
Results in Combined Route Traversal and Collision Avoidance

Stephan Roth, Bradley Hamner, Sanjiv Singh, and Myung Hwangbo

Carnegie Mellon University, 5000 Forbes Ave., Pittsburgh, PA 15213, USA.
{stephan+@cs.cmu.edu, bhamner@andrew.cmu.edu, ssingh@ri.cmu.edu,
myung@andrew.cmu.edu}

Summary. This paper presents an outdoor mobile robot capable of high-speed navigation in outdoor environments. Here we consider the problem of a robot that has to follow a designated path at high speeds over undulating terrain. It must also be perceptive and agile enough to avoid small obstacles. Collision avoidance is a key problem and it is necessary to use sensing modalities that are able to operate robustly in a wide variety of conditions. We report on the sensing and control necessary for this application and the results obtained to date.

Key words: outdoor mobile robot, path following, collision avoidance

1 Introduction

While the use of mobile robots in indoor environments is becoming common, the outdoors still present challenges beyond the state of the art. This is because the environment (weather, terrain, lighting conditions) can pose serious issues in perception and control. Additionally, while indoor environments can be instrumented to provide positioning, this is generally not possible outdoors at large scale. Even GPS signals are degraded in the presence of vegetation, built structures and terrain features. In the most general version of the problem, a robot is given coarsely specified via points and must find its way to the goal using its own sensors and any priori information over natural terrain. Such scenarios, relevant in planetary exploration and military reconnaissance are the most challenging because of the many hazards – side slopes, negative obstacles and obstacles hidden under vegetation – that must be detected. A variant of this problem is for a robot to follow a path that is nominally clear of obstacles but not guaranteed to be so. Such a case is necessary for outdoor patrolling applications where a mobile robot must travel over potentially great distances without relying on structure such as beacons and lane markings. In addition to avoiding obstacles, it is important that the vehicle stay on the designated route as much as possible.

Perception is typically the bottleneck in outdoor navigation, especially at speeds higher than a few meters/sec. This is primarily because perception of small obstacles (as small as 15 cm high) at or beyond the stopping distance ahead of the robot is typically only possible using laser ranging. Laser ranging produces detailed shape of the terrain but is limited in sampling and scanning speed.

Here we report on the perception and guidance that we have developed for an outdoor patrolling robot (Figure 1) that uses two low-cost laser scanners to develop an understanding of the world around it. In specific, we report on methods of obstacle detection and collision avoidance for this robot while it travels at speeds at up to 5 m/s.



Fig. 1. Grizzly is a navigation test-bed built upon a commercially available All Terrain Vehicle (ATV). It uses two laser scanners to perceive the shape of the world. The vehicle is equipped with differential GPS and a six-axis inertial measurement unit that provides accurate attitude.

2 Related Work

There has been a great deal of attention paid to parts of the problem of autonomous operation in semi-structured environments such as in ports [6], underground mines [9], and highways [3]. In some of these cases, the environment can be controlled enough that obstacle detection can be simplified to ensuring that the machines are capable of stopping for people or vehicle sized obstacles. Autonomous machines operating in natural environments, however, must be able to detect several different types of obstacles including side slopes and negative obstacles. This is accomplished by using sensors that can determine the shape of the world around them. Stereo vision [11], color segmentation [1], radar [8] and laser range finders [5] have all been used for obstacle detection. Unfortunately, passive vision suffers from lighting, color constancy, and

dynamic range effects that cause false positives and false negatives. Radar is good for large obstacles, but localization is an issue due to wide beam widths. Single axis laser scanners only provide information in one direction, and can be confounded by unmeasured pitching motion and mis-registration. Two axis scanners are also used, which provide more information, but are very costly.

Several systems have demonstrated off road navigation. The Demo III XUV drives off-road and reaches speeds up to 10 meters per second. The speeds are high, but the testing environments are rolling meadows with few obstacles. Obstacles are given a clearance which is wider than the clearance afforded by extreme routes. When clearance is not available, the algorithm plans slower speeds [5]. Sandstorm, a robot developed for desert racing, has driven extreme routes at speeds up to 22 meters per second, but makes an assumption that it is traveling on slowly varying roads. If an obstacle is encountered in the center of a road, the path cannot change rapidly enough to prevent collision [4].

Our work is related to several previous research themes. The first connection is to the research in autonomous mobile robots for exploration in planetary environments [10][11] that uses traversability analysis to find obstacles that a vehicle could encounter. The second connection is to a method of scanning the environment by sweeping a single-axis laser scanner [2] that allows detection of obstacles even when the vehicle is translating and pitching. A third connection is to a method of collision avoidance that is based on models of human navigation in between discrete obstacles [7].

3 Approach

Here we discuss the two main parts of our approach – obstacle detection and collision avoidance.

3.1 Obstacle Detection

For high speed navigation, the sensors required depend on the vehicle's speed, stopping distance and minimum obstacle size. At higher speeds, where stopping distances are greater, the obstacles must be detected at a greater distance. In order to detect smaller obstacles, the measurement density of the sensor must be correspondingly greater. Our goal is to enable the vehicle to travel at speeds of up to 5 m/s while detecting obstacles as small as 20cm × 20cm. In other work with lower speed vehicles moving at 2 m/s [2], we find that a single sweeping laser is sufficient for detecting obstacles. The sweeping laser system consists of a single Sick laser turned so it is scanning a vertical plane. A motor mechanically sweeps the vertical plane back and forth, thus building a 3-D map of the terrain in front of the vehicle. However, for the higher speed obstacle detection in this application, we find that the sweeping laser alone cannot provide a sufficient density of laser measurements to detect

small obstacles at higher speeds. Accordingly, a second fixed laser is deemed necessary (Figure 2).

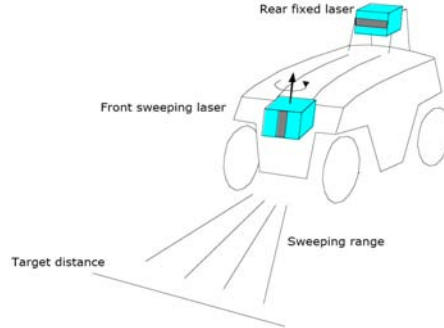


Fig. 2. Configuration of lasers scanners on the vehicle. The fixed laser concentrates its scans 10m in front of the vehicle, giving an early detection system. The sweeping laser concentrates its data closer to the vehicle, giving the ability to track obstacles that are closer to the vehicle.

The addition of a second fixed laser provides several advantages over the single sweeping laser. Primarily, the fixed laser is pointed 10m in front of the vehicle and increases the density of laser data at points far from the front of the vehicle. Now smaller obstacles are detected at a distance sufficient for safe avoidance. The sweeping laser system concentrates its data closer to the vehicle, so obstacles nearer the vehicle are tracked. A second advantage of the two laser system is that they collect orthogonal sets of data. The sweeping laser is best suited for detecting pitch type obstacles, while the fixed laser is best suited for detecting roll type obstacles. The two laser systems complement each other by performing best for these two different types of obstacles.

The addition of a second laser by itself is not enough to guarantee detecting obstacles in all cases. When following curved paths, we find it is not enough to simply sweep the laser in a fixed range. It is necessary to bias the sweeping laser so it points into turns. Figure 4 shows a representation of the number of laser hits that would be received by a $15\text{cm} \times 15\text{cm}$ obstacle located a distance greater than the vehicle's stopping distance from the front of the vehicle. Areas of red indicate a high number (>60) of hits, and areas of blue indicate a lower number (10-20). The first picture shows the number of hits when the laser is swept between a fixed 20 degree range centered about the front of the vehicle.

It is clear from the figure that there is sufficient laser data to detect obstacles along the straight section. However, along the turn the number of hits decreases dramatically. The lower density of laser data increases the chances that an obstacle will not be detected while the vehicle is turning. Figure 4(b)

shows the number of hits when the sweeping laser is biased to point into the turn. Compared to the unbiased case, the number of laser hits on the obstacle greatly increases in the area where the vehicle is turning.

With data from two lasers, we use two obstacle detection algorithms: a traversability analysis and a line scan gradient analysis. In the traversability analysis, data from both lasers is used to produce a point cloud of the terrain in front of the vehicle. Vehicle-sized planar patches are fit to the point cloud data, and the fitted data gives three measures useful in identifying obstacles: plane orientation (roll, pitch), roughness (the residual of the fit) and the height of data points above the plane. These measured values are used to classify areas as *untraversable* or *clear*. While the traversability analysis is a simple way of detecting obstacles, it can produce false positives due to inaccurate calibration of the two lasers and/or incorrect synchronization with positioning. To supplement the traversability analysis, the slope of segments of individual line scans from the sweeping laser is also calculated as in [2]. If the slope of a scan segment is above a given threshold, it is tagged as a *gradient* obstacle. Because the gradient analysis uses piecewise segments of an individual line scan, it is not susceptible to misregistration as the traversability analysis can be.

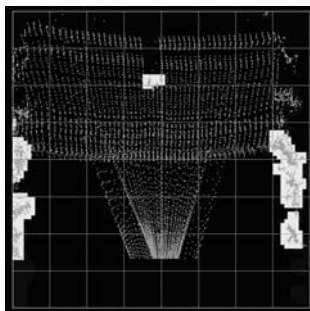


Fig. 3. Overhead view of laser data from from the two scanners. Data over a window of time are registered to a common reference frame and obstacles are found by analyzing traversability and gradient of the individual line scans.

To classify an object as a true obstacle, both the gradient and traversability analyses must agree. The combination of the two obstacle detection algorithms compensates for the weaknesses of the two individual algorithms and dramatically reduces the false obstacle detection rate. Because the gradient analysis looks at only an individual line scan from the sweeping laser, it cannot take advantage of integrating multiple scans over time like the traversability analysis can. However, by only using single line scans, the gradient analysis is relatively immune to mis-registration problems that plague the traversability analysis.

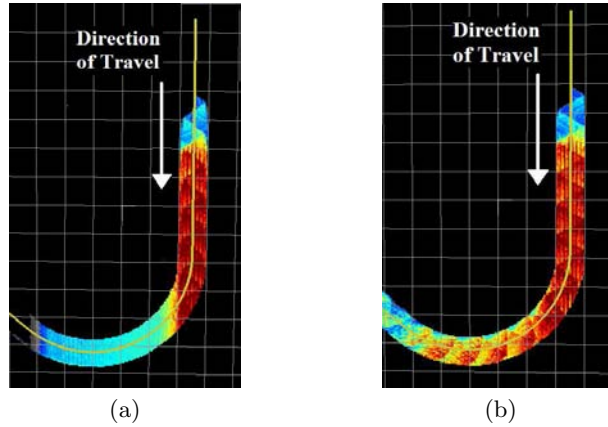


Fig. 4. Grid representation of laser hits by both the fixed and sweeping lasers on a $15\text{cm} \times 15\text{cm}$ obstacle when sweeping with and without biased laser at 4m/s . (a) shows a representation of the number of hits without biasing the laser when going around turns. (b) shows the number of hits when biasing the laser. Areas of blue indicate a low number of laser hits (10-20). Red areas indicate a high number of hits (>60). Biasing the laser when going around turns increases the laser hit density.

3.2 Collision Avoidance

The goal of our collision avoidance system is to follow a path and avoid obstacles along the way. When an obstacle is detected in front of the vehicle, the vehicle should swerve to avoid it and return to the path in a reasonable fashion. If there are multiple obstacles on the path, the vehicle must navigate between them. Sometimes an obstacle may block the entire path. In this case, the vehicle must stop before colliding with it. An ideal collision avoidance algorithm would accept a map of hazards and determine steering and speed to navigate in between these. Since this algorithm must run many times a second, ideally it would have low computational complexity.

Fajen and Warren report a reactive method of collision avoidance based on experiments to determine how humans avoid obstacles [7]. The method uses the positions of a goal point and obstacle points relative to the current vehicle position to derive an instantaneous steering angle. We developed a path-following obstacle avoidance algorithm that extends this method. Since the vehicle simply avoids obstacles without planning a full path, we call the algorithm *Dodger*.

Consider the vehicle and a desired goal point. If the goal is at a large angle to the current vehicle heading, as in Figure 5(a), then the vehicle must steer sharply. Smaller angular differences, as in Figure 5(b), mean that the vehicle does not have to steer as hard. Similarly, for greater distances to the goal, as in Figure 5(c), slight steering is sufficient. Based on these principles, Fajen and Warren develop a goal attraction function,

$$f_a(\psi_g, d_g) = \psi_g(e^{-c_g d_g} + c_s)$$

where d_g is the translational distance to the goal, ψ_g is the angular distance to the goal, c_g is a goal distance decay constant, and c_s is a scale constant to assure the goal attraction is never zero.

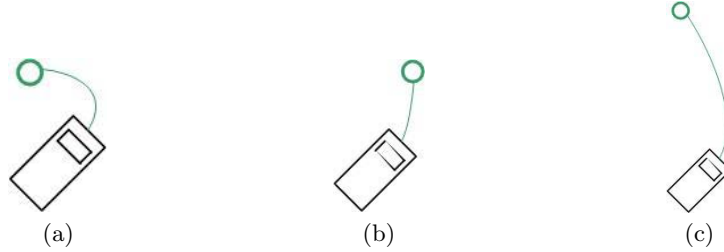


Fig. 5. Three scenarios involving driving to a goal, indicated by the green circles. The vehicle must steer proportionally to the distance and angle to the goal.

Repulsion from obstacles uses similar logic. When an obstacle is at a large angular distance, as in Figure 6(a), the vehicle does not need to turn sharply to avoid it. When the obstacle is far from the vehicle, as in Figure 6(b), a small steering angle is sufficient. The vehicle must steer sharply only when the obstacle is close and in front of the vehicle, as in Figure 6(c). These principles can be combined into a single obstacle repulsion function,

$$f_r(\psi_o, d_o) = \psi_o(e^{-c_{o1}|\psi_o|})(e^{-c_{o2}d_o})$$

where d_o is the translational distance to the obstacle, ψ_o is the angle to the obstacle, c_{o2} is a distance decay constant, and c_{o1} is an angular decay constant.

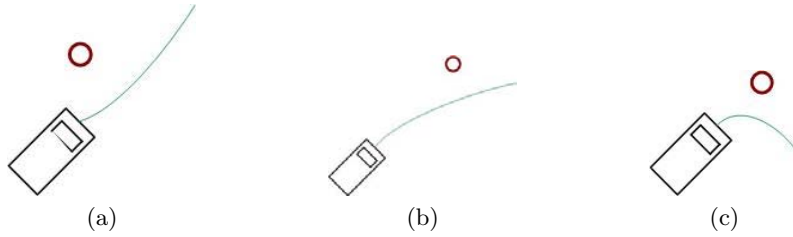


Fig. 6. Three scenarios involving avoiding obstacles, represented by the red circles. The vehicle must steer proportionally to the distance and angle to obstacles.

This function is applied to every obstacle, and the result is summed together. Note that this treats obstacles as individual points. To represent real

obstacles, we discretize them into collections of points spaced ten centimeters apart (Figure 7).

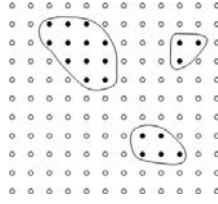


Fig. 7. We represent obstacles as collections of points spaced ten centimeters apart. The obstacle repulsion function is applied to each black obstacle point individually.

The goal attraction and obstacle repulsion are combined to get the control equation:

$$\dot{\phi}^* = -k_g f_a(\psi_g, d_g) + k_o \sum_{o \in O} f_r(\psi_o, d_o)$$

where k_g and k_o are relative weighting constants and $\dot{\phi}^*$ is the commanded steering velocity.

We have extended the original formulation by Fajen and Warren in several ways. First, the original obstacle repulsion function is multiplied by the angle to the obstacle. This means that if the vehicle is headed straight towards an obstacle, the angular repulsion term is zero. The theory is that the vehicle will turn slightly away from the obstacle at first (crossing in front if necessary), the angle will increase, and eventually the vehicle will fully turn away from the obstacle. However, at high speeds, there may not be enough time for that to happen. We modify the function to have high repulsion at small angles, and accept the consequences of getting into local minima more easily. The new obstacle repulsion function becomes

$$f_r(\psi_o, d_o) = \text{sign}(\psi_o)(e^{-c_{o1}|\psi_o|})(e^{-c_{o2}d_o})$$

Another problem occurs in areas of dense obstacles, such as the path illustrated in Figure 8(a). Here, there are obstacles everywhere in front of the vehicle. The leftward repulsion of the obstacles on the right side of the path may be greater than the rightward repulsion of the single obstacle on the path. Were it not for our speed control (see below), the vehicle would collide with the obstacle on the path. The problem is that the base system does not use all of the available information. The obstacles are directly in front of the vehicle, and therefore look threatening, but the path curves away from them. Similarly, the single obstacle may be at a large angular distance, but it is directly between the vehicle and the goal point. We introduce a new term to the obstacle repulsion function, which considers whether the obstacle is blocking the goal,

$$dist(v, g, o) = \frac{|(g_x - v_x)(v_y - o_y) - (v_x - o_x)(g_y - v_y)|}{\|g - v\|}$$

$$f_r(\psi_o, d_o, d_{vgo}) = sign(\psi_o)(e^{-c_{o1}|\psi_o|})(e^{-c_{o2}d_o})(1+c_{o3}(d_{max}-max(d_{max}, d_{vgo}))^2))$$

where d_{vgo} is the perpendicular distance from the obstacle to the vector between the vehicle and the goal calculated by $dist(v, g, o)$, and d_{max} is some maximum distance from that vector. The obstacles to the right are far away from the goal vector, so their repulsion is the same as before. However, now the single obstacle has greater repulsion, assuring that the vehicle will not drive towards it (Figure 8(b)).

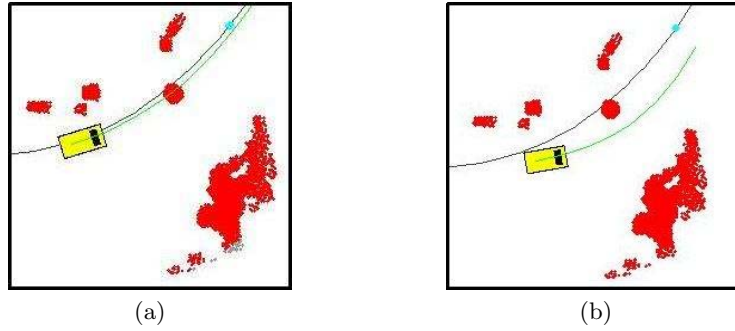


Fig. 8. The dark line is the desired path. The lighter line represents the vehicle's future path when using the Dodger algorithm. The dot on the desired path is the goal point used by Dodger. In (a), without using the goal vector term, the obstacles on the right side of the desired path collectively have a much larger repulsion than the single obstacle that is actually on the path. That problem is corrected in (b), where the goal vector term greatly increases the repulsion by the single obstacle.

Following a path using Dodger is done by first finding the point on the path closest to the vehicle. The goal point is set to a point some distance down the path. When an obstacle appears in front of the vehicle, this distance is increased so as to allow the vehicle to maneuver around the obstacle. Fajen and Warren's experiments showed that humans consistently kept the same speeds as they traveled. However, when obstacles appear, we would like the vehicle to slow down, to allow for greater possible steering angles, and thus greater maneuverability. This is a simple proportional function based on the largest obstacle repulsion. If the largest obstacle score is high enough, that is, if there is an obstacle directly in front of the vehicle, then we stop the vehicle before a collision.

Speed control is also done by predicting the course that Dodger would take in the future. Using Dodger's output steering angle and speed, we run a forward integration of the vehicle model interleaved with the steering control, to predict where the vehicle will be a short amount of time later. We build a

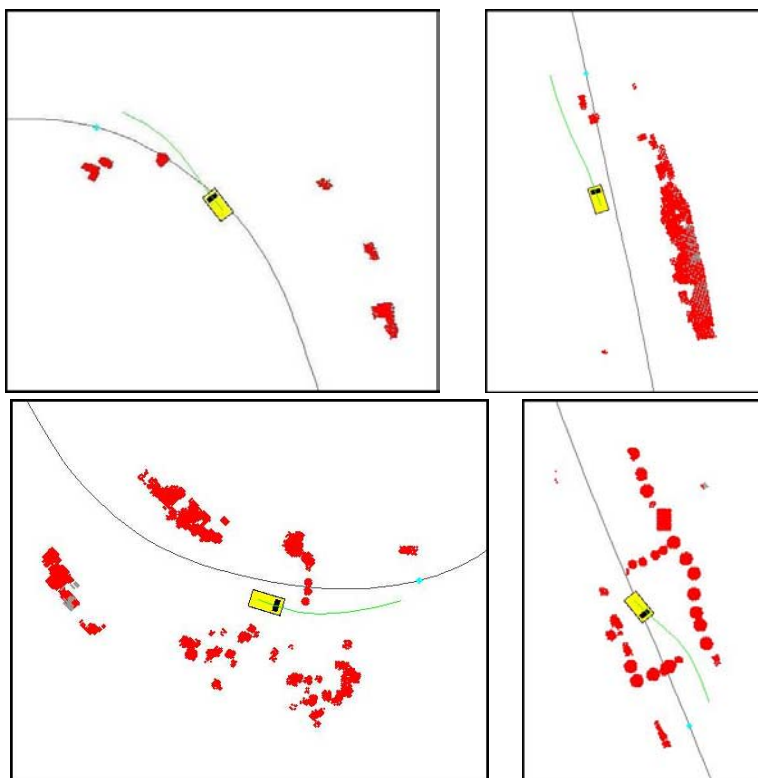


Fig. 9. In these situations, Dodger safely guides the vehicle around the detected obstacles.

path from these predictions over four seconds (shown as the light line extending forward from the vehicle in the figures). This predicted path accounts for curvature limits based on the vehicle’s speed. Then we check along the path for collisions. If there is a collision along the path, then we can slow the vehicle immediately, rather than waiting until it gets closer to the obstacle. Again, the slow-down allows the vehicle more maneuverability and a greater chance of the collision being avoided. Dodger works well for avoiding single obstacles, some situations with multiple obstacles, including slaloms, on straight-aways, and around corners (shown in Figure 9).

However, there are specific situations in which Dodger does not find a path around the obstacle, and the vehicle is forced to stop. When the obstacle is wide, there are points on both sides of the vehicle which counteract each other, so the vehicle never gets all the way around the obstacle (Figure 10(a)). Also, when there is an obstacle around a corner, Dodger prefers to go outside the turn around the obstacle, rather than inside. This is because the obstacle points on the inside of the turn are closer to the goal vector, and therefore have

more repulsion. This causes a problem when the obstacle covers the outside of the corner (Figure 10(b)).

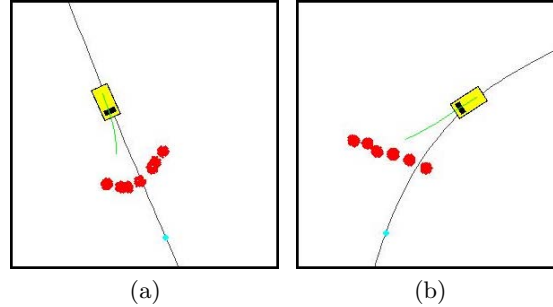


Fig. 10. In (a), due to the curved shape and width of the obstacle, some of the rightward repulsion is cancelled out by a leftward repulsion. Then Dodger does not find a way all the way around the obstacle, and stops before a collision. In (b), there is enough room to avoid this obstacle to the left. However, the obstacle points closer to the goal vector exhibit a larger rightward repulsion. The obstacle is too wide for the vehicle to avoid around the outside, so Dodger stops the vehicle before collision.

Using the predicted path, the system can detect situations in which Dodger fails to direct the vehicle around the obstacle. When the predicted path stops in front of an obstacle, the system invokes a planning algorithm, like D^* , to get a new goal point which will help Dodger around the obstacles. First, the planning algorithm constructs a small map of the area in the vicinity of the vehicle (Figure 11(a)). The goal location passed to D^* is Dodger's goal point. Next, the planning algorithm constructs an optimal path around the obstacles to that goal location. The system then starts at the goal point and walks backwards along the optimal path, stopping when there are no obstacles on a straight line to the vehicle. This unblocked position is selected as a new goal for Dodger, and the Dodger algorithm is run again. The new goal point is closer than the old one, and is off to one side of the problem obstacles, so it has more influence than the original goal point. When Dodger is run again, the new goal point pulls the vehicle to one side of the obstacles. In essence, the planning algorithm chooses a side for Dodger to avoid on. The system continues this hybrid method until Dodger, using its normal goal point, gives a predicted path that safely avoids the obstacles (11(c)). The D^* augmentation to Dodger is especially useful in complex obstacle configurations, as shown in Figure 12. Running Dodger with the planning algorithm takes more computation time, so to be safe, we also slow the vehicle down when the planning algorithm is running.

In both of the above cases, we can detect the impending collision and stop the vehicle in time. However, there are some cases in which Dodger would exhibit undesirable behavior while not actually colliding with an obstacle. For

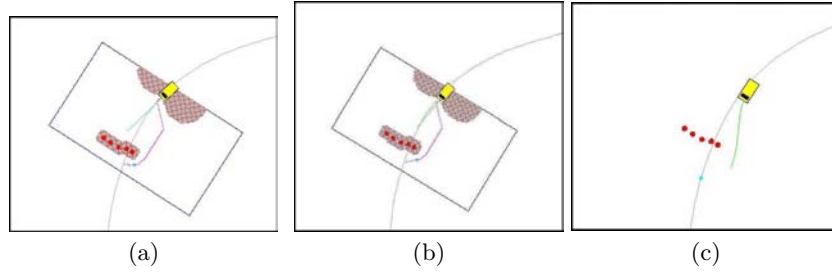


Fig. 11. In (a), the system predicts a collision and invokes D^* . The map covers only a small area between the vehicle and the original goal point. Obstacles are added to the map, and points within the vehicle’s minimum turning radius are also marked as untraversable. The optimal path from D^* goes around the obstacle, and the furthest visible point along the D^* path is set as the new goal point. Dodger is run again using this goal. In (b), the new D^* goal point has pulled the vehicle a little to the left, but not far enough yet, since the system still predicts collision. D^* continues to be invoked. In (c), the vehicle is far enough to the left that the system no longer predicts a collision if the regular goal point is used with Dodger, so D^* is no longer necessary.

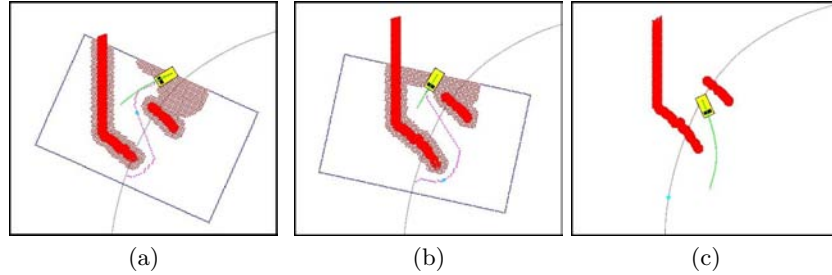


Fig. 12. The D^* augmentation to Dodger can also lead the vehicle through complex configurations of obstacles. In (a), Dodger finds no way around the wall of obstacles, so D^* is invoked. In (b), the goal obtained from the D^* path pulls the vehicle to the left. In (c), Dodger alone can navigate the vehicle past the remaining obstacles.

example, Figure 13(a) shows a case where obstacles on both sides of the path are actually to the left of the vehicle and repel the vehicle off the desired path around the obstacles, even though the desired path is clear. To prevent the vehicle from unnecessarily diverging from the desired path, we use a “ribbon” method. We construct a ribbon of fixed distances down the path and to either side. If there are no obstacles on this ribbon and the vehicle is currently within the ribbon, then we zero any obstacle repulsion. The result is a steering angle entirely based on the goal attraction, and the vehicle successfully tracks the path (Figure 13(b)).

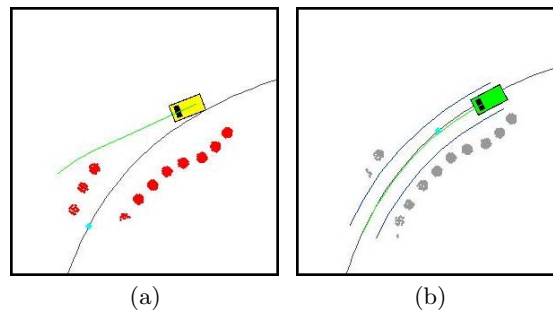


Fig. 13. In (a), the obstacles on both sides of the path repel the vehicle rightward. As a result, the vehicle leaves the path, even though there is no obstacle on the path. In (b), the ribbon method is being used. The dark lines on either side of the path denote the ribbon. There are no obstacles within the ribbon, so the total obstacle repulsion is set to zero, and the vehicle follows the path.

4 Results

The system presented here is able to perform high speed off road navigation at speeds up to 5m/s. The tightly coupled GPS + IMU localization system provides reliable position estimates in areas with limited GPS availability. The combination of two laser systems, one fixed and the other sweeping, enables us to detect obstacles as small as 30cm high and 30cm wide. The obstacle avoidance algorithm allows us to avoid these obstacles even while traveling at 5m/s. The system described here has successfully performed over 100 km of autonomous travel.

5 Conclusions

We have developed a method of obstacle detection and collision avoidance that is composed of low cost components and has low complexity but is capable of state of the art performance. The advantage of being able to actuate the laser scanning is that it provides for an even distribution of laser range data as the path turns.

So far we have used shape to separate obstacles from clear regions. The next step is to allow for recognition of materials so that vegetation can be appropriately recognized.

References

1. P. H. Batavia. and S. Singh. Obstacle detection using adaptive color segmentation and color homography. In *Proceedings of the International Conference on Robotics and Automation. IEEE*, May 2001.

2. P. H. Batavia and S. Singh. Obstacle detection in smooth, high-curvature terrain. In *Proceedings of the International Conference on Robotics and Automation*, Washington, D.C., 2002.
3. T. Jochem C. Thorpe and D. Pomerleau. The 1997 automated highway free agent demonstration. In *IEEE Conference on Intelligent Transportation Systems*, November 1997.
4. M. Clark T. Galatali J.P. Gonzalez J. Gowdy A. Gutierrez S. Harbaugh M. Johnson-Roberson H. Kato P.L. Koon K. Peterson B.K. Smith S. Spiker E. Tryzelaar C. Urmson, J. Anhalt and W.L. Whittaker. High speed navigation of unrehearsed terrain: Red team technology for grand challenge 2004. Technical report.
5. D. Legowik S. A. Murphy D. Coombs, A. Lacaze. Driving autonomously offroad up to 35 km/h. In *Proceedings of the IEEE Intelligent Vehicles Symposium*, 2000.
6. H.F. Durrant-Whyte. An autonomous guided vehicle for cargo handling applications. *International Journal of Robotics Research*, 15, 1996.
7. B. Fajen and W. Warren. Behavioral dynamics of steering, obstacle avoidance, and route selection. *Journal of Experimental Psychology: Human Perception and Performance*, 29(2), 2003.
8. D. Langer and T. Jochem. Fusing radar and vision for detecting, classifying, and avoiding roadway obstacles. In *Proceedings of the IEEE Symposium on Intelligent Vehicles*, 1996.
9. et al. S. Scheduling. An experiment in autonomous navigation of an underground mining vehicle. In *IEEE Transactions on Robotics and Automation*, 1999.
10. et al. S. Singh. Recent progress in local and global traversability for planetary rovers. In *Proceedings of the IEEE International Conference on Robotics and Automation*, April 2000.
11. C. Urmson and M.B. Dias. Vision based navigation for sun-synchronous exploration. In *Proceedings of the International Conference on Robotics and Automation*, May 2002.