2017 NDIA GROUND VEHICLE SYSTEMS ENGINEERING AND TECHNOLOGY SYMPOSIUM
Modeling & Simulation, Testing and Validation (MSTV) Technical Session
August 8-10, 2017 – Novi, Michigan

PERFORMANCE ANALYSIS OF CONSTANT SPEED LOCAL OBSTACLE AVOIDANCE CONTROLLER USING AN MPC ALGORITHM ON GRANULAR TERRAIN

Nicholas Haraus
Dept. of Mechanical Engineering
Marquette University
Milwaukee, WI

Radu Serban
Dept. of Mechanical Engineering
University of Wisconsin - Madison
Madison, WI

Jonathan Fleischmann
Dept. of Mechanical Engineering
Marquette University
Milwaukee, WI

ABSTRACT

A Model Predictive Control (MPC) LIDAR-based constant speed local obstacle avoidance algorithm has been implemented on rigid terrain and granular terrain in Chrono to examine the robustness of this control method. Provided LIDAR data as well as a target location, a vehicle can route itself around obstacles as it encounters them and arrive at an end goal via an optimal route. Using Chrono, a multibody physics API, this controller has been tested on a complex multibody physics HMMWV model representing the plant in this study. A penalty-based DEM approach is used to model contacts on both rigid ground and granular terrain. We draw conclusions regarding the MPC algorithm performance based on its ability to navigate the Chrono HMMWV on rigid and granular terrain.

1 INTRODUCTION

Obstacle avoidance is a crucial capability for Autonomous Ground Vehicles (AGVs) of the future. This refers to a ground vehicle's ability to sense its surrounding environment, develop an optimal path around the obstacles in the environment, generate optimal control commands to satisfy that path, and physically navigate around the obstacles safely and to a desired endpoint. Safety is defined as avoiding collisions as well as enforcing limitations on excessive sideslip or tire lift-off. An ideal control algorithm is one that is capable of pushing a vehicle to its performance limits by using knowledge of its dynamic capabilities and surrounding environmental conditions, while still enforcing strict safety requirements. Although previous work has demonstrated use of Model Predictive Control (MPC) algorithms for obstacle avoidance on wheeled vehicles, more work is required to test the fidelity of these algorithms and determine where improvements are needed. One area in which MPC algo-

rithms have yet to be tested is their ability to control a wheeled vehicle on deformable terrain. Up to this point, the terrain has been assumed to be rigid and flat. Performance of MPC algorithms on deformable terrain raise additional questions. Do the current most commonly used vehicle models perform successfully within an MPC algorithm on deformable terrain? The present work evaluates, through numerical simulation, the robustness and validity of an MPC algorithm with different vehicle models in an environment more similar to what an off-road military vehicle would experience in combat.

The goal of this paper is to present the results of this study to understand how model fidelity of the controller model affects overall performance of the obstacle avoidance controller. Simulating a vehicle on granular terrain in such scenarios, introduces additional challenges related to the sheer number of necessary particles required to properly model the desired terrain patch. This challenge has been addressed by employing a simulation framework for granular terrain described later in this paper.

The present study has the following three objectives:

1. Study the performance of the MPC Controller on granular terrain as compared to that on rigid terrain.

2. Analyze the impact of fidelity of the internal controller model on speed and performance of the obstacle avoidance controller.

3. Showcase the potential of controller testing in a high fidelity virtual test environment with Chrono to assist with initial control algorithm development before physical implementation for vehicular applications.

The remainder of this paper is organized as follows. In Section 2 we provide overviews of the MPC-based local obstacle avoidance algorithm and the Chrono multiphysics package used as the test environment. In Section 3 we describe the Chrono::Vehicle HMMWV vehicle model used in this study as the plant model to be controlled and our proposed method for simulating a vehicle driving on deformable terrain over large distances. Then, the specific MPC LIDAR-based local obstacle avoidance controller used for this study is presented. We introduce the different simplified, lower-fidelity analytical vehicle models used in the MPC algorithm to predict the Chrono vehicle behavior. We also provide descriptions of the test cases considered for this study and a summary of the metrics used to compare performance of the various tested combinations. In Section 4 we present the results of the tests outlined in the previous section and provide comparisons when the internal controller vehicle model is varied and when the terrain is changed from rigid to granular. Section 5 wraps up the paper and presents potential future work based on the results of this study.

2 BACKGROUND

2.1 MPC Based Local Obstacle Avoidance

The concept of MPC is to use an internal model of the system one desires to control to predict and optimize future system behavior from the current system state and inputs [1]. The system behavior is predicted over some defined finite time horizon and the optimal control sequence over the prediction horizon is output. The control sequence is executed for an execution time smaller than the prediction horizon, and the whole process is repeated. The repetition of this process over time creates a feedback loop which continually controls the system, pushing it towards an optimal path.

For this study, the system to be controlled is an AGV. Consider an AGV located in a level environment without roads or any other structures to guide its motion and assume the AGV has a known global target position. Between the target

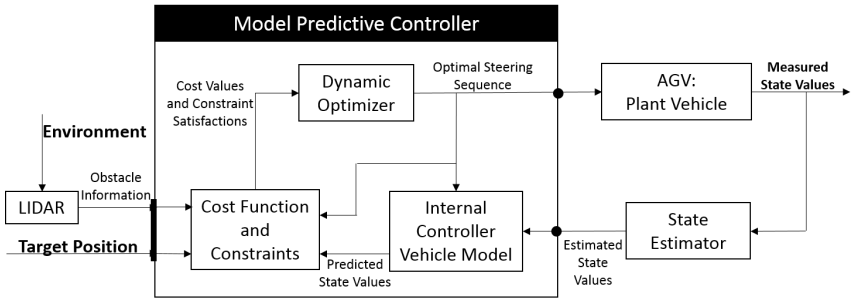
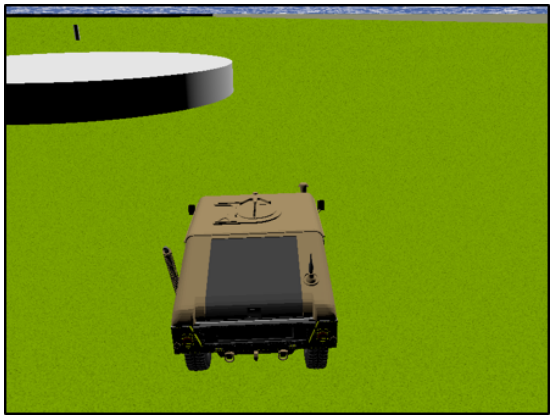


Figure 1: Schematic of MPC LIDAR-Based Constant Speed Local Obstacle Avoidance Controller

position and the current vehicle position there may or may not be obstacles of unknown size. Using the MPC formulation outlined in [2], the vehicle can navigate from the current position to the provided target position while avoiding obstacles as they are encountered. Obstacle information is assumed to be unknown a priori and only obtained through a planar LIDAR sensor. The MPC schematic is presented in Fig. 1.

The planar LIDAR sensor, mounted at the front center location of the vehicle, returns the closest obstacle boundary in all radial directions of the sensor at an angular resolution ϵ. The sensor has a maximum range past which it cannot sense any obstacles. Therefore, if the closest obstacle boundary is greater than the LIDAR radius R_{LIDAR}, then the sensor returns R_{LIDAR}. The LIDAR sensor range is [0°, 180°], with 90° being the vehicle heading direction. Since the AGV considered here is driving along level ground, whether granular or rigid, a planar LIDAR sensor is sufficient. The sensor is assumed to have no delay and zero noise and can therefore instantaneously generate a safe area polygon assembled from the returned points from the LIDAR. An overhead view of the AGV encountering an obstacle and the generated LIDAR safe area polygon are presented in Fig. 2.

For the purpose of these simulations, the only outputs of the MPC algorithm are the steering signals. As shown in Fig. 1, the MPC algorithm is made up of the internal controller vehicle model,



(a) 3D Visualization of LIDAR Encountering Obstacle

(b) LIDAR Sensed Safe Area

Figure 2: Sample obstacle field and LIDAR output

the cost function and constraints, and the dynamic optimizer. The internal controller vehicle model predicts the future states of the AGV for a given steering sequence. Herein, the internal controller vehicle model is varied from test to test between a 2-DOF vehicle model and a 14-DOF vehicle model, as detailed in further sections. Cost functions and constraints are used to formulate the optimal control problem using the equations from the vehicle model, and the resulting optimal control problem is solved using dynamic optimization.

Since the ability of finding and executing an optimal solution, rather than solution speed, is the primary focus here, an exhaustive search is used to find the optimal solution to the problem at hand. With this method, the steering sequence is discretized and a finite set of path possibilities are tested and weighed by a cost function.

2.2 Chrono Multibody Physics Package

The physics modeling and simulation capabilities are provided by the multiphysics open-source package Chrono [3]. The core functionality of Chrono provides support for the modeling, simulation, and visualization of rigid multibody systems, with additional capabilities offered through optional modules. These modules provide support for additional classes of problems (e.g., finite element analysis and fluid-solid interaction), for modeling and simulation of specialized systems (such as ground vehicles and granular dynamics problems), or provide specialized parallel computing algorithms (multicore, GPU, and distributed) for large-scale simulations.

2.2.1 Vehicle Modeling

Built as a Chrono extension module, Chrono::Vehicle [4] is a C++ middleware library focused on the modeling, simulation, and visualization of ground vehicles. Chrono::Vehicle provides a collection of templates for various topologies of both wheeled and tracked vehicle subsystems, as well as support for modeling of rigid, flexible, and granular terrain, support for closed-loop and interactive driver models, and run-time and off-line visualization of simulation results.

Modeling of vehicle systems is done in a modular fashion, with a vehicle defined as an assembly of instances of various subsystems (suspension, steering, driveline, etc.). Flexibility in modeling is provided by adopting a template-based design. In Chrono::Vehicle, templates are parameterized models that define a particular implementation of a vehicle subsystem. As such, a template defines the basic modeling elements (bodies, joints, force elements), imposes the subsystem topology, prescribes the design parameters, and implements the common functionality for a given type of subsystem (e.g., suspension) particularized to a specific template (e.g., double wishbone). Finally, an instantiation of such a template is obtained by specifying the template parameters (hardpoints, joint directions, inertial properties, contact material properties, etc.) for a concrete vehicle (e.g., the HMMWV front suspension).

For wheeled vehicle systems, templates are provided for the following subsystems: *suspension* (double wishbone, reduced double wishbone using distance constraints, multi-link, solid-axle, McPhearson strut, semi-trailing arm); *steering* (Pitman arm, rack-and-pinion); *driveline* (2WD and 4WD shaft-based using specialized Chrono modeling elements, simplified kinematic driveline); *wheel* (simply a carrier for additional mass and inertia appended to the suspension's spindle body); *brake* (simple model using a constant torque modulated by the driver braking input).

In addition, Chrono::Vehicle offers a variety of tire models and associated templates, ranging from rigid tires, to empirical and semi-empirical models (such as Pacejka and Fiala), to fully deformable tires modeled with finite elements (using either an

Absolute Nodal Coordinate Formulation or a co-rotational formulation). Driver inputs (steering, throttle, and braking) are provided from a *driver* subsystem with available options in ChronoVehicle including both open-loop (interactive or data-driven) and closed-loop (e.g., path-following based on PID controllers).

2.2.2 Granular Mechanics

In this work, the focus is on modeling and simulating the terrain and the tire-terrain interaction using high-fidelity, fully-resolved granular dynamics simulations, employing the Discrete element Method (DEM). Meaningful mobility simulations require large enough terrain patches and small enough particle dimensions that result in DEM problems involving frictional contact with millions of degrees of freedom.

Unlike continuum-based deformable terrain modeling approaches, DEM treats all component particles separately, as distinct entities, by maintaining and advancing in time their states while taking into account pair-wise interaction forces due to frictional contact. Broadly speaking, computational methods for DEM at this scale can be categorized into two classes: penalty-based (also known as a compliant-body approach; denoted here by DEM-P) and complementarity-based (also known as a rigid-body approach; denoted here by DEM-C). While differing in the underlying formulation employed for modeling and generating the normal and tangential forces at the contact interface, and thus leading to different mathematical models and different problem sizes, both methods rely crucially on efficient methods for proximity calculation. This common algorithmic step provides a complete geometric characterization of the interaction between neighboring bodies, taking into account the current system state and specification of the contact shapes associated with all interacting bodies.

Penalty methods begin with a relaxation of the rigid body assumption [5]. A regularization approach, DEM-P assumes local body deformation at the contact point. Employing the finite element method to characterize this deformation would incur a stiff computational cost. Instead, an approximation is employed, using information generated during the collision detection stage of the solution, and the local body deformation is related to the depth of inter-penetration between two otherwise rigid contact shapes. In order to apply results from Hertz contact theory, valid for sphere-sphere interactions, the contact shapes are further approximated by their local radius of curvature at the contact point [6]. This approach yields a general methodology for computing the normal and tangential forces at the contact point. For granular dynamics via DEM-P, the equations of motion need not be changed. Indeed, normal and tangential contact forces are treated as any external forces and directly factored in the momentum balance. For details on the specific DEM-P implementation in Chrono, see [7]. In this work, we employed the DEM-P capabilities in Chrono, leveraging the multi-core, OpenMP-based parallelization features offered by the Chrono::Parallel module.

3 TECHNICAL APPROACH AND METHODOLOGY

3.1 Full-Vehicle Multibody Model

The model to be controlled (the *plant*) is a full-vehicle Chrono::Vehicle model of an HMMWV, which includes multi-body subsystems for the suspensions, steering, driveline, and powertrain, and is available in the Chrono package.

This vehicle model (see Fig. 3) has a curb weight of $2,550$ kg. It includes independent front and rear double wishbone suspensions and a Pitman arm steering mechanism. The shock absorber and coil spring, mounted between the lower control arm

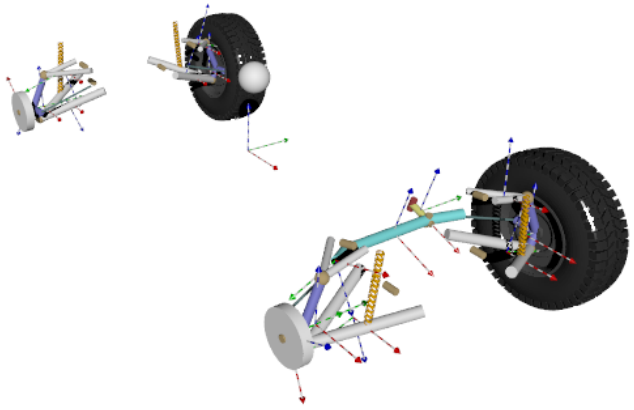


Figure 3: Full-vehicle HMMWV multibody model.

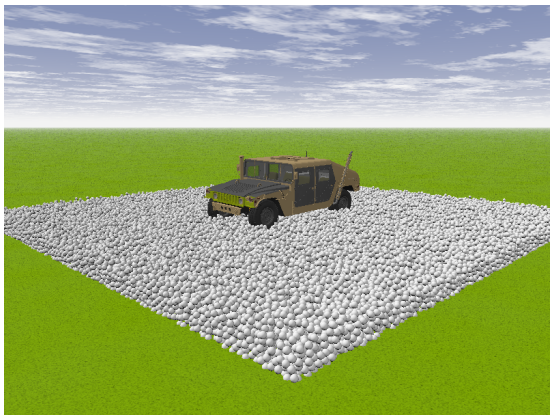


Figure 4: Relocating Granular Patch that follows the vehicle

and the chassis, are modeled with Chrono nonlinear force elements and include the effects of bump stops.

The AWD driveline is modeled using Chrono shaft elements and includes three power splitting elements (a central differential and front/rear differentials), as well as conical gears connected through 1-dimensional shaft elements which carry rotational inertia. The powertrain is also modeled using 1-dimensional shaft elements and includes models for a thermal engine specified through speed-torque maps for power and engine losses, a torque converter specified via maps for the capacity factor and the torque ratio, and an automatic transmission gearbox with three forward gears and a single reverse gear. The connection between powertrain and driveline is a force-displacement interface at the driveshaft (with torque applied from the powertrain and angular velocity provided by the driveline).

For mobility studies on deformable terrain, provided that the tire inflation pressure is comparable or larger than the average ground pressure, according to the postulate by Wong [8] the tire can be considered in a so-called *rigid regime*. As such, tire deformation can be ignored and the vehicle's tires modeled using rigid contact shapes. The HMMWV tire model used here is thus represented by a cylinder with a radius of 0.47 m (18.5 in) and a width of 0.254 m (10 in).

3.2 Relocating Granular Patch

Due to the large-scale nature of the simulations in this study, generating granular particles distributed across the entire obstacle field is both computationally exhausting and unreasonable. Instead, we adopt a moving-patch approach. Consider an AGV on a large terrain patch of 100 by 100 meters. Since we are primarily interested in the vehicle behavior and performance and not explicitly in the terrain behavior, we assume only a small patch of granular terrain underneath and around the vehicle. This idea promoted the development of a relocating granular patch which enables simulations of a vehicle traversing granular terrain over a large area, without the need to generate particles everywhere throughout that area.

This mechanism allows a user to generate granular particles within a specified distance of the vehicle's CoG. Figure 4 presents the patch of granular terrain underneath the Chrono HMMWV that moves with the vehicel. The particles are contained within four rigid walls to prevent them from escaping the desired terrain area. As the vehi-

cle moves across the terrain, this patch maintains granular material underneath the vehicle by consistently relocating particles that are too far away from it. When the vehicle location comes within a certain predefined distance of any of the walls, a band of particles from the opposite end of the granular patch are relocated past the closest wall and all of the walls are shifted in that direction. Each of these relocation steps keeps the vehicle on top of granular terrain at all times. Depending on the vehicle, this granular patch can be resized for both computational efficiency and to guarantee the vehicle maintains a reasonable distance from the walls to prevent any boundary effects. This newly developed method of simulating granular terrain for AGV's is omnidirectional in the ground plane, allowing the vehicle to move in any direction while the terrain patch relocates and responds to that movement, thus enabling simulations over arbitrarily large areas.

For our simulations, the granular patch was maintained at a size of roughly 6.6 by 6.6 meters. This number was assigned as two times the largest dimension of the vehicle. The dimensions of the patch are not constant, however, since the patch expands in the direction of relocation by two times the largest particle radius every time the advancing wall is shifted. This is done in order to avoid particle overlap, since the relocated particles are moved to a position that is shifted one largest particle radius ahead of the particles adjacent to the advancing wall. After relocation, the walls of the granular patch are given a recovery velocity such that the granular patch regains its original dimensions in 0.1 seconds. This recovery time should be small relative to the duration of the simulations, and in general it will also depend on the velocity of the vehicle, since the granular patch should completely recover its dimensions between subsequent relocations.

Within the granular patch, the granular terrain was modeled by $55,931$ uniform spherical particles, with a micro-scale inter-particle sliding friction coefficient of $\mu = 0.8$, and particle diameter of 0.1 m. Previous studies [9] have shown that a randomly packed assembly of as few as $3,000$ - $30,000$ uniform spheres will exhibit macro-scale bulk granular material yield behavior (due to inter-particle sliding) that closely matches the Lade-Duncan yield surface, which is a well-established yield criterion in the field of geomechanics, where the macro-scale friction angle ϕ for the bulk granular material can be determined as a function of the inter-particle friction coefficient μ.

Referring to Fig. 10 of [9], a randomly packed assembly of uniform spheres with an inter-particle friction coefficient of $\mu = 0.8$ will exhibit macro-scale yield behavior corresponding to a bulk granular material with a macro-scale friction angle between roughly $35^\circ \leq \phi \leq 40^\circ$ if particle rotation is allowed (6 DOF particles), or with a macro-scale friction angle between roughly $65^\circ \leq \phi \leq 70^\circ$ if particle rotation is prohibited (3 DOF particles). Since the granular patch used in our simulations contains more than $50,000$ uniform spheres, we can reliably conclude that it is accurately modeling the yield behavior of a true granular material on the macro-scale: either with a macro-scale friction angle of $35^\circ - 40^\circ$ if particle rotation is allowed, which is typical of a wide range of dry natural and crushed sands [10]; or with a macro-scale friction angle of $65^\circ - 70^\circ$ if particle rotation is prohibited, which is typical of crushed or fragmented rock, such as railway track ballast [11].

We note that in the case of the HMMWV, the vehicle tire width is equal to only approximately 2.5 particle diameters in the granular patch. This is less than ideal. However, a more suited value of 10 particle diameters per tire width would increase the number of particles in the granular patch to more than 3 million, which would result in prohibitively slow simulations, particularly for events corresponding to physical time durations on the order of 10 seconds or more.

3.3 MPC LIDAR-Based Local Obstacle Avoidance

The MPC controller as formulated in [2] is used for this study. The cost function and constraints need to be specified to avoid collisions with obstacles and guarantee vehicle dynamical safety. The optimal control problem solved at each MPC time step consists of the following set of equations:

$$J = s_T + wd \tag{1}$$

$$\dot{\xi} = v\left[\xi\left(t\right), \zeta\left(t\right)\right] \tag{2}$$

$$\xi\left(0\right) = \xi_0 \tag{3}$$

$$\tilde{S}\left[x\left(t\right), y\left(t\right)\right] \leq 0 \tag{4}$$

$$\left|\delta_f\left(t\right)\right| \leq \tilde{\delta}_{f,max}\left(U_0\right) \tag{5}$$

$$\left|\varsigma_f\left(t\right)\right| \leq \varsigma_{f,max} \tag{6}$$

$$t \in \left[0, T_P\right] . \tag{7}$$

Equation (1) defines the cost function for this optimal control problem. This equation is a soft requirement which defines how the separate path possibilities are weighed against each other and how to determine which path is actually optimal. The cost function is comprised of two terms defined as:

$$s_T = \sqrt{\left[x_G - x\left(T_P\right)\right]^2 + \left[y_G - y\left(T_P\right)\right]^2} \tag{8}$$

$$d = \int_0^{T_G} \left|\varsigma\left(t\right)\right| dt\,. \tag{9}$$

Here, s_T seeks to minimize the distance between the prediction end-location and the target position. A prediction path that guides the vehicle towards the closest location to the target will have the smaller s_T term. The second term, d, aims to minimize the change in steering angle so that smoother, straighter paths are preferred over windy paths. A weighting factor w is used to scale the influence of d in the total cost.

Equations (2-7) are the constraints for this optimal control problem and represent hard requirements for vehicle safety and collision avoidance. Any paths that violate these constraints are not considered safe paths and are eliminated as potential options in this prediction window. Equation (2) defines a set of differential equations that describe the internal controller vehicle model, with the initial conditions of Eq. (3).

Equation (4) defines a safe area polygon, constructed from LIDAR data and including an additional safety buffer to account for vehicle size and to prevent collisions of the vehicle corners. All points along paths found from Eqs. (2)-(3) must fall inside this safe area polygon.

Equation (5) imposes a maximum limit on the steering angle, based on vehicle speed. This value is obtained from a lookup table which can be generated either experimentally or from simulation. Equation (6) imposes a maximum limit on steering rate. Equation (7) defines the time limits over which this optimal control problem is solved.

As noted above, Eq. (2) refers to an analytical vehicle model expressed as a set of differential equations which allow the controller to predict vehicle performance within the prediction horizon. The accuracy of the internal vehicle controller model should directly influence the driven vehicle's controlled performance. If the internal vehicle controller model poorly predicts a vehicle's response to inputs from the driver or the environment, then these deficiencies will be witnessed when attempting to control the driven vehicle and result in incorrect trajectories that may lead to collisions. On the other hand, a highly complex internal vehicle controller model may provide accurate trajectory and vehicle response predictions, but with an unacceptable time required to solve the optimal control problem. An ideal controller should not only accurately predict vehicle responses, but also do it quickly enough to insure vehicle safety. In this study we consider two separate internal con-

troller vehicle models to approximate the dynamics of the driven Chrono HMMWV vehicle (described in Section 3.1). The first controller uses a simple 2-DOF vehicle model to predict the trajectory for different series of steering sequences and is presented in detail in Section 3.4.1. The second controller, described in Section 3.4.2, uses a more complex 14-DOF vehicle model. These two models are compared in their ability to successfully navigate the driven HMMWV vehicle through two obstacle fields, on rigid flat terrain and then granular terrain.

As mentioned previously, exhaustive search space [2] is used to solve the optimal control problem at each time step. The steering space is discretized into five possible steering angles, while the prediction horizon is split into four intervals. This results in 625 different steering sequence possibilities and therefore 625 different path possibilities that are weighed by the controller to determine the optimal steering sequence and resulting forward path. A sample visualization of the different predicted trajectories from the 2-DOF model is presented in Fig. 5 where, for visualization purposes, only three steering angles over three intervals were considered. A point in polygon algorithm is used to determine which trajectories remain inside of the safe area polygon visualized in Fig. 2 and therefore should be compared to the other possibilities using Eq. (1).

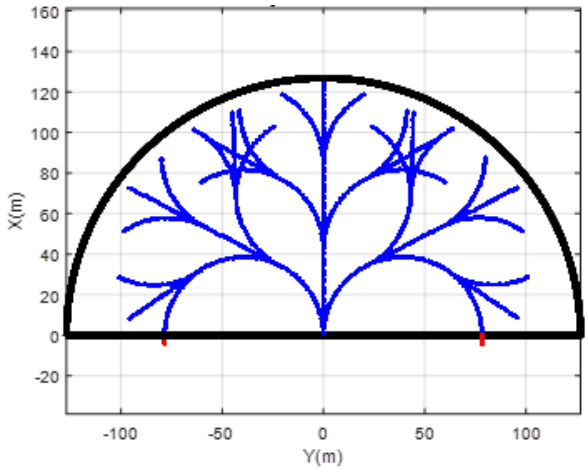


Figure 5: Potential Paths predicted by 2-DOF Internal Vehicle Controller Model

3.4 Internal Controller Vehicle Models

The vehicle models embedded in the MPC are simplifications of the full, multi-body based, Chrono wheeled vehicle model which is being controlled. We consider two such models, providing different levels of fidelity, as described below.

3.4.1 2-DOF Vehicle Model

The standard vehicle model used in recently developed MPC obstacle avoidance algorithms such as [2] is the 2-DOF yaw plane vehicle model. These models normally either assume constant cornering stiffness or the nonlinear Pacejka Magic Formula Tire Model [12] to predict the ground tire interaction forces. For this study, the Pacejka Magic Formula is used to predict tire forces in the vehicle models.

A visual representation of the 2-DOF yaw plane vehicle model can be found in Fig. 6. The 2-DOF model is described by the following first-order ordinary differential equations:

$$\dot{V} = \left(F_{y,f} + F_{y,r}\right)/M - U_0 r \tag{10}$$

$$\dot{r} = \left(F_{y,f} - F_{y,r}\right)/I_{zz} \tag{11}$$

$$\dot{\psi} = r \tag{12}$$

$$\dot{x} = U_0 \cos\psi - (V + L_f r)\sin\psi \tag{13}$$

$$\dot{y} = U_0 \sin\psi + (V + L_f r)\cos\psi\,, \tag{14}$$

where $F_{y,f}$ and $F_{y,r}$ are the lateral tire forces at the front and rear axles, respectively. U_0 and V are the longitudinal speed and lateral speed of the vehicle in the vehicle's coordinate frame. ψ is the yaw angle and r is the yaw rate. (x, y) represent the front center location of the vehicle expressed in global coordinates. M is vehicle mass, I_{zz} is the

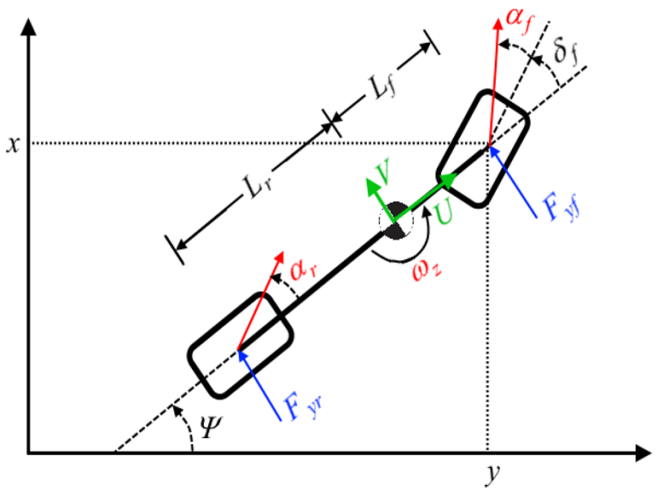


Figure 6: 2-DOF Vehicle Model

moment of inertia of the vehicle, L_f is the distance from the front axle to the vehicle CoG, and L_r is the distance from the rear axle to the vehicle CoG. For this study, the model is constrained to a constant longitudinal speed. Then the planar 3-DOF body with one constraint results in the 2-DOF model used for this study.

3.4.2 14-DOF Vehicle Model

A 14-DOF model is often used in studies such as these to test the obstacle avoidance controller with a higher fidelity vehicle model [2, 13]. A benefit of using the 14-DOF in the controller is the model's ability to predict tire liftoff and account for dynamic effects from suspension systems. For this paper, it is appropriate to also compare performance of the local obstacle avoidance controller running an internal 14-DOF on rigid terrain versus granular terrain.

The 14-DOF vehicle model consists of one sprung mass connected above four unsprung masses [14]. The sprung mass is allowed to roll, pitch, and yaw while also displacing laterally, vertically, and longitudinally. This sprung mass contributes six DOF to the model. Each of the four wheels are allowed to bounce vertically and rotate about the wheel horizontal axis. The front two wheels are also free to steer. Each wheel then contributes two DOF to the fourteen DOF model. The model is constrained at a constant longitudinal speed of 8.1 m/s. The equations used for this model, as well as their derivation can be found in [14].

3.5 Evaluation Metrics

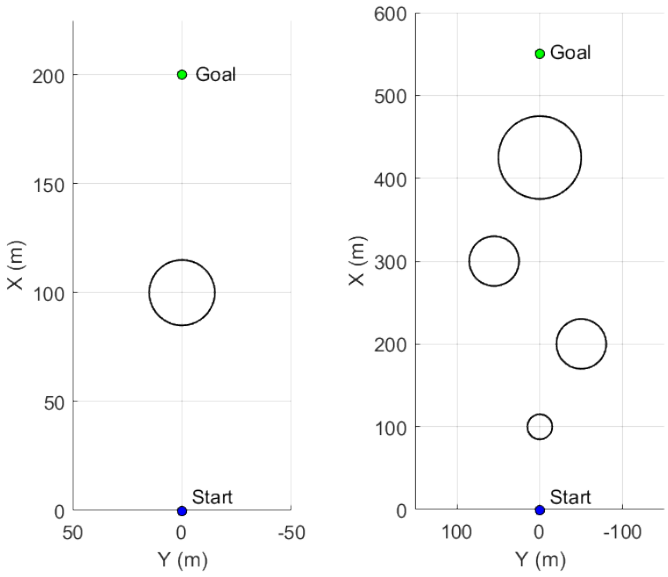
Five evaluation metrics will be used to compare one test performance to the other. First, all test runs will be compared based on the time to reach the target, T_{target} to determine which controller leads the vehicle to the target point quickest. Second, the closest distance the vehicle reaches to any obstacle, d_{min}, will be measured. Third, the control effort will be calculated and compared between test cases. The better test case will have a lower controller effort value. Finally, the maximum and average lateral accelerations will be calculated and compared. The accelerations are calculated at the driver's position in the chassis. Using the DEM-P method of simulation results in noisy acceleration data. To address this, acceleration data is filtered to remove noise. These five evaluation metrics provide a consistent methodology for comparing one test case with another, regardless of the terrain type and underlying analytical controller model.

3.6 Numerical Simulation Setup

Simulations on both rigid and granular terrain were compared to understand how the controller performs on non-ideal surfaces. The two internal controller vehicle models studied in these tests were the previously described 2-DOF and 14-DOF vehicle models. All other controller parameters were kept unchanged over all tests. Referring to Table 1, the effect of model fidelity on controller performance was gauged by comparing test 1 with

Test Number	Terrain Type	Controller Vehicle Model
1	Rigid	2-DOF
2	Rigid	14-DOF
3	Granular	2-DOF
4	Granular	14-DOF

Table 1: Individual Simulation Test Information



(a) Field 1 (b) Field 2

Figure 7: Obstacle fields.

2 and test 3 with 4, for rigid flat and granular terrain, respectively. Similarly, comparing test 1 with 3 and test 2 and 4 allowed evaluating the performance of each internal vehicle model on rigid and granular terrain.

Each of the above four tests consisted of runs on two fields with different obstacle distributions. The first one, depicted in Fig. 7a, contains a single large circular obstacle located along the initial heading of the vehicle, with the target location at a large distance behind the obstacle. The second obstacle field (see Fig. 7b) consists of four circular obstacles of varying sizes placed in the vehicle's initial heading direction, and provides a more realistic obstacle-avoidance scenario. Table 2 summarizes the obstacle locations and dimensions, as well as the target location for the above two scenarios.

	x (m)	y (m)	Radius (m)
Field 1			
Target Location	200.0	0.0	-
Obstacle 1	100.0	0.0	15.0
Field 2			
Target Location	550.0	0.0	-
Obstacle 1	100.0	0.0	15.0
Obstacle 2	200.0	-50.0	30.0
Obstacle 3	300.0	55.0	30.0
Obstacle 4	425.0	0.0	50.0

Table 2: Obstacle Field Parameters

The following vehicle parameters are maintained throughout all of the executed tests. A PID speed controller is used to maintain a near constant speed of 8.1 m/s longitudinally for the simulated plant Chrono wheeled vehicle. This constant speed is also enforced in the analytical models internal to the MPC. The LIDAR sensor has a maximum range $R_{LIDAR} = 129.6$ m and is sampled instantaneously at increments of 2.5°. The vehicle is limited to a maximum steering angle of 10°, with a maximum steering rate of 70°/s.

4 SIMULATION RESULTS

This section summarizes the results of the four tests listed in Table 1.

The trajectories for the four performed tests are presented in Figs. 8a and 9a, for Obstacle Fields 1 and 2, respectively. The associated steering commands generated by the controller are presented in Figs. 8b and 9b. Just from looking at the tra-

Test Number	1	2	3	4
Controller Model	2-DOF	14-DOF	2-DOF	14-DOF
Terrain	Rigid	Rigid	Granular	Granular
Time to Target (s)	26.67	26.15	28.32	28.03
Minimum Obstacle Distance (m)	0.897	5.462	3.491	4.721
Controller Effort	0.0340	0.0340	0.0340	0.0306
Max. Lateral Acceleration (m/s^2)	2.78	1.57	2.47	2.33
Avg. Lateral Acceleration (m/s^2)	0.54	0.51	0.55	0.46

Table 3: Test Evaluation Metrics Summary on Obstacle Field 1

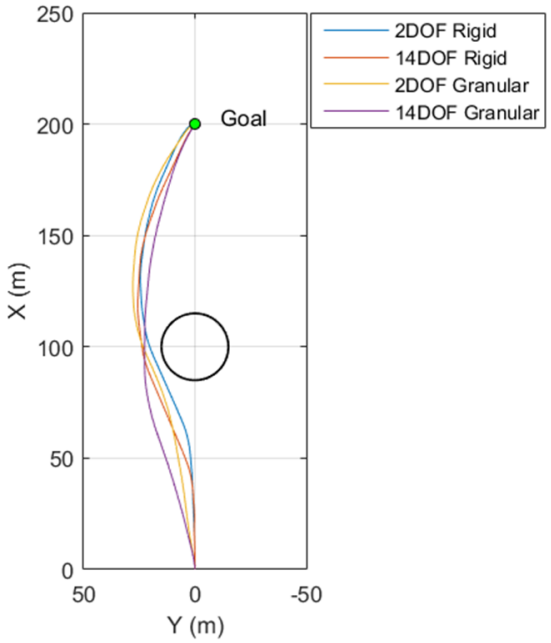
Test Number	1	2	3	4
Controller Model	2-DOF	14-DOF	2-DOF	14-DOF
Terrain	Rigid	Rigid	Granular	Granular
Time to Target (s)	73.85	71.55	76.64	74.70
Minimum Obstacle Distance (m)	0.331	2.599	1.083	1.152
Controller Effort	0.0510	0.0680	0.0714	0.0612
Max. Lateral Acceleration (m/s^2)	2.92	2.51	2.55	2.45
Avg. Lateral Acceleration (m/s^2)	0.41	0.43	0.53	0.58

Table 4: Test Evaluation Metrics Summary on Obstacle Field 2.

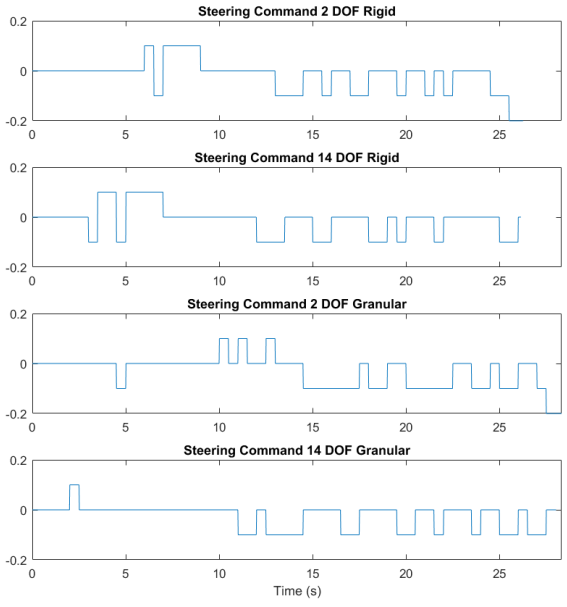
jectories of the vehicles, several observations can be made. The controlled vehicle in each test successfully maneuvers around the single obstacle in Obstacle Field 1. This confirms the findings in [2] that, even though it lacks the fidelity of the 14-DOF model, the 2-DOF yaw plane model can still successfully navigate a vehicle through an obstacle field while moving at a non-extreme speed. The evaluation metrics for each test, as defined in Section 3.5, are tabulated for each obstacle field in Tables 3 and 4.

(Radu) Got here...

Comparing the results of Tests 1 and 2, the controllers that use the 14-DOF internal vehicle model to predict vehicle states guides the Chrono vehicle to the goal marginally quicker than the controllers with the internal 2-DOF vehicle model. In all the cases, the controllers with the 2-DOF internal vehicle model guided the vehicle closer to the obstacles than the controllers with the 14-DOF models. Comparing the controller effort between Tests 1 and 2, on Obstacle Field 1 the controller efforts are equal. The 2-DOF controller inputs more command changes than the 14-DOF controller. On Obstacle Field 2, the 14-DOF controller of Test 2 has a higher controller effort value than the 2-DOF controller by 0.0170. This may be due to the controller's decision to move left around the last obstacle instead of continuing straight around the right side of the obstacle as the 2-DOF controller commands. Finally, the 14-DOF controller results in lower maximum and lateral accelerations. Overall, on rigid terrain the 14-DOF controller performs marginally better than the 2-DOF controller. However considering the largely increased runtimes of the 14-DOF controller, the 2-DOF is suitable for use in an MPC local obstacle avoidance controller for a vehicle on rigid ground terrain traveling at non-extreme speeds. The perfor-

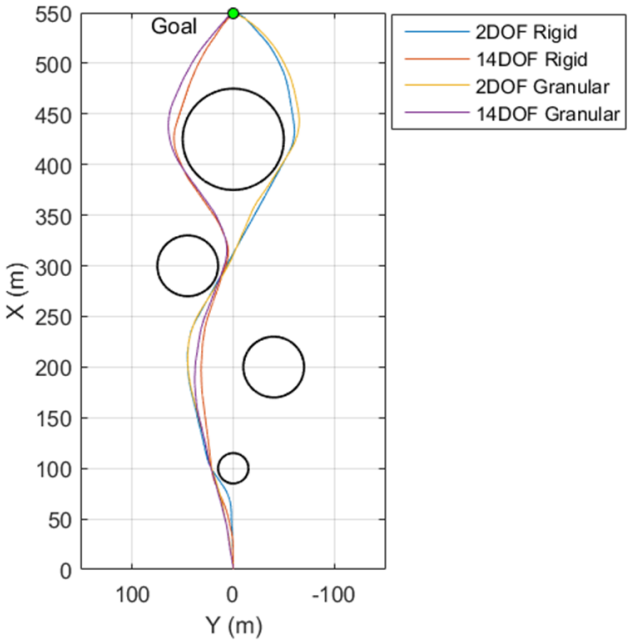


(a) Vehicle trajectories.

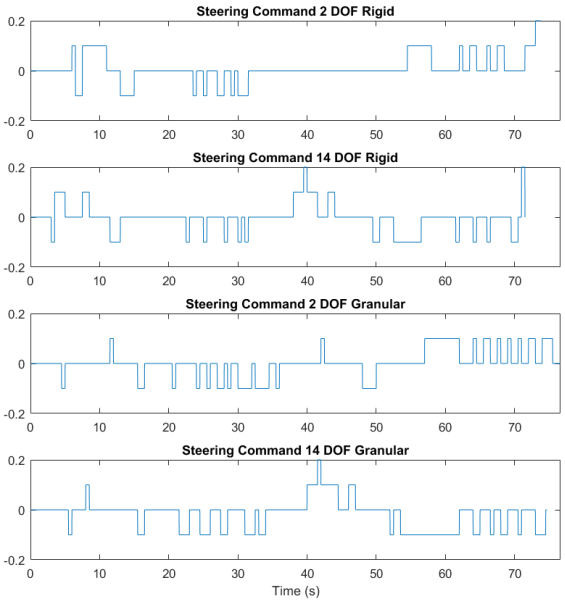


(b) Steering commands.

Figure 8: Test results on obstacle field 1.



(a) Vehicle trajectories.



(b) Steering commands.

Figure 9: Test results on obstacle field 2.

mance drop is slight switching to the 2-DOF vehicle model, but it can still safely navigate the vehicle around an obstacle field to an end goal. These results support the findings and conclusions of [2] on a different AGV.

The same set of comparisons are made between Tests 3 and 4 on granular terrain. The 14-DOF controller navigates the Chrono vehicle to the target slightly quicker than the 2-DOF controller. The 2-DOF controller guides the vehicle closer to obstacles than the 14-DOF controllers. The 14-DOF controller has a lower controller effort value than the 2-DOF controller. On Obstacle Field 1 the 2-DOF has a higher maximum lateral acceleration and average lateral acceleration. On Obstacle Field 2 the 2-DOF controller results in a lower maximum lateral acceleration and average lateral acceleration. Overall, the 14-DOF controller performs marginally better than the 2-DOF controller on granular terrain. The 2DOF and 14DOF internal vehicle models were derived using rigid ground assumptions, yet they are still capable of successfully navigating the Chrono vehicle through an obstacle field safely on this granular terrain. Though vehicle performance marginally drops when using the 2-DOF controller, realistically the software benefits of using the 2-DOF controller outweigh the slight performance drop seen by this study at non-extreme vehicle speeds.

Comparing Tests 2 and 4, the controller used is the 14-DOF controller so we can analyze performance differences between navigating on rigid ground versus granular terrain. In all cases the time to target is lower on rigid ground tests than on granular tests. In Test 4, the vehicle navigates much closer to any obstacle than in Test 2 on rigid ground. On granular terrain, the controller effort is lower than on rigid ground. Comparing the lateral acceleration metrics of Tests 2 and 4 yield no discernable conclusions since neither is significantly nor consistently larger than the other. Comparing Tests 1 and 3 yields similar conclusions. On rigid ground, the 2-DOF controller does a better job of approaching the obstacles more closely. The controller effort is also lower in Test 1 on rigid ground than on granular terrain in Test 3.

The forces the vehicle experiences from driving on granular terrain can be much different than forces on simple rigid ground. Examining Figs. 8a and 9a, the vehicle navigating on granular terrain does not turn as sharply for a given steering command as the vehicle does on rigid ground terrain. To understand the different behavior between a vehicle driving on rigid and granular terrain, a parametric study should be performed analyzing the vehicle driving on a variety of granular terrains with different granular parameters.

Tests 3 and 4 were performed on granular terrain modeled as randomly packed uniform spheres with an inter-particle friction coefficient of $\mu = 0.8$ and a macro-scale friction angle between roughly $65^\circ \leq \phi \leq 70^\circ$ with particle rotation prohibited. This granular material resembles railroad ballast. These same tests were attempted on the the same randomly packed uniform spheres with particle rotation allowed resulting in a macro-scale friction angle between $35^\circ \leq \phi \leq 40^\circ$. This granular material resembles a dry sand. The results of this second set of tests on dry sand are not plotted or tabulated because the vehicle failed to move from the initial position. Instead, the vehicle initializes on the terrain and spins its wheels in place, making no progress forward as the wheels slowly dig themselves down into the granular terrain. To confirm this behavior was not a result of poor modeling techniques, tests were attempted on particles half the size and on terrain twice as thick. These attempts resulted in the same vehicle behavior. Generally the MPC LIDAR-based constant speed local obstacle avoidance controller used in this study is not appropriate for use on all granular terrains. There are situations where a combined speed and steering controller similar to in [15] is more appropriate or even some new speed and steering

controller which accounts for terrain parameters and mobility information to better predict vehicle movement. From this failed test case, neither the rigid ground 2-DOF yaw plane model nor the 14-DOF vehicle models were appropriate for predicting vehicle behavior and performance on dry sand.

5 CONCLUSIONS

In this study, using the multibody physics API Chrono a simulation has been developed of a HMMWV driving through a user specified obstacle field towards a defined target location. Within this simulation, a MPC LIDAR-based local obstacle avoidance controller has been implemented to navigate the vehicle around obstacles as it encounters them. The controller uses a simplified analytical vehicle model to predict the controlled Chrono vehicle states within a finite prediction horizon to determine the optimal path and steering sequence forward from the current vehicle state. Two MPC Algorithms were developed, one that uses a 2-DOF vehicle model to predict Chrono vehicle states and a 14-DOF model. These two controllers were used to navigate a Chrono HMMWV through two obstacle courses on both rigid terrain and granular terrain. The results were compared to understand what improvements need to be made to the MPC LIDAR-based local obstacle avoidance controller to successfully control a vehicle on granular terrain.

As in [2], the controller with the 14-DOF internal vehicle model performs marginally better than the controller with the 2-DOF internal vehicle model in all situations. However, tests with the 2-DOF controller prove that this controller can still successfully navigate a vehicle through an obstacle field when the vehicle is moving at non-extreme speeds. Using the 2-DOF controller also allows for faster calculation of optimal steering sequences which is required for realtime implementation. Both of the controllers navigated the vehicle worse on granular terrain than on rigid ground terrain as expected. This study also highlights the complexities introduced to vehicle modeling when the terrain is no longer rigid. Examining the vehicle trajectories and lateral accelerations on granular terrain, there are clear differences in tire ground interactions. The terrain granular parameters affect the turning characteristics of the vehicle, acceleration abilities, and overall vehicle dynamic performance which is not control predictive with the currently used 2-DOF and 14-DOF analytical vehicle models. In this study, the controller models used the Pacejka Magic Formula to predict tire forces to remain consistent with [2]. However in future studies, a crude deformable terrain model should be included in the controller to predict the ground tire interactions more accurately.

The results of this study are promising. Since the 2-DOF controller is able to navigate the Chrono vehicle on granular terrain modeled as railroad ballast, then there is some small set of granular terrain on which a 2-DOF rigid gound yaw plane model is sufficient for predicting vehicle performance well enough to guide the vehicle successfully and safely to a target point. However, these results also emphasize the need for future research and studies relating to vehicle simulation on granular terrain. This same sort of study should be performed parametrically on a comprehensive set of different granular terrains to understand the difference between vehicle performance on rigid ground and granular terrain. From there, a new intermediate analytical model should be research and developed that is not as complex as the 14-DOF model but does take into account granular parameters such as the friction angle. From running tests on railroad ballast and dry sand, this study also emphasizes the need for a speed and steering controller which uses information about the current terrain and powertrain to better predict vehicle performance on a wider variety of terrains. This sort of controller

would help address situations as seen in this study where the vehicle simply spins its wheels in place on loose dry sand.

Acknowledgments

Chrono development was supported in part by U.S. Army TARDEC Rapid Innovation Fund grant No. W911NF-13-R-0011, Topic No. 6a, "Maneuverability Prediction". Support for the development of Chrono::Vehicle was provided by U.S. Army TARDEC grant W56HZV-08-C-0236.

REFERENCES

[1] F. Allgower and R. Findeisen. An introduction to nonlinear model predictive control. In *21st Benelux Meeting on Systems and Control*, 2002.

[2] J. Liu, P. Jayakumar, J. L. Stein, and T. Ersal. A study on model fidelity for model predictive control-based obstacle avoidance in high-speed autonomous ground vehicles. *Vehicle System Dynamics*, 2016.

[3] A. Tasora, R. Serban, H. Mazhar, A. Pazouki, D. Melanz, J. Fleischmann, M. Taylor, H. Sugiyama, and D. Negrut. Chrono: An open source multi-physics dynamics engine. In T. Kozubek, editor, *High Performance Computing in Science and Engineering – Lecture Notes in Computer Science*, pages 19–49. Springer, 2016.

[4] R. Serban, A. Tasora, and D. Negrut. Chrono::Vehicle – Template-Based Ground Vehicle Modeling and Simulation. *Intl. J. Vehicle Performance*, submitted, 2017.

[5] P. Cundall. A computer model for simulating progressive large-scale movements in block rock mechanics. In *Proceedings of the International Symposium on Rock Mechanics. Nancy, France*, 1971.

[6] K.L. Johnson. *Contact mechanics*. Cambridge University Press, 1987.

[7] J. Fleischmann, R. Serban, D. Negrut, and P. Jayakumar. On the importance of displacement history in soft-body contact models. *Journal of Computational and Nonlinear Dynamics*, 11(4):044502–1–5, 2016.

[8] J. Y. Wong. *Theory of ground vehicles*. J. Wiley, New York, N.Y., 2nd edition, 1993.

[9] J. A. Fleischmann, M. E. Plesha, and W. J. Drugan. Determination of yield surfaces for isotropic non-cohesive particulate materials by the discrete element method. *Geotechnical and Geological Engineering*, 32(4):1081–1100, 2014.

[10] G.-C. Cho, J. Dodds, and J. C. Santamarina. Particle shape effects on packing density, stiffness, and strength: Natural and crushed sands. *Journal of Geotechnical and Geoenvironmental Engineering*, 132(5):591–602, 2006.

[11] B. Indraratna, D. Ionescu, and H. D. Christie. Shear behavior of railway ballast based on large-scale triaxial tests. *Journal of Geotechnical and Geoenvironmental Engineering*, 124(5):439–449, 1998.

[12] H. B. PACEJKA and I. J. M. BESSELINK. Magic formula tyre model with transient properties. *Vehicle System Dynamics*, 27(sup001):234–249, 1997.

[13] J. Liu, P. Jayakumar, J. L. Stein, T. Ersal, and J. L. Overholt. The role of model fidelity in model predictive control based hazard avoidance in unmanned ground vehicles

using lidar sensors. *Dynamic Systems and Control Conference*, 2013.

[14] T. Shim and C. Ghike. Understanding the limitations of different vehicle models for roll dynamics studies. *Vehicle System Dynamics*, 2007.

[15] J. Liu, P. Jayakumar, J. L. Stein, and T. Ersal. An mpc algorithm with combined speed and steering control for obstacle avoidance in autonomous ground vehicles. In *Proceedings of ASME 2015 Dynamic Systems and Control Conference*, 2015.