train limits:*

$$D(\phi, \psi, \varpi, \mu, \kappa) \equiv \{\forall v \mid 0 \le v \le v_{max}\} \tag{2.35}$$

where $v_{max}$ is the maximum vehicle velocity.

### 2.2.5 Dynamic Trajectory Space

**Definition 2.7 (Dynamic Trajectory Space)** *The dynamic trajectory space, Γ, is defined as the intersection of the sideslip, A, rollover, B, steering, C, and drive train, D, spaces:*

$$\Gamma \equiv A \bigcap B \bigcap C \bigcap D \tag{2.36}$$

Figure 2.20 shows an example of the dynamic trajectory space (shaded region). The dynamic trajectory space represents all dynamically feasible velocity and curvature pairs for a particular vehicle on a patch of terrain with given inclination and traction coefficient.

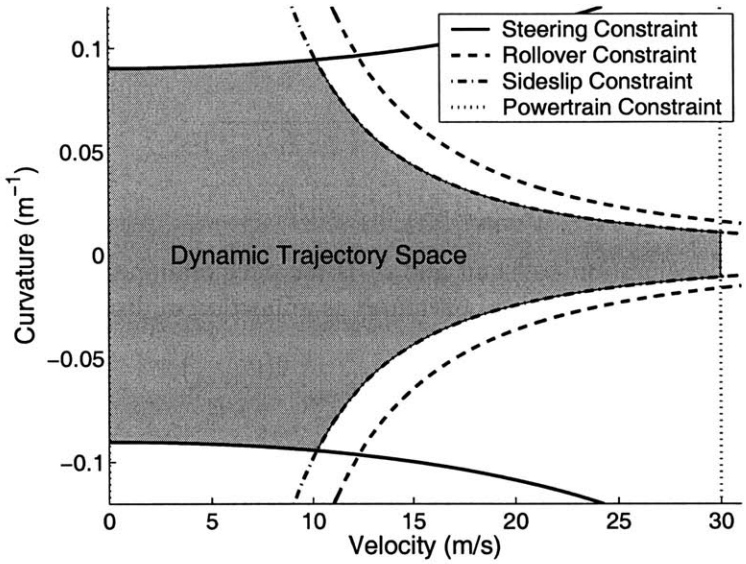


**Figure 2.20: Dynamic trajectory space for an oversteered vehicle on flat terrain**

## 2.3 The Reachable Trajectory Space

The dynamic trajectory space represents the set of velocity and curvature pairs that are dynamically feasible for a given terrain patch. However, not all dynamically admissible trajectories can be reached in a fixed time *t*. Understanding which trajectories can be reached in a fixed time is particularly important when a vehicle is faced with an impending hazard. A UGV's steering rate, acceleration, and deceleration must be considered.

**Definition 2.8 (Reachable Trajectory)** *A reachable trajectory, $\tau_r(s)$, is a trajectory that can be transitioned to, or reached, in a given time, t.*

A reachable trajectory is a function of a UGV's instantaneous curvature and velocity, and its acceleration, braking, and steering characteristics. For a UGV with a location inside the trajectory space of $(v_0, \kappa_0)$, the maximum attainable velocity in time *t* is:

$$v_{reachable}^{\max,\min} = v_0 \pm at \tag{2.37}$$

where *a* is the acceleration/deceleration constant and constant acceleration/deceleration capabilities are assumed. The maximum and minimum attainable curvatures for a front-steered vehicle in time *t* are:

$$\kappa_{reachable}^{\max,\min}(v) = \kappa_0 \pm \dot{\kappa}_{\max} t \tag{2.38}$$

where $\dot{\kappa}_{\max}$ is the maximum rate of change of curvature given as :

$$|\dot{\kappa}_{\max}| = \frac{\tan \dot{\delta}_{\max}}{L} \tag{2.39}$$

where $\dot{\delta}_{\max}$ is the maximum steering rate of change.

Determining the set of reachable trajectories is useful when a UGV is faced with an impending hazard. Consider the situation where a hazard has been identified along the path of a UGV at a distance $d_{hazard}$. The maximum and minimum attainable velocities and curvatures before the UGV reaches the hazard can be computed as a function of distance:

$$\begin{aligned} v_{reachable}^{\max} &= \sqrt{v_0^2 + 2a_p(d_{hazard})} \\ v_{reachable}^{\min} &= \sqrt{v_0^2 + 2a_b(d_{hazard})} \end{aligned} \tag{2.40}$$

where $a_p$ is the acceleration constant and $a_b$ is the deceleration constant. Similarly, the maximum and minimum attainable curvatures as a function of distance are given as:

$$\kappa_{reachable}^{\max,\min}(v) = \kappa_0 \pm \dot{\kappa}_{\max}\left(\frac{2(d_{hazard})}{v + v_0}\right) \tag{2.41}$$

A similar analysis can be performed for a skid-steered vehicle.

**Definition 2.9 (Reachable Trajectory Space)** *The reachable trajectory space, Λ, is the set of admissible velocity and curvature pairs a vehicle can transition to in a given time, t.*

Figure 2.21 shows and example of a reachable trajectory space in addition to a dynamic trajectory space. Here the initial trajectory is $\tau_0 = (20, 0.01)$ and $d_{hazard} = 30$ m. The acceleration constant is $a_p = 1.5\ \text{m/s}^2$ and the braking constant is $a_b = -6.6\ \text{m/s}^2$. This corresponds to $t \approx 1.4$ s if the vehicle accelerates and $t \approx 2.7$ s if it decelerates. These values represent close approximations of actual values of a HMMWV-class vehicle. The maximum rate of change of curvature is $\dot{\kappa}_{\max} = 0.05\ \text{m}^{-1}/\text{s}$ based on an estimation of the vehicles capabilities.

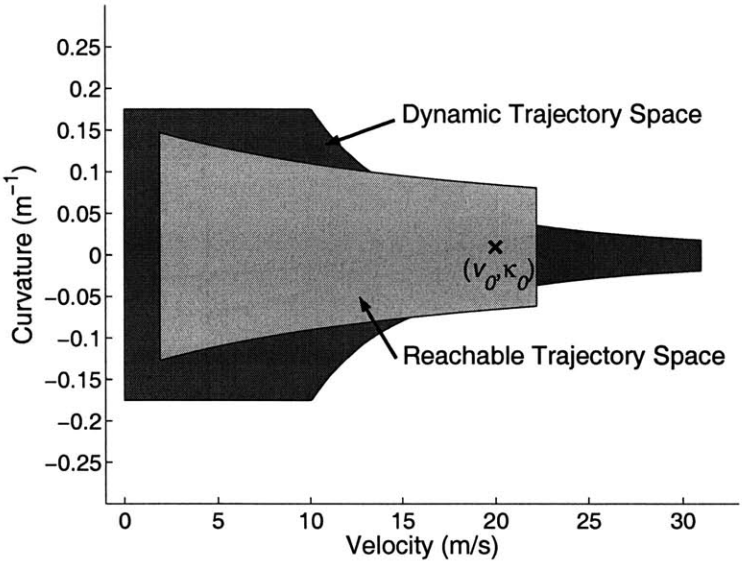


**Figure 2.21: Reachable trajectory space for a HMMWV class vehicle**

Figure 2.22 shows an example of a reachable trajectory space for a skid-steered vehicle. Here the initial trajectory is $\tau_0 = (7, -0.14)$. The acceleration and braking constants for each wheel are both $a = 1.0\ \mathrm{m/s^2}$ and the maximum velocity for each wheel is $v_{\max} = 10.0\ \mathrm{m/s^2}$. The space represents all trajectories that can be reached in $t = 1.5\ \mathrm{s}$.

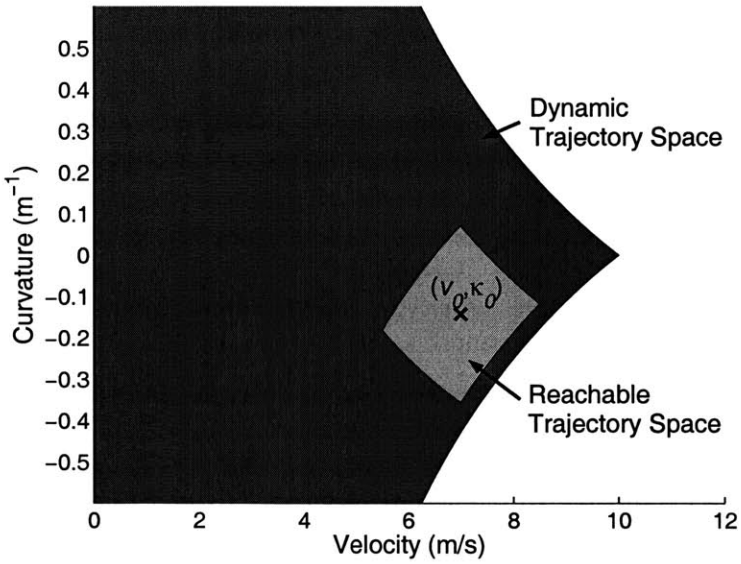


**Figure 2.22: Reachable trajectory space for a skid-steered vehicle**

## 2.4 The Admissible Trajectory Space

**Definition 2.10 (Admissible Trajectory Space)** *The admissible trajectory space, $\Theta$, is the intersection of the dynamic trajectory space and the reachable trajectory space:*

$$\Theta = \Gamma \cap \Lambda \tag{2.42}$$

The admissible trajectory space represents all dynamically safe velocity and curvature pairs that can be transitioned to in a given time. An example is shown in Figure 2.23.

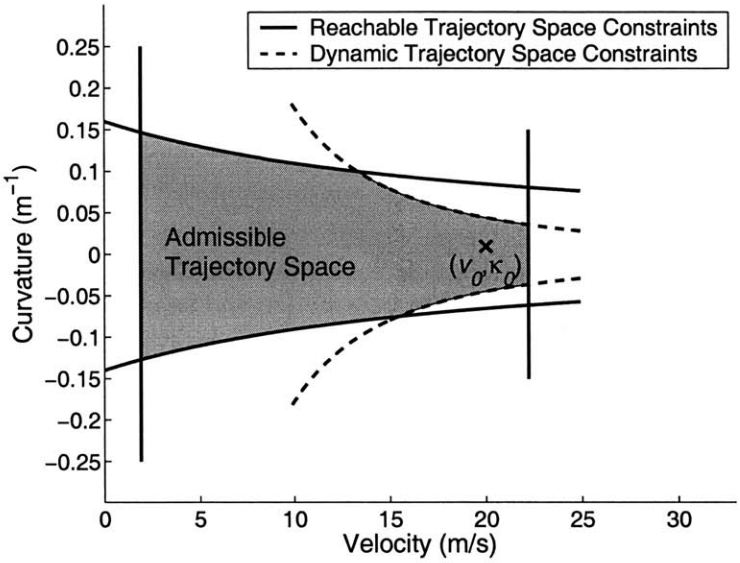


**Figure 2.23: Admissible trajectory space for a HMMWV class vehicle**

## 2.5 The Hazard Trajectory Space

**Definition 2.11 (Hazard)** *A hazard is a discrete object or terrain feature that significantly impedes or halts vehicle motion.*

**Definition 2.12 (Hazard Trajectory Space)** *The hazard trajectory space, $\Omega$, consists of curvatures and velocities that, if maintained from the current vehicle position, would lead to intersection with a hazard.*

$$\Omega(\varpi) \equiv \left\{ \forall (v,\kappa) \,|\, v_{hazard}^{\min} \le v \le v_{hazard}^{\max}, \kappa_{hazard}^{\min} \le \kappa \le \kappa_{hazard}^{\max} \right\} \tag{2.43}$$

Hazards are separated into two types in this thesis: trajectory independent and trajectory dependent. A vehicle cannot safely traverse a trajectory independent hazard at any speed or curvature. Examples include trees and large bushes. In contrast, some hazards can be safely traversed at certain vehicle velocities and curvatures but not others. These are termed trajectory dependent hazards. Examples include ditches and hillocks.

**Definition 2.13 (Trajectory Independent Hazard)** *A trajectory independent hazard is a hazard that a vehicle cannot safely travel across, over, or through.*

A UGV must avoid trajectory independent hazards. Examples of trajectory independent hazards include trees, boulders, and water traps. Figure 2.24 illustrates a trajectory independent hazard that disallows the range of curvatures between $\kappa_{hazard}^{\max}$ and $\kappa_{hazard}^{\min}$ from the current UGV position. Curvatures between $\kappa_{hazard}^{\max}$ and $\kappa_{hazard}^{\min}$ thus belong to the hazard trajectory space. If a UGV maintains a curvature that lies in the hazard trajectory space, the UGV will collide with the hazard.

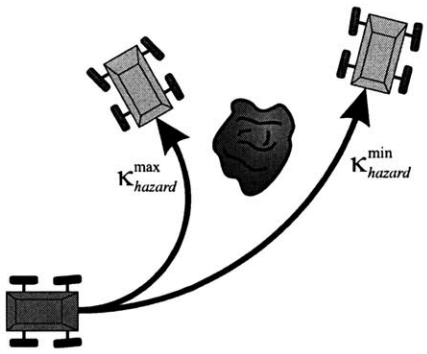


**Figure 2.24: Illustration of maximum and minimum curvatures necessary to avoid impact with a hazard**

To account for vehicle geometry, position estimation error, and path tracking errors, a point vehicle representation is employed. This assumes that a vehicle can be represented as a point, and the perimeter of a hazard is increased proportional to the vehicle size. In this representation a hazard's perimeter may be spatially bounded by a convex polygon, $P_0$, with $n$ edges and vertices. The vertices of an augmented polygon, $P_f$, are formed by increasing the radial distance from the centroid of $P_0$ to each vertex of $P_0$ by $d_{vehicle} + d_{sensor} + d_{control}$ (see Figure 2.25) where:

- $d_{vehicle}$ is the radius of a circle that circumscribes the convex polygon comprised of the outermost points on a vehicle's body;
- $d_{sensor}$ is the radius of the circular error probable (CEP) circle for a UGV's position estimation sensor. The CEP is a measure of horizontal accuracy for a position sensor. It is a circle that encompasses 50% of the sensor's data points taken at a fixed position;
- $d_{control}$ is the maximum lateral path tracking error of the servo-level path tracking controller. The performance can be quantified by the controller's ability to track a desired path. The lateral path tracking error is defined as the distance from the center of a circle that circumscribes the convex polygon comprised of the outermost points on a vehicle's body to the desired path.

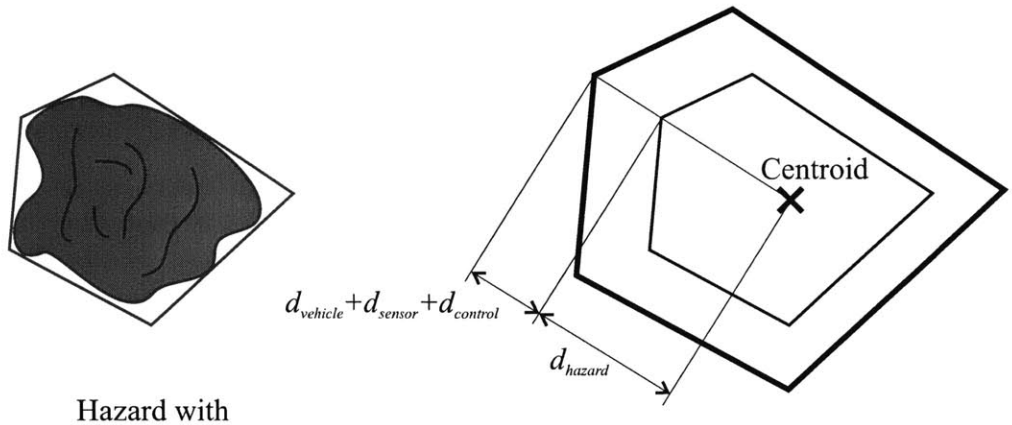


**Figure 2.25: Illustration of hazard size**

The minimum and maximum curvatures associated with a hazard, $\kappa_{hazard}^{\max}$ and $\kappa_{hazard}^{\min}$, are functions of a vehicle's velocity due to the fact that vehicles traveling with a non-zero forward velocity follow a clothoidal path when transitioning between curvatures [65]. This is a result of the fact that a UGV's steering angle can only change with a fixed rate, $\dot{\delta}_{\max}$, and thus the curvature can only change with a fixed rate $\dot{\kappa}_{\max}$. The rate of change in path coordinates is thus given as:

$$\frac{d\kappa}{ds} = \frac{dt}{ds}\frac{d\kappa}{dt} = \frac{\dot{\kappa}}{v} \tag{2.44}$$

Thus at high velocities, a UGV cannot change its curvature, and subsequently its heading, in the same distance as it could at low velocities. This is exemplified in the situation shown in Figure 2.26. In this example the initial vehicle position is $(x_0, y_0) = (0,0)$. Its initial curvature is $\kappa_0 = 0$, and its initial velocity varies from 1 to 7 m/s. The steering rate is fixed such that the rate of change of curvature is $\dot{\kappa}_{\max} = 0.03\ \mathrm{m}^{-1}/\mathrm{s}$. The vehicle is commanded to modify its curvature such that $\kappa_f = 0.05$ while maintaining a constant velocity.

The top subplot of Figure 2.26 shows the resulting curvatures as a function of vehicle path length. The bottom subplot shows the resulting paths. A hazard is shown in the bottom subplot to demonstrate the effect of speed on the hazard trajectory space.

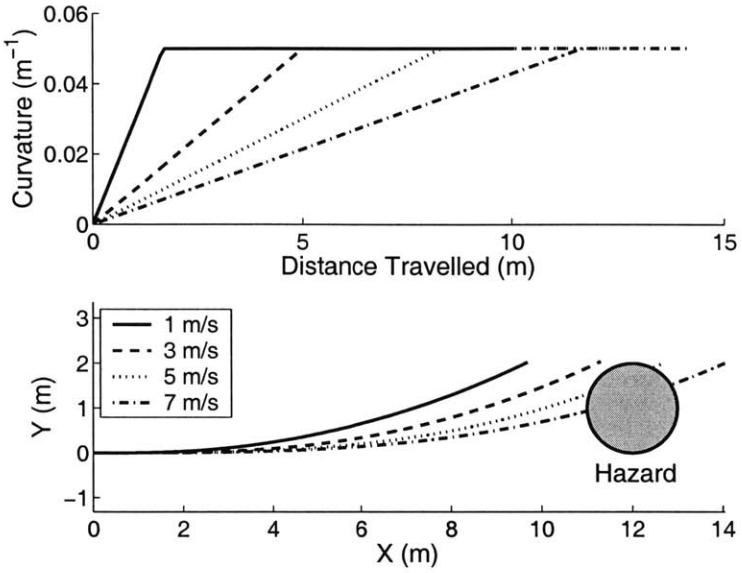


**Figure 2.26: Clothoid paths at varying forward velocities**

An illustration of the hazard trajectory space for a trajectory independent hazard is shown in Figure 2.27. As demonstrated in Figure 2.27, at higher speeds the range of infeasible curvatures increases.

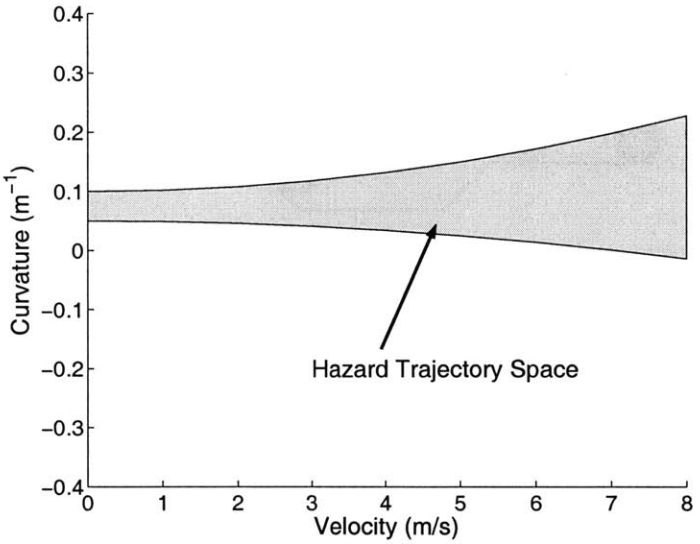


**Figure 2.27: Illustration of hazard trajectory space.**

**Definition 2.14 (Trajectory Dependent Hazard)** *A trajectory dependent hazard is a hazard that a vehicle can travel across, over, or through depending on the vehicle trajectory.*

An example of this is the shallow ditch shown in Figure 2.28.

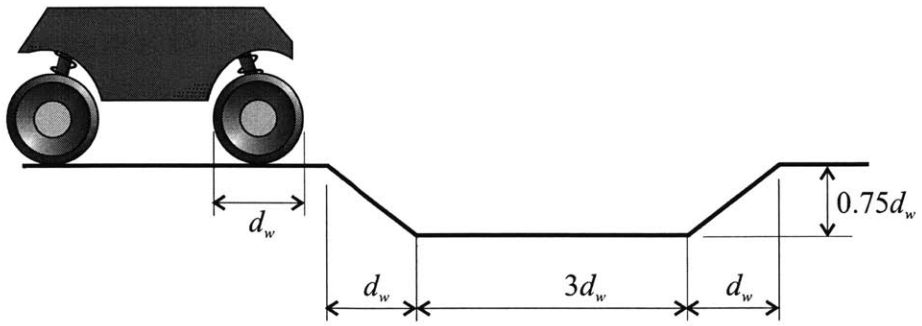


**Figure 2.28: Parameters used for ditch traversal simulation**

At forward velocities less than $v_{low}$, a UGV is able to traverse the hazard (see Figure 2.29). At forward velocities greater than $v_{high}$, the UGV is able to achieve ballistic motion and successfully traverse the ditch. At speeds that lie between $v_{low}$ and $v_{high}$, the UGV will impact the far wall of the ditch [41]. This is considered an unsuccessful traversal.

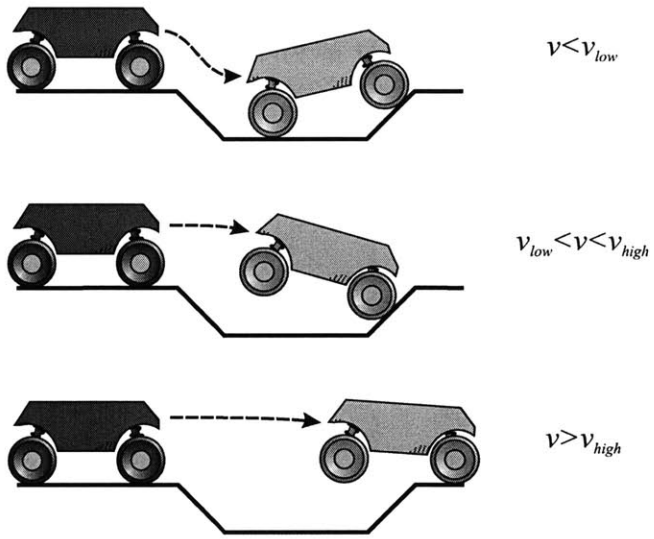


**Figure 2.29: Example of a UGV crossing a shallow ditch at three different forward velocities**

The trajectory space corresponding to such a hazard using the ARTEmiS reference model is shown in Figure 2.30. At speeds below 4.4 m/s the reference model is able to enter the ditch and then climb out. At high speeds (8.8 m/s and greater) there is a set of curvatures that allow ARTEmiS to effectively jump the ditch. Since successful ditch traversal is a function of both vehicle velocity and curvature, at some curvatures ARTEmiS is unable to jump the ditch at any speed.

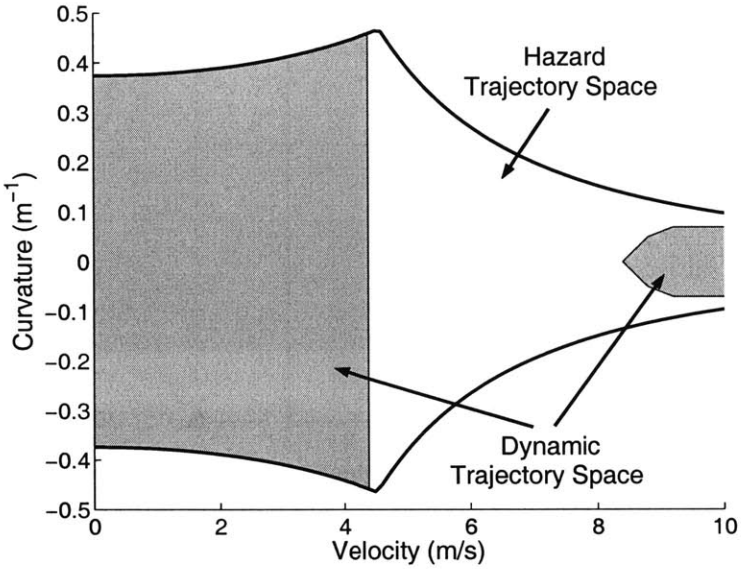


**Figure 2.30: Trajectory space for a trajectory dependent hazard**

The hazard trajectory space is a function of terrain roughness, traction coefficient, angle of approach to the hazard, and hazard geometry for a trajectory dependent hazard. The effect of terrain roughness is investigated in the next section.

## 2.6 Terrain Roughness

This section describes the effect of terrain roughness on the trajectory space constraints presented in the previous sections.

Terrain roughness is dependent on both vehicle speed and geometry. What can be classified as "rough" for a small UGV could be "smooth" for a large UGV. Similarly, terrain features that are considered "rough" for large vehicles may be large enough to be considered hazards for small UGVs. There have been numerous attempts to describe and classify terrain geometry and roughness [4,37,52,55,59,69]. In this thesis, terrain fractal dimension is adopted as a metric for describing terrain roughness. Fractals have the property that they are self-similar [20,53,46]. An object is self-similar if subsets of the object are identical to the object and each other when scaled [85]. It has been shown that natural terrain displays this property [2, 53]. The classic example of this is a coastline that appears similar when viewed from two different distances [53].

Terrain cannot be classified by roughness alone. Roughness is a factor that can exist on either flat or undulating terrain. Terrain roughness is defined here as terrain features on the order of less than one-half the vehicle wheel radius.

### 2.6.1 Fractal Terrain Modeling

In this thesis terrain roughness is generated using the midpoint displacement method [23, 34]. This method begins with a square array of $2^n + 1$ points located a distance $d_t$ apart. The four corner points are set to the same value. An example of this is given for a 5x5 array shown in Figure 2.31.

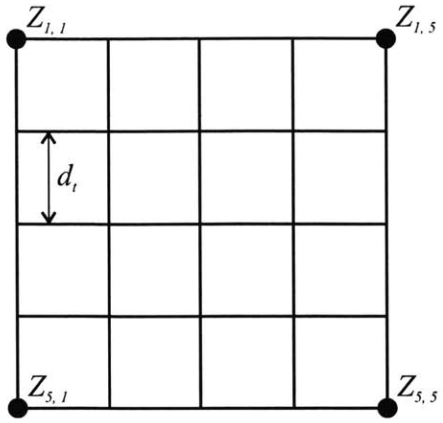


**Figure 2.31: Step one of midpoint displacement method**

The second step generates a height value for the midpoint of the four corners of the square:

$$Z_{3,3} = \frac{1}{4}\left(Z_{1,1} + Z_{1,5} + Z_{5,1} + Z_{5,5}\right) + X_r \tag{2.45}$$

where $X_r$ is computed from a uniform distribution of points with a probability density function of:

$$P(x) = \begin{cases} 0 & \text{for} \quad x < -a \\ \dfrac{1}{2a} & \text{for} \quad -a < x < a \\ 0 & \text{for} \quad x > a \end{cases} \tag{2.46}$$

where:

$$a = \frac{h_t}{2^{(n-1)(3-\varpi)}} \tag{2.47}$$

where $n$ is the step number, $h_t$ is a height scaling factor, and $\varpi$ is the fractal dimension. Thus, as the number of steps increases the value at which the terrain is randomly perturbed is reduced. Fractal dimensions for 2½-dimensional terrain range between 2.0 and 3.0. Higher fractal dimensions equate to rougher terrain.

At this step, five vertices of the array have height values (see right side of Figure 2.32). The white colored vertices represent vertices that have associated height values. Black represents vertices whose height value is being calculated in the current step. The white vertices form a set of "diamonds." At the perimeter of the array, corners of the diamonds can lie on the other side of the array. The height value of the diamond midpoint is the mean of the height values of the corners of a diamond plus a random displacement described by Equations 2.46 and 2.47.

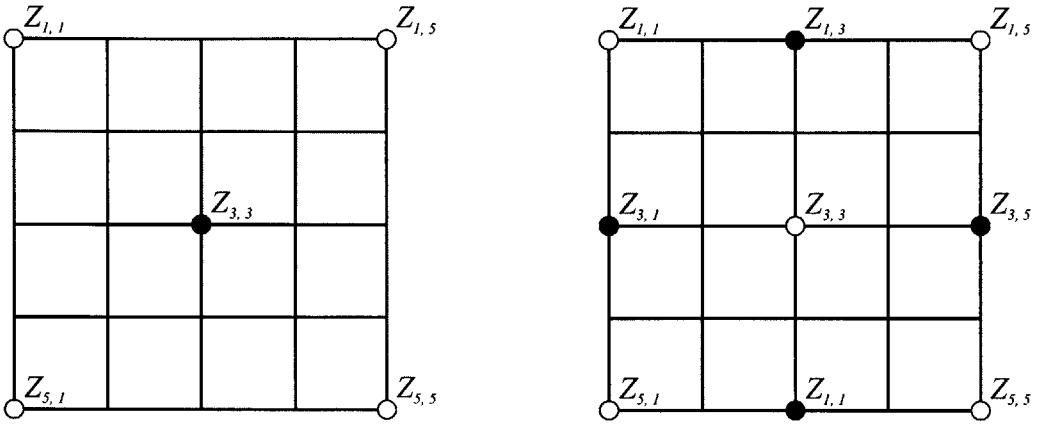


**Figure 2.32: Square step (left) and diamond step (right) for the midpoint displacement method**

These two steps are repeated until all points in the array are given height values as shown in Figure 2.33.

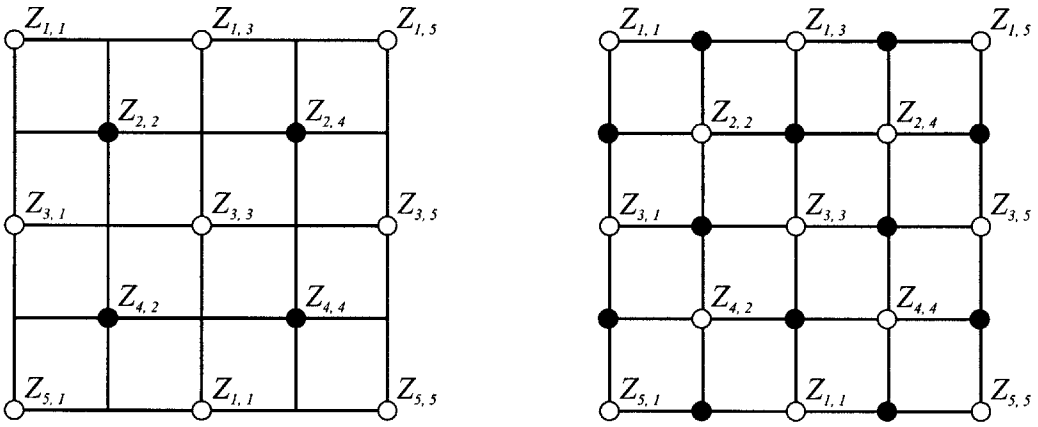


**Figure 2.33: Final steps of the midpoint displacement method**

Figure 2.34 shows six terrains created using this method. In addition to the fractal dimension, the grid spacing, $d_t$, and height scaling, $h_t$, are important. All terrains shown below have a grid spacing of one wheel diameter, and height scaling is proportional to the wheel diameter. The height of terrains shown on the right is scaled five times those shown on the left.

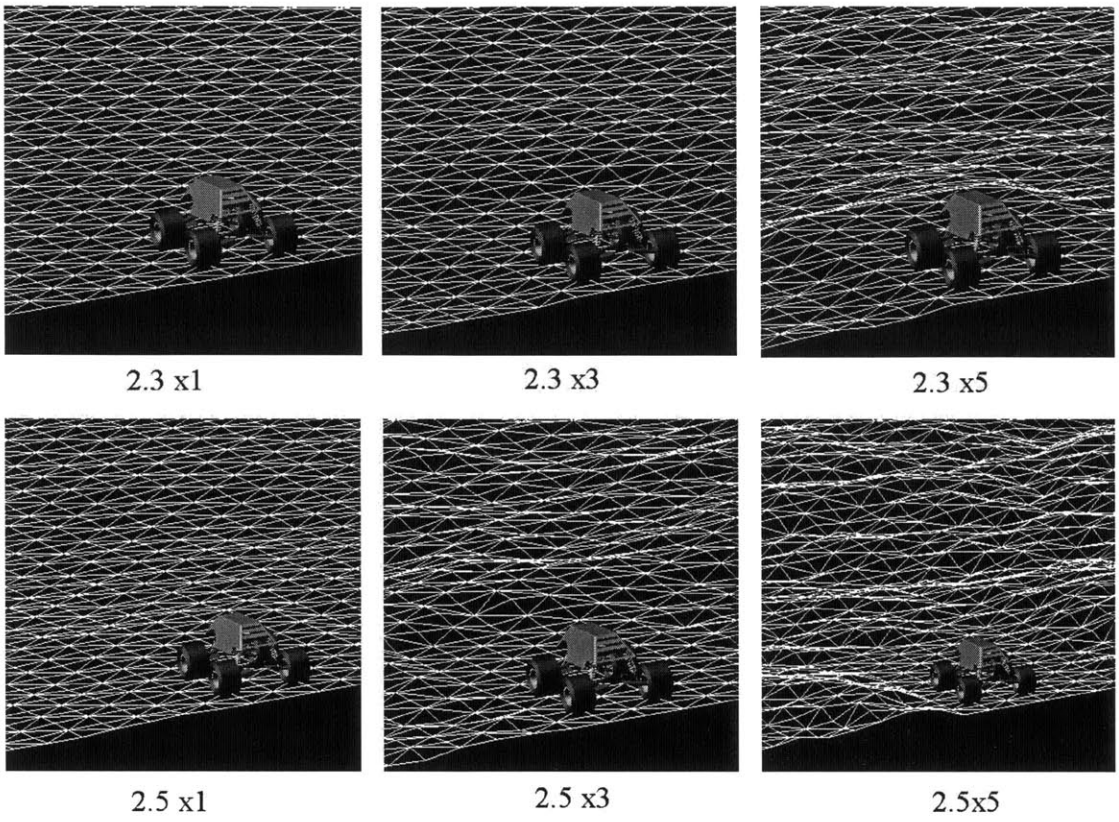


**Figure 2.34: Examples of fractal-generated rough terrain**

To understand the effects of grid spacing and height scaling, it is interesting to examine the power spectral density of the terrain. For one-dimensional fractals, the power spectral density is given as:

$$S_v(f) = Af^{-\varpi} \tag{2.48}$$

where $\varpi$ is the fractal dimension, $f$ is the frequency, and $A$ is proportional to the height scaling. In this case the fractal dimension varies from 1.0 to 2.0. For 2½-dimensional fractals, the power spectral density is given as:

$$S_v(f_x, f_y) = A\left(f_x^2 + f_y^2\right)^{\varpi-4} \tag{2.49}$$

where $f_x$ is the frequency in the x direction and $f_y$ is the frequency in the y direction. The effects of grid spacing and height scaling can be observed in the power spectral density plot of a fractal profile (see Figure 2.35). When the height scale is increased the frequency content of the terrain remains the same, but the power increases. When the grid spacing is reduced, the power spectral density shifts to the right, indicating an increase in power of higher-frequency components.

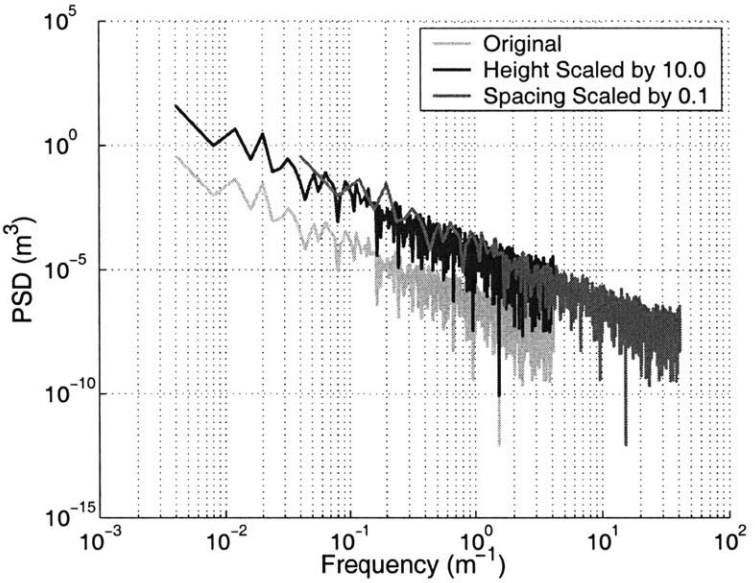


**Figure 2.35: Power spectral density of fractal terrain created with different height scale and grid spacing factors**

Changing the fractal dimension changes the slope of the power spectral density plot. Increasing the fractal dimension causes higher frequency components to become more dominant, as expected (see Figure 2.36).

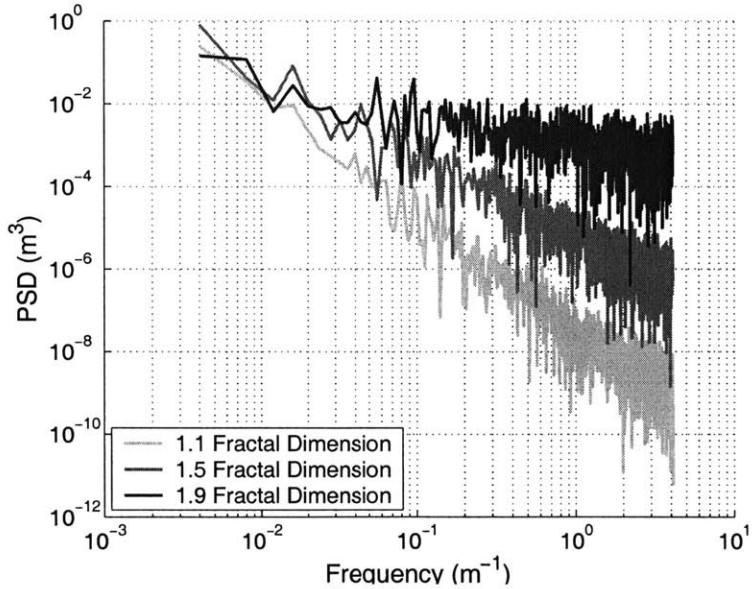


**Figure 2.36: Power spectral density of varying fractal dimensions**

## 2.6.2 The Effect of Roughness on the Hazard Trajectory Space

Following the work of Golda [34] a Monte Carlo method has been developed to study the effects of rough terrain on the hazard trajectory space constraints. The methodology is as follows:

1. Generate a terrain profile of a nominal hazard (e.g. the shallow ditch described in Figure 2.28) with no terrain roughness.
2. Generate a set of $n$ rough terrain profiles using the midpoint displacement method with random seed values. This results in terrains with unique profiles but identical fractal dimensions.
3. Generate a set of $m$ velocities, $\mathbf{v}$, over the interval $[v_{\min}, v_{\max}]$ where $v_{min}$ is the lowest non-zero velocity of interest and $v_{max}$ is the maximum vehicle forward velocity. The size of the set is proportional to the desired resolution of the analysis.
4. Generate a set of $j$ curvatures, $\boldsymbol{\kappa}$, over the interval $[\kappa_{\min}, \kappa_{\max}]$ where $\kappa_{min}$ is the minimum attainable curvature and $\kappa_{max}$ is the maximum attainable curvature given the steering space limits of the UGV. The size of the set is proportional to the desire resolution of the analysis.
5. Overlay the terrain profile of the nominal hazard on to each rough terrain profile. This results in a set of $n$ rough terrain profiles.
6. Command a UGV to traverse each terrain profile for each velocity in $\mathbf{v}$ such that the UGV approaches the hazard with each curvature in $\boldsymbol{\kappa}$. This requires $m \times n \times j$ simulation traversals.
7. Hazard traversal is deemed unsuccessful if the UGV experiences a sideslip angle greater than 20°, or undergoes angular rotation about is roll or pitch axes greater than 90°. In general, successful traversal of a hazard is accomplished if a UGV is able to safely track its desired trajectory from start to finish.

An example of this technique is shown below. Terrain roughness with a fractal dimension of 2.1, a grid spacing of one wheel diameter, and height scaling of two wheel diameters was overlaid onto the hazard illustrated in Figure 2.37.

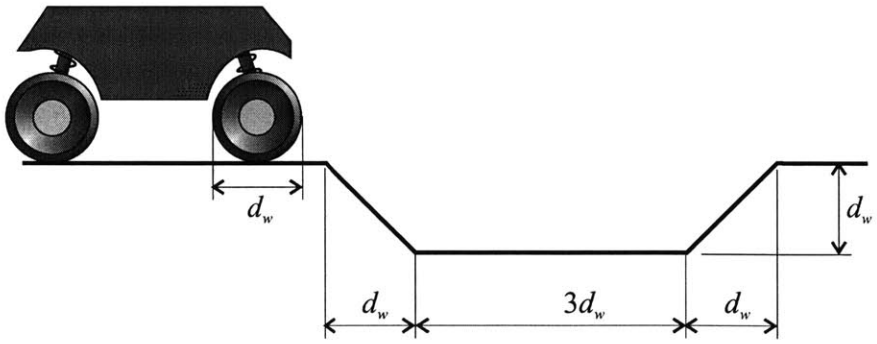


**Figure 2.37: Parameters used for rough terrain ditch traversal simulation**

Fifty separate terrains, $n = 50$, were created with random seed values and identical roughness. Figure 2.38 shows an example of one such terrain.

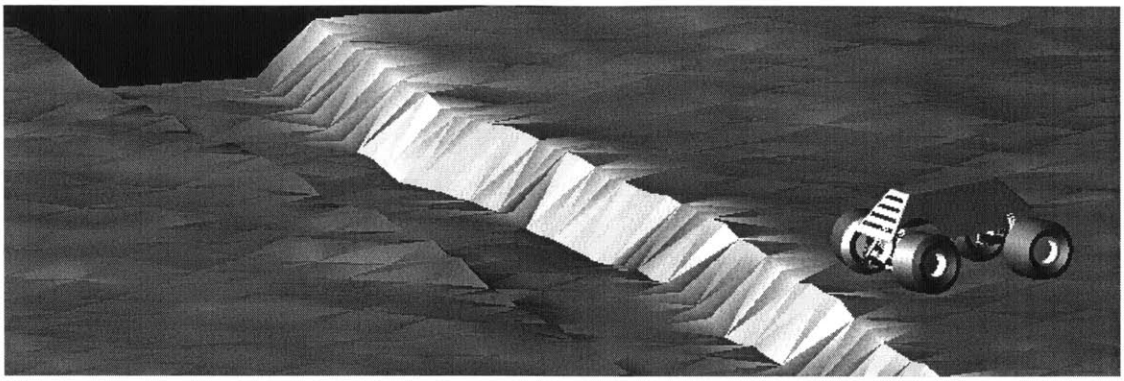

**Figure 2.38: Trajectory dependent hazard (ditch) overlaid with 2.1 fractal roughness.**

A small UGV model (wheelbase = 0.27 m) was commanded to traverse each terrain at $m = 36$ speeds ranging from 0.4 to 7.6 m/s and at a single curvature value of $\kappa = 0$ such that $j = 1$. Figure 2.39 shows a plot of the percentage of successful traversals as a function of speed along with an associated trajectory space representation. The left side of the figure shows a comparison of traversal successes for the "idealized" hazard (i.e. with no terrain roughness) and the rough terrain hazard. For the idealized case, hazard traversal success can be viewed as deterministic (dotted line) since knowledge of the UGV forward velocity specifies success or failure of the traverse. For the rough terrain case, traversal success can be viewed as probabilistic (solid line) since knowledge of the UGV forward velocity does not explicitly specify success or failure of the traverse. Instead, the result of the rough terrain analysis can be viewed as a probability of successful traversal.

The right side of Figure 2.39 shows the dynamic and hazard trajectory spaces corresponding to this example. The hazard trajectory space is represented in grayscale, with white equaling a zero probability of success and black representing a probability of success of one.

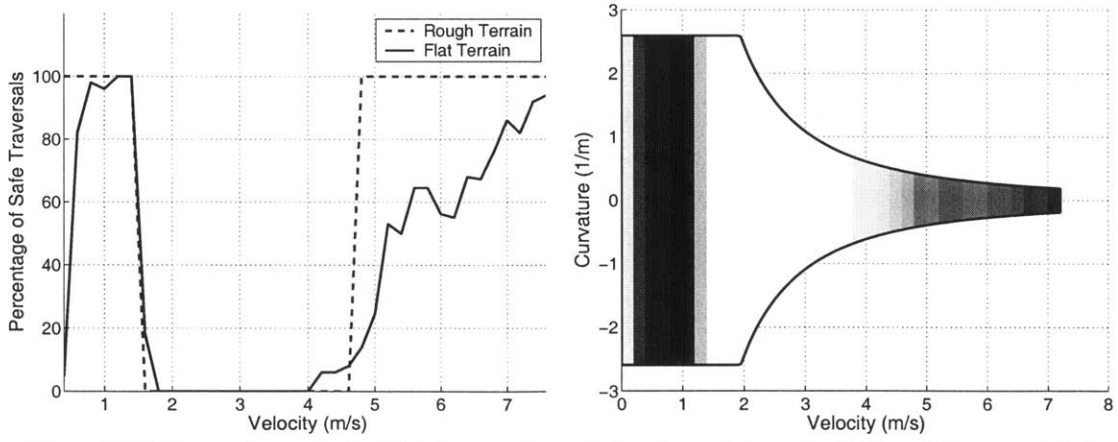


**Figure 2.39: Percentage of safe ditch traversals on flat and rough terrain (left) and the associated rough terrain trajectory space (right).**

At high speeds on rough terrain, UGV success in traversing a hazard is sensitive to small terrain variations. UGV dynamic response to these small variations would be difficult to predict deterministically, and thus a probabilistic representation is more useful

than a "go/no-go" analysis based on the results of a single dynamic simulation. By performing these simulations *a priori*, various canonical types of hazards (i.e. ditches, bumps, hillocks, etc.) can be evaluated at different geometries and gradations of terrain roughness, and their hazard trajectory spaces would be stored in a look-up table.

The hazard trajectory space for rough terrain can be viewed from the standpoint of risk. For example, if speed for a given application is important, it may be reasonable to define the hazard trajectory space constraints where the probability of success is moderate but not extremely high. If safety is critical, the hazard trajectory space constraints might be placed where the probability of success is near 1.

### 2.6.3 The Effect of Roughness on the Dynamic Trajectory Space

The effect of terrain roughness on the dynamic trajectory space rollover constraints was studied through simulations of the ARTEmiS reference model on terrains with fractal roughness dimensions of 2.1, 2.3, and 2.5. The grid spacing and height scaling were set to one wheel diameter. The reference model was commanded to follow a nominal desired path consisting of a 20 m straight line joined with a clothoid described by $\kappa(s) = -0.01s$.

The resulting curvature profile and path are shown in Figure 2.40.

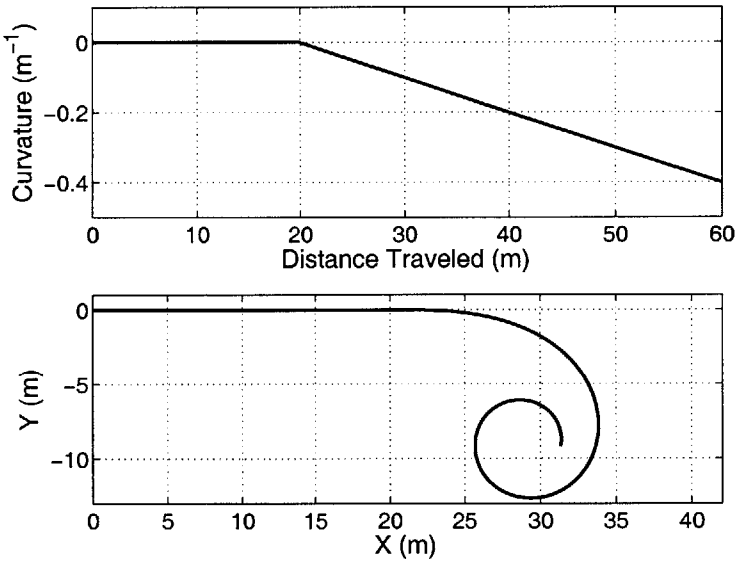


**Figure 2.40: Curvature and path used for rough terrain rollover simulations**

This path allowed the reference model to reach a desired speed of 7 m/s before entering an increasingly sharp turn. The curvature and velocity pair at the point of rollover was recorded. Rollover was defined as initiating when $a_{lateral} = dg/h$ where $a_{lateral}$ is the lateral acceleration of the UGV, $d$ is one-half the UGV track width, $g$ is gravity, and $h$ is the height of the center of mass. Each trial was checked to confirm that rollover was initiated at that moment. The velocity at rollover varied due to errors in the servo-level speed control. The results are shown in Figure 2.41.

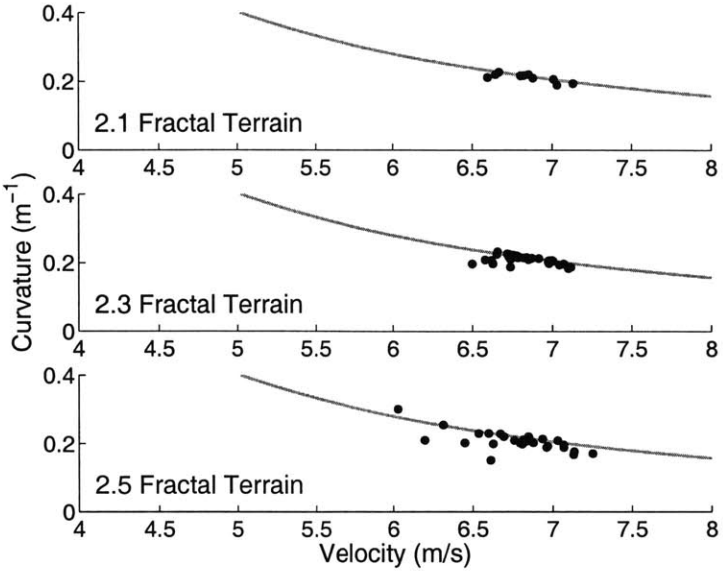


**Figure 2.41: Comparison of rollover constraints on rough terrain**

The effects of the vehicle suspension are included in this analysis by the nature of the Monte Carlo approach. The suspension acts to both allow for increased velocities at a given curvature by reducing vehicle roll and by reducing the acceptable velocity at a given curvature due. However, in general as terrain roughness increases, the variation in velocity and curvature that initiated rollover increases and the velocity at which rollover occurs for a given curvature decreases. The mean and standard deviation of these results are given in Table 2.1.

Table 2.1: Mean and standard deviation for rollover curvature error on rough terrain

| | **2.1 Fractal Dim.** | **2.3 Fractal Dim.** | **2.5 Fractal Dim.** |
|---|---|---|---|
| **Mean** $(m^{-1})$ | 0.005 | 0.007 | 0.013 |
| **Standard Deviation** $(m^{-1})$ | 0.008 | 0.012 | 0.020 |

At low roughness (2.1 fractal dimension), 80% of the rollovers occurred within 5% of the value predicted by the simple model. As the roughness increased (2.3 fractal dimension), that value dropped to 76%. At the highest roughness evaluated here (2.5 fractal dimension) only 52% of the rollovers occurred within 5% of the value predicted using the rigid rollover model.

Clearly the variation in a UGV's dynamic response due to small deviations in terrain geometry makes it difficult to accurately predict rollover using a simple model. However here again, a probabilistic representation is arguably more useful than a deterministic one. In the proposed approach, a constraint is derived from the results of a Monte Carlo analysis. The locations of constraints can be selected according to mission-level objectives. Thus, this type of constraint can be viewed from the perspective of risk. For example, if safety was critical to the mission, it might be reasonable to set the rollover constraints to include $3\sigma$ of the curvature and velocity pairs that have been found to induce rollover on terrain of a given roughness. An example of this is shown in Figure 2.42.

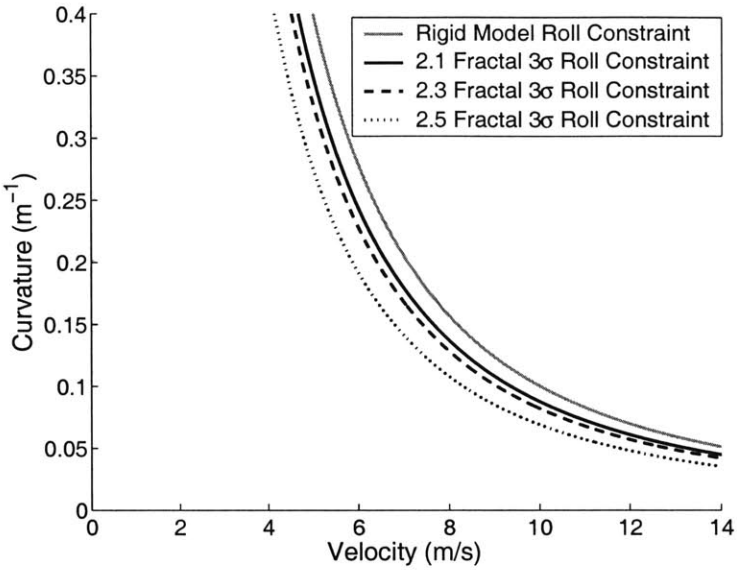


**Figure 2.42: Rollover limits for varying rough terrain**

If speed for a particular application were more important than safety, it might be appropriate to set the rollover constraints to include $1\sigma$ or $2\sigma$ of the curvature and velocity pairs that have shown to induce rollover on terrain of a given roughness. This would allow a UGV to attain higher speeds at a given curvature.

## 2.7 Summary

The trajectory space is a concise representation of a UGV's capabilities on a given terrain patch, and the basis of the hazard avoidance algorithm. It can account for terrain properties such as inclination, roughness, and traction coefficient. The trajectory space can also account for vehicle dynamic effects such as rollover, sideslip, and power train limits and can consider vehicle steering kinematics and dynamics. Furthermore, both trajectory dependent and trajectory independent hazards can be easily represented on the trajectory space.

In this chapter, the construction of the trajectory space constraints is presented. It is shown that the constraints can be computed using simple models, and that these models agree with results of detailed nonlinear reference models on relatively smooth terrain. In addition, the effect of terrain roughness on the trajectory space constraints is demonstrated in a simulation environment using detailed models. A risk-based technique for derivation of trajectory space constraints on rough terrain is briefly discussed.

# CHAPTER 3

# MANEUVERS

Maneuvers are modifications of a UGV's nominal trajectory initiated by the presence of a hazard or dynamically unsafe trajectory. This chapter describes two specific types of maneuvers: hazard avoidance maneuvers and path resumption maneuvers. A hazard avoidance maneuver avoids local hazards not accounted for by a global path planner and prevents a UGV from executing dynamically unsafe trajectories. A path resumption maneuver connects the end of the path generated by a hazard avoidance maneuver to the nominal desired path.

A discussion of hazard detection and conditions for enacting a hazard avoidance maneuver are presented first. Second, a method for selecting a hazard avoidance maneuver is described. These methods are based on the trajectory space analysis described in Chapter 2. Last, a novel method for generating a path resumption maneuver, the curvature matching method, is introduced, and its performance is analyzed.

## 3.1 Hazard Avoidance Maneuvers

During high-speed navigation, emergency situations are likely to occur that require a UGV to rapidly perform a hazard avoidance maneuver. These situations arise when a hazard lies on a vehicle's nominal desired path or a portion of the nominal desired trajectory lies outside of the dynamic trajectory space for a particular terrain patch. These situations can occur for several reasons. First, for outdoor operations, a nominal desired path is typically created using elevation data from topographical maps. If the data is out-of-date, changes in the terrain (e.g. fallen trees or human activity) may render a path unsafe. Moreover, if the elevation data provided by the topographical maps is sparse, the nominal desired path generated by the global path planner may not properly account for local terrain inclination or small hazards. Second, sensors may not have adequate acuity to resolve and identify hazards at a distance. Thus, some hazards may be identified only at close range. This requires quick evasive action from a UGV. Last, the physical terrain properties might be different from those estimated at the time the nominal desired path was generated. For example, weather conditions may change a terrain patch from dry and compact to wet and muddy.

**Definition 3.1 (Maneuver)** *A maneuver, $\chi$, is a transition from one location in the trajectory space to another.*

A maneuver is characterized by the following:

- An initial trajectory, $\tau_0 = (v_0, \kappa_0)$.

- A final trajectory, $\tau_f = (v_f, \kappa_f)$.
- A distance $s$ over which the maneuver occurs.

The requirements of a maneuver are thus given as:

$$\chi = (v_0, v_f, \kappa_0, \kappa_f, s) \tag{3.1}$$

**Definition 3.2 (Hazard Avoidance Maneuver)** *A hazard avoidance maneuver, $\chi_h$, is a transition from a UGV's current location in the trajectory space to a safe location in the trajectory space.*

A description of a "safe" location is given in Section 3.1.2. This definition does not include a designation of the curvature or velocity profiles that compromise the maneuver. Thus, any curvature and velocity profile that end in a safe location in the trajectory space can be construed as a hazard avoidance maneuver.

The hazard avoidance maneuver described in this section is a specific type of hazard avoidance maneuver that is comprised of two segments. This type of hazard avoidance maneuver transitions a UGV to a safe location in the trajectory space in the least amount of time possible. The first segment consists of a transition between the initial, $\tau_0 = (v_0, \kappa_0)$, and final trajectory, $\tau_f = (v_f, \kappa_f)$ at a constant $d\kappa/ds$. The second segment consists of a fixed velocity and curvature over a set distance (see Figure 3.1). The total length of the maneuver is $s = s_f - s_0$

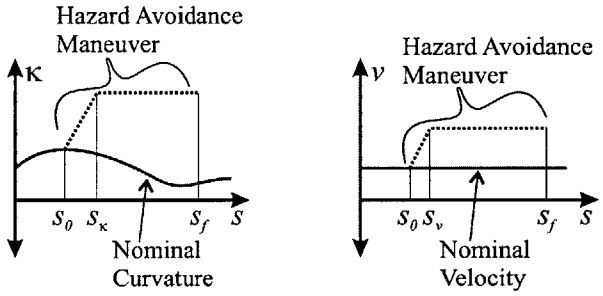


**Figure 3.1: Hazard avoidance maneuver curvature (left) and velocity (right) profiles**

The velocity profile of a hazard avoidance maneuver is described by:

$$\chi_h(v,s) = \begin{cases} v_0 + \dfrac{dv}{ds}(s - s_0) & \text{for} \quad s_0 < s \le s_v \\ v_f & \text{for} \quad s_v < s \le s_f \end{cases} \tag{3.2}$$

The curvature profile of a hazard avoidance maneuver is described by:

$$\chi_h(\kappa,s) = \begin{cases} \kappa_0 + \dfrac{d\kappa}{ds}(s - s_0) & \text{for} \quad s_0 < s \le s_k \\ \kappa_f & \text{for} \quad s_k < s \le s_f \end{cases} \tag{3.3}$$

where $s_0$ is the initial starting point of the maneuver, $s_f$ is the final point of the maneuver, and $s_v$ and $s_\kappa$ are given as:

$$s_v = s_0 \pm \frac{v_f^2 - v_0^2}{2a} \qquad s_\kappa = s_0 \pm \bar{v}\left(\frac{\kappa_f - \kappa_0}{\dot{\kappa}_{\max}}\right) \tag{3.4}$$

where $a$ is the acceleration constant of the vehicle, and $\bar{v}$ is the average velocity over the time it takes to reach the final curvature, $(\kappa_f - \kappa_0)/\dot{\kappa}_{\max}$.

The two fundamental issues regarding hazard avoidance maneuvers are 1) hazard detection, and 2) maneuver selection. These are discussed below.

### 3.1.1 Hazard Avoidance Maneuver Execution

Consider a scenario similar to that illustrated in Figure 3.2. A UGV attempts to follow a pre-planned nominal desired path, $\mathbf{x}_{nominal}$, given by a high-level path planner. As discussed in Section 1.1, the nominal desired path has a corresponding nominal desired trajectory, $\tau_{nominal}$. An on-board forward-looking range sensor measures terrain elevation and detects hazards. The terrain is divided into discrete patches, and the admissible trajectory space corresponding to each patch is computed. The size and number of these patches as well as a method to compute each patch's roll and pitch angles is described in Appendix A.

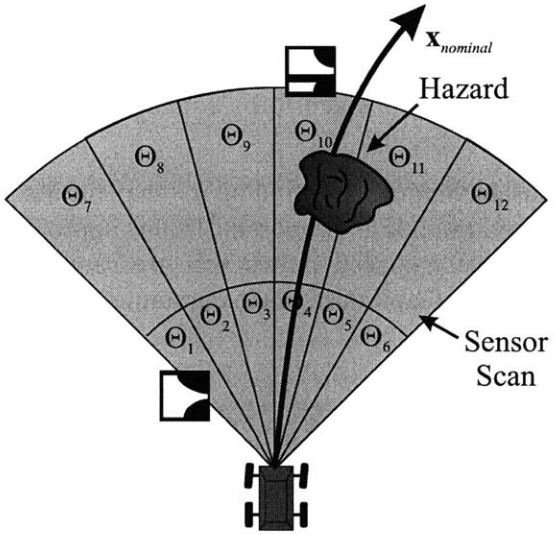


**Figure 3.2: Nominal desired path with terrain patches.**

Let $n$ be the number of terrain patches that $\mathbf{x}_{nominal}$ intersects within the current sensor scan. Let $\Theta_i$ denote the admissible trajectory space for a patch that $\mathbf{x}_{nominal}$ intersects, $i \in [1, n]$.

A maneuver is enacted if one of two situations occurs:

- A hazard lies on the vehicle's current desired path, $\mathbf{x}_{nominal}$.
- An element of $\tau_{nominal}$ whose corresponding path, $\mathbf{x}_{nominal}$, lies in the $i^{th}$ terrain patch violates a constraint on $\Theta_i$. In other words, a UGV is commanded to follow a dynamically inadmissible trajectory.

### 3.1.2 Hazard Avoidance Maneuver Selection

After the decision has been made to execute a hazard avoidance maneuver, a particular maneuver must be selected. This is done using the trajectory space analysis described in Chapter 2. First, a hazard trajectory space is created. Next an admissible trajectory space is created for each terrain patch. The admissible trajectory space for each terrain patch is dependent on the terrain inclination, roughness, and traction coefficient of each individual patch. Let $m$ be the total number of terrain patches, and let $\Theta_j$ denote the admissible trajectory space for each terrain patch, $j \in [1, m]$. Let $p$ be the number of hazards in the current sensor scan, and let $\mathrm{H}_k$ denote the hazard trajectory space for each hazard, $k \in [1, p]$.

**Definition 3.3 (Total Admissible Trajectory Space)** *The total admissible trajectory space,* $\mathrm{Z}$*, is defined as the intersection of all admissible trajectory spaces for the set of terrain patches minus the hazard trajectory space,* $\mathrm{H}$*, for each hazard in the current sensor scan:*

$$\mathrm{Z} \equiv (\Theta_1 \cap \ldots \cap \Theta_m) - \mathrm{H}_1 - \mathrm{H}_2 - \ldots - \mathrm{H}_p \tag{3.5}$$

The total admissible trajectory space is a representation of all feasible UGV trajectories from the UGV's current position. The goal of the hazard avoidance algorithm is to find a maneuver, $\chi_h$, such that $\tau_f = (v_f, \kappa_f) \in \mathrm{Z}$. The maneuver thus transitions a vehicle to a safe location inside the total admissible trajectory space. The length of the hazard avoidance maneuver, $s = s_f - s_0$, is taken to be equal to the Euclidian distance from the UGV's current location to the hazard.

There are numerous techniques for finding $\tau_f$ that results in a "good" maneuver. Here a simple search based method is proposed. In this approach, the trajectory space is first discretized into $n$ closely spaced points that are sampled evenly in the velocity dimension and as the inverse of curvature in the curvature dimension (see Figure 3.3).

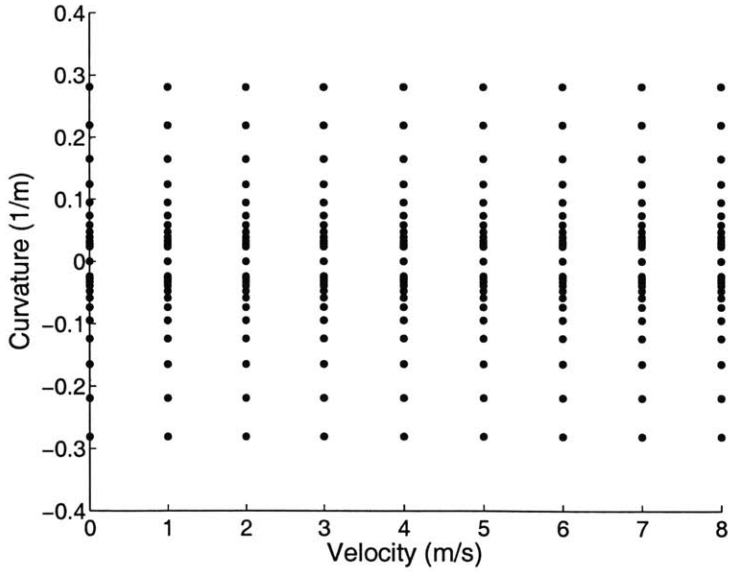


**Figure 3.3: Discrete points on the trajectory space for ARTEmiS**

This non-uniform distribution in the curvature dimension results in a uniform spatial distribution of resulting path curvature. Figure 3.4 shows the resulting possible paths generated from the sampled points shown in Figure 3.3 for two different initial curvatures. When $\kappa_0 = 0$, the feasible UGV maneuvers are symmetric about zero curvature. When the initial curvature is non-zero, the potential maneuvers are asymmetrical due to the resulting asymmetry in the allowable range of steering angles.

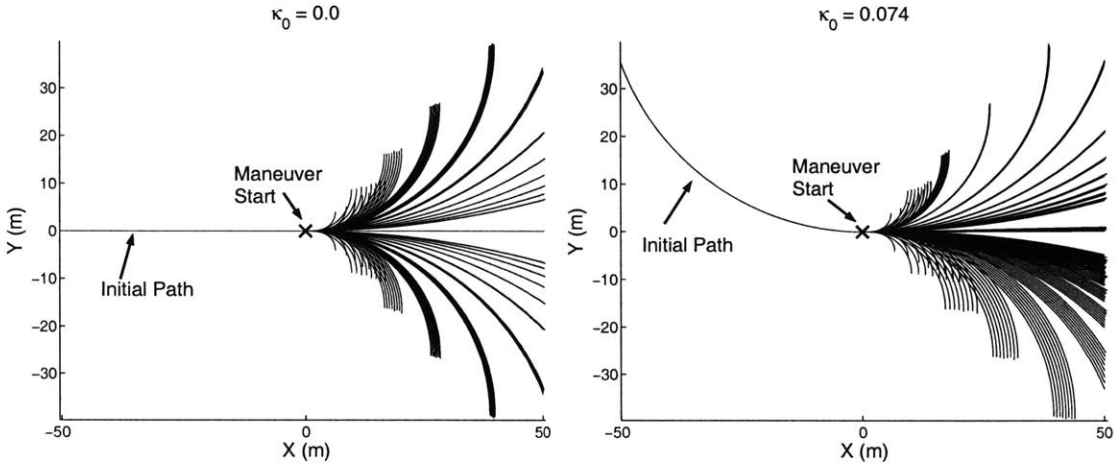


**Figure 3.4: Paths generated from discrete points for $\kappa_0 = 0$ (left) and $\kappa_0 = 0.074$ (right)**

From this discretized trajectory space, $\tau_f$ is chosen as the point that minimizes the distance, $\Delta$, from the current location in the trajectory space, $\tau_0 = (v_0, \kappa_0)$, to a candidate point, $\tau_i = (v_i, \kappa_i)$, $i \in [1, n]$:

$$\Delta = \sqrt{\frac{K_1}{\kappa_{\max} - \kappa_{\min}}(\kappa_0 - \kappa_i) + \frac{K_2}{v_{\max}}(v_0 - v_i)} \tag{3.6}$$

where $K_1$ and $K_2$ are static positive gain factors. These factors affect the relative weighting of changes in velocity and curvature. An exhaustive search is utilized to find $\Delta$. This is feasible due the small size of the search space. This results in a hazard avoidance maneuver being generated on the order of milliseconds using the on-board computation of ARTEmiS.

The resulting $\tau_f$ represents a dynamically admissible curvature and velocity pair that avoids hazards in the region covered by the current sensor scan. A low-level control algorithm is then employed to command the UGV along the new path.

## 3.2 Path Resumption Maneuvers

After a hazard avoidance maneuver is generated, a UGV must plan a kinematically and dynamically feasible path to return to the pre-planned nominal desired path (see Figure 3.5). Two methods for accomplishing this are discussed here for two types of vehicles: front-steered and skid-steered.

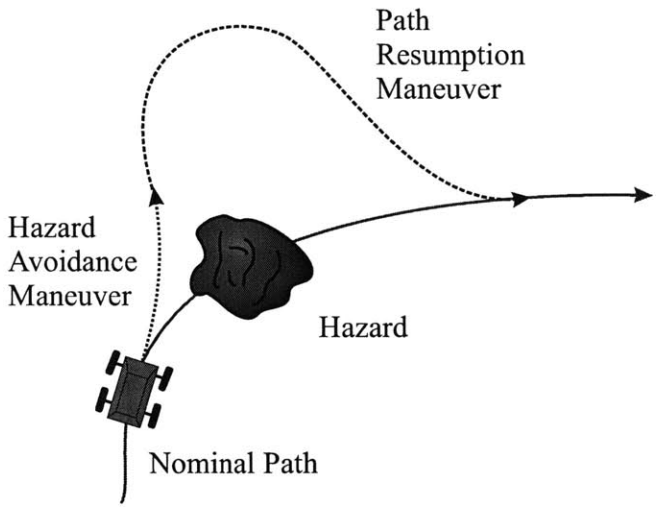


**Figure 3.5: Path resumption maneuver**

The case of a front-steered vehicle (see Figure 2.12) is considered first. Motion of a front-steered vehicle can be described by the following coupled nonlinear equations.

$$\begin{aligned} \kappa(t) &= \tan(\delta(t))/L & \dot{\theta}(t) &= \kappa(t)v(t) \\ \dot{x}(t) &= v(t)\cos\theta(t) & \dot{y}(t) &= v(t)\sin\theta(t) \end{aligned} \tag{3.7}$$

where $\delta(t)$ is the steering input, $\theta(t)$ is the vehicle heading angle, $v(t)$ is the velocity input, and $L$ is the vehicle wheelbase. Assuming that $\delta$ is small and thus $\tan\delta \approx \delta$, the maximum attainable curvature rate of change is:

$$\left|\dot{\kappa}_{\max}\right| = \frac{\dot{\delta}_{\max}}{L} \tag{3.8}$$

The motion of a two wheel skid-steered vehicle (see Figure 2.18) can be described by the following coupled nonlinear equations:

$$\begin{aligned} \kappa(t) &= \frac{v_1(t) - v_2(t)}{2d(v_1(t) + v_2(t))} & \dot{\theta}(t) &= v(t)\kappa(t) \\ \dot{x}(t) &= v(t)\cos\theta(t) & \dot{y}(t) &= v(t)\sin\theta(t) \end{aligned} \tag{3.9}$$

where $d$ is the track width, $v_1$ and $v_2$ are the input velocities, and $v$ is the forward velocity given as:

$$v = \frac{v_1(t) + v_2(t)}{2} \tag{3.10}$$

For a skid-steered vehicle, the maximum curvature rate of change is given as:

$$\frac{d\kappa}{dt} = \frac{d}{dt}\left(\frac{v_1(t) - v_2(t)}{2d(v_1(t) + v_2(t))}\right) \tag{3.11}$$

Given that the maximum rate of curvature change occurs when the wheels accelerate in opposite directions, Equation 3.11 can be simplified to:

$$\left|\dot{\kappa}_{\max}\right| = \frac{a}{2vd} \tag{3.12}$$

where $a$ is the maximum acceleration constant for each wheel.

The equations of motion describing front-steered and skid-steered vehicles (Equations 3.7 and 3.9) are identical except for the curvature input. If the curvature input is described as a general function of time, $u(t)$, integration of Equation 3.7 or Equation 3.9 (with a change of variable from time to distance, and assuming constant velocity) yields:

$$\begin{aligned} \kappa(s) &= u(s) & \theta(s) &= v\int_0^D \kappa(s)ds \\ x(s) &= v\int_0^D \cos\theta(s)ds & y(s) &= v\int_0^D \sin\theta(s)ds \end{aligned} \tag{3.13}$$

This represents the equations of motion for either a front-steered or skid-steered vehicle. The path resumption maneuver can now be generally stated.

Consider the situation illustrated by the plot shown in Figure 3.6. Here the solid line represents a pre-planned nominal trajectory's curvature. A hazard avoidance maneuver is executed at $s_a$, and the maneuver ends at $s_b$. The curvature profiles of the nominal desired path, hazard avoidance maneuver, and path resumption maneuver are defined as $\kappa_1(s)$, $\kappa_2(s)$, and $\kappa_3(s)$. These curvature profiles have associated heading and position profiles, $\theta(s)$, $x(s)$, and $y(s)$. The goal of the path resumption problem is to find $\kappa_3(s)$ in a computationally efficient manner (i.e. completed before the UGV has traversed the hazard avoidance maneuver path) such that:

$$\left(\kappa(s_c),\theta(s_c),x(s_c),y(s_c)\right)_1 = \left(\kappa(s_d),\theta(s_d),x(s_d),y(s_d)\right)_3 \tag{3.14}$$

where $s_c$ is a desired "meeting point" of the replanning maneuver and the nominal trajectory, and $s_d$ is the terminal point of the replanning maneuver. Clearly $s_c$ and $s_d$ need not be coincident.

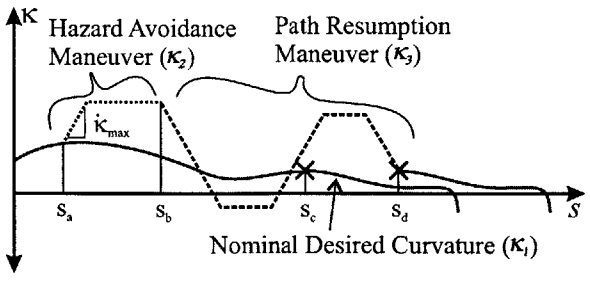


**Figure 3.6: Curvature diagram for path resumption maneuver**

**Definition 3.4 (Path Resumption Maneuver)** *A path resumption maneuver, $\chi_r$, transitions a vehicle from its current trajectory to the nominal desired trajectory.*

A path resumption maneuver has the following properties:

- An initial curvature and velocity equal to the terminal curvature and velocity of a hazard avoidance maneuver, $(v(s_b),\kappa(s_b))_2 = (v(s_b),\kappa(s_b))_3$.
- A length $s_r$ such that $s_r = s_d - s_b$.
- A trajectory profile such that $(\kappa(s_c),\theta(s_c),x(s_c),y(s_c))_1 = (\kappa(s_d),\theta(s_d),x(s_d),y(s_d))_3$.

Two methods for generating a path resumption maneuver are presented here. The first is a novel approach termed the curvature matching method. The second utilizes a simple path tracking control law for nonholonomic vehicles [18,62,63].

### 3.2.1 Curvature Matching Method for Path Resumption

The curvature matching method is outlined as follows:

1. An initial choice of the location of the "meeting point," $s_c$, on the nominal trajectory is made. Here $s_c$ is initially chosen such that path length from the end of the hazard avoidance maneuver to the meeting point is equal to the path length of the hazard avoidance maneuver:

$$s_c = 2s_b - s_a \tag{3.15}$$

2. An initial value of $s_d$ is chosen to be the smallest value such that a vehicle can transition from $\kappa_2(s_b)$ to $\kappa_3(s_d)$ without violating the vehicle's steering rate constraints:

$$s_d = \frac{\kappa_2(s_b) - \kappa_1(s_c)}{v\dot{\kappa}_{\max}} \tag{3.16}$$

Steps 3-6 are designed to find a path resumption maneuver curvature profile, $\kappa_3(s)$, such that the area under the hazard avoidance maneuver curvature profile plus the area under the path resumption maneuver curvature profile is equal to the area under the nominal desired trajectory curvature:

$$\begin{aligned} &\int_{s_a}^{s_c}\kappa_1(s)ds = \int_{s_a}^{s_b}\kappa_2(s)ds + \int_{s_b}^{s_d}\kappa_3(s)ds \\ &\kappa_{\min} \le \kappa_3(s) \le \kappa_{\max} \end{aligned} \tag{3.17}$$

where $\kappa_{min}$ and $\kappa_{max}$ are the minimum and maximum allowable curvatures given by the dynamic constraints of the total admissible trajectory space. Equation 3.17 ensures that the heading angle at the end of the path resumption maneuver is identical to the heading angle of the nominal desired trajectory at the meeting point, $\theta_1(s_c) = \theta_3(s_d)$.

3. Two curvature constraints, $\kappa_{high}$ and $\kappa_{low}$, for the path resumption maneuver are computed (see Figure 3.7). These constraints are defined as the maximum and minimum curvatures that transition the final curvature of the hazard avoidance maneuver, $\kappa_2(s_b)$, to the final curvature of the path resumption maneuver, $\kappa_3(s_d)$, given $\dot{\kappa} = \dot{\kappa}_{\max}$, $\kappa_{high} \le \kappa_{\max}$, and $\kappa_{low} \ge \kappa_{\min}$. The area between the two constraints represents all possible curvatures that can transition $\kappa_2(s_b)$ to $\kappa_3(s_d)$.

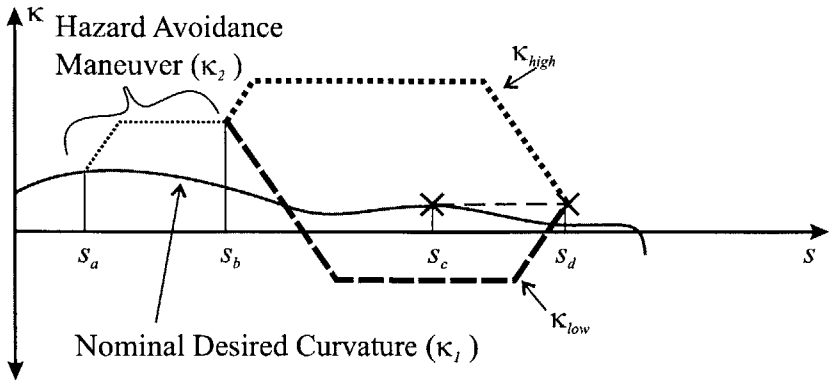


**Figure 3.7: Illustration of curvature matching method constraints**

$\kappa_{high}$ and $\kappa_{low}$ are given as:

$$\kappa_{high} = \begin{cases} \kappa(s_b) + \dfrac{d\kappa}{ds} s & \text{for} \quad s_b < s \le \dfrac{ds}{d\kappa}\left(\kappa_{\max} - \kappa(s_b)\right) \\ \kappa_{\max} & \text{for} \quad \dfrac{ds}{d\kappa}\left(\kappa_{\max} - \kappa(s_b)\right) < s \le \dfrac{ds}{d\kappa}\left(\kappa_{\max} - \kappa(s_d)\right) \\ \kappa_{\max} - \dfrac{d\kappa}{ds} s & \text{for} \quad \dfrac{ds}{d\kappa}\left(\kappa_{\max} - \kappa(s_d)\right) < s \le s_d \end{cases} \tag{3.18}$$

$$\kappa_{low} = \begin{cases} \kappa(s_b) - \dfrac{d\kappa}{ds} s & \text{for} \quad s_b < s \le \dfrac{ds}{d\kappa}\left(\kappa_{\min} - \kappa(s_b)\right) \\ \kappa_{\min} & \text{for} \quad \dfrac{ds}{d\kappa}\left(\kappa_{\min} - \kappa(s_b)\right) < s \le \dfrac{ds}{d\kappa}\left(\kappa_{\min} - \kappa(s_d)\right) \\ \kappa_{\min} + \dfrac{d\kappa}{ds} s & \text{for} \quad \dfrac{ds}{d\kappa}\left(\kappa_{\min} - \kappa(s_d)\right) < s \le s_d \end{cases} \tag{3.19}$$

In general, steps 4-6 generate a curvature profile that either follows the curvature constraints or transitions between them.

4. If the hazard avoidance maneuver has caused a UGV to deviate to the left of the desired nominal path, the path resumption maneuver generates a curvature profile that starts by following the lower curvature constraint, $\kappa_{low}$. This results in a path resumption maneuver that tends toward the right and is thus likely to minimize overall path deviation:

$$\kappa_3(s) = \begin{cases} \kappa_{low}(s) & \text{for} \quad s_b < s \le s_b + \Delta s \\ \kappa_3(s - \Delta s) + \dfrac{ds}{v}\dot{\kappa}_{\max} & \text{for} \quad s_b + \Delta s < s \le s_d \end{cases} \tag{3.20}$$

where $\Delta s$ is the interval defined as:

$$\Delta s = \frac{s_d - s_b}{n} \tag{3.21}$$

where $n$ is a fixed integer that represents the number of subdivisions of the curvature profile. If the UGV is on the right side of the nominal desired path, then the

curvature profile begins by following the upper curvature constraint, $\kappa_{high}$. This results in a path resumption maneuver that tends toward the left:

$$\kappa_3(s) = \begin{cases} \kappa_{high}(s) & \text{for} \quad s_b < s \le s_b + \Delta s \\ \kappa_3(s - \Delta s) - \dfrac{ds}{v}\dot{\kappa}_{\max} & \text{for} \quad s_b + \Delta s < s \le s_d \end{cases} \tag{3.22}$$

5. If at any point the curvature profile of the path resumption maneuver is less than the lower curvature constraint, $\kappa_3(s) < \kappa_{low}(s)$, then the path resumption maneuver curvature is set equal to the lower curvature constraint, $\kappa_3(s) = \kappa_{low}(s)$. Subsequently, if $\kappa_3(s) > \kappa_{high}(s)$ then the path resumption maneuver curvature is set equal to the upper curvature constraint, $\kappa_3(s) = \kappa_{high}(s)$. This ensures the curvature profile does not violate the constraints imposed by the dynamic trajectory space and that the final curvature of the path resumption maneuver, $\kappa_3(s_d)$ is equal to the final curvature of the nominal path, $\kappa_1(s_c)$.
6. If the area under the nominal desired trajectory's curvature profile is within a reasonable tolerance, $\varepsilon_\kappa$, of the area under the hazard avoidance maneuver plus the path resumption maneuver curvature profiles:

$$\int_{s_a}^{s_c} \kappa_1(s)ds - \left( \int_{s_a}^{s_b} \kappa_2(s)ds + \int_{s_b}^{s_d} \kappa_3(s)ds \right) < \varepsilon_\kappa \tag{3.23}$$

then the algorithm continues to step 7. Otherwise $\Delta s$ is increased and steps 4-5 are repeated. This iterative process searches the possible curvature profiles that constitute a path resumption maneuver while maintaining $\dot{\kappa} = \dot{\kappa}_{\max}$ (see Figure 3.8).

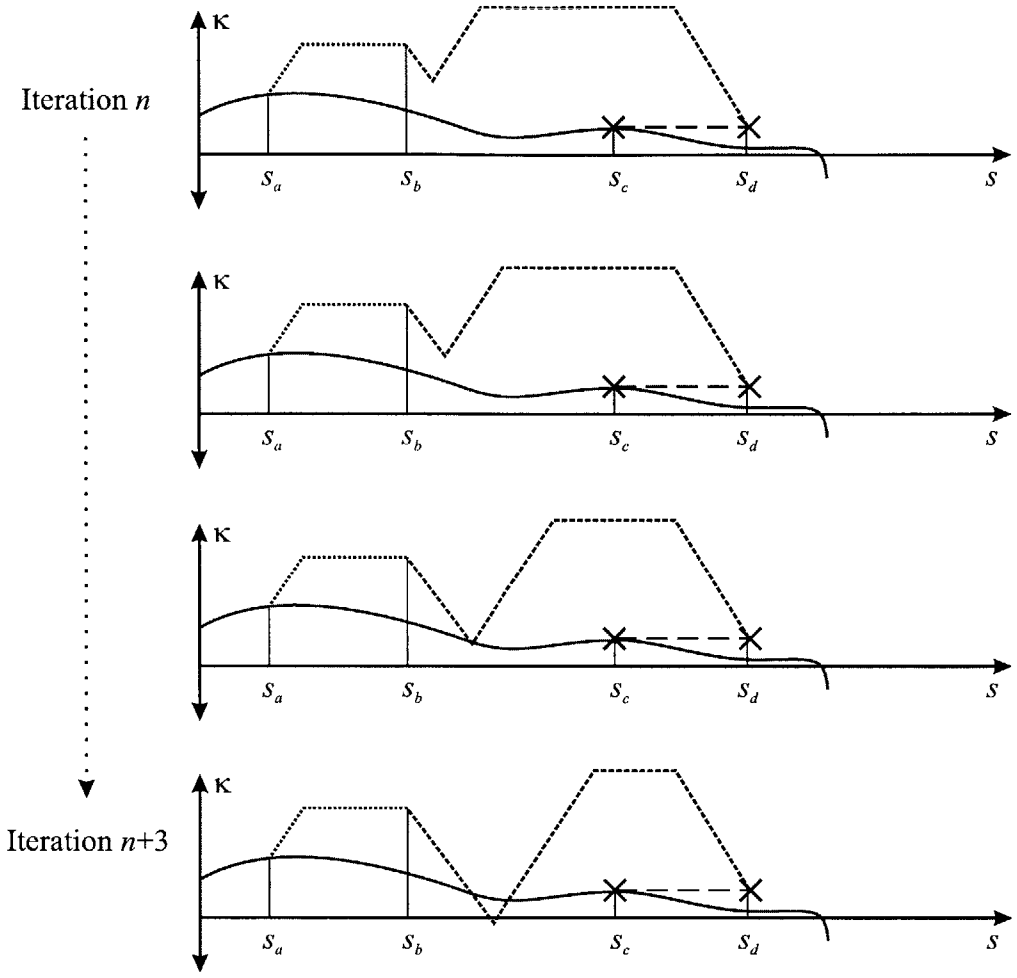


**Figure 3.8: Illustration of iterations of steps 4-5.**

If the entire path resumption maneuver equals either $\kappa_{high}$ or $\kappa_{low}$ before Equation 3.17 is satisfied, the distance for path resumption maneuver, $s_d$, is not large enough. In this case, $\Delta s$ is reset to its original value, $s_d$ is increased by a set length, and the algorithm returns to step 3.

From the path resumption maneuver curvature profile, the heading and position profiles can be generated using Equation 3.13. At this point there is no guarantee that the position of the path resumption maneuver matches the nominal desired path (see Figure 3.9). If they do not match, $s_c$ and $s_d$ are modified based on the Euclidian distance from $(x(s_c), y(s_c))_1$ to $(x(s_d), y(s_d))_3$.

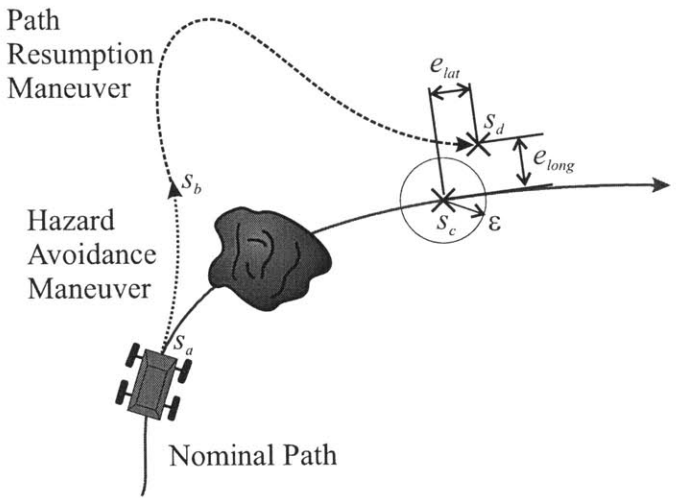


**Figure 3.9: Illustration of curvature matching method path**

7. Define an acceptable threshold for the final position error, $\varepsilon$. If the total position error:

$$e_{total} = \sqrt{(x_1(s_c) - x_3(s_d))^2 + (y_1(s_c) - y_3(s_d))^2} \tag{3.24}$$

lies within a circle of radius ε, then the algorithm is complete. If not, $s_c$ and $s_d$ are adjusted as:

$$\begin{aligned} s_{c_{i+1}} &= s_{c_i} - k_c(e_{lon}) \\ s_{d_{i+1}} &= s_{d_i} - k_d(e_{lat}) \end{aligned} \tag{3.25}$$

where $k_c$ and $k_d$ are static positive gains and $e_{lon}$ and $e_{lat}$ are the longitudinal and lateral error respectively and are given as:

$$e_{lat} = (x_3(s_d) - x_1(s_c))\sin\theta_1(s_c) + (y_1(s_c) - y_3(s_d))\cos\theta_1(s_c) \tag{3.26}$$

$$e_{lon} = (x_1(s_c) - x_3(s_d))\cos\theta_1(s_c) + (y_1(s_c) - y_3(s_d))\sin\theta_1(s_c) \tag{3.27}$$

The algorithm is summarized in a flow chart show in Figure 3.10.

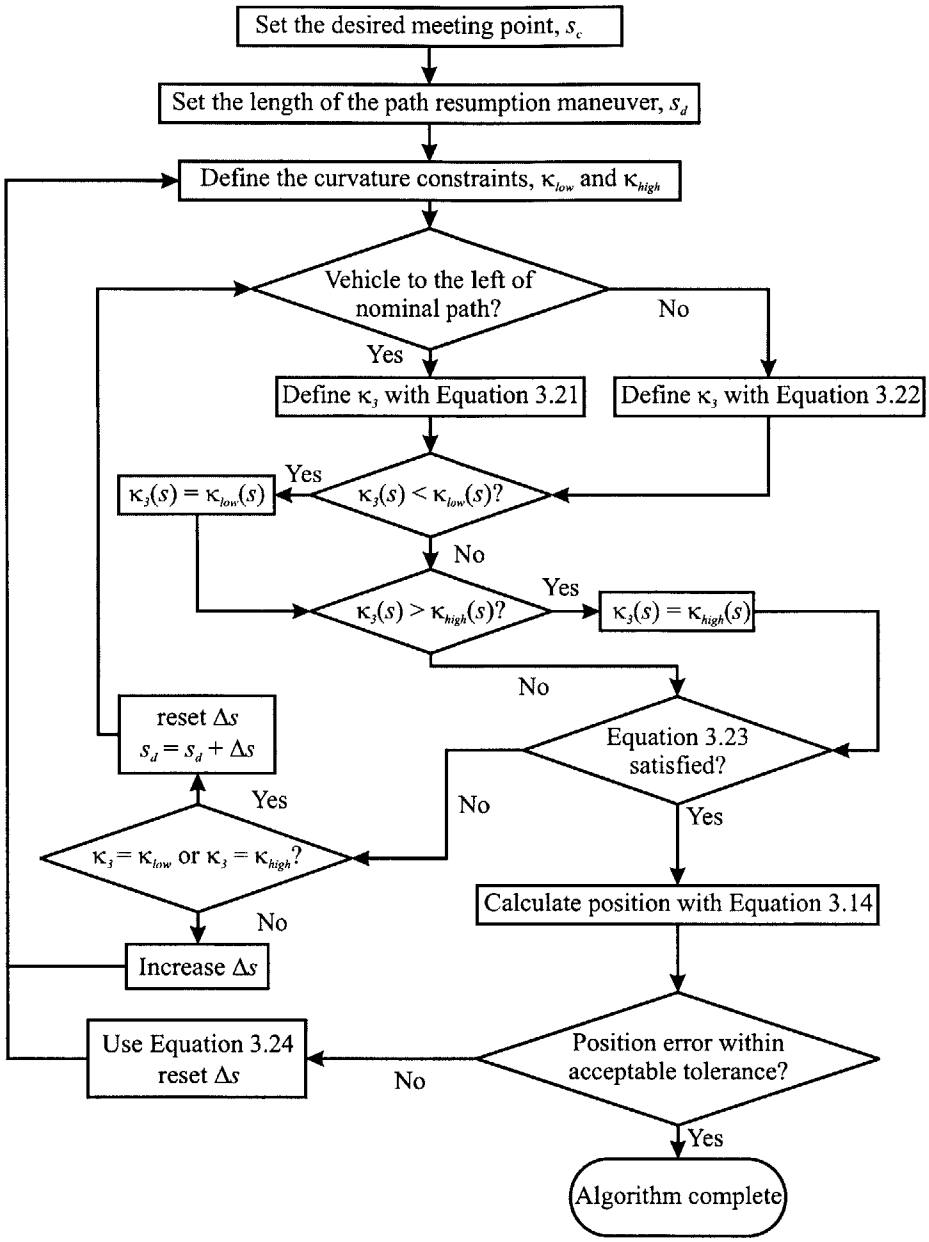


**Figure 3.10: Curvature matching method flow chart**

Two questions regarding this technique addressed here are 1) does the algorithm guarantee convergence? and 2) when the algorithm converges, does it converge to the correct solution?

Due to the fact that the equations of motion are coupled and nonlinear (see Equation 3.13) algorithm convergence cannot be guaranteed. However, the convergence properties have been studied numerically and have yielded 100% convergence for randomly generated paths and randomly generated hazard avoidance maneuvers with a final curvature between -0.2 and 0.2 $m^{-1}$ and length between 4 and 14 m. The results also show

that the algorithm is sufficiently fast for use in high-speed situations. Details of the results are given in Section 3.2.3.

To address the second question, note that from Equation 3.25 as $s_{c_i} \rightarrow s_{c_{i+1}}$ and $s_{d_i} \rightarrow s_{d_{i+1}}$, $e_{lon}$ and $e_{lat}$ approach zero. Using that information and rearranging Equation 3.26 yields:

$$(x_1(s_c) - x_3(s_d)) = \frac{(y_1(s_c) - y_3(s_d))\cos\theta_1(s_c)}{\sin\theta_1(s_c)} \tag{3.28}$$

Combining Equation 3.27 and Equation 3.28 results in:

$$(y_1(s_c) - y_3(s_d))\cos^2\theta_1(s_c) + (y_1(s_c) - y_3(s_d))\sin^2\theta_1(s_c) = 0 \tag{3.29}$$

further reduction yields:

$$y_1(s_c) = y_3(s_d) \tag{3.30}$$

Combining equations 3.26, 3.27, and 3.30 yields:

$$x_1(s_c) = x_3(s_d) \tag{3.31}$$

Thus, when the algorithm converges it converges to the correct solution.

Figure 3.11 shows an example path resumption maneuver generated using the curvature matching method. Note that the nominal path's curvature and heading and the path resumption curvature and heading profiles are identical at points $s_c$ and $s_d$ (upper left and upper right subplots), and points $s_c$ and $s_d$ are coincident along the path (lower subplot).

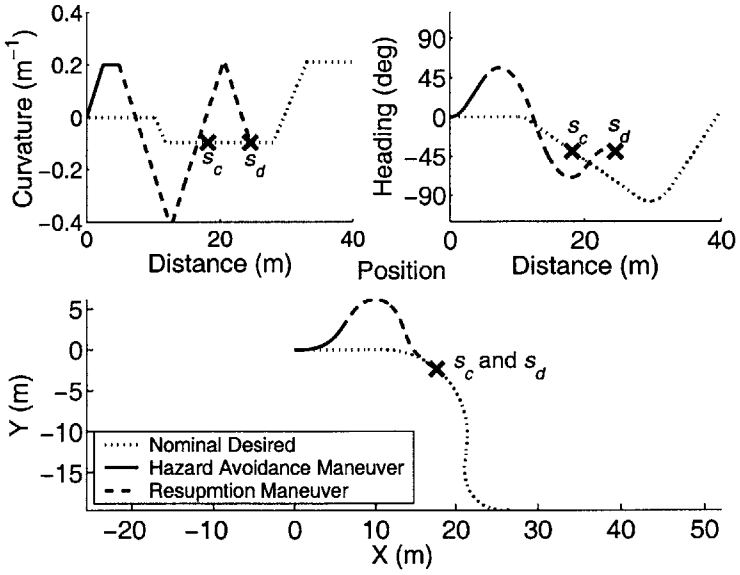


**Figure 3.11: Example of curvature matching method**

In general, the distance required to rejoin the original path is proportional to the maximum curvature rate, $\dot{\kappa}_{\max}$. Compare Figure 3.11 with Figure 3.12 shown below. In

Figure 3.11 the maximum curvature rate was $\dot{\kappa}_{max} = 0.4\ \mathrm{m}^{-1}/\mathrm{s}$. Figure 3.12 shows the same nominal path and hazard avoidance maneuver with $\dot{\kappa}_{max} = 0.2\ \mathrm{m}^{-1}/\mathrm{s}$.

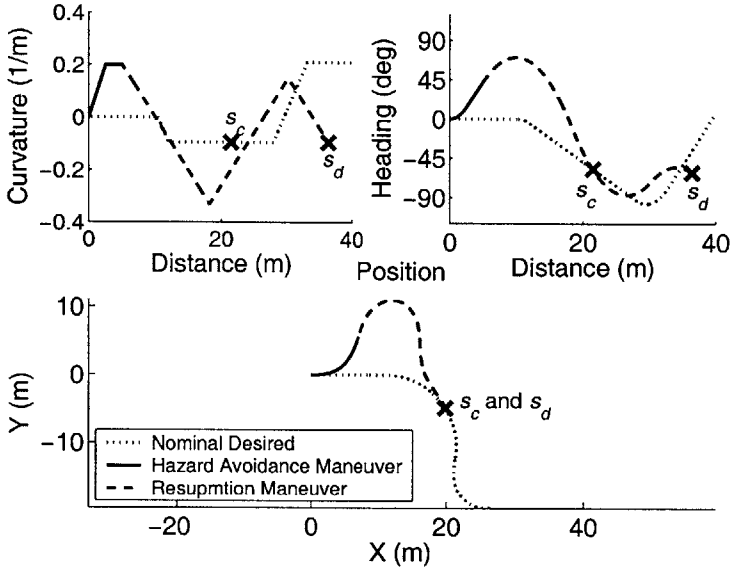


**Figure 3.12: Curvature matching method with low steering rate**

In this particular example, the path resumption maneuver using the slower curvature rate (Figure 3.12) rejoins the nominal path at a different location than the example with the higher curvature rate (Figure 3.11). Also, the resulting path resumption maneuver path is 11.9 m longer in Figure 3.12 than in Figure 3.11.

### 3.2.2 Feedback Control Method for Path Resumption

A second approach to generating a path resumption maneuver employs forward simulation of the model described by Equation 3.13 under a feedback control law. The maneuver is complete when the heading and position errors fall below an acceptable threshold.

Two separate sets of simulation trials were performed. Each utilized a different feedback control law. The first is a linearized feedback control law [18,62,63]:

$$\delta = -K_{p\_\theta}(e_\theta) - K_{p\_l}(e_l) \tag{3.32}$$

where $K_{p\_\theta}$ and $K_{p\_l}$ are static positive gains, $e_\theta$ and $e_l$ are defined as the heading error and lateral error respectively (see Figure 3.13).

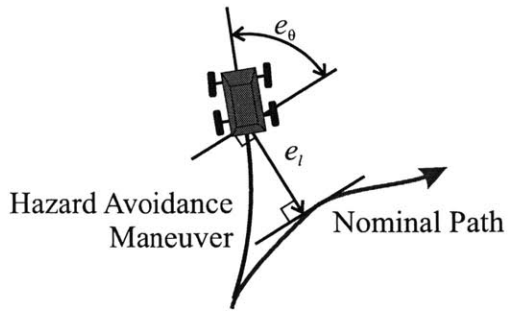


**Figure 3.13: Lateral and position error used in feedback control method**

The curvature rate of change is constrained to be always less than $\dot{\kappa}_{\max}$. It must be noted that the controlled system is only locally asymptotically stable, but it has been shown that the region of asymptotic stability is large [18]. Thus, this method does not guarantee convergence, but results have shown that the convergence properties to be excellent in practice.

The second control law incorporates a damping term to Equation 3.32, which is expected to reduce oscillations on the along the desired path and result in faster convergence times:

$$\delta = -K_{p\_yaw}(e_{yaw}) - K_{d\_yaw}(\dot{e}_{yaw}) - K_{p\_lat}(e_{lateral}) \tag{3.33}$$

Results comparing the two methods are given in the next section.

### 3.2.3 Path Resumption Maneuver Results

This section provides numerical analysis of the performance of the curvature matching method and compares the results to those of the feedback control method. Ten thousand trials of each method were performed on a PIII 1.5 GHz computer. For each trial a nominal desired path was randomly generated with a length of 80 m. A hazard avoidance maneuver was randomly generated with curvature between -0.2 and 0.2 $m^{-1}$ and length between 4 and 14 m. The curvature matching method was deemed complete when:

$$0.4 \geq \sqrt{(x_3(s_d) - x_1(s_c))^2 + (y_3(s_d) - y_1(s_c))^2} \tag{3.34}$$

The feedback control method was deemed complete when:

$$0.4 \geq e_l \quad and \quad 5^\circ \geq e_\theta \tag{3.35}$$

The gains for the linearized feedback controller were $K_{p\_\theta} = 0.1$ and $K_{p\_l} = 0.01$. The gains for the linearized feedback controller with damping were $K_{p\_\theta} = 0.2$, $K_{d\_\theta} = 4.0$, and $K_{p\_l} = 0.02$. These gains were tuned to yield good performance over the given range of nominal desired paths.

Both methods converged to the nominal desired path 100% of the time. This suggests that the convergence properties for the curvature matching method are good in practice. Convergence time results for the two methods are summarized in Table 3.1.

Table 3.1: Convergence computation times

| | Curvature Matching (ms) | Feedback Control (ms) | Feedback Control with Damping (ms) |
|---|---|---|---|
| **Median** | 10 | 1052 | 901 |
| **Mean** | 45 | 994 | 819 |
| **95%** | 171 | 1412 | 1252 |
| **Maximum** | 1522 | 3836 | 5586 |

Consider the example of a HMMWV-class vehicle operating at 20 m/s. If a hazard avoidance maneuver with length $s = 10$ m is executed, the UGV would need to generate a path resumption maneuver in less than 0.5 s. In this common scenario, it is evident from the results shown in Table 3.1 that only the curvature matching method would be suitable.

The difference in convergence time between the two methods is a result of the curvature matching method's relatively few required iterations (average iterations $\approx 10$) compared to the feedback control method (average iterations $\approx 600$). This is partly due to the feedback control method's requirement to modify the curvature based both on heading and position error compared to the curvature matching method's requirement to modify the curvature on position error alone.

Figure 3.14 shows the computation time, path error, and heading error results for the curvature matching method. The top subplot shows a histogram of the convergence time. The bottom left subplot shows a histogram of the position error at algorithm convergence. The bottom right subplot shows the heading error at algorithm convergence.

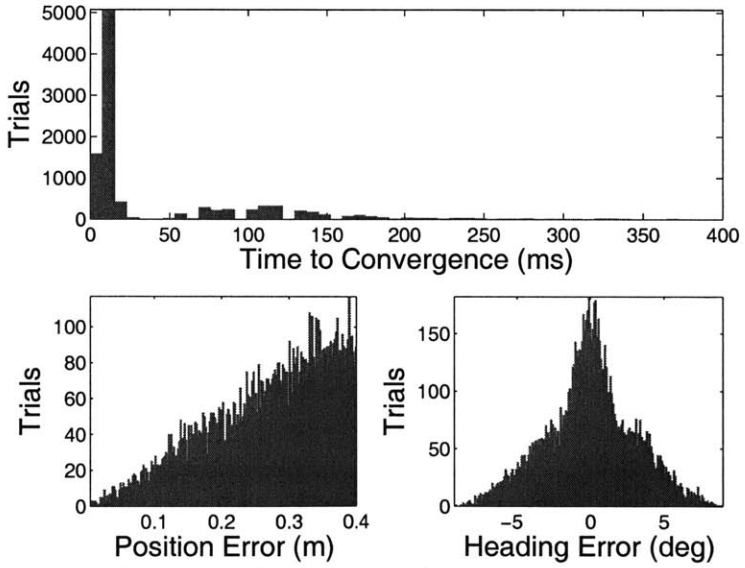


**Figure 3.14: Curvature matching method results**

When the position error falls below 0.4 the algorithm terminates. This explains why many trials have a final position error close to 0.4 m. Furthermore, the algorithm guarantees that the heading error is zero; however, there exists some heading error at convergence due to numerical error.

Figure 3.15 shows similar plots for the feedback control method using each control law. Since the algorithm attempts to satisfy both heading and position errors, the resulting errors lie close to their respective tolerances.

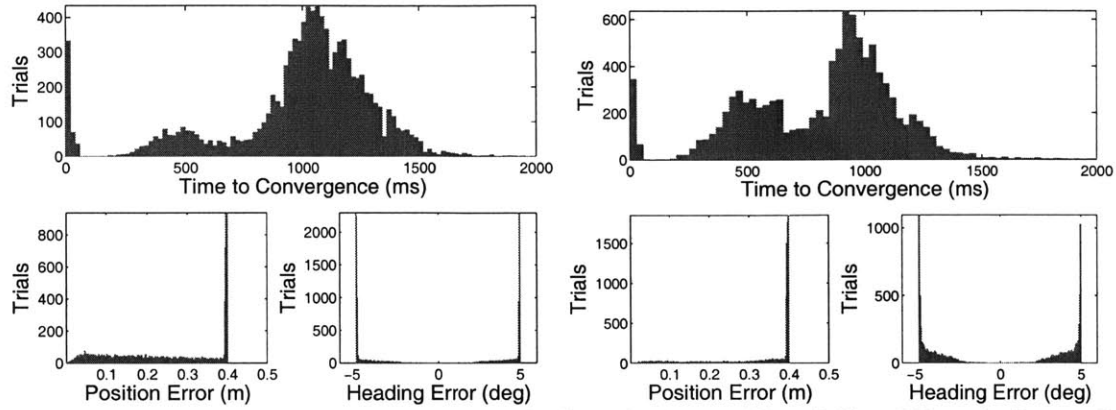


**Figure 3.15: Feedback control method results for linearized control law (left) and linearized control law with damping (right).**

A typical result for the generated paths is shown in Figure 3.16. The figure compares the feedback control method using the control law described in Equation 3.33 and the curvature matching method.

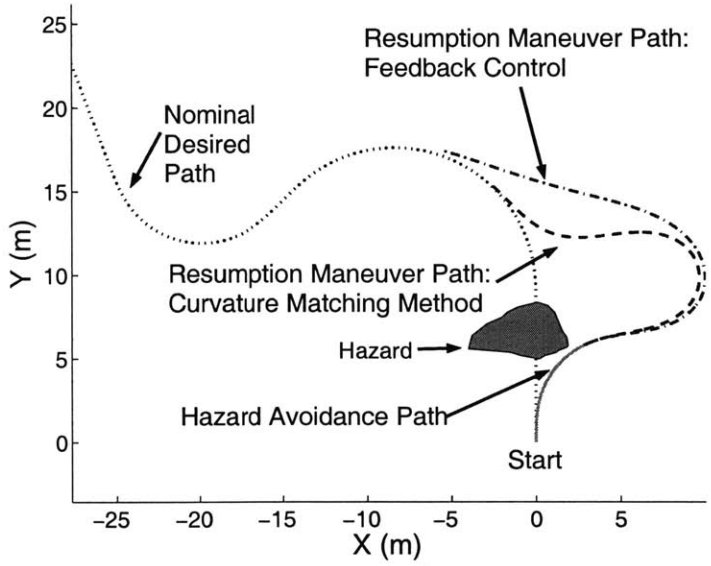


**Figure 3.16: Comparison of curvature matching and feedback control with damping technique**

## 3.3 Summary

In this chapter a hazard avoidance algorithm is presented that utilizes the trajectory space to generate a dynamically feasible maneuver to allow a UGV to quickly avoid local hazards. A hazard avoidance maneuver is enacted if a hazard lies on the nominal desired path or if the nominal desired trajectory commands a UGV to perform a dynamically unfeasible maneuver on a given terrain patch. A particular hazard avoidance maneuver is

selected by minimizing a cost function that measures distance from the current location in the trajectory space to a set of candidate locations.

After the hazard avoidance maneuver is selected, a maneuver that returns a UGV to the nominal desired path is required. Two methods for computing a path resumption maneuver are introduced and compared. The first is the curvature matching method, a novel and computationally efficient technique based on the concept of matching the integral of the curvature profile of the nominal desired trajectory with the integral of the curvature profile of the hazard avoidance and path resumption maneuvers. The second uses a simple feedback control law to generate the desired path. A comparison of the two methods demonstrates that both have excellent convergence properties, but the curvature matching method exhibits superior convergence times.

# CHAPTER 4

# SIMULATION AND EXPERIMENTAL RESULTS

This chapter describes simulation and experimental results that validate the hazard avoidance algorithm presented in Chapters 2 and 3. The chapter begins with a description of the experimental UGV system, ARTEmiS. Results are then presented that compare experimentally measured rollover velocity and curvature pairs to the constraints derived from theoretical models presented in Chapter 2. Next, experimental and simulation results are introduced that validate the hazard avoidance algorithm. Finally, simulation results illustrate the advantages of the trajectory space-based algorithm over a local path search technique.

## 4.1 Experimental System

Experimental trials were conducted on the Autonomous Rough Terrain Experimental System (ARTEmiS); see Figure 4.1. ARTEmiS is a front-steer rear-wheel drive UGV that measures 0.88 m long, 0.61 m wide, and 0.38 m high. It has a 0.56 m wheelbase and 0.25 m diameter pneumatic tires. It is equipped with a 2.5 Hp Zenoah G2D70 gasoline engine, Crossbow AHRS-400 inertial navigation system (INS), Novatel differential global positioning system (DGPS) capable of 0.2 meter resolution (circular error probable), Futaba S5050 servos for steering, brakes, and throttle, and a PIII 700 MHz PC104 computer. ARTEmiS is not equipped with forward-looking range sensors. Instead, using knowledge of ARTEmiS' position, hazard locations are only revealed once they are within the range of a "virtual sensor."

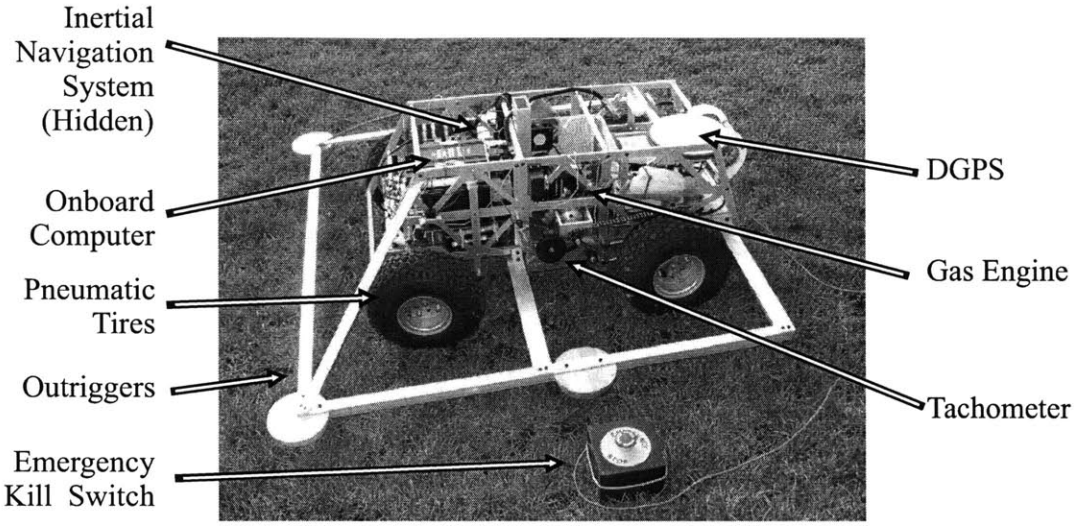


**Figure 4.1: Diagram of ARTEmiS**

Further information regarding the physical parameters and sensor capabilities of ARTEmiS as well as a description of the servo-level control laws is contained in Appendix B. Appendix B also includes all parameters used to create the simulation reference model using the MSC.ADAMS software package.

Details concerning the generation of ARTEmiS' trajectory space constraints can also be found in Appendix B. Because ARTEmiS exhibits only slight oversteer, for the purpose of the simulations and experiments presented in this chapter the steering constraints were considered to be derived from a neutral-steered vehicle. Also, note that the center of mass of ARTEmiS does not bisect the track width of the vehicle. Thus, the rollover constraints are not symmetric about zero curvature.

## 4.2 Validation of Trajectory Space Constraints

The accuracy of the model-derived trajectory space rollover constraints was studied experimentally on flat terrain at speeds up to 8 m/s (see Figure 4.2). ARTEmiS was commanded to follow a nominal desired path consisting of a straight line followed by a clothoid segment (see Figure 2.40). Rollover was determined exeperimentally as occurring when:

$$a_{lateral} \geq \frac{gh}{d} \tag{4.36}$$

where $a_{lateral}$ is the lateral acceleration of the vehicle, $g$ is gravity, $h$ is the height of the vehicle center of mass and $d$ is the distance from the outside tire to the center of mass [10,35]. At the instant of rollover, the velocity and path curvature were recorded. Due to the high traction coefficient between the synthetic turf surface and rubber tires ($\mu \approx 1.3$), rollover occurred before excessive sideslip in these experiments.

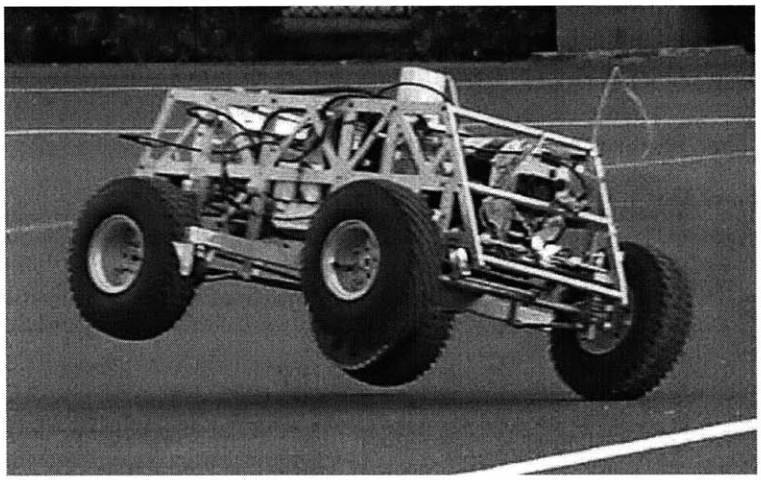

**Figure 4.2: ARTEmiS at the initiation of rollover**

Figure 4.3 compares experimental rollover data with the rollover constraint generated using the full suspension rollover model given in Equation 2.10 and illustrated in Figure 2.7.

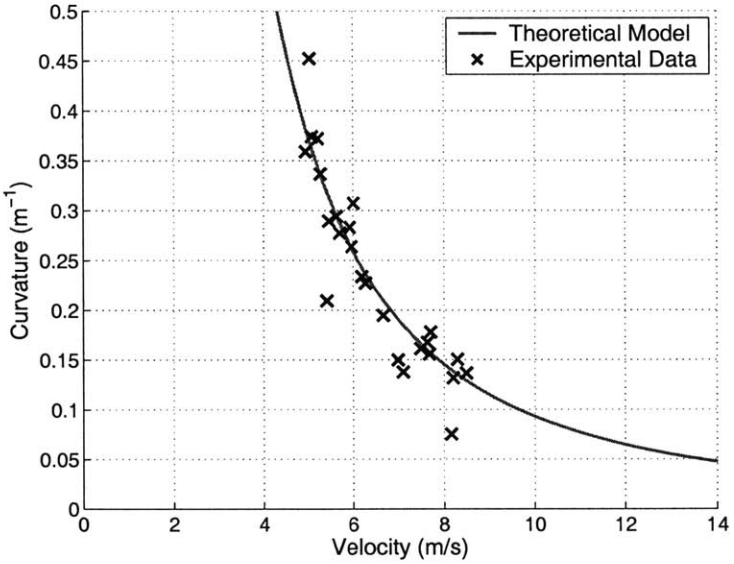


**Figure 4.3: Experimentally validated trajectory space for flat terrain**

The experimental data and predicted rollover constraint match closely. The mean of the error is 0.005 $m^{-1}$ and the standard deviation is 0.037 $m^{-1}$. The most prevalent source of error is from the calculation of the vehicle's path curvature, which can be highly sensitive to position measurements calculated from GPS and INS data.

This experiment also verifies what is considered to be "high-speed" for ARTEmiS. As discussed in Chapter 1, one definition of "high-speed" is a speed at which a vehicle can exhibit rollover. Since the steering constraints of ARTEmiS are located at $|\kappa| = 0.4\ m^{-1}$, speeds above 5 m/s are considered high.

## 4.3 Validation of Hazard Avoidance Maneuver Algorithm

The hazard avoidance maneuver algorithm was validated through both simulation and experimental analysis. This section provides the results from six experiments. Comparisons are made between two of the experimental results and simulation results. Table 4.1 lists each experiment and its purpose.

Table 4.1: Experiment title and purpose

| Experiment Title | Purpose |
|---|---|
| Multiple Hazards I | Demonstrate high speed avoidance of sequential hazards |
| Multiple Hazards II | Demonstrate high speed avoidance of sequential hazards |
| Multiple Hazards III / Sensor Range | Demonstrate avoidance of multiple hazards; show that sensor range affects maneuver selection |
| Trajectory Space Weighting Functions | Trajectory space weighting function affects maneuver selection |
| Sloped Terrain | Sloped terrain affects maneuver selection |
| Rough Terrain | Demonstrate algorithm on rough terrain |

For each experiment, ARTEmiS was placed in an initial starting location, $(x_0, y_0)$, and commanded to follow a nominal desired trajectory, $\tau_{nominal}$, with a corresponding path, $\mathbf{x}_{nominal}$. Hazards consisted of traffic cones placed in various configurations. The locations of hazards were known *a priori*, but were not revealed until they were within range of a "virtual sensor." The range of the sensor varied among experiments from 12 m to 20 m (21 to 35 times the vehicle wheelbase). Once a hazard was in range it was assumed that the hazard geometry was known. All experiments used the curvature matching method to generate a path resumption maneuver. All experiments also used the maneuver selection cost function given in Equation 3.6 with $K_1 < K_2$ unless otherwise noted.

### 4.3.1 Multiple Hazard Simulation and Experimental Results

The results from two experimental trials are presented in this section. Both results illustrate the ability of the algorithm to account for multiple sequential hazards. Thus, in these experiments only one hazard appears in the total admissible trajectory space at a particular instant. This section also contains simulation results for comparison to one of the experimental trials.

Figure 4.4 shows three "snapshot" subplots of an experiment for high speed avoidance of two hazards. The experiment was performed on a grass and dirt field at a desired velocity of 6 m/s. The nominal desired path was a 100 m long straight path. ARTEmiS detected the first hazard at $x = 16.4$ m. This is shown in the top subplot of Figure 4.4. At this point a hazard avoidance and path resumption maneuver were executed. ARTEmiS followed the modified path until a second hazard was detected at $x = 43.2$ m. This is shown in the middle subplot of Figure 4.4. A second maneuver was

then executed and ARTEmiS successfully completed the path, as shown in the lower section of Figure 4.4.

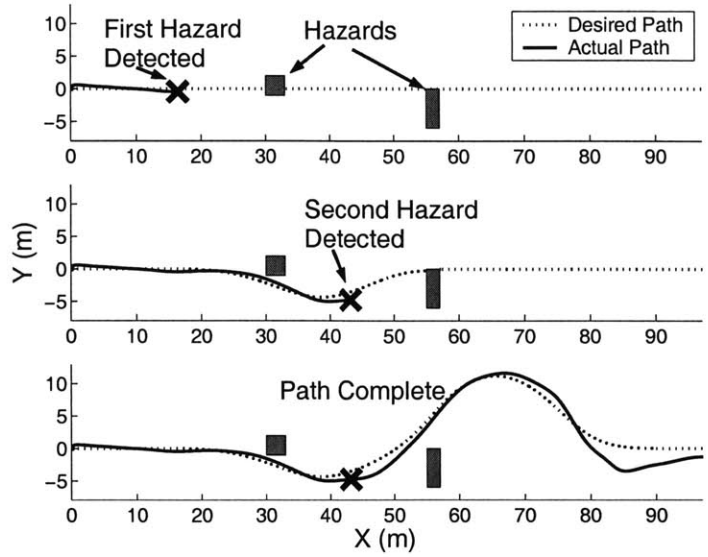


**Figure 4.4: Hazard avoidance maneuvers executed for multiple hazards**

Figure 4.5 shows the trajectory spaces at the two instants that the hazards were detected. An **x** marks ARTEmiS' location in the trajectory space. For the first hazard, ARTEmiS modified its trajectory from $\tau_0 = (6.0, 0.00)$ to $\tau_f = (6.0, -0.03)$. For the second hazard ARTEmiS modified its trajectory by maintaining its current location inside the trajectory space until it passed the hazard. In this case, a hazard avoidance maneuver was enacted because $\mathbf{x}_{\text{nominal}}$ intersected a hazard despite the fact that the current position in the trajectory space did not violate a constraint.

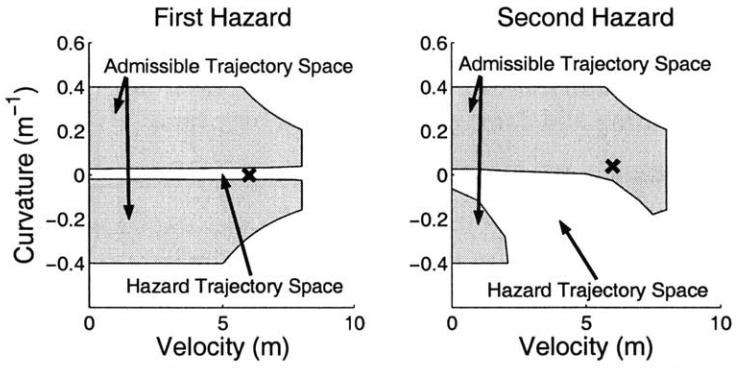


**Figure 4.5: Trajectory spaces when the hazards were detected**

Figure 4.6 compares experimental and simulation results. The top subplot displays the simulation results and the lower subplot shows the experimental results. The two results agree although the simulation generated a slightly different maneuver than the experiment for the second hazard. This is due to differences in ARTEmiS' position when the second hazard was identified. This can be attributed to position estimation and path tracking errors that are present in the experimental system.

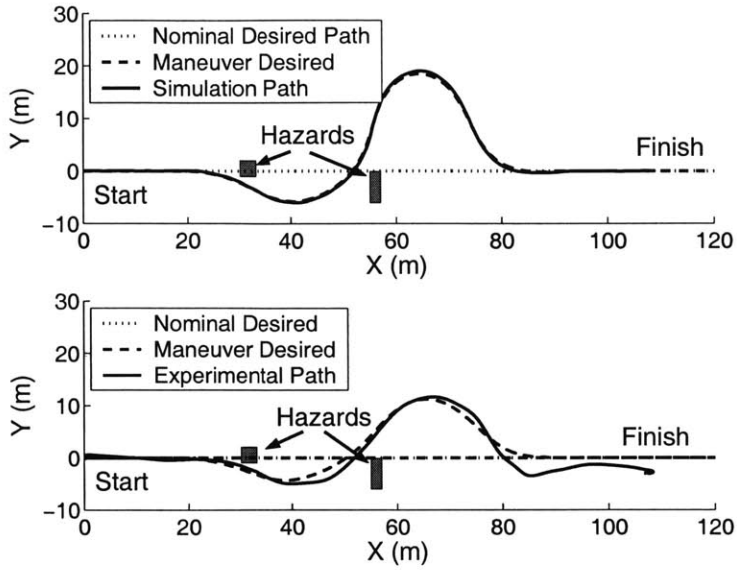


**Figure 4.6: Comparison of simulation (top) and experimental (bottom) results**

The simulation reference model tracked the desired path more accurately than did the physical system. This is due to the experimental system's position estimation errors and issues regarding ARTEmiS's steering mechanism, which are outlined as follows:

- ARTEmiS' steering servos are backdrivable. Thus, bumps in the terrain cause unwanted disturbances to the steering angle. Although the terrain used in this experiment was considered flat, there still existed a small amount of terrain roughness.
- ARTEmiS' steering servos are underpowered. At high speeds, large lateral forces develop on the tires. Thus, it was sometimes difficult for the steering servos to attain the desired steering angle.
- There is a moderate amount of mechanical slop in the front wheel hubs. This is due both to manufacturing and damage accumulated over time.

The simulation reference model does not account for these limitations. Thus, the simulation exhibits more accurate path tracking.

Another example of successful implementation of the hazard avoidance algorithm is shown in Figure 4.7. This experiment was conducted under the same terrain and initial conditions as the previous experiment. The only difference was the second hazard differs in location and size. Figure 4.7 shows three "snapshot" subplots. ARTEmiS detected the first hazard at $x = 16.4\,\text{m}$. A hazard avoidance and path resumption maneuver were enacted. ARTEmiS followed the modified path until a second hazard was detected at $x = 34.1\,\text{m}$. This is shown in the middle subplot of Figure 4.7. A second maneuver was then executed and ARTEmiS successfully completed the path, as shown in the lower subplot of Figure 4.4.

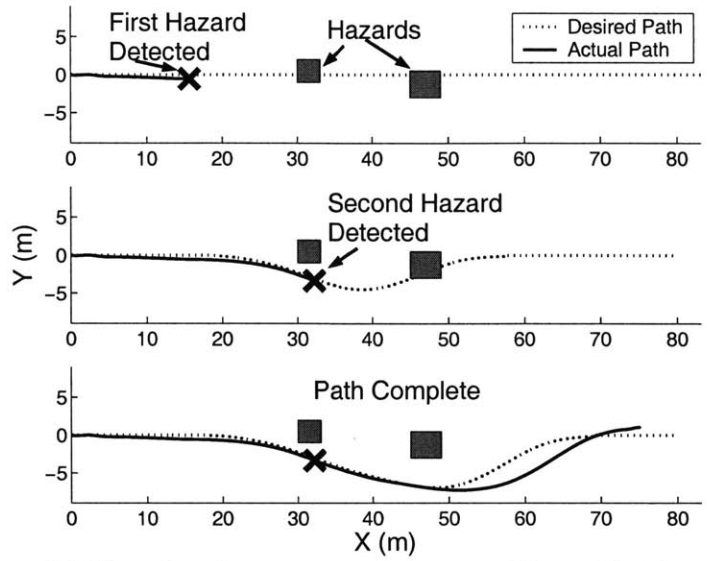


**Figure 4.7: Hazard avoidance maneuvers executed for multiple hazards**

The two trajectory spaces associated with this experiment are show in Figure 4.8. Again, An **x** marks ARTEmiS' location in the trajectory space when the hazards were identified.

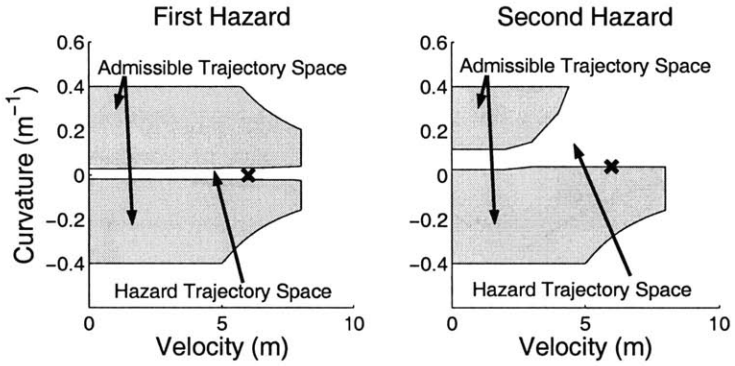


**Figure 4.8: Trajectory spaces when the hazards were detected**

This section presented results of two experiments with multiple sequential hazards. The experimental and simulation results demonstrated three points:

- The hazard avoidance algorithm has the ability to quickly and effectively generate a dynamically feasible maneuver for a vehicle traveling at "high-speed." The experiments were conducted at 6 m/s. At this speed ARTEmiS has the potential to rollover. The ability to experience rollover was given as one criteria of what is considered "high-speed" in Chapter 1.
- The hazard avoidance algorithm can account for several sequential hazards. After each hazard avoidance and path resumption maneuver is enacted, the new path and trajectory become the nominal desired path and trajectory. Thus the algorithm can always account for new hazards whenever they appear.

- Simulation results closely match the experimental results. Differences between the two were accounted for in position estimation errors and limitations of ARTEmiS' steering mechanism.

### 4.3.2 Multiple Hazard and Sensor Range Experimental Results

The range of the forward-looking range sensor can affect the selection of the hazard avoidance maneuver. Figure 4.9 compares two paths generated with the same hazards and initial UGV conditions, but with different sensing ranges. The experiments were performed on a grass-covered field at 4 m/s. The first hazard did not lie on the nominal desired path. The second hazard did lie on the nominal desired path and thus a hazard avoidance maneuver was required. The nominal desired path was a 60 m long straight path. The subplot on the left shows experimental results with a sensor range of 18 m. ARTEmiS detected the hazard at $x = 7.5$ m, and at this point hazard avoidance and path resumption maneuvers were executed. The resulting hazard avoidance maneuver had a final trajectory of $\tau_f = (4.0, -0.03)$. The subplot on the right shows results with a sensor range of 25 m. ARTEmiS detected the hazard at $x = 0.0$ m, and at this point a hazard avoidance and path resumption maneuver with a final trajectory of $\tau_f = (4.0, 0.02)$ was enacted.

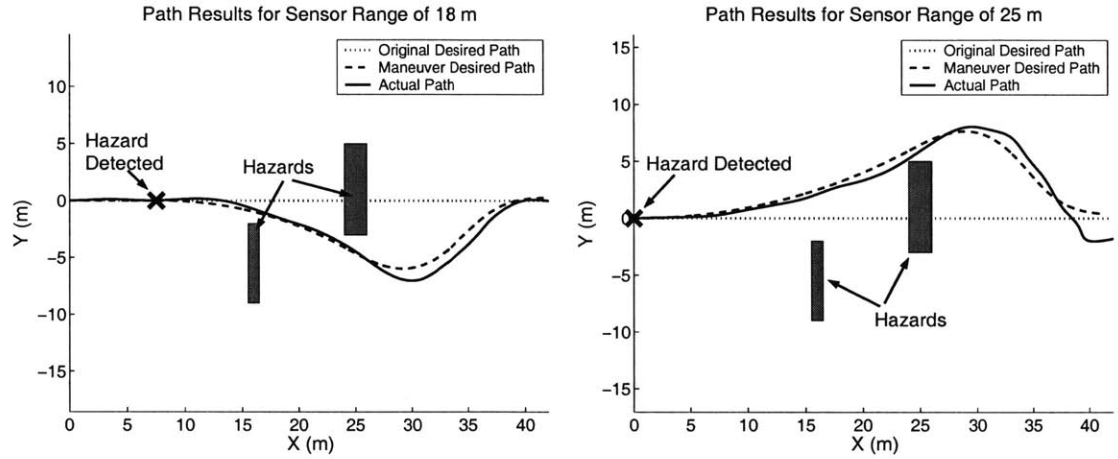


**Figure 4.9: Hazard avoidance maneuvers executed for varying sensor ranges**

The experimental results given in this section demonstrated two points:

- In general, while using the current maneuver selection cost function, longer sensor ranges result in a maneuver that deviates less from the current trajectory.
- The hazard avoidance algorithm can account for multiple hazards in the trajectory space at the same time. This is evident by examining the trajectory space for ARTEmiS at $x = 7.5$ m (see Figure 4.10).

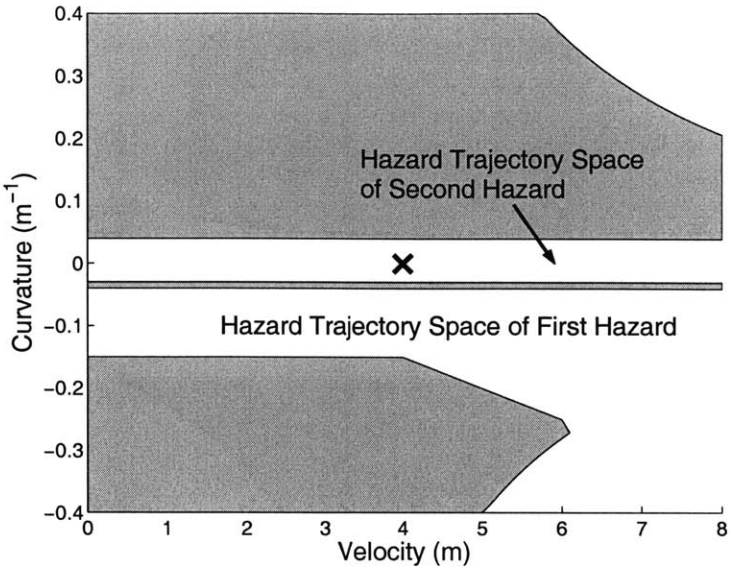


**Figure 4.10: Trajectory space for multiple hazards**

### 4.3.3 Trajectory Space Weighting Function Experimental Results

Different hazard avoidance maneuvers result from changing the gains, $K_1$ and $K_2$, in Equation 3.6, repeated here:

$$\Delta = \sqrt{\frac{K_1}{\kappa_{\max} - \kappa_{\min}}(\kappa_0 - \kappa_i) + \frac{K_2}{v_{\max}}(v_0 - v_i)} \tag{4.37}$$

The experiments in this section illustrate this fact.

In this set of three experiments, ARTEmiS was commanded to follow a nominal desired path that consisted of a 100 m long straight path. The desired speed was 8 m/s. The terrain consisted of a flat grass and dirt field. A single hazard was placed 29.7 m along the path. ARTEmiS located this hazard 18.2 meters along the path. At this point a velocity and curvature pair was chosen from the trajectory space shown in Figure 4.11.

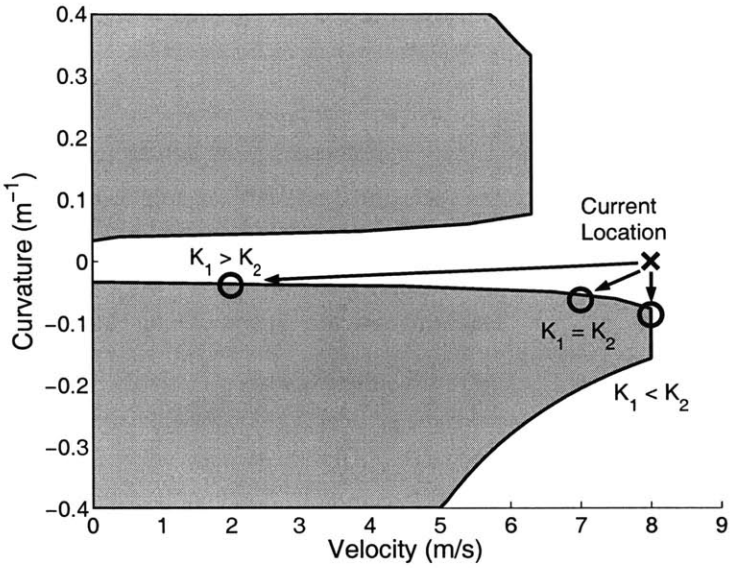


**Figure 4.11: Weighting function gains affect maneuver selection**

If $K_1 > K_2$, the algorithm will chose a hazard avoidance maneuver that will change the current velocity before changing curvature. If $K_1 < K_2$, the hazard avoidance algorithm will find a location in the admissible trajectory space that minimizes changes in velocity. If $K_1 = K_2$ the selected maneuver minimizes changes in both curvature and velocity. The resulting final trajectories for different gain values for this case are summarized in Table 4.2. Since the maneuver selection cost function normalizes $K_1$ and $K_2$, only the relative magnitude of the two affects maneuver selection.

Table 4.2: Resulting trajectories from various gain values

| Experiment Number | Gain Values | Selected Trajectory |
|---|---|---|
| **1** | $K_1 > K_2$ | $\tau_f = (2,-0.04)$ |
| **2** | $K_1 = K_2$ | $\tau_f = (7,-0.06)$ |
| **3** | $K_1 < K_2$ | $\tau_f = (8,-0.12)$ |

Figure 4.12 shows the resulting paths that ARTEmiS executed for the three experiments.

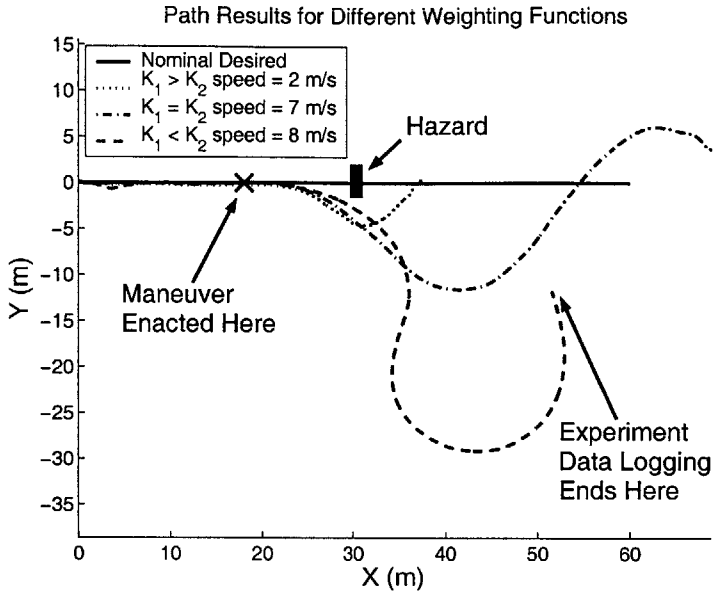


**Figure 4.12: Weighting function experimental results**

Experiment 3 stopped logging data before the experiment concluded; however, ARTEmiS did return to the nominal desired path in this case. Also, experiment 2 did not exhibit good path tracking as evident by the path overshoot. Furthermore, notice the large deviation from the nominal desired path when ARTEmiS maintained its speed of 8 m/s. This is due to the low steering rate of change of ARTEmiS. Had ARTEmiS' steering rate been faster, the path deviation would be much less.

This set of three experiments illustrates two main points:

- The hazard avoidance algorithm is capable of generating a dynamically feasible maneuver at vehicle speeds of up to 8 m/s.
- The choice of gains for the maneuver selection cost function can have a large effect on the resulting path. The choice of gains should be a mission-level decision dependent on a desire to either maximize speed or minimize deviation from the nominal desired path. Further discussion of the choice of gains and the possible use of other cost functions is given in Section 5.2.1.

### 4.3.4 Sloped Terrain Experimental and Simulation Results

An important property of the hazard avoidance algorithm is its ability to consider the effects of terrain inclination. In this experiment, the results of two trials are compared. The first was performed on flat terrain and the second was on terrain with a 15-18° slope with the fall line perpendicular to the initial direction of vehicle travel (see Figure 4.13).

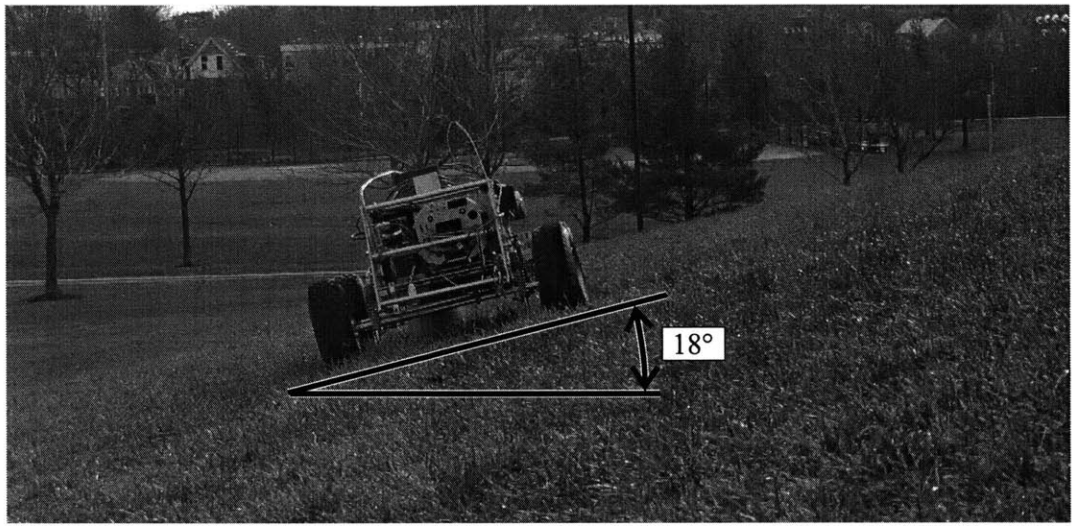


**Figure 4.13: ARTEmiS on sloped terrain**

The experiments were performed at a desired speed of 8 m/s. The initial desired path for each trial was a 100 m long curved path. This experiment was designed to illustrate how terrain inclination can affect hazard avoidance maneuver selection. For both experiments ARTEmiS traversed the path until it detected a hazard 16.4 m along the path. At the time the hazard was identified, ARTEmiS selected a hazard avoidance maneuver of $\tau_f = (8.0, 0.12)$ on flat terrain. On the sloped terrain ARTEmiS selected $\tau_f = (7.0, -0.06)$. This is due to the fact that on sloped terrain, $\tau_f = (8.0, 0.12)$ violated a trajectory space constraint, as is evident in Figure 4.14.

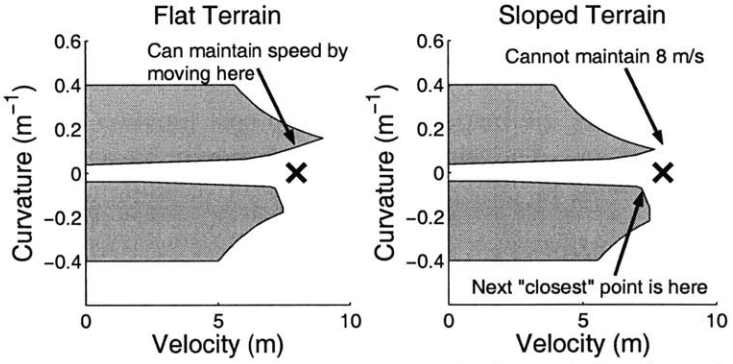


**Figure 4.14: Trajectory space comparison for flat and sloped terrain**

The resulting paths are compared in Figure 4.15. There is some path tracking overshoot in the flat terrain case. Although the sloped terrain path caused ARTEmiS to slow down, it was clearly a better maneuver for this particular nominal desired path. This occurs because the maneuver selection cost function does not consider the future path when selecting a maneuver. Possible alternative cost functions are addressed in Section 5.2.1. Again, this experiment was designed to illustrate that the trajectory space-based hazard avoidance algorithm yields a dynamically safe path (if one exists) as a function of the terrain properties.

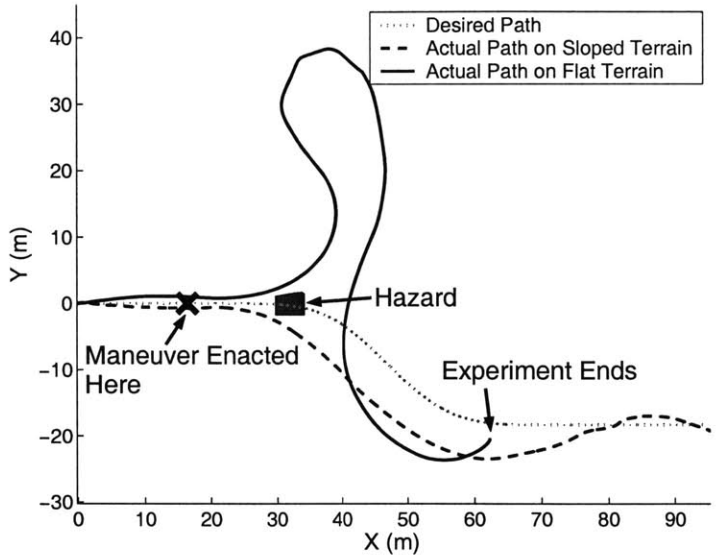


**Figure 4.15: Hazard avoidance maneuver enacted on flat and sloped terrain**

Figure 4.16 compares simulation and experimental results of ARTEmiS performing a hazard avoidance maneuver on sloped terrain. The top subplot shows the simulation results, and the bottom subplot shows the experimental results. Both subplots show the nominal desired path, the path created from the hazard avoidance maneuver, and the actual path that ARTEmiS followed.

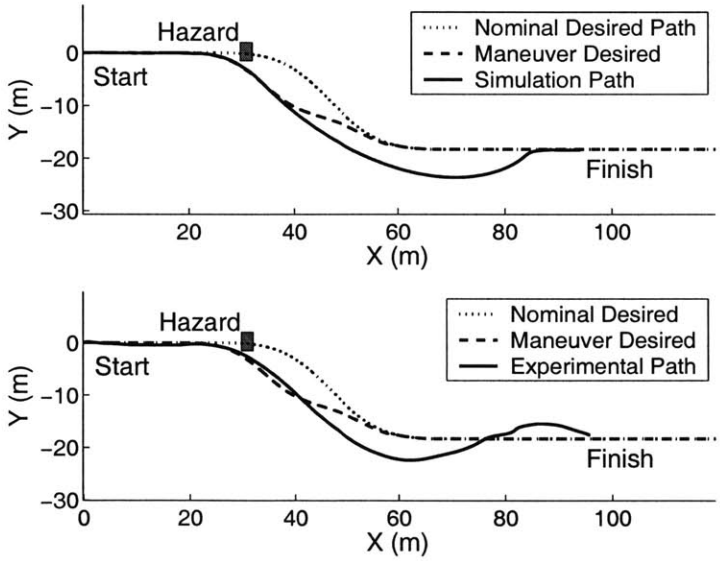


**Figure 4.16: Comparison of simulation (top) and experimental (bottom) results**

The same hazard avoidance and path resumption maneuvers were generated for the simulation and experimental trials. The simulation and experimental trials demonstrated good path tracking at the beginning of the hazard avoidance maneuver path, but some overshoot exists at the end of the maneuver.

This set of experiments and simulations illustrates two main points.

- Differences in terrain inclination change the rollover constraints of a vehicle. The trajectory space-based hazard avoidance algorithm considers this effect, and modifies the resulting maneuver accordingly.
- Simulations using the reference model accurately model the experimental system on sloped terrain.

### 4.3.5 Rough Terrain Experimental Results

Experiments on rough terrain were performed at Minute Man National Historic Park. The terrain consisted of an uncut grass field. The geometric roughness of the terrain was not measured explicitly, but terrain features tended to be on the order of one-half the wheel radius. Figure 4.17 illustrates the roughness of the terrain by comparing the vertical acceleration measured by the INS of the terrain at Minute Man National Historic Park with that taken on flat non-rough terrain. Both experiments were performed at 7 m/s.

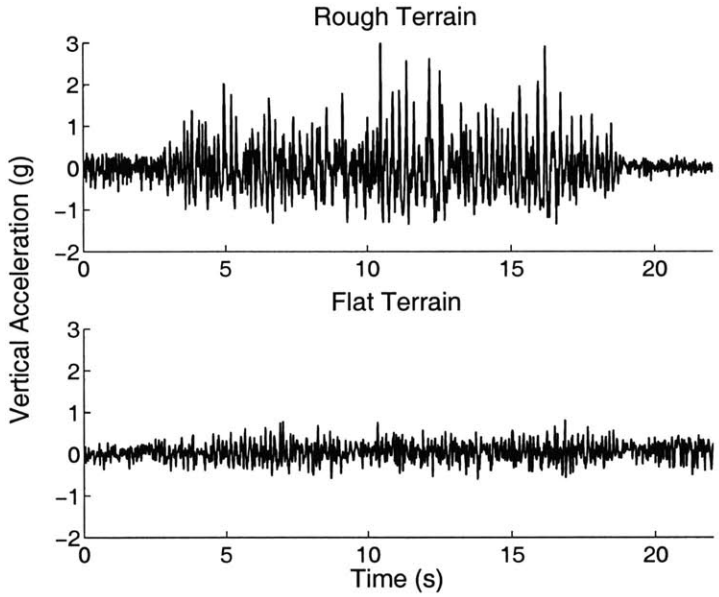


**Figure 4.17: Vertical acceleration comparison on rough and flat terrain**

The roughness of the terrain is also exemplified in Figure 4.18, which shows ARTEmiS traversing the terrain at 7 m/s. The clips are taken at $1/30^{th}$ of a second. The total time between frame 1 and frame 20 is 1.66 seconds. The figure shows ARTEmiS achieving ballistic motion twice. The first occurs between frames 1 and 9. The second occurs between frames 14 and 19. In frame 2, ARTEmiS' front wheels are no longer in contact with the terrain. In frames 4-5 none of ARTEmiS' wheels are in contact with the ground. In frame 8, the front wheels regain contact with the ground. Frames 14-19 clearly illustrate the rear wheels of ARTEmiS not in contact with the terrain.

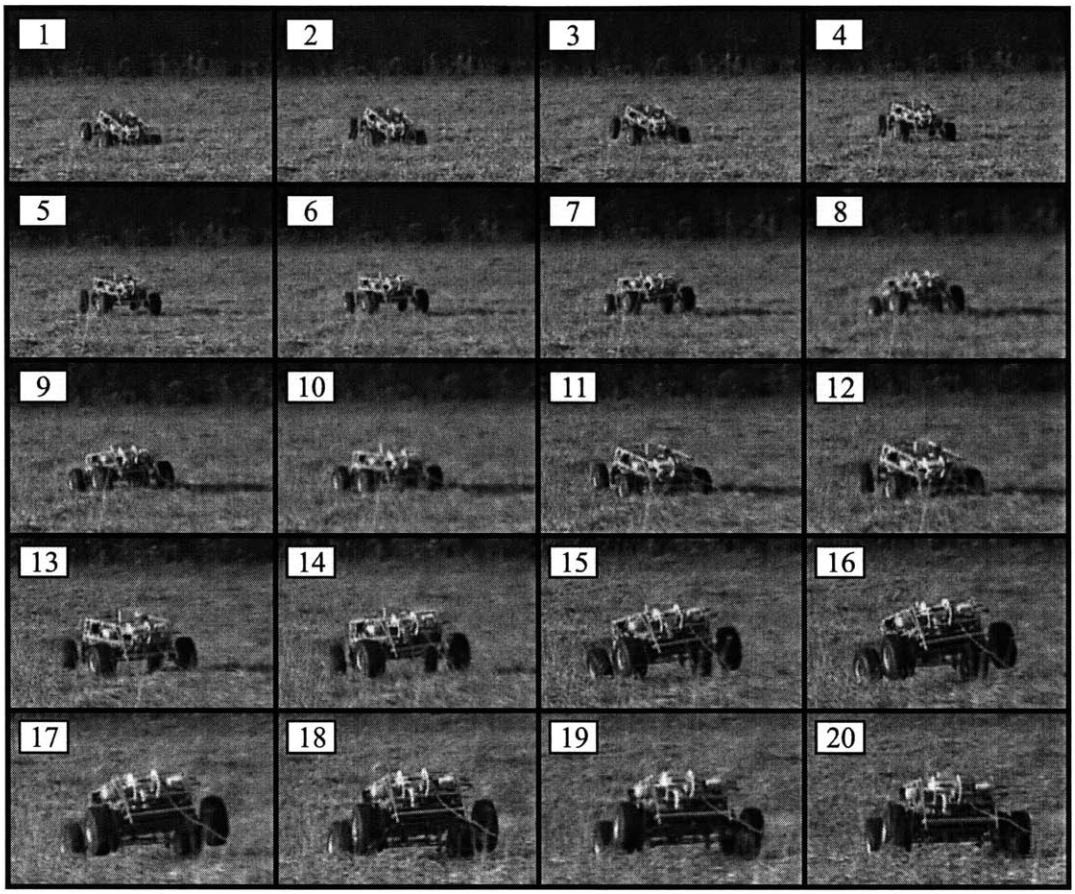


**Figure 4.18: Example of ballistic motion on rough terrain**

Figure 4.19 shows the experimental site. The nominal desired path is a 100 m long straight path. ARTEmiS is pictured at the start of the path. The goal location is obstructed from view by the hazard. The hazard consists of a cluster of tall brushes, plants, and small trees.

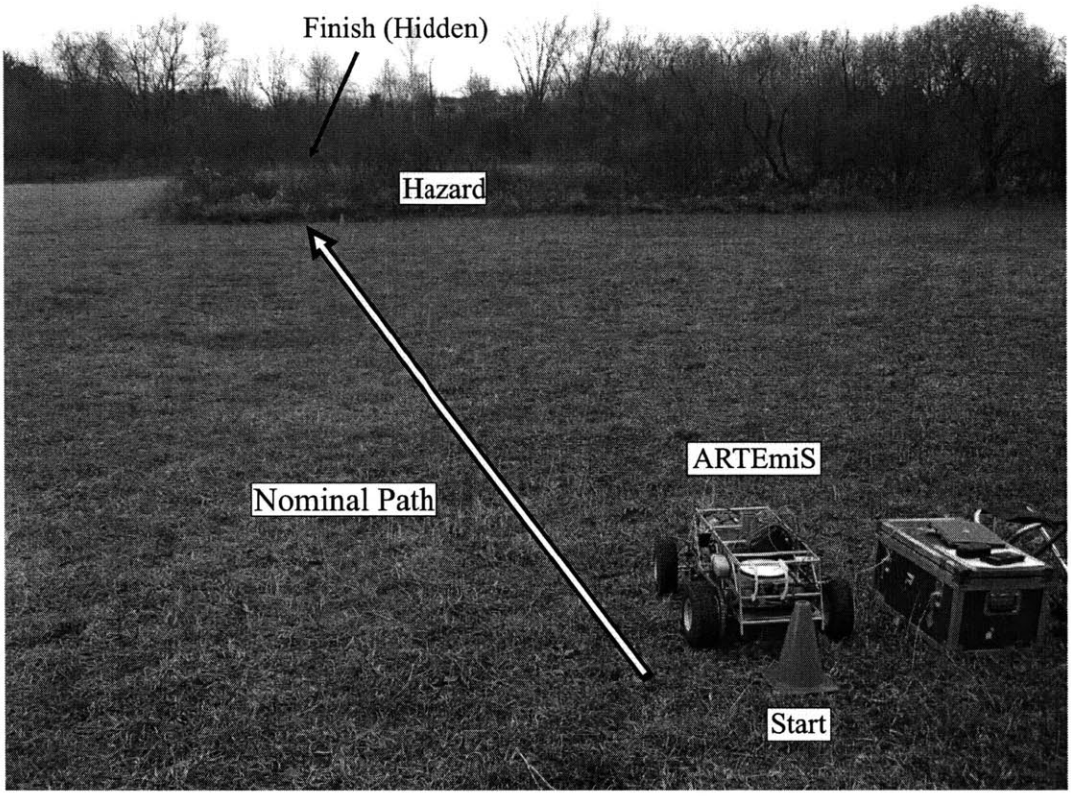


**Figure 4.19: Rough terrain experimental setup**

Figure 4.20 shows three "snapshot" subplots of the experiment. The experiment was performed at a speed of 7 m/s. ARTEmiS detected the first hazard at $x = 10.4$ m. This is shown in the top subplot of Figure 4.20. At this point hazard avoidance and path resumption maneuvers were executed, as shown in the middle subplot of Figure 4.20. The lower section of Figure 4.20 shows the completed path.

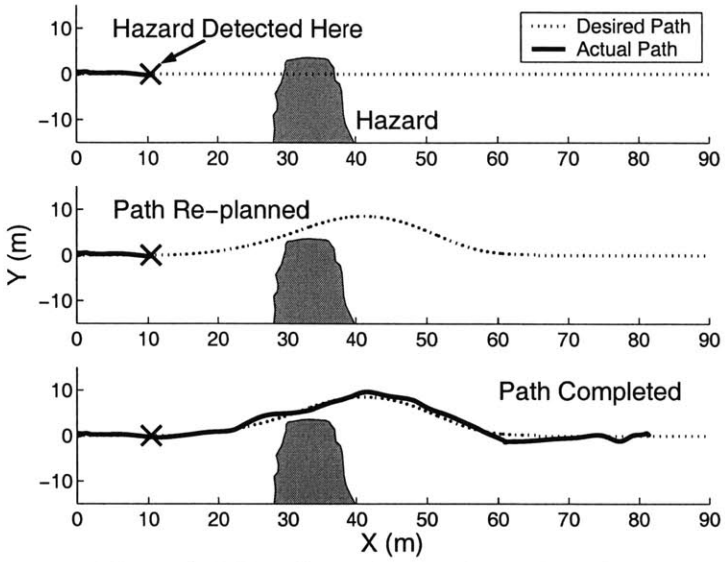


**Figure 4.20: Rough terrain experimental results**

Figure 4.21 shows the trajectory space at the time the hazard was detected. The dynamic rollover limits were manually adjusted to compensate for the effects of terrain roughness. When the hazard was detected, ARTEmiS modified its trajectory from $\tau_0 = (7.0, 0.00)$ to $\tau_f = (7.0, 0.03)$.

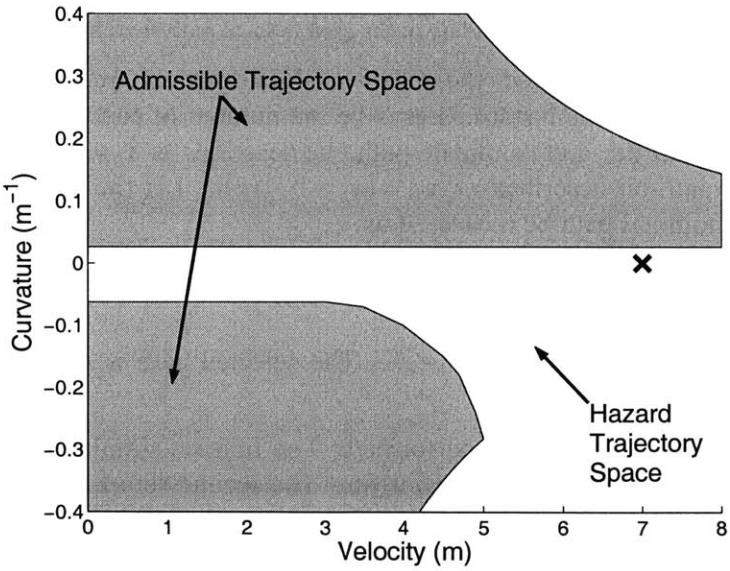


**Figure 4.21: Rough terrain trajectory space**

This experiment demonstrated one main point:

- A vehicle using the trajectory space-based hazard avoidance algorithm is capable of avoiding hazards while operating at high speeds on rough terrain. These are real world conditions that are expected to be close to actual operating conditions for applications requiring high-speed UGVs on natural terrain.

### 4.3.6 Comparison of Trajectory Space Technique with Other Methods

In this section, the hazard avoidance maneuver is compared to a local path search technique (see Section 1.2.1). Algorithm efficiency is not addressed in this comparison. The local path search technique employed here used 700 candidate paths. A sample of 60 candidate paths is shown in Figure 4.22. They are symmetric about the Y-axis. The nominal desired path used in these simulations is a straight line.

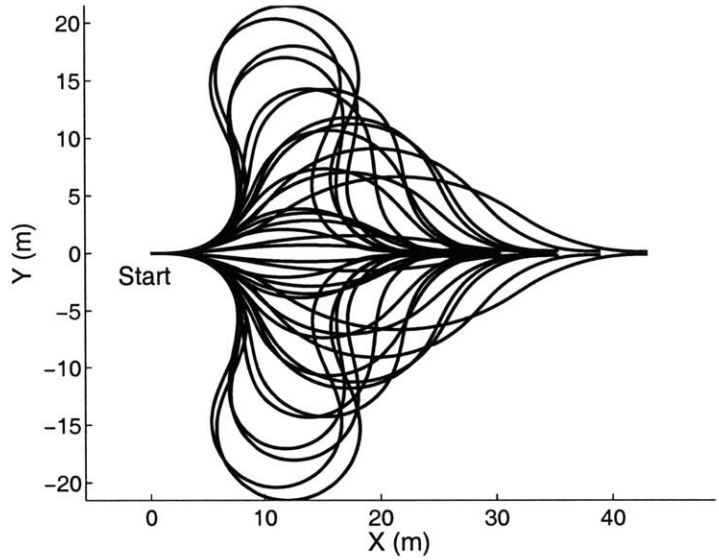


**Figure 4.22: Sample candidate paths used in local path search comparison**

When a hazard is identified, the set of candidate paths is reduced by removing all paths that intersect with the hazard. Let $N$ be the number of candidate paths that do not intersect with the hazard. Let candidate path $i$ be describes as $\hat{\mathbf{x}}_i = (\hat{x}_i(s), \hat{y}_i(s))$, $i \in [1, N]$. Let the desired path be described as $\mathbf{x}_D = (x_D(s), y_D(s))$. Let the deviation of candidate path $i$ from the nominal path be measured as:

$$a_i = \sqrt{\left\langle (\hat{y}_i - y_D)^2 \right\rangle} \tag{4.38}$$

Let $A$ be the set of all deviations, $a_i \in A$. The selected path is the candidate path that minimizes the deviation, $\min A$.

Two sets of simulations were performed. The first set simulates a single hazard on terrain with two different traction coefficients. The second set of simulations attempts to avoid a large hazard on flat terrain. Each simulation is compared to results using the hazard avoidance algorithm introduced in this thesis.

Figure 4.23 shows the results of the local path search technique on flat terrain with two different terrain coefficients. The reference model was commanded to follow a 100 m long straight path. A hazard was placed 29.7 m along the path. The initial desired speed was 8 m/s. The subplot on the left shows the results using terrain with a traction coefficient of $\mu = 1.3$. The local path search generates an acceptable path and the reference model is able to accurately track it. The subplot on the right shows the results on terrain with a traction coefficient of $\mu = 0.6$. Since the local path search algorithm

does not account for terrain conditions, the chosen path for hazard avoidance is the same. However, in this case the UGV is not able to accurately track the path due to excessive side slip.

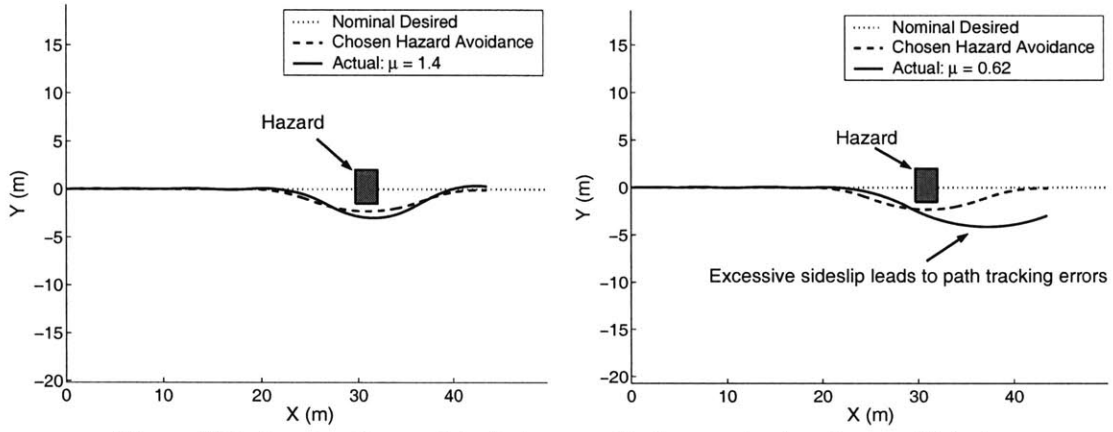


**Figure 4.23: Local path search technique results for varying traction coefficients.**

Figure 4.24 plots the results of the trajectory space-based hazard avoidance algorithm on terrain with a low traction coefficient, $\mu = 0.6$. The reference model is able to successfully avoid the hazard because the algorithm considers the sideslip constraint and lowers its speed accordingly.

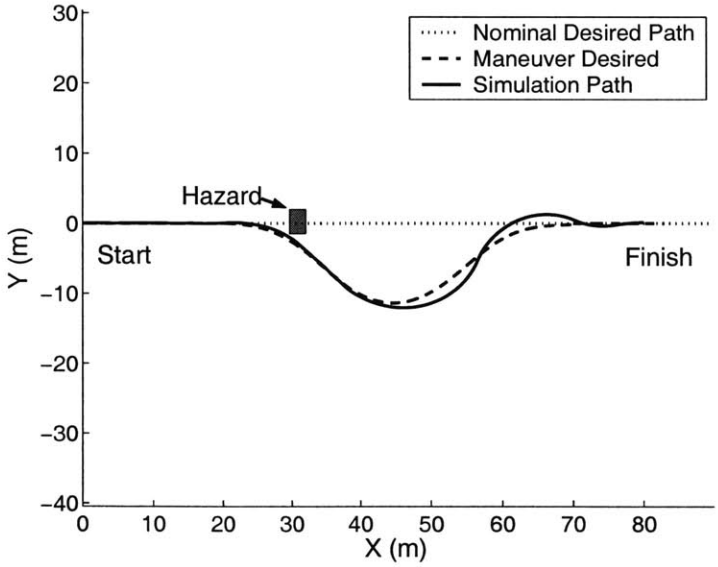


**Figure 4.24: Results using the trajectory space-based hazard avoidance algorithm for low friction**

Another example is shown in Figure 4.25. Here the traction coefficient is high enough such that a vehicle will rollover before experiencing excessive sideslip. The path generated by the local path search method does not account for the effects of rollover. Thus, while attempting to maintain its desired speed of 8 m/s, the reference model rolls over while executing a sharp turn.

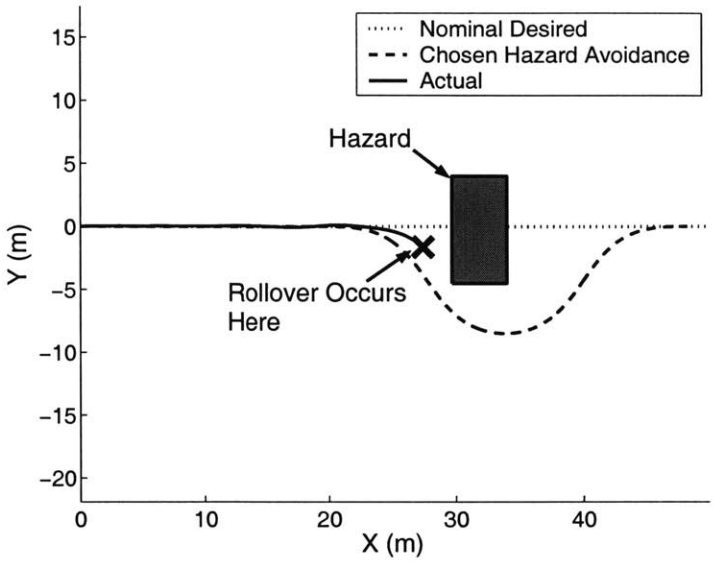


**Figure 4.25: Local path search results in vehicle rollover at 8 m/s**

In contrast, using the hazard avoidance algorithm based on the trajectory causes the vehicle to reduce its speed to 3 m/s to avoid the hazard (see Figure 4.26).

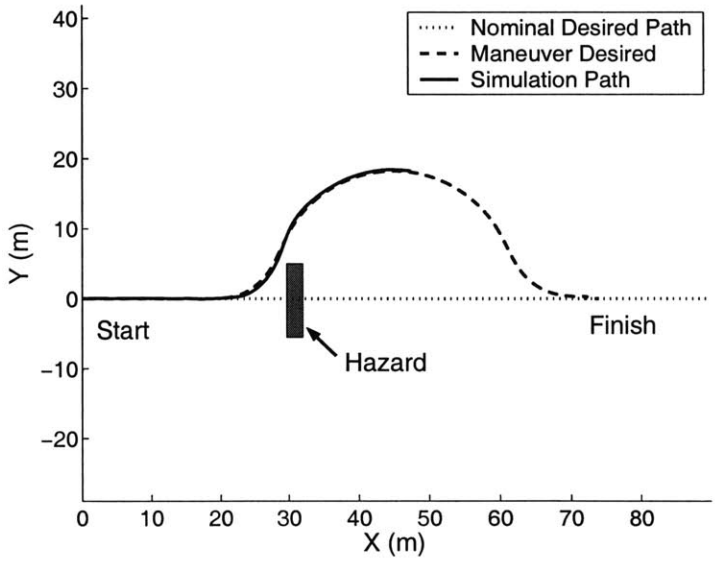


**Figure 4.26: Simulation results using the trajectory space based hazard avoidance algorithm for a large hazard**

These two examples demonstrate some of the advantages the trajectory space method over techniques that do not consider the important aspects of terrain/tire interaction and vehicle dynamics.

## 4.4 Summary

This chapter provides simulation and experimental results that validate the trajectory space rollover constraints and the hazard avoidance algorithm. First, details concerning ARTEmiS' physical parameters and sensing capabilities are introduced. Second, the

rollover constraints of ARTEmiS on flat terrain are validated experimentally and compared to the rollover constraints generated using the models presented in Chapter 2. Third, experimental results validate the hazard avoidance algorithm's ability to avoid local hazards while traveling at high speed over a variety of terrain conditions. Specifically, the experimental results demonstrate the ability of the algorithm to account for multiple hazards, sloped terrain, and rough terrain. Simulation results of two of the experimental trials are presented. In general, the simulation results closely match the experimental results; the major difference is that the simulation reference model demonstrates better path tracking than the experimental system. Finally, the hazard avoidance algorithm is compared to a local path search method. It is shown that the local path search technique yields valid results on flat terrain with a high traction coefficient; however, the technique fails to generate dynamically safe paths in situations where the traction coefficient is low or the commanded curvature too high.

# CHAPTER 5

# CONCLUSIONS

This thesis presents an algorithm for hazard avoidance for high-speed unmanned ground vehicles operating on rough, natural terrain. The algorithm accounts for dynamic effects such as vehicle sideslip, rollover, and over/understeer, as well as vehicle steering dynamics, drive train properties, terrain geometry, and vehicle/terrain interaction. The method is computationally efficient (operating on the order of milliseconds), and thus suitable for on-board implementation. Extensive simulation and experimental results demonstrate the algorithm's effectiveness.

## 5.1 Summary

In Chapter 1 the problem of high speed hazard avoidance on rough terrain is introduced, including motivation, potential applications of the work, and key assumptions. Potential applications include reconnaissance, surveillance, and material transportation. The motivation focuses on the ability of high speed, rough terrain UGVs to perform tasks more rapidly, thus limiting exposure to hazardous environments, and increasing potential applications by traversing a wide range of outdoor environments.

Chapter 1 also provides information regarding related literature. Most previous research focuses on hazard avoidance at relatively slow speeds on benign terrain. Local search techniques have proven promising; however, they often do not consider vehicle dynamics or vehicle/terrain interactions. Local reactive behavior methods have been shown to work well in indoor and some outdoor environments at low speeds, but they also do not take into account vehicle dynamics and vehicle/terrain interaction. Global path planning techniques are becoming more computationally efficient, but still are generally too slow for high speed hazard avoidance. Other techniques that do consider vehicle dynamics have been shown to be effective, but not necessarily applicable to high speed unmanned ground vehicles.

In Chapter 2 the basis of the hazard avoidance maneuver, the trajectory space, is described. The chapter focuses on methods for calculating constraints on the trajectory space. Four major subspaces of the trajectory space are presented: the dynamic trajectory space, reachable trajectory space, admissible trajectory space, and hazard trajectory space. The dynamic trajectory space represents all possible velocities and curvature pairs that do not induce rollover or excessive vehicle sideslip and are attainable considering a UGV's cornering and power train properties. The reachable trajectory space represents all velocities and curvature pairs that a UGV can attain from a given point in the trajectory space in a fixed time. The admissible trajectory space consists of the intersection of the dynamic and reachable trajectory spaces. The hazard trajectory space represents all

velocities and curvature pairs that, if maintained from a UGV's current position, would result in failure to traverse or avoid a hazard. Chapter 2 also presents work pertaining to rough terrain modeling, and investigates the effects of terrain roughness on trajectory space constraints.

Chapter 3 focuses on maneuvers, which transition a UGV from one location in the trajectory space to another. The hazard avoidance maneuver is introduced first. A hazard avoidance maneuver transitions a UGV from an upcoming unsafe location in the trajectory space to a safe location. It generates a dynamically safe path from the nominal desired path. Issues regarding 1) when to enact a hazard avoidance maneuver and 2) which maneuver to enact are discussed. Chapter 3 also introduces the path resumption maneuver, which generates a path from the end of the hazard avoidance maneuver back to the nominal path. Two methods for creating a path resumption maneuver are discussed. The first is the curvature matching method, a novel technique that generates a path resumption maneuver in a computationally efficient manner. The second utilizes a simple feedback control law to generate a path resumption maneuver. A comparison of the two methods demonstrates that the curvature matching method provides results in a much shorter time.

In Chapter 4, simulation and experimental results demonstrate the effectiveness of the hazard avoidance maneuver algorithm. Speeds of up to 8 m/s (18 mph) were attained for a small gasoline powered vehicle traversing varying types of natural terrain. First, the trajectory space rollover constraints are experimentally validated on flat terrain. Second, experiments demonstrate the ability of the algorithm to effectively consider multiple hazards. Third, experiments show the effect of adjusting the gains of the maneuver selection cost function on the resulting path. Fourth, experiments performed on sloped terrain demonstrate how terrain inclination can affect maneuvers by disallowing dynamically infeasible solutions that may have been valid on flat terrain. Fifth, experiments on rough terrain demonstrate the ability of the algorithm to perform in a challenging real-world environment. Simulation results are presented that closely match the results of experimental trials. Last, the chapter concludes by comparing the hazard avoidance method based on the trajectory space with a local path search technique. Results show that the search technique performs well on flat, smooth terrain with high traction coefficients, but can easily result in vehicle failure when the traction coefficient is low or the commanded curvature too high.

## 5.2 Suggestions for Future Work

This section presents ideas for possible extensions of this work. These include improvements to the maneuver selection process, using the trajectory space to increase vehicle performance online, adapting the trajectory space for global path planning, and employing the trajectory space analysis for driver's aids in passenger vehicles.

### 5.2.1 Maneuver Selection

The maneuver selection method introduced in Chapter 3 is limited by the fact that the maneuver selection cost function depends on two user-selectable gains. Depending on mission-level objectives, an operator can choose to maximize speed, minimize

deviation from the nominal path, or some combination of the two. Experiments have shown (see Section 4.3.3) that for a particular desired nominal trajectory, one set of gains may be considered more appropriate than another set. However, for a different nominal desired trajectory that set of gains may change. It would be interesting to study the connection between chosen gain values and the nominal desired trajectory.

Another limitation is that if the current trajectory lies in the hazard trajectory space when a hazard avoidance maneuver is enacted, the resulting chosen maneuver always lies on the boundary of that space. This has both advantages and disadvantages. The main advantage is that the resulting trajectory requires a minimum amount of change from the nominal desired trajectory. The main disadvantage is that it does not benefit from using the entire admissible trajectory space. For example, even though a curvature, $\kappa_1$, avoids a hazard, it is feasible that a sharper curvature, $|\kappa_2| > |\kappa_1|$ may yield better results. For example consider the situation illustrated in Figure 5.1 where $\kappa_0 = 0$. In this situation the nominal desired path does not bisect the hazard. The resulting path created from the hazard avoidance maneuver would be $\mathbf{x}_1$. However, in this situation it is obvious it would be advantageous to select a different hazard avoidance maneuver and follow path $\mathbf{x}_2$.

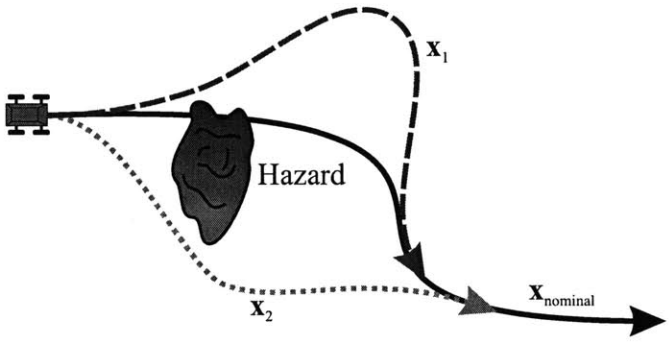


**Figure 5.1: Illustrative hazard avoidance situation**

Experimenting with different cost functions might address this limitation.

A third limitation is that hazard avoidance maneuvers are chosen from a discrete set of velocity and curvature pairs. Other methods that do not require this could be employed. One possible method currently being investigated applies a potential field to the trajectory space [73]. In this method, a UGV is given a series of waypoints instead of a nominal desired trajectory. A potential field is generated inside the trajectory space where the constraints act as sources, and the trajectory that leads to the next waypoint is a sink. The UGV then changes its trajectory accordingly.

It would also be interesting to compare the maneuver selection to the response of a human driver. An experienced human driver could provide valuable insight about the profile of an optimal hazard avoidance maneuver. This could then be used as a metric to evaluate different cost functions or maneuver selection techniques.

### 5.2.2 Increasing UGV Performance

It may be possible to use the trajectory space in conjuncture with local terrain features to yield better performance. For example, a hazard avoidance maneuver is only enacted when failure is impending. However, as new information about the terrain profile and composition is acquired online, it may be feasible to not only use the trajectory space to avoid hazards, but also to increase speed on a given path.

### 5.2.3 Global Path Planning

The hazard avoidance algorithm is designed to avoid local hazards. In its current form, the algorithm would not be appropriate for large-scale path planning due to the nature of the total admissible trajectory space. For example, when creating the total admissible trajectory space, an estimation of the UGV heading angle for each terrain patch is necessary. It is not obvious how this would be accomplished on a global scale. However, if it were possible, this method has the potential to be a computationally efficient global path planning algorithm.

### 5.2.4 Driver's Aid

The tools developed in this thesis could be applied to passenger vehicles to prevent rollover and other accidents. It is foreseeable that future on-road vehicles might employ drive-by-wire systems for throttle and steering control. If these systems become prevalent, it would be feasible to integrate some level of computer assistance with the driver commands. The trajectory space could then be used to predict impending accidents, help decide what actions a vehicle should take, or limit an overzealous driver.

# REFERENCES


1. ADAMS Version 12 Software Documentation. MSC Software, Inc., 2002.
2. Arakawa, K., and E. Krotkov. "Fractal Surface Reconstruction for Modeling Natural Terrain." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (1993): 314-320.
3. Beer, F., and E. Johnston. Vector Mechanics for Engineers: Statics and Dynamics. New York: McGraw-Hill, 1988.
4. Bekker, M. Introduction to Terrain-Vehicle Systems. Ann Arbor, Michigan: University of Michigan, 1969.
5. Brock, O., and O. Khatib. "High-Speed Navigation Using the Global Dynamic Window Approach." Proceedings of the IEEE International Conference on Robotics and Automation (1999).
6. Brooks, R. A. "A Robust Layered Control System for a Mobile Robot." IEEE Journal of Robotics and Automation 2.1 (1986): 14-23.
7. Buckingham, E. "Model Experiments and the Form of Empirical Equations." Transactions of the American Society of Mechanical Engineers 37 (1915).
8. Canudas de Wit, C., B. Siciliano, and G. Bastin, eds. Theory of Robot Control. London: Springer-Verlag, 1996.
9. Chanclou, B., and A. Luciani. "Global and Local Path Planning in Natural Environment by Physical Modeling." Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (1996): 1118-1125.
10. Chen, B.C., and H. Peng. "Rollover Warning of Articulated Vehicles Based on a Time-to-Rollover Metric." Proceedings of the ASME International Congress and Exposition (1999).
11. Cherif, M., and C. Laugier. "Dealing with Vehicle/Terrain Interaction when Planning the Motions of a Rover." Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems 1 (1994): 579-586.
12. Cherif, M., and C. Laugier. "Motion Planning of Autonomous Off-Road Vehicles under Physical Interaction Constraints." Proceedings of the IEEE International Conference on Robotics and Automation 2 (1995): 1687-1693.
13. Committee on Army Unmanned Ground Vehicle Technology, Technology Development for Army Unmanned Ground Vehicles National Academies Press, (2002).
14. Connolly, C., J. Burns, and R. Weiss. "Path Planning Using Laplace's Equation." Proceedings of the IEEE International Conference on Robotics and Automation (1990): 2102-2106.
15. Coombs, D., et al. "Driving Autonomously Off-Road up to 35 km/h." Proceedings of the IEEE Intelligent Vehicle Symposium (2000): 186-191.

16. Daily, M. et al. "Autonomous Cross-Country Navigation with the ALV." Proceedings of the IEEE International Conference on Robotics and Automation 2 (1988): 718-726.
17. "DARPA Outlines Plans for Grand Challenge at Competitors' Conference: Press Release." DARPA Official Home Page. 2003. Defense Advanced Research Projects Agency. 22 February 2003. <http://www.darpa.mil>.
18. De Luca, A., G. Oriolo, and C. Samson. "Feedback Control of a Nonholonomic Car-Like Robot." Robot Motion Planning and Control. Ed. J.-P. Laumond. London: Springer-Verlag, 1998.
19. Dixon, J. Tires, Suspension, and Handling. Warrendale, PA: Society of Automotive Engineers, 1996.
20. Dudgeon, J., and R. Gopalakrishnan. "Fractal-Based Modeling of 3D Terrain Surfaces." Proceedings of the IEEE Conference on Bringing Together Education, Science, and Technology (1996): 246-252.
21. Fiorini, P. and Z. Shiller. "Motion Planning in Dynamic Environments Using Velocity Obstacles." International Journal of Robotics Research 17.7 (1998): 760-772.
22. Fish, S. "Overview of UGCV and PerceptOR Status." Proceedings of SPIE Unmanned Ground Vehicle Technology V, 5083 (2003): 336-339.
23. Fournier, A., D. Fussell, and L. Carpenter. "Computer Rendering of Stochastic Models." Communications of the Association of Computing Machinery 25.6 (1982): 371-384.
24. Fox, D., W. Burgard, and S. Thrun. "The Dynamic Window Approach to Collision Avoidance." IEEE Robotics & Automation Magazine 4.1 (1997): 23–33.
25. Fraichard, T., and C. Laugier. "Dynamic Trajectory Planning, Path-Velocity Decomposition and Adjacent Paths." Proceedings of the Joint Conference on Artificial Intelligence 2 (1993): 1592-1597.
26. Fraichard, T., and C. Laugier. "Path-Velocity Decomposition Revisited and Applied to Dynamic Trajectory Planning." Proceedings of the IEEE International Conference on Robotics and Automation 2 (1993): 40-45.
27. Fraichard, T., and C. Laugier. "Online Reactive Planning for a Nonholonomic Mobile in a Dynamic World." Proceedings of the IEEE International Conference on Robotics and Automation (1991): 432-437.
28. Frazzoli, E. "Robust Hybrid Control for Autonomous Vehicle Motion Planning." Diss. Massachusetts Institute of Technology, 2001.
29. Frazzoli, E., M. Dahleh, and E. Feron. "Maneuver-based Motion Planning for Nonlinear Systems with Symmetries." To appear in IEEE Transactions of Robotics and Automation.
30. Freitag, D., R. Schafer, and R. Wismer. "Similitude Studies of Soil-Machine Systems." Journal of Terramechanics 7.2 (1970): 25-59.

31. Gerhart, G., R. Goetz, and R. Gorsich. "Intelligent Mobility for Robotic Vehicles in the Army after Next." Proceedings of the SPIE Conference on Unmanned Ground Vehicle Technology 3693 (1999).
32. Gifford K., and R. Murphy. "Incorporating Terrain Uncertainties in Autonomous Vehicle Path Planning." Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (1996): 1134-1140.
33. Gillespie, T. Fundamentals of Vehicle Dynamics. Warrendale, PA: Society of Automotive Engineers, 1992.
34. Golda, D. "Modeling and Analysis of High-Speed Mobile Robots Operation on Rough Terrain." Diss. Massachusetts Institute of Technology, 2003.
35. Hac, A. "Rollover Stability Index Including Effect of Suspension Design." SAE Vehicle Dynamics and Simulation 2002-01-0965 (2002).
36. Haddad, H. et al. "Reactive Navigation in Outdoor Environments using Potential Fields." Proceedings of the IEEE International Conference on Robotics and Automation (1998): 1232-1237.
37. Howard, S., and H. Seraji. "Real-Time Assessment of Terrain Traversability for Autonomous Rover Navigation." Proceedings of the IEE/RSJ International Conference on Intelligent Robots and Systems 1 (2000): 58-63.
38. Iagnemma, K. and S. Dubowsky. Mobile Robots in Rough Terrain. New York: Springer, 2004.
39. Iagnemma, K., S. Kang, C. Brooks, and S. Dubowsky. "Multi-Sensor Terrain Estimation for Planetary Rovers." Proceedings of the 7th Int. Symposium on Artificial Intelligence, Robotics, and Automation in Space, i-SAIRAS (2003).
40. Iagnemma, K., F. Genot, and S. Dubowsky. "Rapid Physics-Based Rough-Terrain Rover Planning with Sensor and Control Uncertainty." Proceedings of the IEEE International Conference on Robotics and Automation 3 (1999): 2286-2291.
41. Iagnemma, K., D. Golda, M. Spenko, and S. Dubowsky. "Experimental Study of High-Speed Rough-Terrain Mobile Robot Models for Reactive Behaviors." Proceedings of the 8th International Symposium on Experimental Robotics (2002).
42. Kavraki, L., et al. "Probabilistic Roadmaps for Path Planning in High-Dimensional Configuration Spaces." IEEE Transactions of Robotics and Automation 12.4 (1996): 566-580.
43. Kelly, A. "An Intelligent, Predictive Control Approach to the High-Speed Cross-Country Autonomous Navigation Problem." Diss. Carnegie Mellon University, 1995.
44. Kelly, A., and A. Stentz. "Rough Terrain Autonomous Mobility – Part 1: A Theoretical Analysis of Requirements." Autonomous Robots 5 (1998): 129-161.
45. Khatib, O. "Real-time Obstacle Avoidance for Manipulators and Mobile Robots." International Journal of Robotics Research 5.1 (1986): 90-98.

46. Kube, P., and A. Pentland. "On the Imaging of Fractal Surfaces." IEEE Transactions on Pattern Analysis and Machine Intelligence 10.5 (1988): 704-707.

47. Lacaze, A., et al. "Path Planning for Autonomous Vehicles Driving Over Rough Terrain." Proceedings of the IEE ISIC/CIRA/ISAS Joint Conference (1998): 50-55.

48. Langer, D., J. Rosenblatt, and M. Hebert. "A Behavior-Based System for Off-Road Navigation." IEEE Transactions of Robotic and Automation 10.6 (1994): 776-783.

49. Latombe, J. Robot Motion Planning. Boston: Kluwer, 1991.

50. Laumond, J.P., ed. Robot Motion Planning and Control. Great Britain: Springer-Verlag, 1998.

51. Lavalle, S., and J. Juffner. "Randomized Kinodynamic Planning." Proceedings of the IEEE International Conference on Robotics and Automation (1999): 473-479.

52. Leal, J., S. Scheding, and G. Dissanayake. "Stochastic Simulation in Surface Reconstruction and Application to 3D Mapping." Proceedings of the IEEE International Conference on Robotics and Automation (2002).

53. Mandelbrot, B. B. The Fractal Geometry of Nature. New York: W. H. Freeman, 1977.

54. Manduchi, R., A. Castano, A. Talukder, and L. Matthies. "Obstacle Detection and Terrain Classification for Autonomous Off-Road Navigation." Autonomous Robots 18 (2005): 81-102.

55. Mortenson, M. Geometric Modeling. New York: Wiley, 1985.

56. Nissen, N. Principles of Artificial Intelligence. Palo Alto, CA: Tioga, 1980.

57. Olin, K., and D. Tseng. "Autonomous Cross-Country Navigation: An Integrated Perception and Planning System." IEEE Expert 6.4 (1991): 16- 30.

58. Pacejka, H. "The Tire as a Vehicle Component." XXVI FSITA Congress Prague, (1996).

59. Pai, K., and L. Reissell. "Multiresolution Rough Terrain Motion Planning." IEEE Transactions on Robotics and Automation 14.1 (1998).

60. Philippsen, R., and R. Siegwart. "Smooth and Efficient Obstacle Avoidance for a Tour Guide Robot." Proceedings of the IEEE International Conference on Robotics and Automation (2003).

61. Ryu, B., and H. Yang. "Integration of Reactive Behaviors and Enhanced Topological Map for Robust Mobile Robot Navigation." IEEE Transactions of Systems, Man, and Cybernetics- Part A: Systems and Humans 29.5 (1999): 474-485.

62. Samson, C., and K. Ait-Abderrahim. "Mobile Robot Control Part 1: Feedback Control of a Nonholonomic Wheeled Cart in Cartesian Space." Research Report 1288 Program 6 Robotics, Image, and Vision (1990).

63. Samson, C., and K. Ait-Abderrahim. "Feedback Control of a Nonholonomic Wheeled Cart in Cartesian Space." Proceedings of the IEEE International Conference on Robotics and Automation (1991): 1136-1141.

64. Sanjiv, S., and A. Kelly. "Robot Planning in the Space of Feasible Actions: Two Examples." Proceedings of the International Conference on Robotics and Automation (1996): 3309-3316.

65. Scheuer, A., and T. Fraichard. "Continuous-Curvature Path Planning for Car-Like Vehicles." Proceedings of the International Conference on Intelligent Robotics and Systems (1997): 997-1003.

66. Schouwenaars, T., J. How, and E. Feron. "Decentralized Cooperative Trajectory Panning of Multiple Aircraft with Hard Safety Guarantees." American Institute of Aeronautics and Astronautics Guidance, Navigation, and Control Conference (2004).

67. Schouwenaars, T., J. How, and E. Feron. "Receding Horizon Path Planning with Implicit Safety Guarantees." American Control Conference (2004)

68. Sekhavat S., and M. Chyba. "Nonholonomic Deformation of a Potential Field for Motion Planning." Proceedings of the IEEE International Conference on Robotics and Automation (1999): 817-822.

69. Seraji, H. "Traversability Index: A New Concept for Planetary Rovers." Proceedings of the IEEE International Conference on Robotics and Automation 3 (1999): 2006-2013.

70. Shiller, Z., and J. Chen. "Optimal Motion Planning of Autonomous Vehicles in Three Dimensional Terrains." Proceedings of the IEEE International Conference on Robotics and Automation 1 (1990): 198-203.

71. Shiller, Z., and Y.-R. Gwo. "Dynamic Motion Planning of Autonomous Vehicles." IEEE Transactions on Robotics and Automation 7.2 (1991): 241-249.

72. Shiller, Z., and S. Sunda. "Emergency Lane-Change Maneuver of Autonomous Vehicles." Journal of Dynamic Systems, Measurement, and Control 120 (1998): 37-44.

73. Shimoda, S., Y. Kuroda, and K. Iagnemma. "Potential Field Navigation of High Speed Unmanned Ground Vehicles on Uneven Terrain." To appear in the Proceedings of the IEEE International Conference on Robotics and Automation (2005).

74. Shoemaker, C., and J. Bornstein. "Overview of the Demo III UGV Program." SPIE Conference on Robotic and Semi-Robotic Ground Vehicle Technology 3366 (1998).

75. Shoemaker, C., and J. Bornstein. "Overview and Update of the Demo III Experimental Unmanned Vehicle Program." Proceedings of SPIE Unmanned Ground Vehicle Technology II 4024 (2000): 212-220

76. Simmons, R. "The Curvature-Velocity Method for Local Obstacle Avoidance." Proceedings of the 1996 IEEE International Conference on Robotics and Automation (1996): 3375-3382.

77. Song, G., and N. Amato. "Randomized Motion Planning for Car-Like Robots with C-PRM." Technical Report TR01-002 Department of Computer Science Texas A&M University (2001).

78. Spenko, M. "ARTEmiS Hardware, Computer, and Electronics Manual." Field and Space Robotics Laboratory, Massachusetts Institute of Technology, 2005.

79. Stentz, A. "Optimal and Efficient Path Planning for Partially–Known Environments." Proceedings of the IEEE International Conference on Robotics and Automation 4 (1994): 3310-3317.

80. Stentz, A., and M. Hebert. "A Complete Navigation System for Goal Acquisition in Unknown Environments." Autonomous Robots 2.2 (1995).

81. Szirtes, T. Applied Dimensional Analysis and Modeling. New York: McGraw-Hill, 1997.

82. Tomizuka, M., J. Hedrick, and H. Pham. "Integrated Maneuvering Control for Automated Highway Systems Based on a Magnetic Referencing/Sensing System." Publication of Path Project, ITS, UC Berkeley, UCB-ITS-PRR-95-12.

83. Valavanis, K., et al. "Mobile Robot Navigation in 2-D Dynamic Environments Using an Electrostatic Potential Field." Proceedings of the IEEE Transactions on Systems, Man, and Cybernetics – Part A: Systems and Humans 30.2 (2000): 187-196.

84. van Maren, C., and J. Sika. "Scaled Vehicle Dynamics of DaVinci Project." TU Deflt. 2001. <http://euler.fd.cvut.cz/~sika/papers.html>.

85. Voss, R. Fractals in Nature: Characterization, Measurement, and Simulation SIGGRAPH course notes #15, 1987.

86. Walker, J. "Unmanned Ground Combat Vehicle Contractors Selected: News Release." DARPA Official Home Page. 2001. Defense Advanced Research Projects Agency. 7 February 2001. <http://www.darpa.mil>.

87. Warren, C. "Fast Path Planning Using Modified A* Method." Proceedings of the IEEE International Conference on Robotics and Automation 2 (1993): 662-667.

88. White, F. Fluid Mechanics. New York: McGraw-Hill, 1994.

89. Yamauchi, B. "PackBot: A Versatile Platform for Military Robotics." Proceedings of SPIE conference on Unmanned Ground Vehicle Technology VI 5422 (2004): 228-237.

# APPENDIX A

# TERRAIN PATCH CALCULATIONS

This appendix addresses questions regarding terrain patch size and shape as well as the calculation of a terrain patch's inclination. Requirements concerning UGV range sensor throughput and acuity are not addressed here. An analysis of these subjects in the context of UGV navigation is provided in [43]. It is assumed that terrain elevation measurements are taken from on-board sensors as well as other potential sources such as topographical map data, previous sensor scans, or sensor data from unmanned aerial vehicles (UAVs) or other UGVs.

The size of the forward range sensor's field of view should be large and broad enough to cover a reasonable set of potential hazard avoidance paths. The size of the sensor range is proportional to the reachable trajectory space. Thus as the sensor range becomes smaller, so does the reachable trajectory space.

The number of terrain patches is related to the desired accuracy of the terrain inclination estimation used to compute the total admissible trajectory space. As the number of terrain patches increases, the estimated terrain roll and pitch for each patch becomes more accurate. This is partly due to the fact that terrain inclination is dependent on the UGV heading angle over a particular terrain patch. Consider the terrain patches described in Figure A.1. The right side of Figure A.1 displays the terrain patches overlaid with potential paths. Each potential path crosses each terrain patch at a different heading angle, and that heading angle is not constant across the terrain patch. However, examining a single terrain patch reveals that the potential heading angles over that patch are relatively similar.

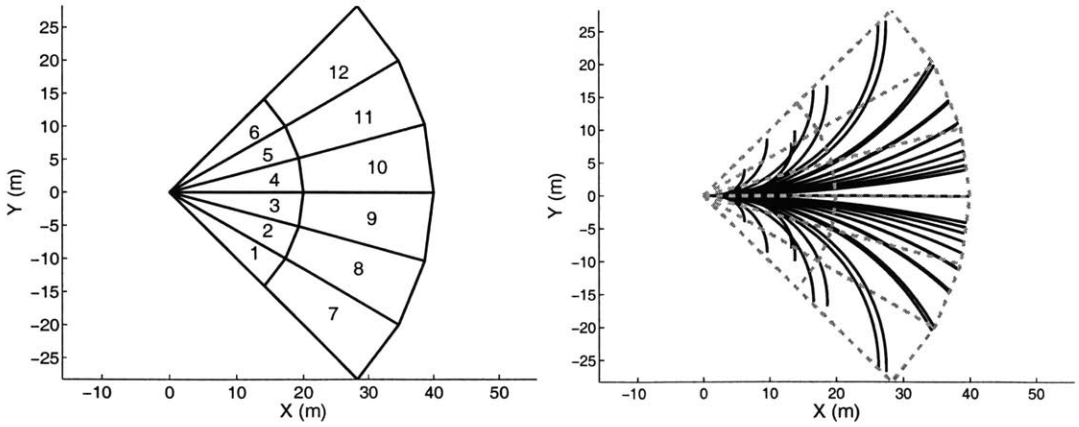


**Figure A.1: Numbered terrain patches (left) and with potential paths for $\kappa_0 = 0$ (right)**

Consider terrain patch number 8 shown with a subset of potential paths (see Figure A.2).

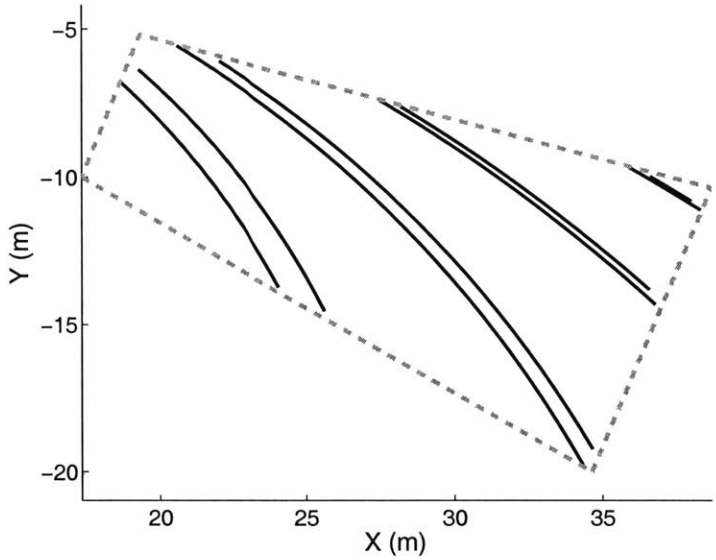


**Figure A.2: Potential paths crossing terrain patch #8**

Since the heading angles of the potential paths that cross a terrain patch are similar, the angle at which a UGV is likely to cross that patch can be estimated *a priori* by computing the mean heading angle of the potential paths. This is shown for terrain patch number 8 in Figure A.3, where the arrow represents the mean heading angle of the potential paths.

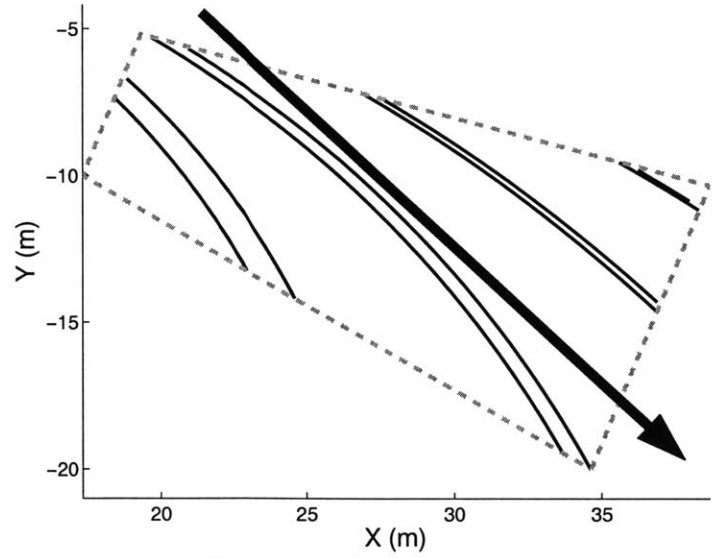


**Figure A.3: Terrain patch with mean heading angle**

The resulting heading angles for all terrain patches with $\kappa_0 = 0$ are shown in Figure A.4. In practice, this analysis is performed for a number of candidate initial curvatures and the results are stored in an online lookup table.

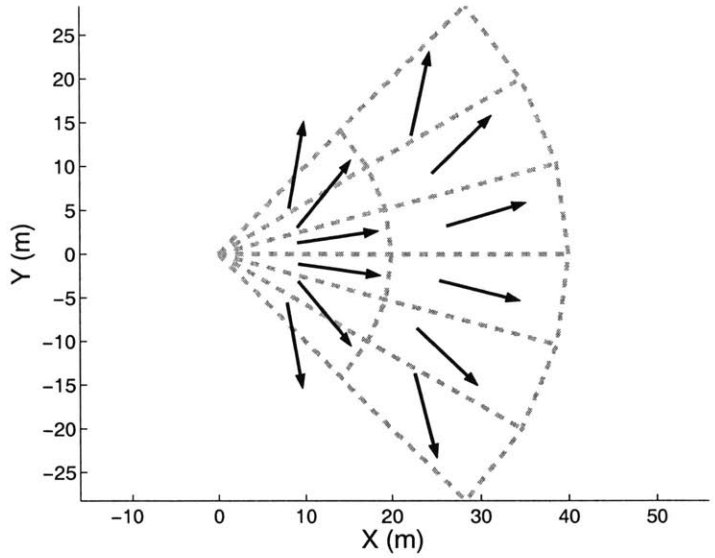


**Figure A.4: Comparison of average heading angles for terrain patches**

Given the average heading angle that a UGV will cross a terrain patch, the average roll and pitch is calculated by finding the plane that minimizes the least square error of the elevation points inside the patch. This is done as follows:

The equation of a plane is given as:

$$a_1(x)+a_2(y)+a_3(z)=b \tag{A.1}$$

where $\langle a_1, a_2, a_3 \rangle$ is the normal vector to the plane. The roll angle of the plane is given as:

$$\phi = \tan^{-1}\left(-\frac{a_2}{a_3}\right) \tag{A.2}$$

The pitch angle of the plane is given as:

$$\psi = \tan^{-1}\left(-\frac{a_1}{a_3}\right) \tag{A.3}$$

To find the plane, let the set of $n$ measured elevation points be described by $\mathbf{X}=[\mathbf{x},\mathbf{y},\mathbf{z}]$ where $\mathbf{x}=[x_1,x_2,...,x_n]^T$, $\mathbf{y}=[y_1,y_2,...,y_n]^T$, and $\mathbf{z}=[z_1,z_2,...,z_n]^T$ given in inertial coordinates. For terrain patch $i$, a coordinate transformation from inertial coordinates to the centroid of the terrain patch is performed:

$$\begin{bmatrix} \tilde{x}_j \\ \tilde{y}_j \\ 1 \end{bmatrix} = \begin{bmatrix} \cos(\bar{\theta}_i) & \sin(\bar{\theta}_i) & -(x_c\cos(\bar{\theta}_i)+y_c\sin(\bar{\theta}_i)) \\ -\sin(\bar{\theta}_i) & \cos(\bar{\theta}_i) & x_c\sin(\bar{\theta}_i)-y_c\cos(\bar{\theta}_i) \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_j \\ y_j \\ 1 \end{bmatrix} \tag{A.4}$$

where $(x_c, y_c)$ is the location of the centroid of the terrain patch, and $\bar{\theta}_i$ is the mean heading angle associated with terrain patch $i$. Let $\mathbf{A}$ be the $n$ x 3 matrix:

$$\mathbf{A} = \begin{bmatrix} 1 & \tilde{x}_1 & \tilde{y}_1 \\ 1 & \tilde{x}_2 & \tilde{y}_2 \\ \vdots & \vdots & \vdots \\ 1 & \tilde{x}_n & \tilde{y}_n \end{bmatrix} \tag{A.5}$$

The coefficients of the normal vector to the plane that minimizes the error in a least squares sense are given as:

$$\begin{bmatrix} a_1 & a_2 & a_3 \end{bmatrix}^T = \left(\mathbf{A}^T\mathbf{A}\right)^{-1}\mathbf{A}^T\mathbf{z} \tag{A.6}$$

Each terrain patch now has an associated roll and pitch angle that can be used to compute the corresponding admissible trajectory space.

# APPENDIX B

# ARTEMIS' PARAMETER IDENTIFICATION

This appendix presents measured values of parameter identification experiments performed to compute the trajectory space constraints and generate the reference model. The system parameters include mass and inertia of the sprung mass, mass and inertia of the wheels, center of mass location, suspension damping and stiffness, tire stiffness, and engine torque.

## B.1 Measurement of ARTEmiS' Inertial Parameters

### B.1.1 Mass Measurements

The inertia and mass of ARTEmiS' sprung mass and wheels were determined experimentally using a scale and are presented in Table B.1.

Table B.1: ARTEmiS mass properties

| Part | Mass (kg) |
|---|---|
| Sprung mass | 28.0 |
| Wheels (each) | 1.85 |
| Total | 35.4 |

### B.1.2 Center of Mass Location Measurements

Methods for calculating the location of the center of mass of an arbitrary geometry can be found in [34]. The location of ARTEmiS' center of mass is shown in Figure B.1.

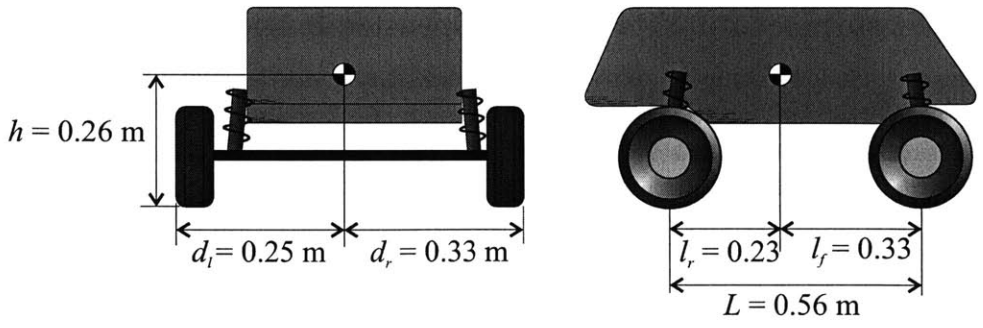


**Figure B.1: ARTEmiS location of center of mass**

### B.1.3 Wheel and Sprung Mass Inertia Measurements

The moments of inertia of the wheels were experimentally measured by hanging the wheel from a metal cable and recording the period of angular oscillation about the desired axis [34]. The moment of inertia about the λ axis is given as:

$$T = 2\pi\sqrt{\frac{I_\lambda}{K_{cable}}} \tag{B.1}$$

where $K_{cable}$ is the linear rotational inertia of the cable. $K_{cable}$ was computed using a mass of known inertia. It is assumed that damping is not significant and thus can be ignored. Experiments for a single wheel were done for the two principle axes. The inertia tensor for the wheel was determined to be:

$$I_w = \begin{bmatrix} 8.24 & 0 & 0 \\ 0 & 8.24 & 0 \\ 0 & 0 & 1.39\times10^4 \end{bmatrix} \text{kg}\cdot\text{mm}^2 \tag{B.2}$$

Due to difficulties in attaching the sprung mass to metal cable, the moments of inertia for the sprung mass were estimated using a CAD solid model of the system. The inertia tensor for the sprung mass was determined to be:

$$I_s = \begin{bmatrix} 5.33\times10^5 & 1.80\times10^4 & 1.20\times10^4 \\ 1.80\times10^4 & 1.03\times10^6 & 2.36\times10^4 \\ 1.20\times10^4 & 2.36\times10^4 & 1.36\times10^6 \end{bmatrix} \text{kg}\cdot\text{mm}^2 \tag{B.3}$$

One experiment was performed to verify the validity of the solid model. The yy-component of the inertial tensor, $I_s$, was experimentally measured and compared to the values provided by the solid model. The experimentally measured value is $I_{yy} = 1.10\times10^6$ kg·m$^2$ which yields a 6.7% error with the value generated from the solid model. Thus, the solid model is considered to provide an accurate representation of ARTEmiS' inertia.

## B.2 Measurement of ARTEmiS' Suspension Properties

The suspension members of ARTEmiS consist of a spring and damper. The total length of deflection of the suspension member is 25.4 mm. Thirteen load cell measurements were taken at seven different displacements of the spring. The spring constant of the suspension member was measured experimentally to be 3.8 N/mm (see Figure B.2).

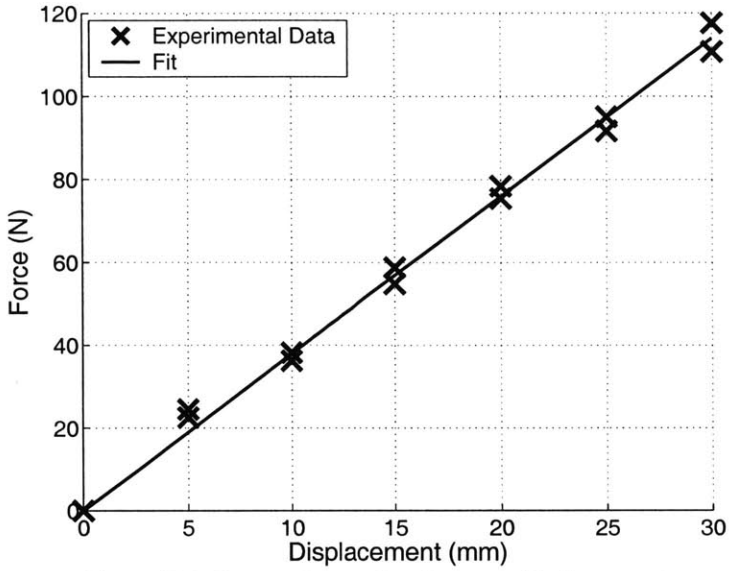


**Figure B.2: Suspension spring force vs. displacement**

The damper of the suspension mechanism consists of an oil filled chamber. However during field trials, it was noted that oil would easily escape from the chamber. Thus to maintain uniform damping coefficient across all experiments, the damping chambers are kept empty. This does not appear to detrimental to the experimental trials. Measurement of the damping coefficient using a linear variable displacement transducer (LVDT) was attempted, but the lack of oil in the chamber prevented an accurate measurement.

The dynamic response of the suspension member to a step input is shown in Figure B.3. In this experiment, a LVDT was attached to the suspension member, ARTEmiS was displaced until the suspension member reached full depression, and then ARTEmiS was released. As expected the dynamic response resembles an underdamped second order system.

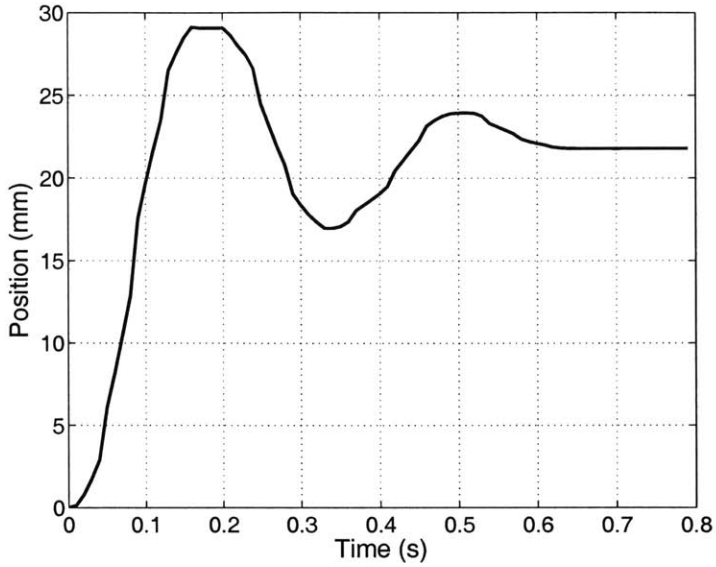


**Figure B.3: Frequency response of one of ARTEmiS' suspension members**

Estimates of the suspension member's dynamic response are given in Table B.2.

Table B.2: Dynamic response of suspension member

| Parameter | Value |
|---|---|
| Rise time (s) | 0.07 |
| Peak time (s) | 0.18 |
| Settling time (s) | 0.64 |
| Damping ratio | 0.53 |
| Damped natural frequency (Hz) | 3.10 |

## B.3 Measurement of ARTEmiS Tire Stiffness

ARTEmiS employs Carlisle 4.10-4NHS pneumatic tires. The tire is modeled as a series of radial springs emanating from the hub. The vertical (normal to the ground plane) stiffness was measured experimentally to be 24 N/mm (see Figure B.4).

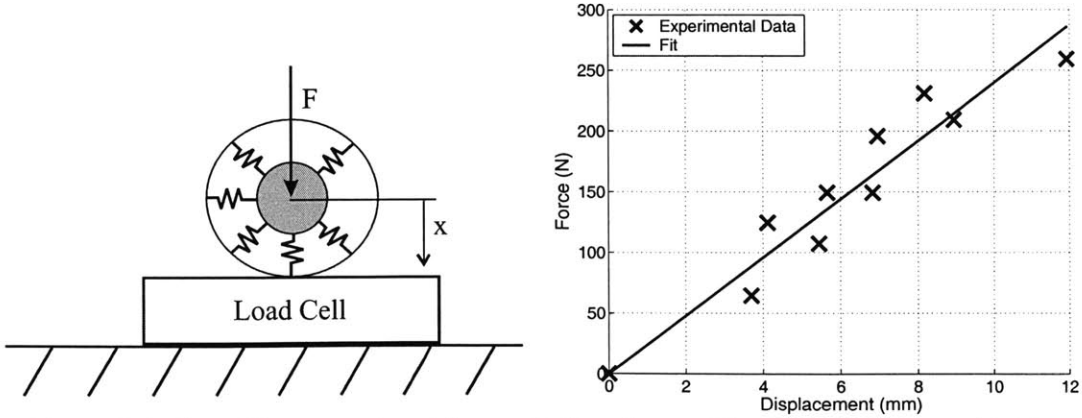


**Figure B.4: Tire stiffness experimental setup (left) and normal force vs. displacement (right)**

## B.4 Estimation of ARTEmiS Torque Profile

The 2.5 Hp gasoline Zenoah G2D70 engine's torque profile was visually extrapolated from data provided by Zenoah (www.zenoah.net at the time of publication). For use in the simulation reference model, the engine's torque profile was estimated as (see Figure B.5):

$$T(\omega) = -0.0014\omega^2 + 2.78\omega + 190 \tag{B.4}$$

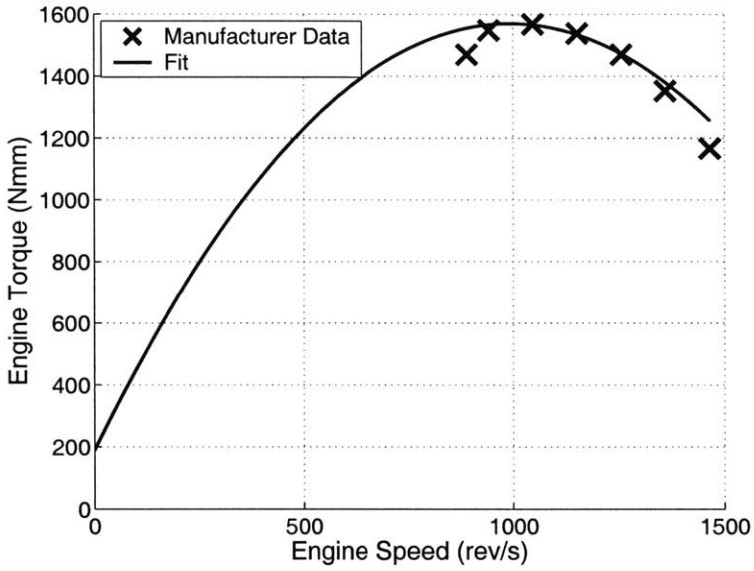


**Figure B.5: Engine torque profile**

## B.5 Sensing, Computation, and Actuation

The specifications of ARTEmiS' computer are given in Table B.3:

Table B.3: Computer specifications.

| | |
|---|---|
| Motherboard | Electronic Equipment Produktion and Distribution (E.E.P.D) Profive CPU-C3VE PC104+ |
| Processor | Intel PIII 700 MHz |
| RAM | 256 MB 133 MHz |
| Hard Drive | 500 MB Compact Flash Drive |
| DIO expansion board | Diamond Systems Garnet-MM |
| A2D/D2A expansion board | Diamond Systems Diamond-MM-32-AT |

The DIO expansion board interfaces with an expansion card fabricated specifically to provide control commands to the servos [78]. The A2D/D2A expansion board interfaces with the Crossbow AHRS-400CC-200 inertial navigation system. Key specifications of the INS are given in Table B.4.

Table B.4: Inertial navigation parameters

| | |
|---|---|
| Update rate (Hz) | 60 |
| Attitude dynamic accuracy (deg rms) | ± 2.5 |
| Heading dynamic accuracy (deg rms) | ± 4 |
| Angular rate range (deg/s) | ± 200 |
| Angular rate bandwidth (Hz) | >25 |
| Acceleration bandwidth (Hz) | >10 |

The differential global positioning system utilizes a Novatel ProPak-G2 receiver that is capable of < 0.2 m position accuracy (Circular Error Probable) and a bandwidth of 20 Hz. ARTEmiS' velocity is measured with a MicroMo 1616 DC tachometer. Steering and braking are accomplished with Futaba S5050 servomotors. The servo properties are given in Table B.5 (www.futaba.com at time of publication). The nominal voltage used during operation is 6.0 V.

Table B.5: Servo properties

| Voltage (V) | Torque (Nm) | Speed (deg/s) |
|---|---|---|
| 4.8 | 1.5 | 240 |
| 6.0 | 1.9 | 300 |

## B.6 Tire/Terrain Traction Coefficient Measurements

The traction coefficient was experimentally measured by pulling a wheel with a spring-scale along a given terrain (see Figure B.7).

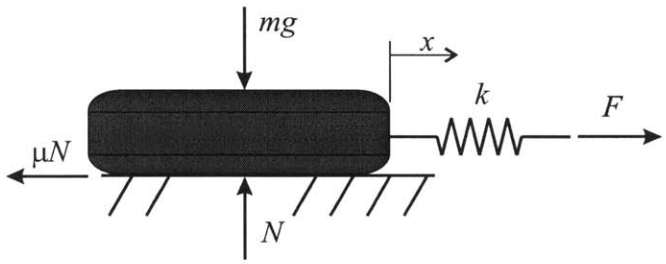


**Figure B.6: Tire/terrain traction coefficient experiment**

The traction coefficient is given as:

$$\mu = \frac{kx}{mg} \tag{B.5}$$

where $k$ is the stiffness of the spring, which was measured experimentally. The tire/terrain traction coefficient was measured experimentally for grass and artificial turf in dry conditions.

| Terrain | Traction Coefficient |
|---|---|
| Artificial turf | 1.3 |
| Natural grass | 1.5 |

## B.7 ARTEmiS Tire Cornering Stiffness Estimation

Tire cornering forces for pneumatic tires can exhibit a wide range of values, but typically range from 650 to 950 N/rad [19]. The steering constraints for ARTEmiS with $C_k = 650$ and $C_k = 950$ N/rad are shown in Figure B.7.

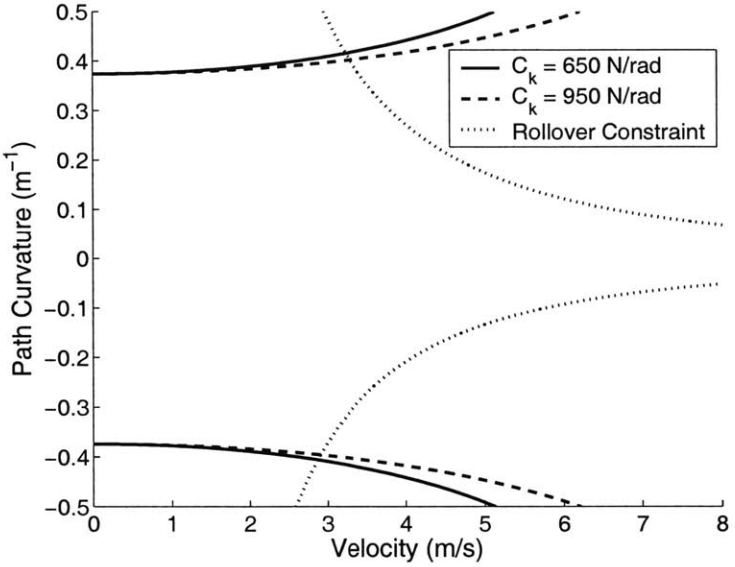


**Figure B.7: Comparison of steering constraints for varying cornering stiffness values**

Since the steering constraints do not vary significantly at speeds under 3 m/s, for the purposes of the experiments and simulations conducted here, it is assumed that ARTEmiS is a neutrally steered vehicle with a maximum/minimum curvature of $\pm 0.4\ \mathrm{m}^{-1}$.

## B.8 ARTEmiS Acceleration/Deceleration and Steering Rates

ARTEmiS' acceleration and deceleration constants were measured experimentally by recording the velocity while at full throttle and at full brake. The maximum rate of curvature change was estimated based on ARTEmiS ability to track a clothoid path at 8 m/s. The values are given in Table B.6.

Table B.6: Acceleration, deceleration, and curvature rate of change

| Parameter | Value |
|---|---|
| Acceleration, $a_p$ (m/s$^2$) | 1.2 |
| Deceleration, $a_b$ (m/s$^2$) | -1.6 |
| Curvature rate of change, $\dot{\kappa}_{max}$ (m$^{-1}$/s) | 0.07 |

# APPENDIX C

# SERVO-LEVEL CONTROL

This appendix describes the models and feedback control laws used for ARTEmiS path tracking and velocity control.

## C.1 Path Tracking

The material described in this section is taken from [8,50]. The problem of path tracking is equivalent to finding a feedback control law for the steering input [8]:

$$\delta = k(s, e_l, e_\theta) \tag{C.1}$$

such that:

$$\lim_{t\to\infty} e_l(t) = 0 \qquad \lim_{t\to\infty} e_\theta(t) = 0 \tag{C.2}$$

where $s$ is the distance along the nominal desired path, $e_l$ is the distance from the vehicle to point on the path that minimizes the Euclidian distance from the path to the vehicle, and $e_\theta$ is difference between the vehicle heading angle, $\theta$, and the angle of the tangent to the path at that same point, $\theta_t$ (see Figure C.1).

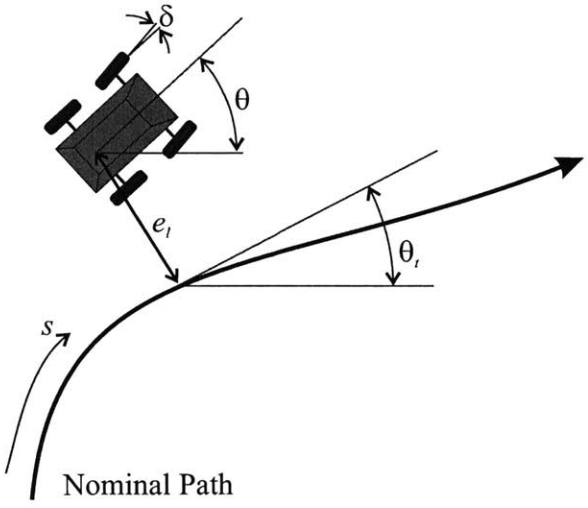


**Figure C.1: Path coordinates for vehicle model (adapted from [50]).**

A kinematic model of a front-steered vehicle in path coordinates (see Figure C.1) can be given as [50]:

$$\begin{aligned}\dot{s} &= v\cos e_\theta\left(\frac{1}{1-L\kappa(s)}\right)\\ \dot{e}_l &= v\sin e_\theta\\ \dot{e}_\theta &= \frac{v}{e_l}\tan\delta - \frac{v\kappa(s)\cos e_\theta}{1-L\kappa(s)}\end{aligned} \tag{C.3}$$

The control variable, $u$, can be introduced as:

$$u = \frac{v}{e_l}\tan\delta - \frac{v\kappa(s)\cos e_\theta}{1-L\kappa(s)} \tag{C.4}$$

Thus, Equation C.3 can be given as:

$$\begin{aligned}\dot{s} &= v\cos e_\theta\left(\frac{1}{1-L\kappa(s)}\right)\\ \dot{e}_l &= v\sin e_\theta\\ \dot{e}_\theta &= u\end{aligned} \tag{C.5}$$

It is assumed that both $e_l$ and $e_\theta$ are small and thus $\sin e_l \approx e_l$, $\cos e_l \approx 1$, $\sin e_\theta \approx e_\theta$, and $\cos e_\theta \approx 1$. Linearizing around $e_l = 0$ and $e_\theta = 0$ modifies the last two equations in Equation C.5 to be:

$$\begin{aligned}\dot{e}_l &= ve_\theta\\ \dot{e}_\theta &= u\end{aligned} \tag{C.6}$$

The control law is defined as:

$$u = \left(-K_\theta(e_\theta) - K_l(e_l)\right) \tag{C.7}$$

where $K_\theta > 0$ and $K_l > 0$. Combining Equations C.6 and C.7 yields:

$$\ddot{e}_l + K_\theta\dot{e}_l + \tilde{K}_l e_l = 0 \tag{C.8}$$

where:

$$\tilde{K}_l = vK_l \tag{C.9}$$

In order to regulate the system, velocity scaling is employed such that $K_l = K_l(v)$ [8]. Thus, $K_l$ is decreased as velocity increases. This control law has proven to be effective for ARTEmiS path tracking in both simulations and experiments.

## C.2 Velocity Control

It can be assumed that ARTEmiS' engine can impart a force $F_{powertrain}$, as shown in Figure C.2. For simplicity, it can be assumed that the drag and rolling resistance forces are linear functions of vehicle velocity.

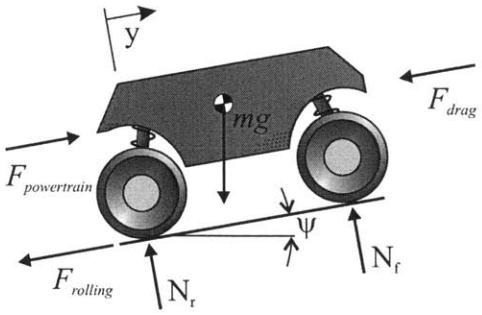


**Figure C.2: Free body diagram of a vehicle**

Summing the forces in the y-direction yields:

$$m\ddot{y} = F_{powertrain} - b\dot{y} - mg\sin\psi \tag{C.10}$$

where the damping coefficient, $b$, represents the combined effect of rolling resistance and aerodynamic drag. Rearranging Equation C.10 yields:

$$m\dot{v} + bv = F_{powertrain} - mg\sin\psi \tag{C.11}$$

A new input function is introduced:

$$F = F_{powertrain} - mg\sin\psi \tag{C.12}$$

Combining Equations C.11 and C.12 yields:

$$\frac{V(s)}{U(s)} = \frac{1}{ms + b} \tag{C.13}$$

A proportional controller is thus suitable; however, a derivative term is added to decrease overshoot:

$$u = K_p e_v + K_d \dot{e}_v \tag{C.14}$$

where $K_p$ and $K_d$ are static positive gains, $v_d$ is the desired velocity, and $e_v = v_d - v$. This control law has proven to effectively regulate vehicle speed in both simulations and experiments.

# APPENDIX D

# SPEED COMPARISON WITH FULL SIZED VEHICLE

The work presented in this thesis uses speed as a metric. Thus, it is important to understand how speeds obtained in experiments with ARTEmiS compare to experiments with full sized vehicles. This appendix describes a methodology for comparing the speeds obtained with ARTEmiS to speeds obtained with a HMMWV-class vehicle. HMMWV-class vehicles have been commonly used by other researchers [43,44,15].

To accurately compare the speeds of two differently sized vehicles, the vehicles must exhibit geometric, dynamic, and kinematic similarity [81]. Two objects are considered geometrically similar if the smaller could be enlarged to match the size of the larger. Although ARTEmiS is not a scale replica of a HMMWV class vehicle, if it were enlarged such that its wheelbase equaled a HMMWV's wheelbase, the two vehicles would be considered geometrically similar. Note that terrain also scales geometrically.

Dynamic similarity occurs when corresponding points on the two vehicles experience the same forces at corresponding times [81]. Dynamically similarity can be shown by employing the Buckingham Pi Theorem [7]. This theorem calculates a mathematical relationship between a two objects that describes how a parameter of one object is proportional to the same parameter of the other object. This relationship is shown below for velocity.

Three known parameters of each vehicle (ARTEmiS and a HMMWV) for use in the Buckingham Pi analysis are gravity, vehicle wheelbase length, and vehicle mass. A fourth parameter is velocity. This is the $\pi$ term. Table D.1 displays each parameter of concern along with its fundamental dimensions.

Table D.1: Buckingham Pi Variables

| Parameter | Symbol | Fundamental Dimension |
|---|---|---|
| Characteristic length | $L$ | $L$ |
| Mass | $M$ | $FT^2/L$ |
| Acceleration of gravity | $g$ | $L/T^2$ |
| Linear velocity | $v$ | $L/T$ |

All terms are placed inside the following matrix with the $\pi$ term on the right:

| | $L$ | $M$ | $g$ | $v$ |
|---|---|---|---|---|
| $F$ | 0 | 1 | 0 | 0 |
| $L$ | 1 | −1 | 1 | 1 |
| $T$ | 0 | 2 | −2 | −1 |

First, the known parameters are modified to form an identity matrix, and then force, length, and time are eliminated from the matrix:

| | $Mg$ | $L$ | $\left(\frac{L}{g}\right)^{1/2}$ | $v$ |
|---|---|---|---|---|
| $F$ | 1 | 0 | 0 | 0 |
| $L$ | 0 | 1 | 0 | 1 |
| $T$ | 0 | 0 | 1 | −1 |

$\rightarrow$

| | $Mg$ | $L$ | $\left(\frac{L}{g}\right)^{1/2}$ | $\frac{v}{L^{1/2}g^{1/2}}$ |
|---|---|---|---|---|
| $F$ | 1 | 0 | 0 | 0 |
| $L$ | 0 | 1 | 0 | 0 |
| $T$ | 0 | 0 | 1 | 0 |

The $\pi$ term is now given on the right hand side of the matrix:

$$\pi = \frac{V}{L^{1/2}g^{1/2}} \tag{D.1}$$

Thus, the relationship between velocities for the two vehicles can be given as:

$$\frac{v_F}{\sqrt{L_F g}} = \frac{v_A}{\sqrt{L_A g}} \tag{D.2}$$

where $F$ refers to the full size vehicle and $A$ refers to ARTEmiS. Solving for $v_A$ gives:

$$v_A = v_F \sqrt{\frac{L_A}{L_F}} \tag{D.3}$$

This corresponds to the Froude number which is a relationship that determines dynamic similarity. The two vehicles are now considered to be dynamically similar. Since kinematic similarity is obtained if the system exhibits both geometric and dynamic similarity, ARTEmiS and the full sized vehicle are now geometrically, kinematically, and dynamically similar [30]. Figure D.1 shows a comparison of equivalent speeds for ARTEmiS and a full size vehicle. ARTEmiS has performed hazard avoidance maneuvers at speeds up to 8 m/s (18 mph) which roughly corresponds to a full size vehicle traveling at 20 m/s (45 mph). More information about dynamic similarity as it applies to small mobile ground vehicles can be found in [84].

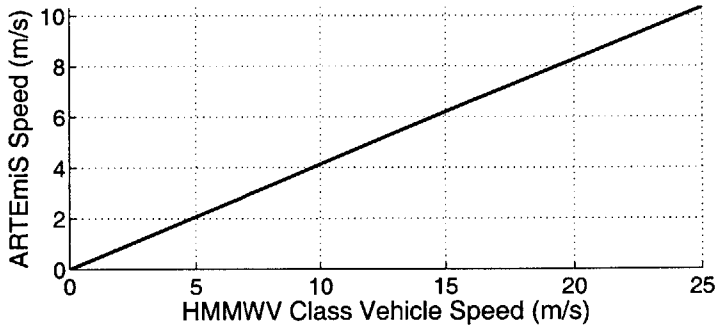


**Figure D.1: Comparison of equivalent speeds for ARTEmiS and a HMMWV**