Online Speed Adaptation using Supervised Learning for High-Speed, Off-Road Autonomous Driving

David Stavens, Gabriel Hoffmann, and Sebastian Thrun
Stanford Artificial Intelligence Laboratory
Stanford, CA 94305-9010
{stavens,gabeh,thrun}@robotics.stanford.edu

Abstract

The mobile robotics community has traditionally addressed motion planning and navigation in terms of steering decisions. However, selecting the best speed is also important – beyond its relationship to stopping distance and lateral maneuverability. Consider a high-speed (35 mph) autonomous vehicle driving off-road through challenging desert terrain. The vehicle should drive slowly on terrain that poses substantial risk. However, it should not dawdle on safe terrain. In this paper we address one aspect of risk – shock to the vehicle. We present an algorithm for trading-off shock and speed in real-time and without human intervention. The trade-off is optimized using supervised learning to match human driving. The learning process is essential due to the discontinuous and spatially correlated nature of the control problem – classical techniques do not directly apply. We evaluate performance over hundreds of miles of autonomous driving, including performance during the 2005 DARPA Grand Challenge. This approach was the deciding factor in our vehicle's speed for nearly 20% of the DARPA competition – more than any other constraint except the DARPA-imposed speed limits – and resulted in the fastest finishing time.

1 Introduction

In mobile robotics, motion planning and navigation have traditionally focused on steering decisions. This paper presents *speed* decisions as another crucial part of planning – beyond the relationship of speed to obstacle avoidance concerns, such as stopping distance and lateral maneuverability. Consider a high-speed (35 mph) autonomous vehicle driving off-road through challenging desert terrain. We want the vehicle to drive slower on more dangerous terrain. However, we also want to minimize completion time. Thus, the robot must trade-off speed and risk in real-time. This is a natural process for human drivers, but it is not at all trivial to endow a robot with this ability.

We address this trade-off for one component of risk: the shock the vehicle experiences. Minimizing shock is important for several reasons. First, shock increases the risk of damage to the vehicle, its mechanical actuators, and its electronic components. Second, a key perceptive technology, laser range scanning, relies on accurate estimation of orientation. Shock causes the vehicle to shake violently, making accurate estimates difficult. Third, shocks substantially reduce traction during oscillations. Finally, we demonstrate that shock is strongly correlated with speed and, independently, with subjectively difficult terrain. That is, minimizing shock implies slowing on challenging roads when necessary – a crucial behavior to mitigate risk to the vehicle.

Our algorithm uses the linear relationship between shock and speed which we derive analytically. The algorithm has three states. First, the vehicle drives at the maximum allowed speed until a shock threshold is exceeded. Second, the vehicle slows immediately to bring itself within the shock threshold using the relationship between speed and shock. Finally, the vehicle gradually accelerates. It returns to the first state, or the second if the threshold is exceeded during acceleration.

To maximize safety, the vehicle must react to shock by slowing down immediately, resulting in a discontinuous speed command. Further, it should accelerate cautiously, because rough terrain tends to be clustered, a property we demonstrate experimentally. For these reasons, it is not possible to determine the optimal parameters using classical control techniques. The selection of parameters cannot be tuned by any physical model of the system, as the performance is best measured statistically by driving experience.

Therefore, we generate the actual parameters for this algorithm – the shock threshold and acceleration rate – using supervised learning. The algorithm generates a reasonable match to the human teacher. Of course, humans are proactive, also using perception to make speed decisions. We use inertial-only cues because it is very difficult to assess terrain roughness with perceptive sensors. The accuracy required exceeds that for obstacle avoidance by a substantial margin because rough patches are often just a few centimeters in height. Still, experimental results indicate our reactive approach is very effective in practice.

Our algorithm is part of the software suite developed for Stanley, an autonomous vehicle entered by Stanford University in the 2005 DARPA Grand Challenge [DARPA, 2004]. Stanley is shown in Figure 1. The Grand Challenge was a robot race organized by the U.S. Government. In 2005, vehicles had to navigate 132 miles of unrehearsed desert ter-

Figure 1: Stanley won the 2005 DARPA Grand Challenge by completing a 132 mile desert route in just under 7 hours. The algorithm described in this paper was the deciding factor in Stanley's speed for nearly 20% of the race – more than any other constraint except the DARPA-imposed speed limits.

rain. The algorithm described here played an important role in that event. Specifically, it was the deciding factor in our robot's speed for nearly 20% of the competition – more than any other constraint except the DARPA imposed speed limit. It resulted in the fastest finishing time of any robot.

2 Related Work

There has been extensive work on terramechanics [Bekker, 1956; 1969; Wong, 1989], the guidance of autonomous vehicles through rough terrain. That work generally falls into two categories. One focuses on high-speed navigation. However, it requires that terrain maps and topology be known beforehand [Shimoda *et al.*, 2005; Spenko *et al.*, 2006]. The other analyzes terrain ruggedness in an online fashion by driving over it [Sadhukhan *et al.*, 2004; Brooks and Iagnemma, 2005; Iagnemma *et al.*, 2004]. However, vehicle speed is very slow, less than 1.8 mph. Further, classification focused on terrain type, not roughness. Our work combines the strengths of both categories: online terrain roughness assessment and high-speed navigation.

Successful obstacle detection systems have been built for high-speed autonomous driving [Thrun *et al.*, 2006b; 2006a; Urmson *et al.*, 2006; 2004; Kelly and Stentz, 1998]. However, as we mentioned in Section 1, the precision needed for determining terrain ruggedness exceeds that required for obstacle avoidance by a substantial margin. Thus, while these systems are excellent at detecting and avoiding static obstacles, they do not protect the vehicle from the effects of road roughness.

Some work has explicitly considered speed as important for path planning. For example, [Fox *et al.*, 1996] presents a method for trading off "progress" and "velocity" with regard to the number and type of obstacles present. That is, slower speeds are desirable in more cluttered environments. Faster speeds are better for open spaces. This concept, applied to desert driving, is analogous to slowing for turns in the road and for discrete obstacles. Our vehicle exhibits those behaviors as well. However, they are separate from the algorithm we describe in this paper.

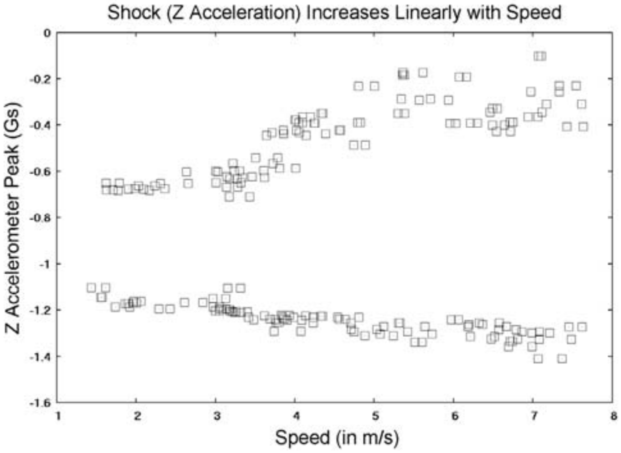

Figure 2: We find empirically that the relationship between perceived vertical acceleration amplitude and vehicle speed over a static obstacle can be approximated by a linear function in the operational region of interest.

Other entrants in the 2005 DARPA Grand Challenge realized the importance of adapting speed to rough terrain. For example, two teams from Carnegie Mellon University (CMU) used "preplanning" for speed selection [Gutierrez *et al.*, 2005; Urmson *et al.*, 2006; 2004; NOVA, 2006]. For months, members of the CMU teams collected extensive environment data in the general area where the competition was thought to be held. Once the precise route was revealed by DARPA, two hours before the race, the team preprogrammed speeds according to the previously collected data. The end-result of the CMU approach is similar to ours. However, our fully-online method requires neither human intervention nor prior knowledge of the terrain. The latter distinction is particularly important since desert terrain readily changes over time due to many factors, including weather.

3 Speed Selection Algorithm

Our algorithm is derived in three steps. First, we derive the linear relationship between speed and shock. Second, we leverage this relationship to build an algorithm for determining an appropriate speed. Finally, we tune the parameters of the algorithm using supervised learning.

3.1 Z Acceleration Mapping and Filtering

The mapping between vehicle speed and shock experienced is fundamental to our technique. Here, shock is measured by the amplitude of acceleration in the vertical, z-direction, of the vehicle's main body. The vehicle's speed is v and position along the path is p. The ground varies in height, $z_g(p)$, pushing stiffly on the car's tires. The ground is modeled to be the sum of sinusoids of a large number of unknown spatial frequencies and amplitudes. Consider one component of this summation, with a spatial frequency of ω_s, and an amplitude A_{z,g,ω_s}. Due to motion of the vehicle, the frequency in time of z_{g,ω_s} is $\omega = v\omega_s$. By taking two derivatives, the acceleration, $\ddot{z}_{g,\omega_s}$, has an amplitude of $v^2\omega_s^2 A_{z,g,\omega_s}$ and a frequency of $v\omega_s$. This is the acceleration input to the vehicle's tire.

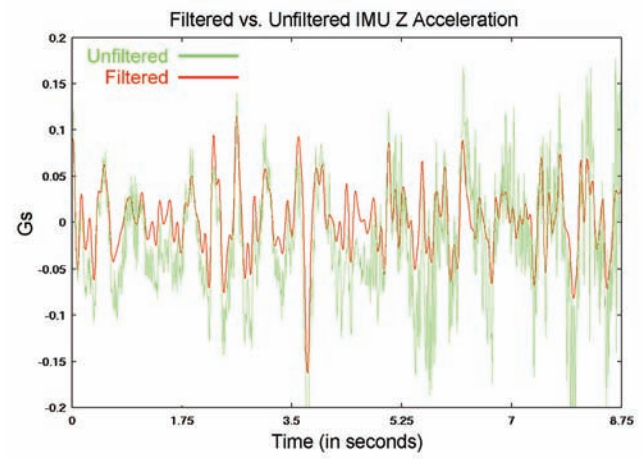

Figure 3: Filtered vs. unfiltered IMU data. The 40-tap FIR band pass filter extracts the excitation of the suspension system, in the 0.3 to 12 Hz band. This removes the offset of gravity, as terrain slope varies, and higher frequency noise, such as engine system vibration, that is unrelated to oscillations in the vehicle's suspension.

To model the response of the suspension system, we use the quarter car model [Gillespie, 1992]. The tire is modeled as a connection between the axle and the road with a spring with stiffness k_t, and the suspension system is modeled as a connection between the axle and the vehicle's main body, using a spring with stiffness k_s and a damper with coefficient c_s. Using this model, the amplitude of acceleration of the main body of the vehicle can be easily approximated, by assuming infinitely stiff tires, to be

$$A_{\ddot{z},\omega_s} = v^2\omega_s^2 A_{z,g,\omega_s}\sqrt{\frac{(c_s v\omega_s)^2 + k_s^2}{(c_s v\omega_s)^2 + (k_s - mv^2\omega_s^2)^2}} \qquad (1)$$

where m is the mass of one quarter of the vehicle, and the magnitude of the acceleration input to the tire is taken to be $v^2\omega_s^2 A_{z,g,\omega_s}$, as described above. The resonant frequency of a standard automobile is around 1 to 1.5 Hz. Below this frequency, the amplitude increases from zero at 0 Hz, slowly, but exponentially, with frequency. Above this frequency, the amplitude increases linearly, approaching $\frac{c_s v\omega_s}{m} A_{z,g,\omega_s}$. Modern active damping systems further quench the effect of the resonant frequency, leading to a curve that is nearly linear in frequency throughout, to within the noise of the sensors, which also measure the effect of random forcing disturbances.

The frequency, ω, is directly proportional to speed v, so the amplitude of acceleration response of the vehicle is also approximately a linear function of the velocity. Now, by superposition, considering the sum of the responses for all components of the terrain profile at all spatial frequencies, the amplitude of the acceleration response of the vehicle, $A_{\ddot{z}}$, is the sum of many approximately linear functions in v, and hence is approximately linear in v itself. This theoretical analysis can also be verified experimentally as in Figure 2.

This approximation assumed infinitely stiff tires. In reality, the finite stiffness of tires leads to a higher frequency resonance in the suspension system around 10 to 20 Hz, followed by an exponential drop off in $A_{\ddot{z}}$, to zero.

Although the acceleration is approximately a linear function of v, a barrier to the use of pure z-accelerometer inertial measurement unit (IMU) data is that acceleration can reflect other inputs than road roughness. First, the projection of the gravity vector onto the z-axis, which changes with the pitch of the road, causes a large, low frequency offset. Second, the driveline and non-uniformities in the wheels cause higher frequency vibrations.

To remove these effects, we pass the IMU data through a 40-tap, FIR band pass filter to extract the excitation of the suspension system, in the 0.3 to 12 Hz band. This band was found, experimentally, to both reject the offset of gravity and eliminate vibrations unrelated to the suspension system, without eliminating the vibration response of the suspension.

A comparison of the raw and filtered data is shown in Figure 3. The mean of the raw data, acquired on terrain with no slope, is about 1G, whereas the mean of the filtered data is about 0G. Therefore, solely for presentation in this figure, 1G was subtracted from the raw data.

3.2 Generating Speed Recommendations

Our speed selection algorithm (the velocity controller) has three states. Each road is assumed to have a speed limit (γ, in mph) that represents the upper bound on vehicle speed along that road. First, the vehicle travels at the speed limit until a road event generates a shock that exceeds an acceptable threshold (α, in Gs). Second, it reduces speed such that the event would have been less than α. Finally, it accelerates (at rate β, in mph/s) back to the speed limit.

More formally, recall that, for a given rough segment of road, the shock experienced by the vehicle is linear with respect to speed. Every dt seconds, the vehicle takes a reading at its current location p along the route. Filtered shock $\ddot{z}_p$ and speed v_p are observed. The optimal speed at p should have been

$$v_p^* \quad = \quad \alpha\frac{v_p}{|\ddot{z}_p|} \qquad (2)$$

That is, v_p^* is the velocity that would have delivered the maximum acceptable shock (α) to the vehicle.

Our instant recommendation for vehicle speed is v_p^*. Notice that the vehicle is slowing reactively. Reactive behavior has no effect if shock is instant, random, and not clustered. That is, our approach relies on one shock event "announcing" the arrival of another, momentarily. Experimental results on hundreds of miles of desert indicate this is an effective strategy.

We consider v_p^* an observation. The algorithm incorporates these observations over time, generating the actual recommended speed, v_p^r:

$$v_p^r \quad = \quad \min\left[\gamma, v_p^*, v_{p-1}^r + \beta dt\right] \qquad (3)$$

For our purposes, we also clamp v_p^r such that it is never less than 5 mph. We call the v_p^r series a velocity plan.

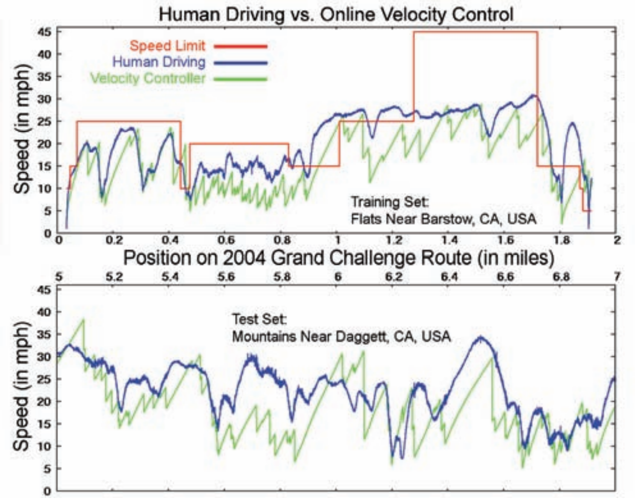

Figure 4: The results of coordinate descent machine learning are shown. The learning generated the race parameters of .25 Gs and 1 mph/s, a good match to human driving. Note that the human often drives under the speed limit. This suggests the importance of our algorithm for desert driving.

3.3 Supervised Machine Learning

We use supervised machine learning to select α and β. Some might argue we should select these parameters using explicit hardware lifetime calculations rather than matching a human teacher. However, an autonomous vehicle has hundreds of components. Analyzing the effect of shock on each would require a great deal of time. It would also be costly – each component would have to break before we could determine its limit. We believe that matching a human's speed provides a useful and more efficient bound on acceptable shock.

The velocity plan generated by a specific parameterization is scored against a human driver's velocity decisions at all p. The parameters are then iteratively modified according to a scoring function and coordinate descent.

Let v_p^h be the speed of the human driver at position p. The objective/scoring function we wish to minimize is

$$\int_p \psi \left| v_p^h - v_p^r \right| \left(1 + \alpha\beta^{-1}\right) \tag{4}$$

where $\psi = 1$ if $v_p^r \leq v_p^h$ and $\psi = 3$ otherwise. The ψ term penalizes speeding relative to the human driver. The value 3 was selected arbitrarily. The $\alpha\beta^{-1}$ term penalizes parameterizations that tolerate a great deal of shock or accelerate too slowly.

For this experiment, a human driver drove 1.91 miles of the previous year's (2004) Grand Challenge route. The vehicle's state vector was estimated by an Unscented Kalman Filter (UKF) [Julier and Uhlmann, 1997]. The UKF combined measurements from a Novatel GPS with Omnistar differential corrections, a GPS compass, the IMU, and wheel speed encoders. The GPS devices produced data at 10Hz. All other devices and the UKF itself produced data at 100Hz. All data is logged with millisecond accurate timestamps.

We do not model driver reaction time, and we correct for the delay due to IMU filtering. Thus, in this data, driver velocity at position p is the vehicle's recorded velocity at position p, according to the UKF. The algorithm's velocity at position p is the decision of the velocity controller considering all data up to and including the IMU and UKF data points logged at position p.

The parameterization learned was (α, β) = (.27 Gs, .909 mph/s). This is illustrated in Figure 4. The top and bottom halves of the figure show the training and test sets, respectively. The horizontal axis is position along the 2004 Grand Challenge route in miles. The vertical axis is speed in mph. For practical reasons we rounded these parameters to .25 and 1, respectively, which are essentially identical in terms of vehicle performance. The robot used these parameters during the Grand Challenge event and they are evaluated extensively in future sections.

We notice a reasonable match to human driving. Without the penalty for speeding, a closer match is possible. However, that parameterization causes the vehicle to drive too aggressively. This is, no doubt, because a human driver also uses perceptual cues to determine speed.

Notice that the speed limits are often much greater than the safe speed the human (and the algorithm) selects. This demonstrates the importance of our technique over blindly following the provided speed limits. Notice also, however, both the human and the algorithm at times exceed the speed limit. This is not allowed on the actual robot or in the experiments in Section 4. However, we permit it here to simplify the scoring function, the learning process, and the idea of matching a human driver's speed.

4 Experimental Results

4.1 Qualitative Analysis

In our first experiment, we evaluate the algorithm on sections of off-road driving that we identify (qualitatively) as particularly easy and, alternatively, as challenging. The purpose of this experiment is to spot-check performance. We seek to verify that our algorithm, as tuned by the learning, does not drive too slowly on easy terrain or too fast on difficult terrain.

We select two seven mile stretches from the 2005 Grand Challenge route. The first stretch includes miles 0-7. This is very easy terrain consisting of wide dirt roads and dry lakes. The second includes miles 122-129. This terrain is very challenging: a narrow mountain road with a 200 ft drop on one side. Both sections were graded (smoothed with a large plow) not long before the Grand Challenge event. Based on video collected by the vehicle and by DARPA, the mountain road is clearly much more challenging to drive. However, neither has obviously greater potential for vertical shock.

Figure 5 presents results from the first seven miles of the course – the easy example. In the figure, points represent a single shock event along the route. The vertical axis plots shock the vehicle experiences while driving the route with the velocity controller. The horizontal axis plots shock experienced driving the speed limits alone. We notice two things. First, shock above our .25G threshold is very rare. This reflects on the particularly straightforward driving conditions.

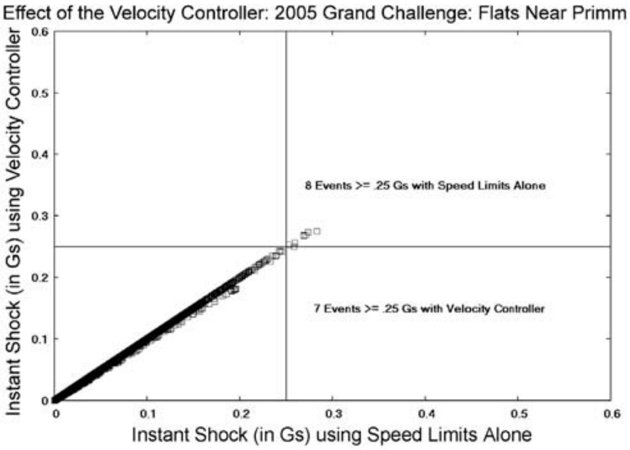

Figure 5: Each point indicates a specific shock event with both the velocity controller (vertical axis) versus speed limits alone (horizontal axis). There are very few high shock events on this easy terrain. The velocity controller does not reduce shock or speed much.

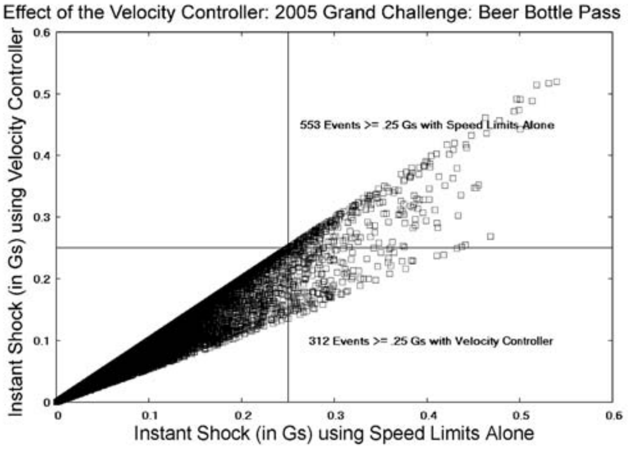

Figure 6: Each point indicates a specific shock event with both the velocity controller (vertical axis) versus speed limits alone (horizontal axis). There are many high shock events on this difficult terrain. The velocity controller reduces shock and speed a great deal.

Second, we notice that the shock experienced with the speed limits alone is, for all events, roughly the same as that experienced with the velocity controller. Thus, because shock is linear with speed, the velocity controller did not significantly reduce speed on this easy terrain. This supports the idea that the algorithm, with the learned parameters, does not cause the vehicle to drive too slowly on easy terrain. Figure 6 is the same type of plot, with the same axis ranges, as Figure 5. However, here, we plot data from the more difficult segment, miles 122-129. In this terrain, poor perceptive sensing or loss of traction – both due to excessive shock – could result in total vehicle loss. The velocity controller is very active, lowering vehicle speed. The lower speed results in reduced shock, better traction, and generally more favorable conditions for successful navigation of this challenging terrain.

4.2 Quantitative Analysis

Experimental Setup

Having verified the basic function of the algorithm, we now proceed to analyze its performance over long stretches of driving. For long stretches, it becomes intractable to compare algorithm performance to human decisions (as in Section 3.3) or relative to qualitatively assessed terrain conditions (as in Section 4.1). Therefore, we must consider other metrics. Over long stretches of driving, we use the aggregate reduction of large shocks to measure algorithm effectiveness.

However, to evaluate the reduction of large shocks, we cannot simply sum the amount of shock. Extreme shock is rare. Less than .3% of all readings (2005 Grand Challenge, Speed Limits Alone) are above our maximum acceptable shock (α) threshold. Thus, the aggregate effect of frequent, small shock could mask the algorithm's performance on rare, large shock. To avoid this problem, we take the 4th power ("L4 Metric") of a shock before adding it to the sum. This accentuates large shocks and diminishes small ones in the aggregate score.

Whenever the vehicle travels along a road – regardless of whether under human or autonomous control – it can map the road for velocity control experiments. Each raw IMU reading is logged along with the vehicle's velocity and position according to the UKF. Then, offline, each IMU reading can be filtered and divided by the vehicle's speed. This generates a sequence of position tagged road roughness values in the units Gs-felt-per-mile-per-hour. We use these sequences in this section to evaluate the velocity controller.

We calculate completion time by summing the time spent traversing each route segment (ie: from p_i to p_{i+1}) according to the average speed driven during that segment. $1/dt$ is 100Hz. Because the velocity controller may change speed instantly at each p_i, we track the velocities it outputs – for these simulations only – with a simple controller. It restricts the maximum amount of change in velocity between successive p_i's, approximating vehicle capabilities. The limits are .02 mph for increases and .09 mph for decreases. We use the controller output for completion time and shock calculations.

Time vs. Shock for Four Desert Routes

We consider the trade-off of shock versus completion time on four desert routes. See Figure 8. The horizontal axis plots completion time. The vertical axis plots shock felt. Both are for the algorithm normalized against speed limits alone. Each of the four curves represents a different desert route: the 2005 Grand Challenge route, the 2004 Grand Challenge route, and two routes outside of Phoenix, Arizona, USA that we created for testing purposes. Individual points on the curves depict the completion time versus shock trade-off for a particular set of (α, β) values. For these curves, α is fixed and β is varied. The asterisk on each curve indicates the trade-off generated by the parameters selected in Section 3.3. The fifth asterisk, at the top-left, indicates what happens with speed limits alone. We notice that, for all routes, completion time is increased between 2.5% and 5% over the baseline. However,

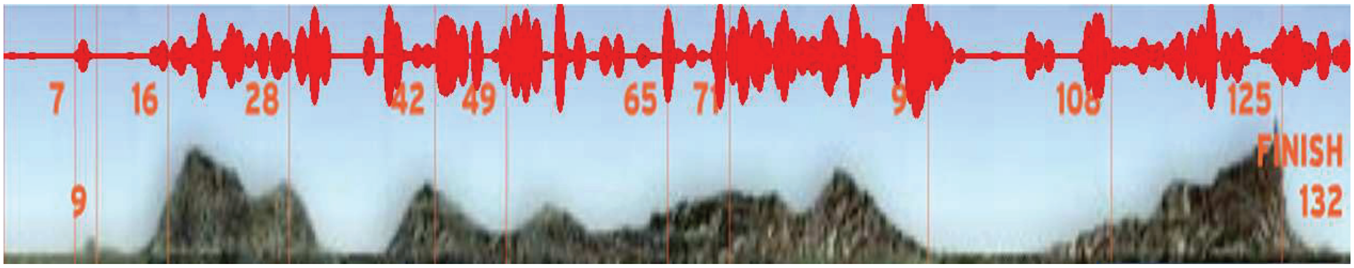

Figure 7: Here we show where the algorithm reduced speed during the 2005 Grand Challenge event. The velocity controller modified speed during 17.6% of the race. Height of the ellipses shown at the top is proportional to the maximum speed reduction for each 1/10th mile segment. Numbers represent mile markers. Simulated terrain at the bottom approximates route elevation.

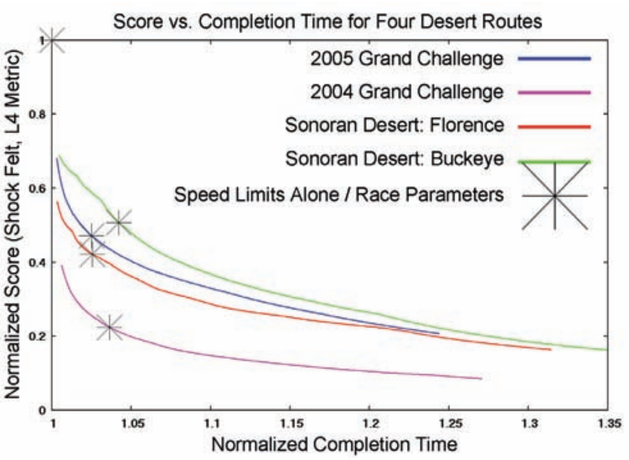

Figure 8: Here we plot the trade-off of completion time and shock on four desert routes. α is constant (.25) and β is varied along each curve. With the single set of parameters learned in Section 3.3, the algorithm reduces shock by 50% to 80% while increasing completion time by 2.5% to 5%.

shock is reduced by 50% to 80% compared to the baseline, a substantial savings.

It appears that the algorithm is most effective along the 2004 Grand Challenge route. There are two reasons for this. First, we prepared for the 2005 event along the 2004 course. Therefore, our algorithm is especially optimized for that route. However, the effect is mostly due to a different phenomenon.

The 2004 course had sections that were in extremely bad condition when this data was collected (June 2005). Such extreme off-road driving was not seen on the other three routes. Because the core functionality of our algorithm involves being cautious about clustered, dangerous terrain, it was particularly effective for the 2004 course. This implies our approach may be more beneficial as terrain difficulty increases.

Finally, we notice occasional abrupt, unsmooth drops in shock in the figure. This is the result of a parameterization that is just aggressive enough to slow the vehicle before a major rough section of road. Using the parameterization immediately prior did not slow the vehicle in time.

4.3 Results from the 2005 Grand Challenge

In Figure 7, we indicate where the algorithm reduced speed during the 2005 Grand Challenge. Numbers and vertical lines are mile markers. We depict terrain at the bottom that approximates elevation data from the route. At the top of the figure, ellipse height indicates the maximum speed reduction generated by the algorithm for that portion of the route. For the purposes of this figure, the route was divided into 1/10th mile segments. Log files taken by the vehicle during the Grand Challenge event indicate the velocity controller was controlling vehicle speed for 17.6% of the entire race – more than any other factor except speed limits.

We notice that the algorithm was particularly active in the mountains and especially inactive in plains. We also notice that miles 122-129 – which we identified in Section 4.1 as the most challenging part of the route – do not represent the greatest speed reductions the velocity controller made. This is because speed limits (given by DARPA) were already low in those regions, and the amount of speed reduction is, of course, related to the accuracy of the original speed limit as a good speed guide.

5 Discussion

Speed decisions are a crucial part of planning – even beyond their relationship to obstacle avoidance and lateral maneuverability. We have presented an online approach to speed adaptation for high-speed off-road autonomous driving. Our approach addresses shock, an important component of overall risk when driving off-road. Our method requires no rehearsal or remote imaging of the route.

The algorithm has three states. First, the vehicle drives at the speed limit until an acceptable shock threshold is exceeded. Second, the vehicle reduces speed to bring itself within the threshold. Finally, the vehicle gradually increases speed back to the speed limit. The algorithm uses the linear relationship between speed and shock to determine the amount of speed reduction needed to stay within the limit. This limit and the acceleration parameter from the third state are tuned using supervised learning to match human driving.

Experimental results allow us to draw numerous conclusions. First, our algorithm seems to be more cautious on more difficult terrain, although this is difficult to prove in the general case (Figures 5 and 6). Second, by slowing, we can substantially reduce high shock events with minimal effect on route completion time (Figure 8). Finally, the algorithm had significant influence on our vehicle's behavior during the 2005 Grand Challenge (Figures 5, 6, 7, and 8). It reduced shock by 52% and slowed the vehicle in difficult terrain, all while adding less than 10 minutes to the completion time. The algorithm also had more influence than any other factor – except speed limits – on our robot's speed. Finally, it generated the fastest completion time despite being fully automated, unlike some other competitors in the Grand Challenge [Gutierrez *et al.*, 2005; Urmson *et al.*, 2006; 2004].

Acknowledgments

David Stavens' doctoral study is supported by the David Cheriton Stanford Graduate Fellowship (SGF). The authors gratefully acknowledge the Stanford Racing Team who helped make this work possible, especially Michael Montemerlo, Hendrik Dahlkamp, and Sven Strohband.

References

[Bekker, 1956] G. Bekker. *Theory of Land Locomotion*. University of Michigan, 1956.

[Bekker, 1969] G. Bekker. *Introduction to Terrain-Vehicle Systems*. University of Michigan, 1969.

[Brooks and Iagnemma, 2005] C.A. Brooks and K. Iagnemma. Vibration-based terrain classification for planetary exploration rovers. *IEEE Transactions on Robotics*, 21(6):1185–1191, 2005.

[DARPA, 2004] DARPA. DARPA Grand Challenge Rulebook, 2004. http://www.darpa.mil/grandchallenge05/.

[Fox *et al.*, 1996] D. Fox, W. Burgard, and S. Thrun. Controlling synchro-drive robots with the dynamic window approach to collision avoidance. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 1996.

[Gillespie, 1992] T. Gillespie. *Fundamentals of Vehicle Dynamics*. Society of Automotive Engineers, 1992.

[Gutierrez *et al.*, 2005] A. Gutierrez, T. Galatali, J. P. Gonzalez, C. Urmson, and W. L. Whittaker. Preplanning for high performance autonomous traverse of desert terrain exploiting a priori knowledge to optimize speeds and to detail paths. Technical Report CMU-RI-TR-05-54, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, December 2005.

[Iagnemma *et al.*, 2004] K. Iagnemma, S. Kang, H. Shibly, and S. Dubowsky. On-line terrain parameter estimation for wheeled mobile robots with application to planetary rovers. *IEEE Transactions on Robotics and Automation*, 2004. To appear.

[Julier and Uhlmann, 1997] S. Julier and J. Uhlmann. A new extension of the Kalman filter to nonlinear systems. In *International Symposium on Aerospace/Defense Sensing, Simulate and Controls*, Orlando, FL, 1997.

[Kelly and Stentz, 1998] A. Kelly and A. Stentz. Rough terrain autonomous mobility, part 2: An active vision, predictive control approach. *Autonomous Robots*, 5:163–198, 1998.

[NOVA, 2006] NOVA. The Great Robot Race (Documentary of the DARPA Grand Challenge), 2006. http://www.pbs.org/wgbh/nova/darpa/.

[Sadhukhan *et al.*, 2004] D. Sadhukhan, C. Moore, and E. Collins. Terrain estimation using internal sensors. In *Proceedings of the 10th IASTED International Conference on Robotics and Applications (RA)*, Honolulu, Hawaii, USA, 2004.

[Shimoda *et al.*, 2005] S. Shimoda, Y. Kuroda, and K. Iagnemma. Potential field navigation of high speed unmanned ground vehicles in uneven terrain. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, Barcelona, Spain, 2005.

[Spenko *et al.*, 2006] M. Spenko, Y. Kuroda, S. Dubowsky, and K. Iagnemma. Hazard avoidance for high-speed mobile robots in rough terrain. *Journal of Field Robotics*, 23(5):311–331, 2006.

[Thrun *et al.*, 2006a] S. Thrun, M. Montemerlo, and A. Aron. Probabilistic terrain analysis for high-speed desert driving. In G. Sukhatme, S. Schaal, W. Burgard, and D. Fox, editors, *Proceedings of the Robotics Science and Systems Conference*, Philadelphia, PA, 2006.

[Thrun *et al.*, 2006b] S. Thrun, M. Montemerlo, H. Dahlkamp, D. Stavens, A. Aron, J. Diebel, P. Fong, J. Gale, M. Halpenny, G. Hoffmann, K. Lau, C. Oakley, M. Palatucci, V. Pratt, P. Stang, S. Strohband, C. Dupont, L.-E. Jendrossek, C. Koelen, C. Markey, C. Rummel, J. van Niekerk, E. Jensen, P. Alessandrini, G. Bradski, B. Davies, S. Ettinger, A. Kaehler, A. Nefian, and P. Mahoney. Stanley: The robot that won the darpa grand challenge. *Journal of Field Robotics*, 2006.

[Urmson *et al.*, 2004] C. Urmson, J. Anhalt, 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, and W.L. Whittaker. High speed navigation of unrehearsed terrain: Red team technology for grand challenge 2004. Technical Report CMU-RI-TR-04-37, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, 2004.

[Urmson *et al.*, 2006] C. Urmson, J. Anhalt, D. Bartz, M. Clark, T. Galatali, A. Gutierrez, S. Harbaugh, J. Johnston, H. Kato, P. Koon, W. Messner, N. Miller, A. Mosher, K. Peterson, C. Ragusa, D. Ray, B. Smith, J. Snider, S. Spiker, J. Struble, J. Ziglar, and W. L. Whittaker. A robust approach to high-speed navigation for unrehearsed desert terrain. *Journal of Field Robotics*, 23(8):467–508, August 2006.

[Wong, 1989] J. Wong. *Terramechanics and Off-Road Vehicles*. Elsevier, 1989.