



**Politecnico  
di Torino**

Master degree's course in Automotive engineering autonomous and connected vehicle  
Academic year 2024/2025  
Graduate session March 2025

# **Tire-ground interaction analysis**

A comparative study of DEM and SCM approaches within Project  
Chrono environment

Supervisors:

Prof. Alessandro Vigliani  
Prof. Enrico Galvagno  
Dott. Domenico Angelo Vella  
Dott. Luca Zerbato

Candidate

Fiammengo Davide

## Summary

<b>Abstract .....</b>	<b>3</b>
<b>1. Introduction.....</b>	<b>4</b>
<b>2. Chrono overview .....</b>	<b>6</b>
<b>2.1 Chrono Vehicle .....</b>	<b>8</b>
<b>3. Contact model .....</b>	<b>13</b>
<b>2.1 The “Complementarity” Approach.....</b>	<b>15</b>
<b>3.2 The “Penalty” Approach .....</b>	<b>18</b>
<b>4. Test scenario.....</b>	<b>20</b>
<b>5. Simulation processing .....</b>	<b>22</b>
<b>6. Postprocessing.....</b>	<b>25</b>
<b>7. Sensitivity analysis .....</b>	<b>25</b>
<b>7.2 Torque analysis .....</b>	<b>31</b>
<b>7.3 Rigid Terrain .....</b>	<b>32</b>
<b>8. SCM terrain model .....</b>	<b>34</b>
<b>7.1 Test scenario .....</b>	<b>38</b>
<b>7.2 Simulation processing.....</b>	<b>39</b>
<b>8. Sensitivity analysis .....</b>	<b>42</b>
<b>8.1 LETE Sand Wong terrain .....</b>	<b>48</b>
<b>8.2 Road North Gower Clayey Loam .....</b>	<b>50</b>
<b>8.3 Snow road .....</b>	<b>52</b>
<b>9. Lateral slip .....</b>	<b>55</b>
<b>9.1 Combined slip .....</b>	<b>57</b>
<b>10. Full vehicle simulation .....</b>	<b>59</b>
<b>10.1 Step acceleration manoeuvre .....</b>	<b>63</b>
<b>10.2 Step braking manoeuvre .....</b>	<b>68</b>
<b>10.3 Step steer manoeuvre .....</b>	<b>73</b>
<b>10.4 Ramp steer manoeuvre .....</b>	<b>78</b>
<b>11. Conclusion .....</b>	<b>81</b>
<b>12. Bibliography .....</b>	<b>83</b>
<b>Appendix .....</b>	<b>84</b>

## Abstract

This thesis explores tire-ground interaction using the open-source simulation program ProjectChrono, developed in C++. The study begins by examining the fundamental features and capabilities of Chrono, providing an overview of its structure and functionalities. A key focus has been made on the two primary contact modelling approaches implemented within the software: the Smooth Contact Method (SMC) and the Non-Smooth Contact (NSC) method. Their mathematical formulation and application principles are analysed to establish a solid basis for understanding the subsequent simulations. To evaluate tire-terrain interaction, an existing tire test rig demo in ProjectChrono was adapted, allowing for a comparative analysis of two terrain models: granular terrain and the Soil Contact Model (SCM). The DEM approach was applied to the granular terrain, while the SCM model was based on the semi-empirical Bekker-Wong formulation. A sensitivity analysis was conducted for both terrain types to assess how different soil parameters influence wheel-terrain interaction. This analysis provided insights into how variations in particle properties in the granular terrain and soil parameters in the SCM model affect force transmission and wheel sinkage. Due to the high computational demands of DEM simulations, hardware limitations restricted the number of particles used, reducing accuracy. In contrast, the SCM model offered a computationally efficient alternative. The study further extends to a full vehicle analysis, where an HMMWV model was tested on SCM terrain under various driving conditions, including acceleration, braking, ramp steer, and step steer maneuvers. These tests provided a comprehensive evaluation of vehicle dynamics and terrain interaction, offering a detailed understanding of the strengths and limitations of each modelling approach.

## 1. Introduction

The way a vehicle's wheels interact with the terrain significantly influences its dynamic performance, particularly on soft or deformable surfaces. Over the years, numerous computational methods have been introduced in terramechanics to better simulate wheel-soil interaction as presented in [14]. These models are developed to enhance off-road mobility predictions through simulation, minimizing the reliance on physical prototypes. Based on their computational efficiency and accuracy, terrain-vehicle interaction models can be categorized into three types: empirical models, continuous representation models (CRMs), and discrete element models (DEMs). Empirical models use simplified mathematical equations to represent wheel-soil interaction, offering high computational efficiency. However, their validity is limited to specific conditions, requiring recalibration when wheel geometry or terrain properties change. Since these models are not derived from fundamental physical principles, they struggle to accurately predict behaviour outside their calibration range, making them less adaptable to varying scenarios. DEM-based terramechanics, instead, adopts a physics-driven approach by modelling each individual soil grain as a separate entity that interacts through frictional contact with neighbouring particles or external objects such as wheels. While this method provides a highly detailed representation of deformable terrain, its major drawback is computational inefficiency. Tracking the motion of every particle significantly increases simulation time, even for relatively simple cases. In contrast, the CRM approach also follows a physics-based framework but offers a more practical trade-off between computational speed and accuracy[13]. This balance makes it a suitable choice for many engineering applications where detailed terrain interaction modelling is required without the excessive computational cost of DEM simulations. Empirical terramechanics models have undergone continuous refinement for over a century and are widely used for analysing wheel-soil interaction in off-road mobility. Recent studies established a relationship between wheel load and terrain sinkage, leading to the development of initial mathematical formulations. Among these, Bekker and Wong introduced a more accurate pressure-sinkage equation by distinguishing between frictional and cohesive components of soil resistance. Their formulation, extensively validated through experimental testing, has become a fundamental tool in terramechanics for predicting wheel behaviour on deformable surfaces. Further improvements were made in modelling the shear stress between the wheel and terrain, particularly with the work of Janosi and Hanamoto, who enhanced Bekker's original approach to better capture frictional forces. More recently, the Soil Contact Model (SCM) has integrated these empirical formulations, combining the pressure-sinkage relationship with shear stress modelling to provide a more versatile approach to terrain interaction [2]. This model has demonstrated real-time computational efficiency in various scenarios, particularly when wheel sinkage is minimal, slip remains within moderate levels, and the wheel shape is close to cylindrical. Despite its advantages, the SCM model requires precise parameter calibration, which is typically performed using a bevameter test rig. This testing procedure helps determine key soil properties, but its application can be cumbersome, especially in environments where terrain conditions change frequently. While empirical models remain widely used due to their efficiency, their reliance on experimental calibration and limited adaptability to highly variable terrains present challenges in certain applications. The CRM-based terramechanics approach models soil as a continuum, significantly reducing the number of degrees of freedom (DOF) compared to the discrete element method (DEM). It relies on solving partial differential equations governing mass and momentum balance, supplemented by closure conditions such as the Jaumann equation for the Cauchy stress tensor. Spatial discretization is typically achieved through the finite element method (FEM), allowing for stress distribution and soil deformation analysis. However, FEM struggles with large deformations due to costly re-meshing requirements and convergence issues. To overcome these limitations, meshless methods like the Material Point Method (MPM) and Smoothed Particle Hydrodynamics (SPH) replace traditional grids with Lagrangian particles, effectively handling significant deformations. Thanks to its reduced DOF count, CRM achieves computational efficiency, often cutting

simulation times by one to two orders of magnitude compared to DEM. The DEM approach provides a highly detailed representation of soil by incorporating the shape and material properties of individual grains, making it particularly effective for capturing wheel-soil interactions. This precision has led to its widespread use in terramechanics, especially for complex mechanical systems. However, real-world applications involve an immense number of soil particles—a single cubic meter of sand can contain billions of grains—making fully-resolved DEM simulations computationally impractical for large-scale vehicle studies. In many scenarios, absolute accuracy is not always necessary. When simulations are used to validate traction control algorithms, optimize path planning, or analyse human-in-the-loop interactions, computational efficiency becomes the priority. Under these conditions, empirical models like SCM offer a practical alternative, balancing performance and realism while significantly reducing computational demands. A potential enhancement to DEM-based simulations is the implementation of the DEM-engine, a GPU-accelerated solver that operates in co-simulation with Chrono. In this setup, Chrono manages the dynamics of the machine system, while the DEM simulation focuses on granular interactions. This configuration allows for the efficient modelling of complex mechanical systems, such as rovers and construction equipment navigating granular terrains. The DEM-engine employs a dual-threaded structure, with one thread (the "kinematics thread") updating active contacts and another (the "dynamics thread") solving the equations of motion. To maximize efficiency, the software utilizes two dedicated NVIDIA GPUs—one for each thread. Although the cost of such high-performance GPUs is considerable, their potential benefits for advanced simulations and research applications may justify the investment. Further details can be found in [6]. This thesis explores tire-ground interaction using ProjectChrono, an open-source simulation software developed in C++. The research focuses on evaluating different contact modeling approaches to determine their accuracy, computational efficiency, and applicability in vehicle mobility simulations. Specifically, it takes into account two terrain modeling techniques: the Discrete Element Method (DEM) for granular terrain and the Soil Contact Model (SCM) based on the semi-empirical Bekker-Wong formulation. Through this comparison, the study aims to assess how different soil modeling approaches influence force transmission, wheel sinkage, and overall vehicle-terrain interaction. A sensitivity analysis is conducted to evaluate the impact of soil parameters on simulation accuracy, providing insights into how changes in terrain properties affect wheel behaviour. To address these objectives, the study begins with an overview of ProjectChrono, detailing its structure and the two primary contact modeling techniques: the Smooth Contact Method (SMC) and the Non-Smooth Contact Method (NSC). A tire test rig demo available within the software was adapted to facilitate a comparative analysis of the granular terrain and SCM models. The DEM approach, which represents soil as a collection of discrete particles, was applied to granular terrain, while the SCM model provided a simplified yet computationally efficient alternative based on empirical formulations. The study also considers the computational feasibility of each method, taking into account hardware limitations and the trade-offs between accuracy and efficiency. Beyond isolated wheel-soil interaction, the research extends to a full vehicle simulation on SCM terrain using an HMMWV model, assessing its performance under various driving conditions such as acceleration, braking, ramp steer, and step steer maneuvers. These tests provide a broader evaluation of vehicle dynamics and the capability of each modeling approach to represent real-world conditions. The results highlight the strengths and limitations of the DEM and SCM methods.

## 2. Chrono overview

Chrono is an open-source software designed to analyse the systems time evolution governed by sets of differential-algebraic equations, ordinary differential equations, and/or partial differential equations. Chrono can currently be used to simulate:

1. the dynamics of large systems of connected bodies governed by differential-algebraic equations;
2. controls and other first-order dynamic systems governed by ordinary differential equations;
3. fluid-solid interaction problems governed, in part, by the Navier-Stokes equations;
4. the dynamics of deformable bodies governed by partial differential equations.

Its simulation engine is built on five foundational components that deliver essential functionalities: equation formulation, solution, collision detection and proximity calculations, parallel computing support, and pre/post-processing. The first component, Equation Formulation, allows for versatile modelling of extensive systems composed of both rigid and flexible bodies, including basic fluid-structure interaction (FSI) scenarios. The second, Equation Solution, offers the necessary algorithms to numerically resolve the equations of motion derived from the system. The third component focuses on proximity calculations, which are critical for detecting collisions and computing short-range interaction forces. The fourth enhances parallel computing capabilities. The software runs on all operating systems available, optimized for various parallel computing platforms, including GPU acceleration with CUDA, multi-core processing via OpenMP, and multi-node computing using MPI. OpenMP is an API for parallel programming on shared-memory systems, such as multi-core CPUs. It distributes workloads across processor cores using directives within the code, minimizing structural changes. MPI, on the other hand, is a library set designed for distributed-memory systems like computer clusters. It enables communication between processes running on different networked nodes through message passing. Lastly, the fifth component, pre- and post-processing, employs real-time support through Irrlicht and OpenGL, along with tools like POV-Ray and ParaView for offline, high-quality visualizations. Developed primarily in C++, Chrono is compiled into a library that third-party applications can utilize, positioning it as middleware software that facilitates customized solutions involving additional user code or external software. Users can access Chrono's functionality through an Application Programming Interface (API) available in both C++ and Python. It is organized into modules which consist in additional libraries that can be optionally used to expand the features of Chrono, depending on users' needs. This allows to:

- reduce the number of dependencies to the minimum, requiring only those strictly needed by the specific user;
- have smaller size libraries;
- have a better isolation during the development.

Each module can be enabled or disabled through the corresponding "ENABLE\_MODULE\_XXX" in the CMake configuration. In particular these modules provide support for additional classes of problems (e.g., finite element analysis and fluid-solid interaction), for modelling and simulation of specialized systems (such as ground vehicles and granular dynamics problems), or for providing specialized parallel computing algorithms (multi-core, GPU, and distributed) for large-scale simulations.

Fig 1 illustrates how modules can depend on external libraries, whereas the core system of Chrono depends on the underlying operating system only.

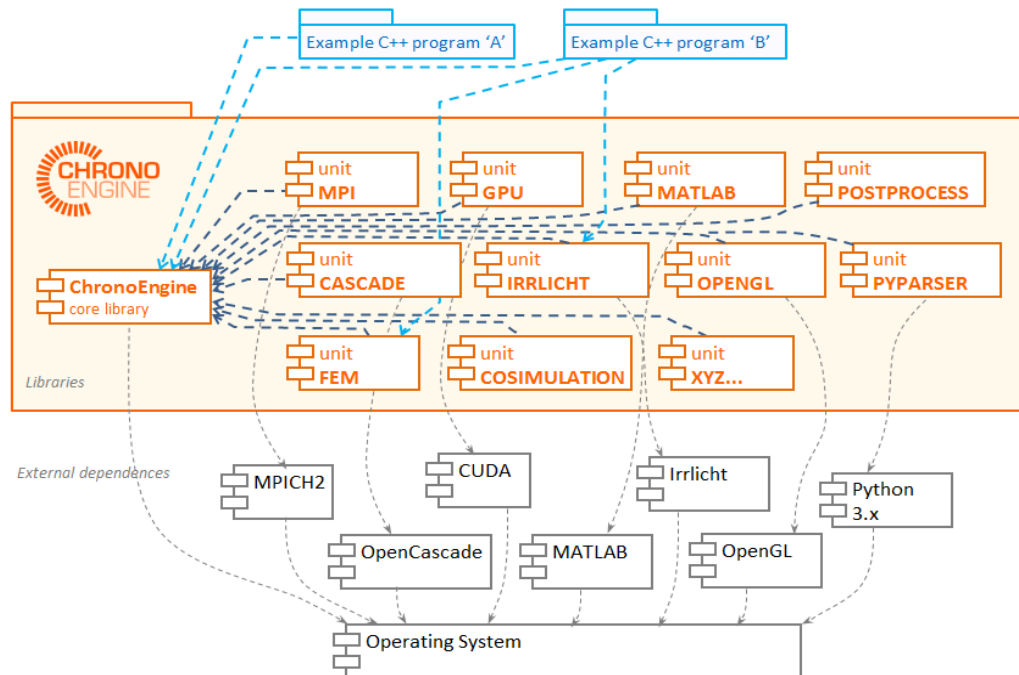


Figure 1: Example of use of Chrono modules.

The core module offers essential functionalities necessary for simulating the dynamics of mechanical systems that include bodies, kinematic joints, force elements, and one-dimensional shaft elements, among others. A key advantage of Chrono is its capability to consider the geometry of the elements within the simulated mechanical system. This feature enables users to analyse interactions between bodies where the relationship between shape and frictional contact forces plays a critical role in the system's dynamics. An example of module which facilitate modelling and simulation in multiphysics scenario is Chrono::FEA (Finite Element Analysis). It is specifically tailored to tackle challenges associated with simulating dynamic systems that may undergo significant displacements, rotations, or deformations. Flexible bodies within this framework can interact with other elements through forces, friction, contact, and constraints. Chrono supports three approaches to finite element analysis:

- the absolute nodal coordinate formulation (ANCF), suitable for large deformations along with arbitrary displacements and rotations;
- the co-rotational formulation, which is effective for small deformations coupled with large displacements and rotations;
- preliminary support for traditional Lagrangian finite elements applicable to scenarios involving substantial deformations, displacements, and rotations.

The API documentation for the primary modules of Chrono is created from their annotated C++ source code using Doxygen. Only the selected modules of interest are compiled into libraries that can be linked to user applications.

## 2.1 Chrono Vehicle

Chrono::Vehicle is a module within the Chrono simulation package designed for modelling, simulating, and visualizing wheeled and tracked vehicle multibody systems. It enables the direct representation of mechanical systems using fundamental elements such as bodies, joints, and force components. However, constructing complex or large-scale models, especially those lacking a clear structure, can introduce challenges and potential errors. Ground vehicle systems, in particular, are highly intricate, including multiple interconnected components that require precise configurations, as in tracked vehicles with their detailed subsystem arrangements. Despite this complexity, vehicle systems typically follow standard topologies and hierarchical structures, shaped by operational and manufacturing constraints that limit the design choices for key sub-assemblies like suspensions, steering, and track systems. For this reason, Chrono::Vehicle [2] adopts a so-called template-based approach which provides a library of vehicle system and subsystem models that are fully parameterized (and as such, only define the topology and interface of any given subsystem) for various topologies of both wheeled and tracked vehicle subsystems, as well as support for closed-loop and interactive driver models, and run-time and off-line visualization of simulation results. A concrete vehicle is defined by specifying actual parameters (geometry, inertia properties, force elements) in a set of templates that correspond to the particular vehicle being modelled. As such, the resulting vehicle model is a complex multibody system that accounts for all principal vehicle moving parts and includes full models of the engine, transmission, and driveline. Template-based modelling simplifies the creation of new vehicle models and enhances reusability. Chrono::Vehicle follows this approach, offering a modular design, versatile API, and a variety of templates. Its structure enables easy system replacement and smooth integration with third-party libraries. The module supports both integrated and co-simulation methods for vehicle-terrain interactions. In co-simulation, systems such as the vehicle, terrain, engine, and driver work in parallel, periodically exchanging data, such as in a force-displacement framework, as illustrated in fig. 2.

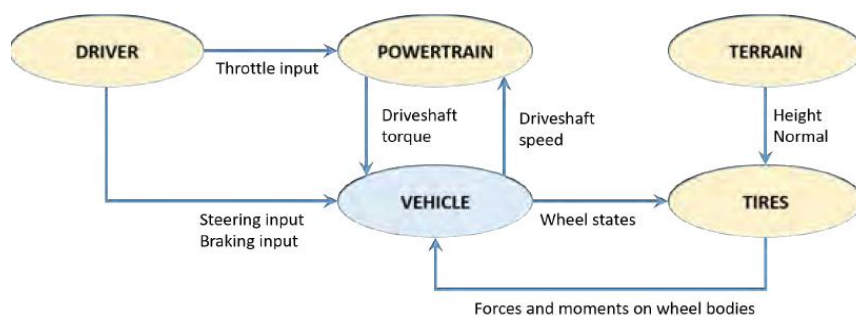


Figure 2: Main systems and exchange data flow for the wheeled vehicles.[2]

Each vehicle subsystem is defined within its own reference frame, requiring all hardpoint locations in a subsystem template to be specified accordingly. A complete vehicle model, whether wheeled or tracked, is assembled by combining instances of these templates for each component. This involves positioning and orienting each part relative to the overall vehicle reference frame while establishing connectivity details, such as linking a steering mechanism to an axle or suspension in a wheeled vehicle. Chrono::Vehicle follows the ISO vehicle axes convention, employing a right-handed coordinate system where the X-axis points forward, the Z-axis points upward, and the Y-axis extends to the vehicle's left. Although these components are



technically part of the vehicle, Chrono::Vehicle organizes the engine, torque converter, and transmission box as a separate system from the driveline. This modular approach enhances modelling flexibility and facilitates the integration of advanced third-party engine models, as illustrated in Figure 3. Multiple powertrain templates are available, offering varying levels of complexity to accommodate different simulation needs. The simplest powertrain template follows a basic engine torque-speed relationship, functioning similarly to a DC motor and operating without a transmission. A more advanced template incorporates a kinematic model based on user-defined torque-speed engine curves, along with a simplified manual or automatic transmission that includes both forward and reverse gears. The most sophisticated powertrain template in Chrono::Vehicle utilizes 1-D shaft elements, which represent components carrying only rotational inertia, and specialized constraints linking these elements to rigid bodies or other shafts. This model features an engine based on speed-torque curves, accounting for power output and losses, a torque converter defined by capacity factor and torque ratio curves, and a transmission system configurable with multiple forward gear ratios and a single reverse gear. The engine block can be installed either longitudinally or transversally and is connected to the chassis to account for torque transfer effects.

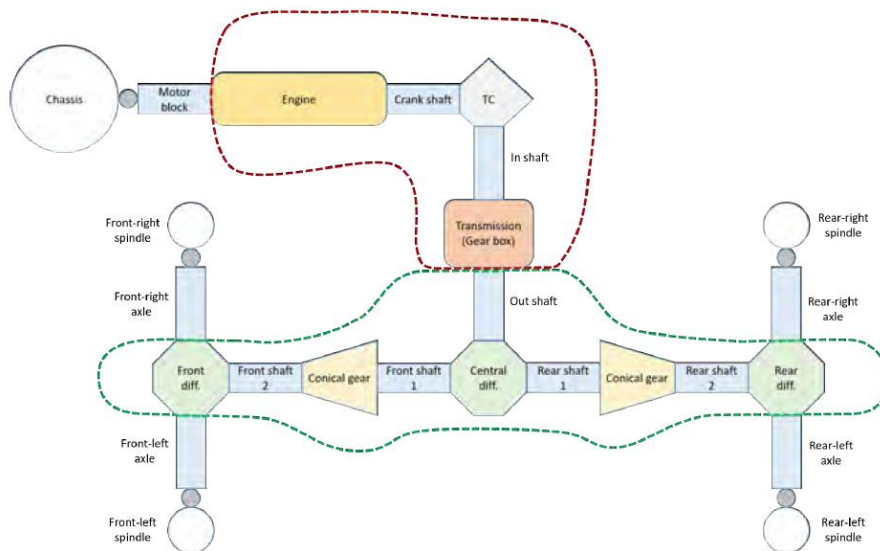


Figure 3: vehicle powertrain schematization [2]

Chrono::Vehicle takes advantage of Chrono's high-performance and parallel computing capabilities, making it well-suited for complex multiphysics simulations. This includes scenarios such as flexible tires on granular terrain, fluid-solid interactions, and autonomous vehicle testing, where computational efficiency and accuracy are crucial. Moreover, it provides a comprehensive set of vehicle subsystem templates (for tires, suspensions, steering mechanisms, drivelines, sprockets, track shoes, etc.), templates for external systems (for powertrains, drivers, terrain models), and additional utility classes and functions for vehicle visualization, monitoring, and collection of simulation results. As a C++ middleware library, Chrono::Vehicle requires the user to provide classes for a concrete instantiation of a particular template. An optional Chrono library provides complete sets of such concrete C++ classes for a few ground vehicles and Chrono provides both wheeled and tracked pre-defined VEHICLE models which currently contains:

- Wheeled vehicle models
- HMMWV: off-road 4-wheel vehicle

- Sedan: generic passenger car
- Citybus: passenger bus
- UAZ: minibus model of the UAZ-452 vehicle
- M-role: multi-purpose wheeled vehicle model
- MAN: truck models (3 different variants: 5t, 7t, and 10t)
- Kraz: semi-trailer truck model
- FMTV: models of trucks from the Family of Medium Tactical Vehicles
- Gator: small utility vehicle model
- RC: remote-controlled vehicle model
- Tracked vehicle models
- M113: M113 tracked vehicle
- Marder: Marder tracked vehicle

An alternative mechanism for defining concrete instantiation of vehicle system and subsystem templates is based on input specification files in the JSON format. Together with all other input files it refers to, this JSON file could completely describes a wheeled vehicle with two axles, using double wishbone suspensions both in front and rear, a Pitman arm steering mechanism attached to the front axle, and a rear-wheel drive line. As already mentioned before, wheeled vehicle is defined as a collection of subsystems (see Fig. 4). It contains a chassis subsystem, a driveline subsystem, and an arbitrary number of axles which, by convention, are numbered starting at the front of the vehicle). This base wheeled vehicle configuration can be extended through user code to different topologies.

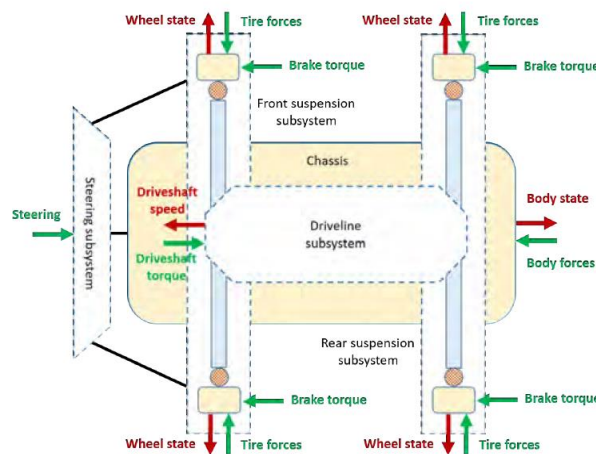


Figure 4: Decomposition of a wheeled vehicle into its subsystems. There are also illustrated the data exchanged with the systems external to the vehicle itself (driver, powertrain, and terrain).[2]

To support both independent and dependent suspension assemblies within a unified API, the Chrono::Vehicle convention defines a suspension subsystem as including both the left and right sides. For symmetric components, only the left side needs to be specified in the suspension template, with the right side automatically mirrored. Additionally, the suspension subsystem is defined in relation to a local reference frame that aligns with the vehicle frame, meaning that only an offset is required to position the suspension assembly within the vehicle system. Each suspension template defines its own set of bodies, joints, and the overall mechanical system topology. The template parameters include the locations of body centers of mass, hardpoint positions, and unit vectors for joint orientations. Other parameters include body masses, inertia tensors, and rotational inertias of the two axle shafts. For the double wishbone suspension (fig 5), the modelling components consist of two spindle bodies, two upright bodies, and four control arms (lower and

upper), connected through spherical and revolute joints. The tie rods are modelled using distance constraints. Users have full flexibility in defining the springs and shocks, which do not necessarily need to be collinear. The suspension templates support both linear and nonlinear spring and damper force elements, with nonlinear elements defined either through lookup tables or as arbitrary functions.

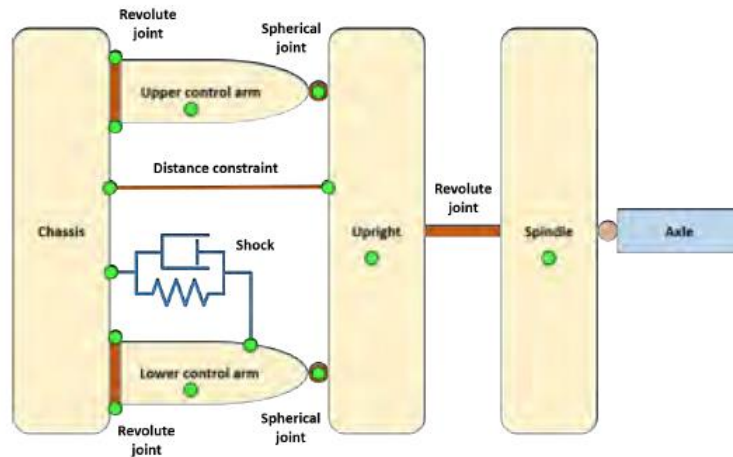


Figure 5: Double wishbone suspension template schematic model [2].

It is also provided templates for multi-link, MacPherson strut, solid-axle, semi trailing arm, Hendrickson PRIMAXX, and Polaris independent rear suspensions. As far as steering mechanisms is concerned, there could be of two types:: Pitman arm and rack-pinion. Unlike suspension templates, a steering mechanism can be offset and rotated when placed within a vehicle system. Multi-steer vehicles are supported by allowing either an arbitrary number of steering mechanisms (which are connected to different axles) or by allowing multiple steerable axles to be connected to the same steering mechanism. Chrono::Vehicle provides a simplified driveline model, suitable for 4WD vehicles, which uses a constant front/rear torque split and simple models of Torsen limited-slip differentials. In addition, two other templates, both using Chrono 1-D shaft elements and specialized connecting elements are provided, for 2WD and 4WD vehicles. Driveline subsystem is connected to one or two of the vehicle's axles. Besides the main components mentioned above, there are also included templates for the brake subsystem (simplified model using a braking-input proportional torque), anti-roll bar subsystem (modelled with two bars connected through a rotational spring-damper), and the wheel subsystem (which is simply a carrier of additional mass and inertia to be compounded with that of the associated suspension spindle). Chrono::Vehicle currently supports three different classes of tire models: rigid, semiempirical, and finite element. The rigid tires are the simplest of the three tire classes offered. The assumption for these models is that the tire is completely rigid and it interacts with the ground and any other rigid objects through the same underlying friction and contact algorithms as the other rigid bodies in Chrono. The contact geometry for these tires can be as simple as a cylinder or as complex as a 3D triangular mesh. These models are not only useful for debugging the overall vehicle model, but they can also be used in cases where runtime is important, the terrain is much softer than the tire, and a highly detailed model of the tire is unnecessary. In such scenarios, including SCM, granular, or FEA-based models, rigid or FEA-based tire models are currently the only two available options implemented in Chrono::Vehicle. The second class of tires models offered are the semi-empirical ones commonly used for vehicle handling. Chrono::Vehicle currently supports TMeasy, Fiala tire model, and a Pacejka based model. Finally, the third class are full finite element representations of the tire. While these models have the potential to be the most accurate due to their detailed physical model of the tire, they are also the most computationally expensive of the tire model. Unlike the rigid or semi-empirical tire models, the finite element-based tire models are able to account for the flexibility in both the tire and in the ground at the same time, which is an important

characteristic for many types of off-road mobility and vehicle dynamics studies. Chrono::Vehicle includes a variety of terrain and soil models, each offering different levels of detail and computational complexity. These range from rigid surfaces to more advanced representations. The simplest model assumes a completely rigid terrain, well-suited for most on-road vehicle simulations. In this setup, the tire-terrain interaction may either rely on a basic height-normal map or use Chrono's internal frictional contact mechanics, depending on the tire model in use. The rigid terrain can be defined as a flat plane, an arbitrary triangular mesh (in a Wavefront OBJ file format), or a height-map represented by a grayscale BMP image. Another terrain choice is provided by a semi-empirical, deformable soil model based on Bekker parameters and the Soil Contact Model (SCM). Unlike the traditional SCM approach with regular grids, this model extends to triangular meshes including an automatic mesh refinement to enhance detail specifically in areas where vehicle tires or track shoes interact with the soil. This refinement improves both memory and computational efficiency. By utilizing Chrono's collision engine, this model efficiently computes vehicle-soil contact forces. See Sec. 7 for further details. Using the Chrono::FEA and Chrono::Granular modules, the deformable soil can be simulated with either finite element or granular material approaches that incorporate contact, friction, and cohesion properties. The finite element method models soil as a continuum, applying multiplicative plasticity theory combined with the Drucker-Prager failure criterion and using a custom 9-node brick element to mitigate locking issues common in standard 8-node elements, without relying on techniques like enhanced assumed strain. For granular-based deformable soil modelling, Chrono uses the Discrete Element Method (DEM), which handles each particle as an individual entity, updating its state over time and applying interaction forces from frictional contacts. In simulations of vehicle mobility on granular terrain, both penalty-based compliant body approach and complementarity-based approach are supported, depending on the specific requirements of the scenario. As these granular dynamics simulations often involve millions of particles, high computational demands are addressed through parallel processing, either by coupling vehicle and terrain simulations. Driver inputs such as steering, throttle, and braking are managed by a driver subsystem with various options, including interactive, data-driven, and closed-loop (e.g., path-following with PID controllers). The base C++ driver system class in Chrono::Vehicle requires minimal implementation details from a driver system template, specifically the ability to return normalized throttle input (from 0 to 1), steering input (from -1 to +1, with negative values indicating a left turn), and braking input (also normalized from 0 to 1). Moreover, the driver system can receive information from other systems, such as the vehicle's state, through its Synchronize method and may include internal dynamics managed by the Advance method. Specific driver templates may add extra input options, such as gear selection for manual transmissions or enabling/disabling cross-drive capabilities in tracked vehicles. Additionally, under the Chrono::CAVE module, Chrono::Vehicle provides closed-loop controllers for autonomous driving, using inputs from sensors like LiDAR, GPS, and IMU. These controllers are designed for connected and autonomous vehicle simulations and support advanced model-predictive controllers for tasks such as obstacle avoidance. Visualization support is available for both real-time interactive simulations and high-quality post-processing rendering for animation generation. Currently, runtime visualization builds upon the Chrono::Irrlicht module for sequential simulations, while the more computationally efficient, though functionally limited, Chrono::OpenGL module is used for parallel simulations involving large-scale granular terrain representations.

### 3. Contact model

The Discrete Element Method (DEM) is a computational strategy for simulating the mechanical behaviour of granular materials by tracking each element's motion and modelling interactions in detail. Granular, many-body, or discrete element problems result in large sets of generalized coordinates  $q$ . For example, granular flow in a hopper can lead to millions or even billions of entries in  $q$ . Such large-scale problems are common, with over 50% of materials processed in industry being granular. Understanding their dynamics is essential in a wide range of applications, including additive manufacturing, terramechanics, nanoparticle self-assembly, composite materials, pyroclastic flows, the formation of asteroids and planets, and meteorite cratering. This knowledge is also valuable in industries such as pharmaceuticals, chemical and biological engineering, food processing, agriculture, manufacturing, construction, and mining. It's important to note that "granular dynamics" isn't limited to micro- or meso-scales. For instance, avalanche dynamics and planetary formation involve large bodies but still qualify as granular dynamics problems, where large collections of bodies interact through friction and contact forces, with their motion influenced by their individual shapes. Chrono is equipped to compute and simulate interactions between objects, including handling collisions and contact responses. The contact simulation process involves two main components:

- Collision Detection: Identifying pairs of points that may come into contact in the immediate future.
- Contact Formulation: Calculating the reaction forces between the shapes that come into contact.

Chrono offers a comprehensive multibody simulation framework, with specialized support for handling frictional contact. This functionality includes both non-smooth (NSC) and smooth (SMC) contact models. The NSC approach uses a complementarity formulation combined with a Coulomb friction model, resulting in a Differential Variational Inequality (DVI) that governs the equations of motion. In contrast, the SMC method, or penalty method, models contact forces based on localized deformations at the contact point, effectively regularizing the contact interaction. These two methods differ significantly in terms of modelling capacities, parameter requirements, computational complexity, and suitability for parallel processing. For frictional contact and granular simulations, where each particle interacts through contact and friction, Chrono applies a variant of the Discrete Element Method (DEM). Typically, the term DEM refers to SMC-type models, so Chrono distinguishes between DEM-C (for complementarity, using NSC) and DEM-P (for penalty, using SMC). A major distinction between NSC and SMC methods, is in how equations of motion are formulated: DEM-C requires solving a global optimization problem derived from DVI equations, while DEM-P allows frictional contact forces to be calculated individually per contact point, making DEM-P more conducive to parallelization, particularly in distributed computing setups.

Both DEM-P and DEM-C methods rely on an efficient collision detection mechanism that generates a geometric profile of interactions between colliding shape pairs in a given configuration. This profile typically includes the closest points, the contact normal, and the radii of curvature at the contact location. A straightforward approach to collision detection tests all possible shape pairs which quickly becomes impractical for large systems. To improve efficiency, most collision detection frameworks employ a two-phase process. The initial phase, known as the broad-phase, filters out pairs of shapes that are clearly not in proximity. This phase uses bounding shapes (such as spheres or boxes aligned to axes or objects) and specialized data structures like dynamic trees or hierarchical grids, enabling the system to focus only on nearby shape pairs. The subsequent narrow-phase examines these filtered pairs in greater geometric detail. This phase may use analytical methods for simpler shapes or algorithms like Gilbert-Johnson-Keerthi (GJK) or Minkowski Portal Refinement (MPR) to compute contact details for pairs of general convex shapes. For concave shapes, a pre-processing step that decomposes them into convex components is often necessary to enable accurate collision handling.

Chrono can set up two different types of systems based on the contact method:

1. ChSystemNSC (Non-Smooth Contact, NSC):

- Contacts are treated as hard/stiff constraints.
- No stiffening terms are introduced, allowing for longer time steps.
- Specialized variational inequality (VI) solvers are required.
- Only the ChSolverADMM (Alternating Direction Method of Multipliers) solver, which is specifically designed to handle both non-smooth contacts and finite element models (FEA), making it especially useful for scenarios involving deformable materials.

2. ChSystemSMC (Smooth Contact, SMC):

- This system takes a different approach, handling contacts as "soft" or smooth interactions. Here, contact forces are calculated based on interpenetration between objects in contact, scaled by a stiffness term (compliance/stiffness) similar to a spring.
- Since contact stiffness affects the response, shorter time steps are often needed to ensure stability, especially with higher stiffness values.
- No VI solvers are required, so calculations may be simpler to implement

Users should carefully choose the appropriate system for their needs, as time steps may differ by up to two or three orders of magnitude between the two methods. However, both systems are fully equivalent if no collisions are involved. In Chrono, each ChBody (and any other object derived from ChContactable) contains information about its collision properties through a ChCollisionModel object. This ChCollisionModel can include multiple ChCollisionShapes, each of which includes a geometric shape and the collision surface material.

Each ChBody can therefore contain the following structure:

- ChCollisionModel: holds several ChCollisionShape, each containing
  - ChGeometry: defines the geometric shape (e.g., the size of a box or sphere) that forms the collision boundary.
  - ChContactMaterial: specifies the material properties of the collision surface, such as friction and restitution, affecting how the object interacts upon impact with other bodies

As with many other objects in Chrono, items within the ChCollisionModel are stored as pointers, allowing them to be easily shared across different bodies. This is especially useful for ChContactMaterial class objects.

## 2.1 The “Complementarity” Approach

The model which is presented is an approximation of the Coulomb friction model. A system with  $n$  bodies in three dimensions is represented by  $m = 6n$  coordinates. Two rigid bodies should not penetrate, and, if they are in contact, there should be friction acting at the interface. To enforce the nonpenetration constraint, we assume that there exists a function  $\Phi(q)$ , which is called the gap function, that satisfies the following requirements:

$$\Phi(q) = \begin{cases} > 0 & \text{If the bodies are separated} \\ = 0 & \text{If the touch each other} \\ < 0 & \text{If the bodies are interpenetrating} \end{cases} \quad 1$$

If a position  $q$  is feasible and the contact is active, that is  $\Phi(q) = 0$ , then at the contact there is a normal force and a tangential force. Let  $n$  be the normal at the contact, pointing toward the exterior of the body, and let  $t_1$  and  $t_2$  be the tangents at the contact. Here  $n$ ,  $t_1$  and  $t_2$  are mutually orthogonal vectors of length one in three dimensions. Moreover, they are function of the position  $q$  and the subscripts  $u, v$  refer to quantities related to the two linearly independent tangential directions at a given contact.

$$\mathbf{F} = \mathbf{F}_N + \mathbf{F}_T = \widehat{\gamma}_n \mathbf{n} + \widehat{\gamma}_u \mathbf{t}_1 + \widehat{\gamma}_v \mathbf{t}_2 \quad 2$$

The DEM-C framework utilizes a complementarity condition that establishes a unilateral constraint preventing penetration (3a). For any potential contact  $i$  within the active contact set  $A(q(t))$ , either the gap  $\phi_i$  between the two geometrical bodies is equal to zero, resulting in a positive normal contact force  $\widehat{\gamma}_{i,n}$ , or the opposite is true. The Coulomb model consists of the following assumptions:

$$\widehat{\gamma}_n \geq 0, \quad \Phi(q) \geq 0, \quad \Phi(q) \widehat{\gamma}_n = 0 \quad 3a$$

$$\mu \widehat{\gamma}_n \geq \sqrt{\widehat{\gamma}_u^2 + \widehat{\gamma}_v^2}, \quad \|v_T\| \left( \mu \widehat{\gamma}_n - \sqrt{\widehat{\gamma}_u^2 + \widehat{\gamma}_v^2} \right) = 0 \quad 3b$$

$$\langle F_T, v_T \rangle = -\|F_T\| \|v_T\| \quad 3c$$

where  $v_T$  is the relative tangential velocity at contact. The effect of the friction over the dynamical system is defined by the friction coefficient  $\mu \in R^+$ , that typically has a value between 0 and 1 for most materials. The constraint in 3c ensures that the reaction force is dissipative, meaning it acts to resist to the motion. A useful way to express this constraint is through the maximum dissipation principle, which states that the system will adjust the magnitudes of the tangential forces to minimize energy loss during contact (4).

$$(\widehat{\gamma}_u, \widehat{\gamma}_v) = \underset{\sqrt{\widehat{\gamma}_u^2 + \widehat{\gamma}_v^2} \leq \mu \widehat{\gamma}_n}{\operatorname{argmin}} (\widehat{\gamma}_u \mathbf{t}_1 + \widehat{\gamma}_v \mathbf{t}_2)^T v_T \quad 4$$

By applying this principle, the system effectively finds the optimal values for the friction forces components the of the friction force  $(\overline{\gamma}_{i,w}, \overline{\gamma}_{i,u})$  for contact  $i$ .

To effectively integrate these contact forces into dynamic models, it is employed a mapping technique that translates the vectors  $n$ ,  $t_1$  and  $t_2$  from contact coordinates into generalized coordinates. For a three-dimensional vector  $v$ , the mapping to generalized coordinates is

$$v \rightarrow \begin{cases} v \\ r_1 x v \\ -v \\ r_2 x v \end{cases} \quad 5$$

$r_1$  and  $r_2$  refer to the relative positions of the contact point in relation to the centres of mass of the two bodies. By employing this mapping, we can express the generalized vector forms of  $n$ ,  $t_1$  and  $t_2$  as  $D_n$ ,  $u$ , and  $D_v$ . However, a drawback of using generalized coordinates is that they are no longer mutually orthogonal in these new coordinates. If  $v$  is the generalized velocity, the tangential velocity satisfies the following conditions:

$$t_1^T v_T = v^T D_u, \quad t_2^T v_T = v^T D_v \quad 6$$

In generalized coordinates, the Coulomb model becomes:

$$F_N = \widehat{\gamma}_n D_n, \quad F_T = \widehat{\gamma}_v D_v + \widehat{\gamma}_u D_u \quad 7$$

$$(\widehat{\gamma}_u, \widehat{\gamma}_v) = \underset{\sqrt{\widehat{\gamma}_u^2 + \widehat{\gamma}_v^2} \leq \mu \widehat{\gamma}_n}{\operatorname{argmin}} (\widehat{\gamma}_u D_u + \widehat{\gamma}_v D_v)^T v_T \quad 8$$

Consequently, this results in a differential variational inequality (DVI) problem. Solving this problem numerically presents significant challenges and remains a focus of ongoing research. Various approaches to numerical discretization have been explored but the method applied here follows the approach outlined in [5]. After discretizing time and relaxing certain kinematic constraints, the problem is formulated as a conically constrained quadratic optimization problem. Specifically, the aim is to minimize:

$$\min q(\gamma) = \frac{1}{2} \gamma^T N_\gamma + p^T \gamma \quad 9$$

Subject to  $\gamma_i \in Y_i$  for  $i = 1, 2, \dots, n_c$  where:

- $n_c$  is the number of active contacts; i.e., the number of elements in  $A(q(t))$ .
- $Y$  is the friction cone for each contact  $i$ .
- $\gamma = [\gamma_1^T, \gamma_2^T, \dots, \gamma_{n_c}^T]^T$  is a vector where each  $\gamma_i = [\gamma_{i,n}, \gamma_{i,w}, \gamma_{i,u}]^T \in R^3$  with  $h$  as the time step for simulation.
- $p \in R^{3n_c}$  is a vector and  $N \in R^{3n_c \times 3n_c}$  is a positive semi-definite matrix. Both  $p$  and  $N$  vary at each time step but remain independent of the Lagrange multipliers  $\gamma$ .

The detailed expressions for  $p, N$ , as well as an in-depth explanation of how the differential variational inequality problem is reformulated into the conic constraint optimization problem, are provided in [5]. Chrono handles this optimization problem by allowing the selection of various solvers, each tailored to meet specific needs for accuracy and convergence depending on the complexity of the simulation:



### 1. PSOR

- Suitable for low-precision needs but may experience slow convergence, especially with irregular mass ratios.
- Works with Differential Variational Inequalities (DVIs), allowing hard contacts through complementarity constraints.
- Commonly used in smaller problems where high solution precision isn't critical.

### 2. APGD

- Excellent convergence properties, making it a preferred choice in simulations that demand high accuracy.
- Also supports DVIs, handling hard contacts with complementarity conditions.

### 3. Barzilai-Borwein

- Offers reliable convergence and supports DVIs for hard contacts with complementarity.
- Comparable to APGD, though it may be more effective in scenarios with high mass ratios.

The DEM-C approach shows several advantages: it requires only a few parameters, such as the coefficients for friction and cohesion; it allows for a larger simulation time step  $h$ , as it does not rely on model stiffness; and it is flexible enough to handle bodies of any geometry. However, DEM-C also has limitations. At each time step, an optimization problem must be solved, increasing the complexity of the process. It also expands the problem size, adding three unknowns ( $\gamma_{i,n}, \gamma_{i,w}, \gamma_{i,u}$ ) for each active contact, which can be challenging for granular dynamics simulations. DEM simulations are often computationally intensive due to two primary factors. The elements modelled can be very stiff or small, which necessitates very small-time steps for numerical stability. Additionally, the collision detection process is resource-demanding. To tackle these challenges and enhance simulation speed, DEM has been accelerated through parallel computing techniques on CPUs with OpenMP, distributed memory clusters with the MPI standard, and hybrid MPI-OpenMP approaches. Another effective parallel computing architecture is the Graphics Processing Unit (GPU), which has seen increasing adoption in DEM applications. In the demonstration “demo\_VEH\_Multicore\_TireTestRig,” an NSC system is used because the tire and ground particles are modelled as rigid bodies, making the complementarity approach ideal for contact interactions. This setup also uses the Multicore collision detection system to optimize performance and the APGD solver.

### 3.2 The “Penalty” Approach

DEM-P utilizes a regularization technique that relaxes the assumption of purely rigid bodies, assuming instead that the bodies experience slight deformation at contact points. Fully modelling this deformation with finite element methods would be computationally intensive. To approximate this effect, DEM-P generates a surrogate deformation for bodies in contact during each time step, based on the detected interpenetration depth and the shapes of the contact bodies. While complex geometries may complicate the process, this surrogate deformation is often simplified using Hertzian contact theory, which applies well to basic forms like sphere-to-sphere or sphere-to-plane contacts. This approach provides a generalized method for calculating normal ( $F_n$ ) and tangential forces ( $F_t$ ) at contact points. When bodies  $i$  and  $j$  come into contact, the interaction is modeled using a spring–dashpot system to represent the normal force  $F_n$ . The tangential friction force  $F_t$  is determined by the materials' characteristics and local micro-deformations, and it is constrained by the Coulomb friction condition, which is defined by the friction coefficient  $\mu$ . In particular:

$$\mathbf{F}_n = \sqrt{\bar{R}\delta_n}(K_n\delta_n - C_n\bar{m}\mathbf{v}_n) \quad 10a$$

$$\mathbf{F}_t = \sqrt{\bar{R}\delta_n}(-K_t\delta_t - C_t\bar{m}\mathbf{v}_t), \quad |\mathbf{F}_t| \leq \mu|\mathbf{F}_n| \quad 10b$$

$$\bar{R} = \frac{R_i R_j}{R_i + R_j} \quad 10c$$

$$\bar{m} = \frac{m_i m_j}{(m_i + m_j)} \quad 10d$$

The subscripts  $n$  and  $t$  represent normal and tangential directions, respectively. In this context,  $\delta$  denotes the overlap between the two interacting bodies, while  $\bar{R}$  and  $\bar{m}$  refer to the effective radius of curvature and mass. The variable  $\mathbf{v}$  indicates the relative velocity at the point of contact. For the materials in contact, the normal and tangential stiffness values  $K_n, K_t$  and the damping coefficients  $C_n, C_t$  are derived from various constitutive laws based on measurable physical properties such as Young's modulus, Poisson's ratio, and the coefficient of restitution. A key assumption in this model is that the geometries can experience slight penetration  $\delta_n$  at the contact point. The normal penetration vector is represented as  $\boldsymbol{\mu}_n = \delta_n \mathbf{n}$ . The relative velocity at the contact point, denoted as  $\mathbf{v}_{rel} = \mathbf{v}_n + \mathbf{v}_t$ , is determined as follows:

$$\mathbf{v}_{rel} = \mathbf{v}_j + \boldsymbol{\omega}_j \times \mathbf{r}_j - \mathbf{v}_i - \boldsymbol{\omega}_i \times \mathbf{r}_i \quad 11a$$

$$\mathbf{v}_n = (\mathbf{v}_{rel} \cdot \mathbf{n}) \mathbf{n} \quad 11b$$

$$\mathbf{v}_t = \mathbf{v}_{rel} - \mathbf{v}_n \quad 11c$$

In this context,  $\mathbf{v}_i, \mathbf{w}_i$  and  $\mathbf{v}_j, \mathbf{w}_j$  represent the linear velocities and angular velocities of bodies  $i$  and  $j$ , respectively. The position vectors  $\mathbf{r}_i$  and  $\mathbf{r}_j$  extend from the centres of mass of bodies  $i$  and  $j$  to the point of contact between them. The friction force  $F_t$  is influenced by the history of tangential micro-displacement  $\mu_t$ , which is updated incrementally during each time step for the duration of the contact event, based on the tangential velocity  $\mathbf{v}_t$ . If  $\mu_t'$  denotes the newly updated tangential micro-displacement, then:

$$\mathbf{u}' = \mathbf{u}_t + h\mathbf{v}_t \quad 12a$$

$$\mathbf{u}'_t = \mathbf{u}' - (\mathbf{u}' \cdot \mathbf{n})\mathbf{n} \quad 12b$$

$h$  refers to the size of the time step. The method used to update  $\mu_t'$  is based on the approach outlined in [10]. Once the update is performed, it might be necessary to adjust the tangential micro-displacement  $\mu_t'$  to determine the final value of  $\mu_t$  for the subsequent time step, ensuring that the constraint  $||F_t|| < \mu||F_n||$  is met:

$$\mathbf{u}' = \begin{cases} \mathbf{u}'_t, & \text{if } |F_t| \leq \mu|F_n| \\ \frac{\mu||F_n||}{k_t} \frac{u'_t}{||u'_t||} & \text{otherwise} \end{cases} \quad 13$$

The motion equations for element  $i$  are expressed as follows:

$$m_i \frac{d\mathbf{v}_i}{dt} = m_i \mathbf{g} + \sum_{j=1}^{n_c} (\mathbf{F}_n + \mathbf{F}_t) \quad 14a$$

$$I_i \frac{d\boldsymbol{\omega}_i}{dt} = \sum_{j=1}^{n_c} (\mathbf{r}_j \times \mathbf{F}_t) \quad 14b$$

DEM-P is frequently utilized in the fields of soft-matter physics and geomechanics due to several appealing features. For instance, there is a substantial body of literature that provides valuable insights from previous successful applications. Additionally, this method manages friction and contact without increasing the complexity of the numerical problem, and its algorithm is relatively easy to implement in software. However, DEM-P also presents some limitations. Determining the model parameters can be quite difficult, especially in large, heterogeneous granular systems. The need for small integration time steps arises from the high values of the contact stiffness coefficients. Furthermore, accurately calculating the friction force requires tracking the history of local tangential deformations for each contact. There are also challenges in addressing contacts between complex-shaped bodies when the fundamental assumptions of sphere-to-sphere or sphere-to-plane contacts do not hold, which may compel users to resort to ad-hoc methods to establish appropriate values for effective radius and mass.

## 4. Test scenario

In order to study the tire-terrain interaction on a granular surface using the Discrete Element Method (DEM), the “demo\_VEH\_Multicore\_TireTestRig” from Project Chrono has been chosen. This setup, shown in Fig 1, was built using the ChTireTestRig class, which integrates both the tire and the underlying system modelled with a non-smooth contact (NSC) approach. To efficiently manage collision dynamics, the Chrono::Multicore module, which is an OpenMP-parallelized solver specifically designed for granular and multibody dynamics for a CPU-based solution, has been employed. Although its performance may not match that of a GPU implementation, Chrono::Multicore allows for a fully integrated vehicle simulation on granular terrain without requiring a co-simulation setup.

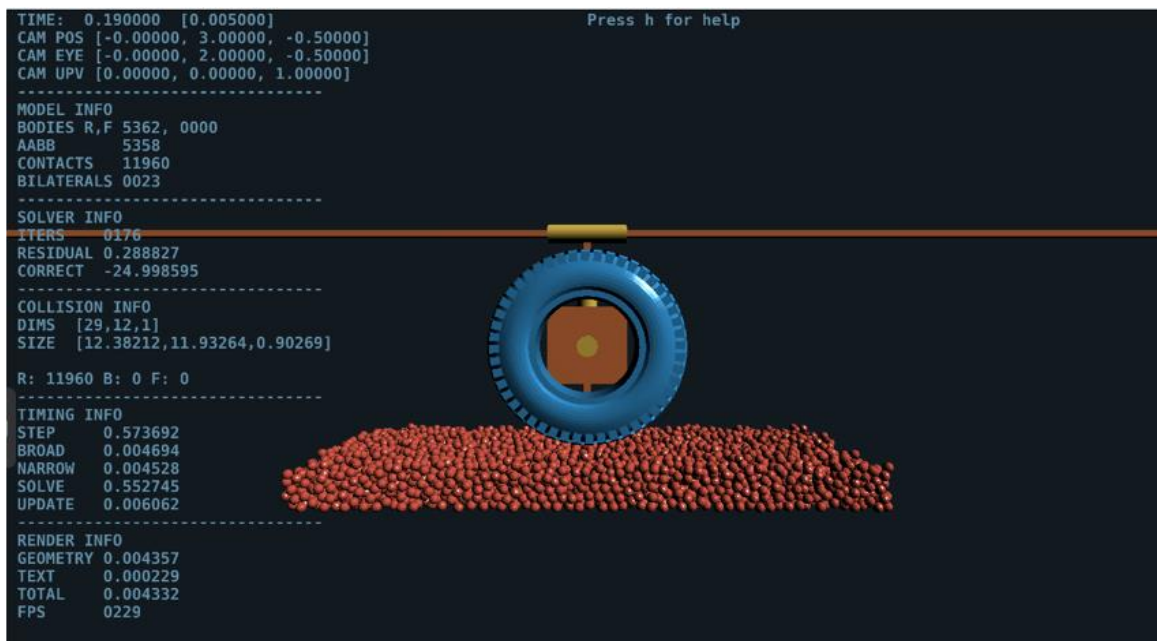


Figure 6: Tire test rig. Beginning of the simulation.

This module also accommodates arbitrary particle shapes, although it should be noted that more complex geometries may impact performance. However, it's not possible to use different geometries within the same simulation, therefore the terrain is made up of monolithic spheres which have all the same radius. Additionally, a set of mathematical solver parameters has been defined consistently to a good resolution of equations which govern the collision events.

From a mechanical point of view, the system is made up of:

- Carrier: the cylindrical shaped element to which a linear speed motor is linked in order to allow the longitudinal tire motion.
- Rig slip: cylindrical element connected to the carrier through a prismatic joint which make possible the descent of the tire on the ground;
- Chassis: square element completely constrained to the rig slip;

- Spindle: cylindrical element connected to the chassis through a revolute joint which allows the rotation of the wheel around its axis.
- Off-road tire: it is a HMMWV (High Mobility Multipurpose Wheeled Vehicle) wheel, solidal to the spindle. It has been imported from an .obj file which was already present in the downloaded material after the program installation. It is implemented as a rigid tire with radius equal to 0.467 m and a normal load of 2000 N is applied on it. In fig 7 it is possible to notice the tire tread which has only visualisation purpose, its geometry doesn't affect the interaction with the soil;

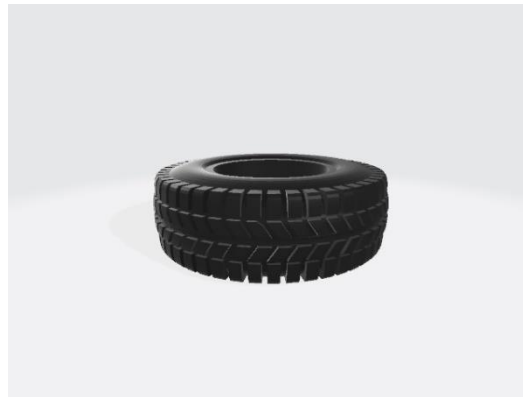


Figure 7: HMMWV Tire

The properties of the tire are reported in the figure 8 which are extracted from the json included, again, in the Chrono download procedure. As illustrated, there are included geometrical parameters such as the radius, the mass and inertia, but also the contact material ones such as CoR, Young Modulus, stiffness, and damping coefficients, which are used in case of system SMC.

```
{
  "Name": "HMMWV Rigid Tire",
  "Type": "Tire",
  "Template": "RigidTire",

  "Radius": 0.467,
  "Width": 0.254,
  "Mass": 37.6,
  "Inertia": [3.84, 6.69, 3.84],

  "Contact Material": {
    "Coefficient of Friction": 0.9,
    "Coefficient of Restitution": 0.1,

    "Properties": {
      "Young Modulus": 2e7,
      "Poisson Ratio": 0.3
    },

    "Coefficients": {
      "Normal Stiffness": 2e5,
      "Normal Damping": 40.0,
      "Tangential Stiffness": 2e5,
      "Tangential Damping": 20.0
    }
  }
}
```

Figure 8: HMMWV tire properties

None of the tire models from the family of “force element”-type models (i.e., derived from ChForceElementTire, including Pacejka, Fiala, TMeasy) carry any collision geometry and as such cannot be used on deformable terrain. As a result, the tire which is used is rigid.

- Ground body: it is defined as a rigid box container where the granular terrain is generated through the function “rig.SetTerrainGranular”. Although it is not visible during the simulation, it holds all the particles. The parameters, which the function takes as inputs and define the terrain, are reported in table 1.

Soil parameters	Soft terrain
Particle radius [m]	0.02
Number of layers	6
Density [kg/m3]	2000
Inter-particle friction	0.9
Inter-particle cohesion pressure [Pa]	10e3
Young Modulus [Pa]	1e7

Table 1: Default soil parameters

## 5. Simulation processing

Several attempts were made before defining the optimal test parameters for the simulation. Initially, the simulation has been carried out over 600 iterations, corresponding to 3 seconds, with a fixed time step of 0.005 s. The terrain subsystem, measuring 2.33 m × 0.2 m × 1 m, included 13.879 identical spherical particles arranged in layers. The simulation begins with the tire being dropped onto the terrain, requiring some time to settle before stable conditions are reached. An initial assessment period of 1 second was set, though this can be adjusted as needed. After stabilization, the wheel is pulled forward by the carrier while rotating at a predefined angular velocity, controlled through the motion functions illustrated in Figure 10. This analysis aims to evaluate how variations in soil properties influence interaction dynamics. To this end, simulations were conducted using the soil parameters listed in Table 2, where the percentages indicate the reduction in each parameter’s value.

	Particle radius [m]	Number of layers	Density [kg/m3]	Inter-particle friction (ipf)	Inter-particle cohesion pressure [Pa] (ipcp)	Young Modulus [Pa]
Soft Terrain (20% reduction)	0.016	6	1600	0.72	8e3	8e6
Soft Terrain (50% reduction)	0.01	6	1000	0.45	5e3	5e6

Table 2: soil parameters modified.

The computational time increased reducing particle radius, due to the consequent large particle number, from 5 minutes for the terrain with the default values shown in tab 1, up to 40 minutes for the Soft Terrain (50%). Instead, simulation with lower values of radius particle than 0.01 m didn't work at all, due to sudden crash of the program caused by the large computational effort. For this reason, there aren't available results to analyse for this kind of terrain.

To ensure that the tire remains in contact with the particles rather than the ground box, the camera angle was changed from a side view to a bottom-up perspective. This allowed for a clearer observation of any inconsistencies caused by particle displacement under the tyre. During the simulations, it was observed that the tire displaced a significant number of particles, reflecting the influence of changing soil parameters. However, as shown in Figure 10, the tire did not maintain full contact with the soil particles but also interacted with the rigid ground body. This unwanted contact could affect the accuracy of the simulation results.

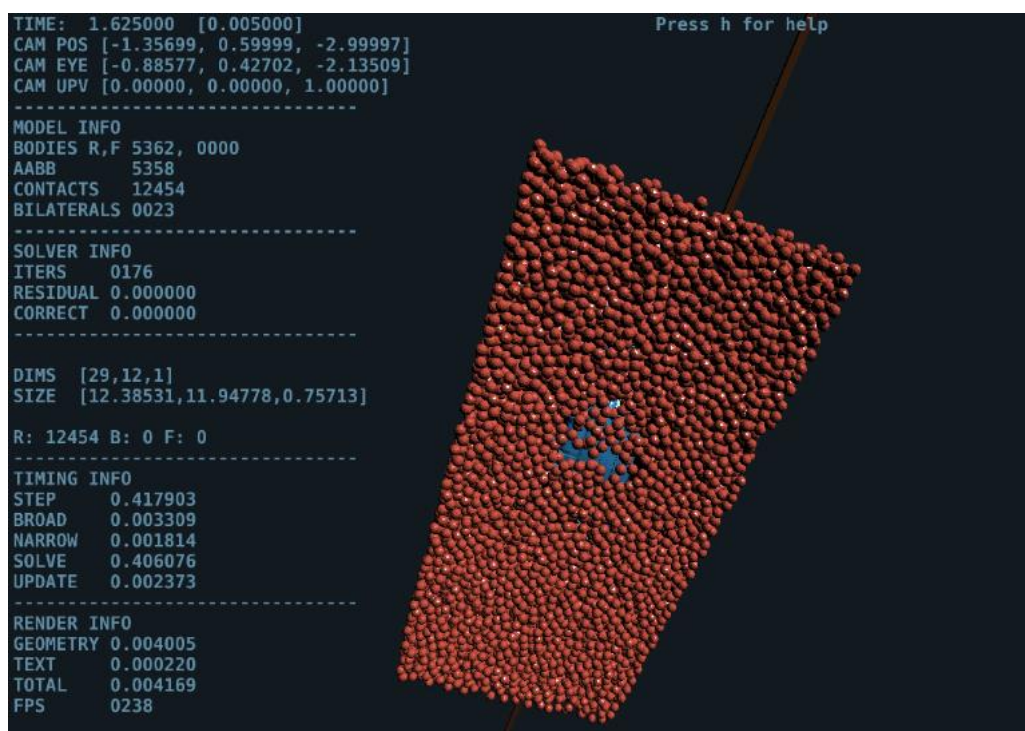


Figure 9: View from below the terrain in case of 6 particles layer.

To mitigate the scattering of terrain particles due the tire motion, which is particularly evident in configurations with lower particle density and smaller radii, some adjustments have been made. The main reason is that the reduced particle number beneath the tire could impact the realism of tire-soil interaction, potentially affecting key outputs like traction force and sinkage depth. To address this, the particle layer count was increased from 6 to 8 layers, offering a denser particle field to support the tire more effectively. Additionally, the simulation duration was extended from 3 to 6 seconds to allow sufficient time for the system to reach a steady state, moving away from initial disturbances and allowing more reliable measurements. These changes help to ensure that the interaction between the tire and terrain is more realistic, improving the simulation's accuracy.

The data extracted from the simulation are:

- Tire forces and moments;

- Centre of mass position of the wheel;
- Drawbar pull force;
- Longitudinal slip
- Wheel angular speed;

Tire force and torque are computed using the function “ReportTireForce()” which returns force and moment along the three directions. In particular, they are defined as follow:

- Tire force: The force exerted on the wheel spindle as a result of the interaction between the tire and the terrain.;
- Moment: it is the moment that is applied to the wheel spindle due to the tire-terrain interaction.

Thus, the force application point and the centre of mass of the wheel coincide.

The Drawbar pull force, instead, has been found using the function “GetDBP()” which returns the drawbar force as the reaction force generated by the linear motor which is in charge to maintain the specified longitudinal speed of the rig.

The longitudinal slip is characterised by the following formula:

$$s = \frac{\omega r}{|V|} - 1 \quad 15$$

So, for  $s < 0$  the simulation will perform the breaking condition, instead for  $s > 0$  the driving one. Different simulations can be performed keeping fixed the longitudinal speed of the carrier and varying the angular speed or the longitudinal slip, using the functions reported in figure 10.

```
// Scenario: prescribe all motion functions
rig.SetLongSpeedFunction(chrono_types::make_shared<ChFunction_Const>(2));
//rig.SetAngSpeedFunction(chrono_types::make_shared<ChFunction_Const>(30));
rig.SetConstantLongitudinalSlip(0.9);
```

Figure 10: Motion functions

In particular, imposing only the first motion’s function, the longitudinal speed of the carrier is set, letting the wheel free to rotate thanks to the revolute joint which links the spindle to the wheel, allowing the rotation around the y-axis. Moreover, also the angular speed of the wheel could be set using the second motion function. Thanks to the prismatic joint which links the ground body to the carrier, the displacement along the longitudinal axis is allowed. As third option, a constant longitudinal slip can be imposed as well as a constant speed through the function “rig.SetConstantLongitudinalSlip”. The longitudinal slip follows the formula 15. It’s worth to point out that, the third equation of motion has been applied in all simulations to ensure the correct value of a specific longitudinal slip. Finally, the camber angle and the side slip angle have been imposed equal to zero radians.

Since simulating a sufficiently long terrain composed of particles is computationally demanding, the GranularTerrain object in Chrono::Vehicle provides support for a "moving patch" approach. This technique extends the terrain by adding new sections of the surface as the wheel approaches the end of the current simulated area. Currently, the moving patch can only be extended in the forward x-direction.



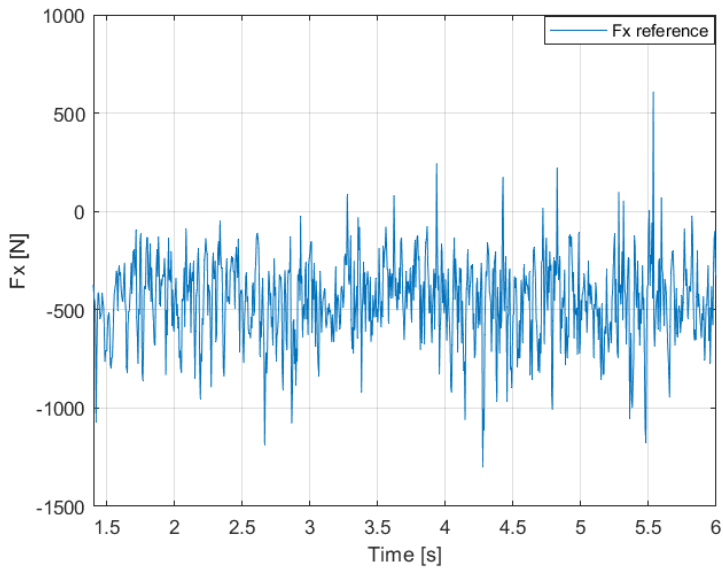
As final step, all these data have been saved in a file txt, called “TireForce\_Torque\_history”, during the simulation running.

## 6. Postprocessing

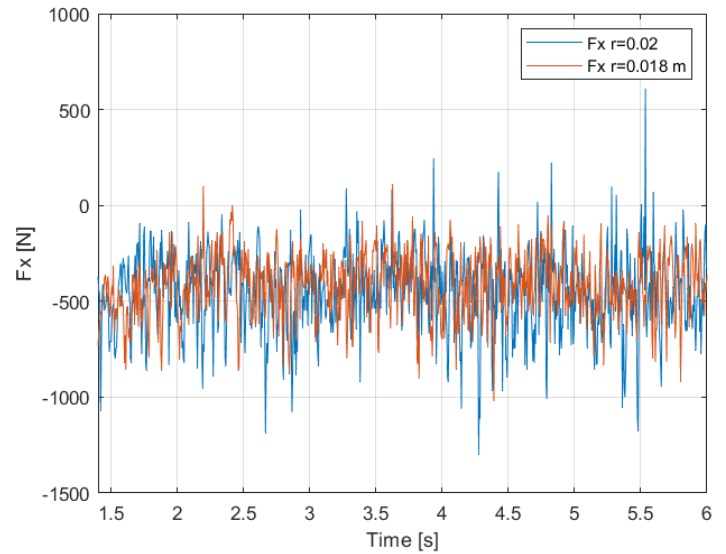
The postprocessing process has been carried out in MATLAB, importing the data saved in the file txt mentioned in paragraph 5, and then saved in MAT format, making possible to graphically spot the differences among several conditions. More details can be found in the Appendix.

## 7. Sensitivity analysis

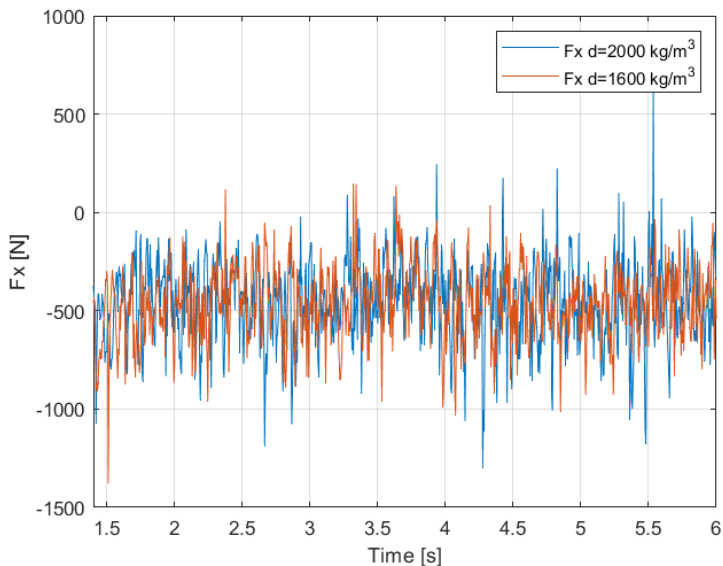
A sensitivity analysis, consisting of modifying a single parameter at a time, has been carried out and figures 11 and 12 illustrate the results of while keeping the wheel angular speed fixed at 5 rad/s and the carrier longitudinal speed at 2 m/s. This setup produces a longitudinal slip of 0.2, as calculated using the formula outlined in paragraph 4. This method allows for the isolation of each parameter's effect, clearly highlighting its specific influence on overall behaviour.



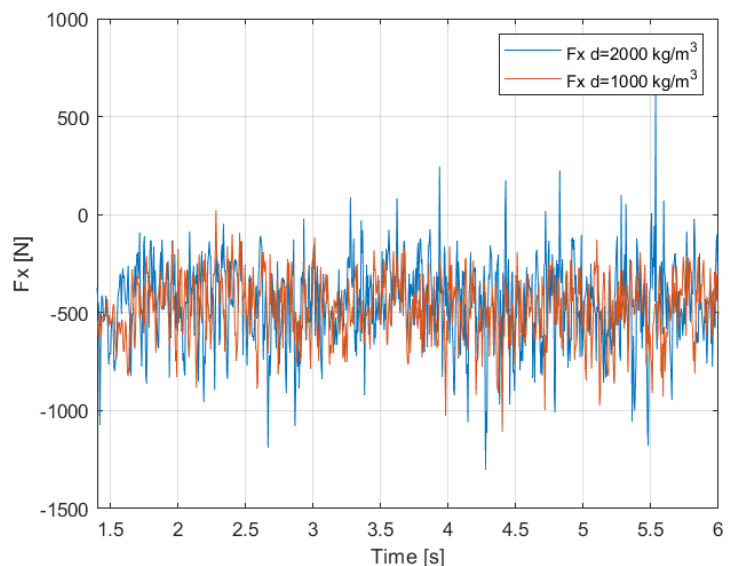
a)



b)



c)



d)

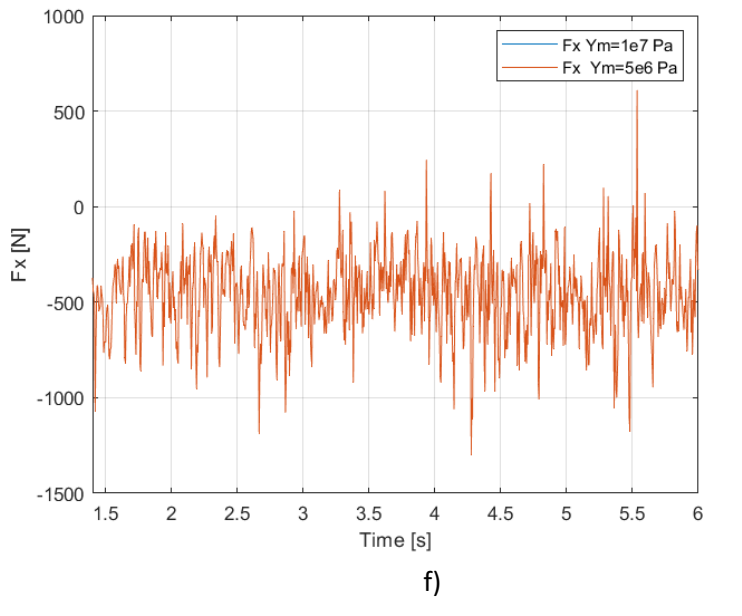
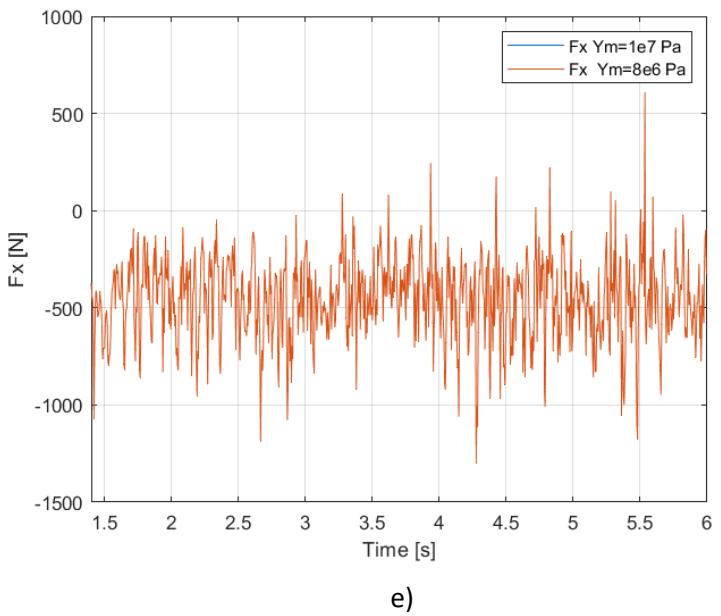
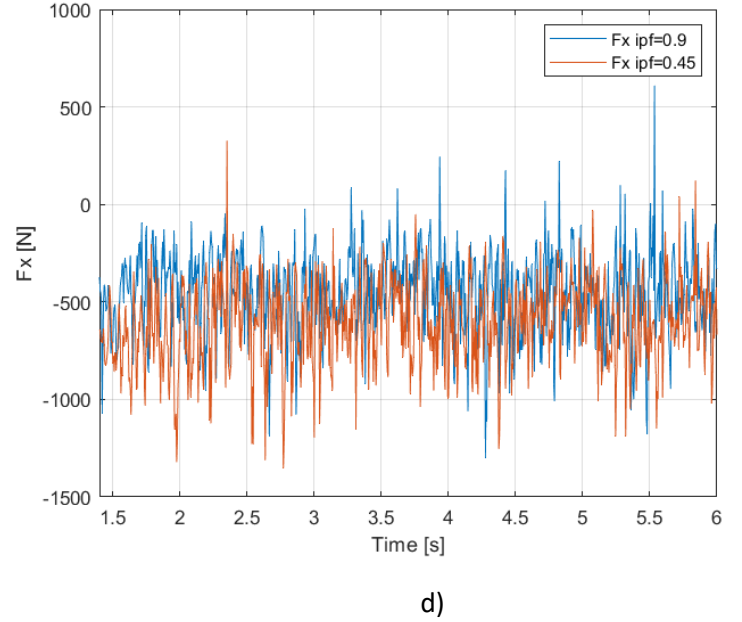
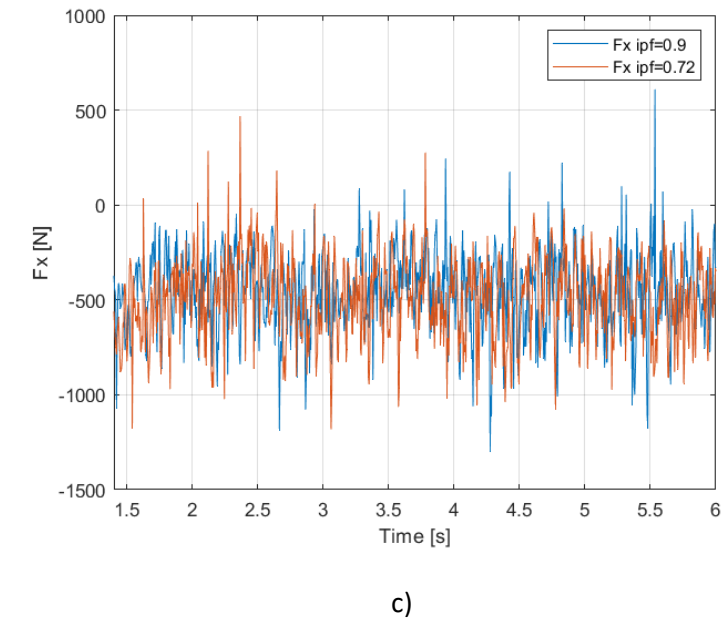
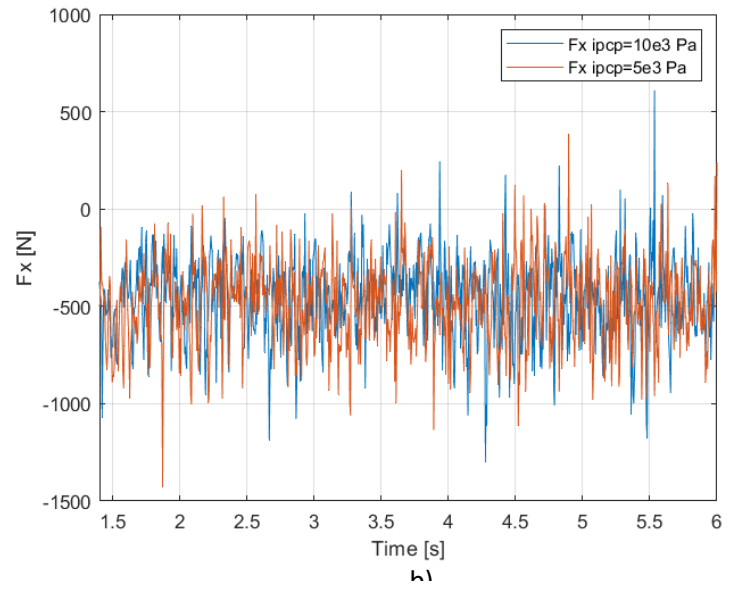
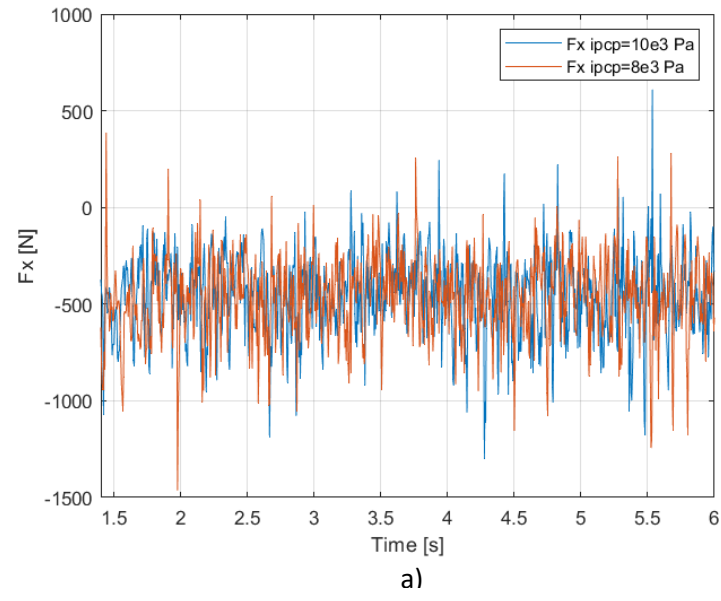


Figure 12:  $F_x$  analysis on a terrain with: a)  $ipcp$  reduction of 20% b)  $ipcp$  reduction of 50% c)  $ipf$  reduction of 20%,  $ipf$  reduction of 50% d)  $Y_m$  reduction of 20% e)  $Y_m$  reduction of 50%

Each plot shows curves with presenting irregular shapes, resulting in numerous spikes caused by soil particle interactions. The behaviour is noisy across all cases, making it difficult to spot the true impact of each modification but providing a starting point for more refined analyses. To address this, the average  $F_x$  was calculated for each case, as illustrated in Fig. 13, and compared in a bar plot. A detailed explanation is provided in the following points:

- **Radius Reduction:** it appears to be the most significant modification, as it's the only one that has increased the mean  $F_x$  value in absolute terms. This suggests that using smaller particles would yield more accurate measurements, making it possible to simulate soft soils, like sand, with greater real-world accuracy.
- **Density Reduction:** This adjustment didn't show substantial improvements and instead led to a slight decline in the mean  $F_x$  value.
- **Inter Particle Friction (IPF) Reduction:** While a minor reduction in IPF results in only a slight decrease in performance, a significant reduction (50% less) causes a considerable drop in the mean  $F_x$  value. This behaviour aligns with the findings in [9]. The inter particle friction refers to the resistance that develops between the surfaces of solid particles when they interact each other under certain loads. When inter-particle friction is high, particles tend to resist sliding over more effectively, leading to the formation of more stable structures. At the contrary, lower inter-particle friction allows for greater particle mobility, resulting in a less stable and more deformable structure. This might explain the  $F_x$  behaviour;
- **Inter Particle Cohesion Pressure (IPCP):** it seems to have a low impact on  $F_x$  like density reduction case.
- **Young's modulus ( $Y_m$ ):** As shown in Figures 12.e and 12.f, the two curves overlap completely, which is further confirmed by identical  $F_x$  mean values. Young's modulus indicates how much a material will stretch or compress under an applied force. Since the NSC system operates with rigid bodies, this could explain why this factor has no impact on  $F_x$ .

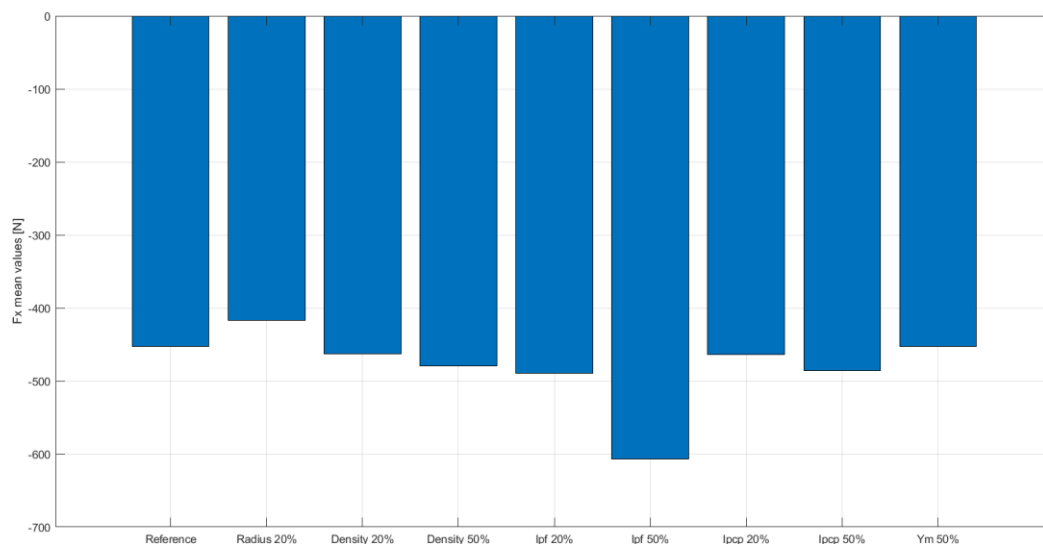


Figure 13:  $F_x$  mean values according to different soil parameters.

Fx mean reference	-452.66 N
Fx mean radius (20%)	-417.04 N
Fx mean density (20%)	-462.96 N
Fx mean density (50%)	-479.16 N
Fx mean ipf (20%)	-489.41 N
Fx mean ipf (50%)	-606.57 N
Fx mean ipcp (20%)	-464.02 N
Fx mean ipcp (50%)	-485.52 N
Fx mean Ym (20%)	-452.66 N
Fx mean Ym (50%)	-452.66 N

Table 3: Fx mean values

Moreover, it is worth noticing that measurements are not taken from time zero due to two factors:

- The simulation begins with the drop of the tire, and it takes some time before the tire settles and assessments can be made.
- The function 'rig.SetTimeDelay()' introduces a delay of 1 second before the motion function is applied. However, the speed follows a ramp function, reaching a steady value of 2 m/s after 1.3 seconds. Consequently, measurements have been taken starting from that point.

In summary, it appears that as the soil becomes softer, the magnitude of Fx decreases. To gain a more comprehensive understanding of this behaviour, the most affecting characteristics identified in fig 13 (i.e. particle radius, density, inter particle friction and inter particle cohesion pressure) were analysed across varying longitudinal slip conditions. The resulting plots are shown in Figures 14, 15, 16, and 17.

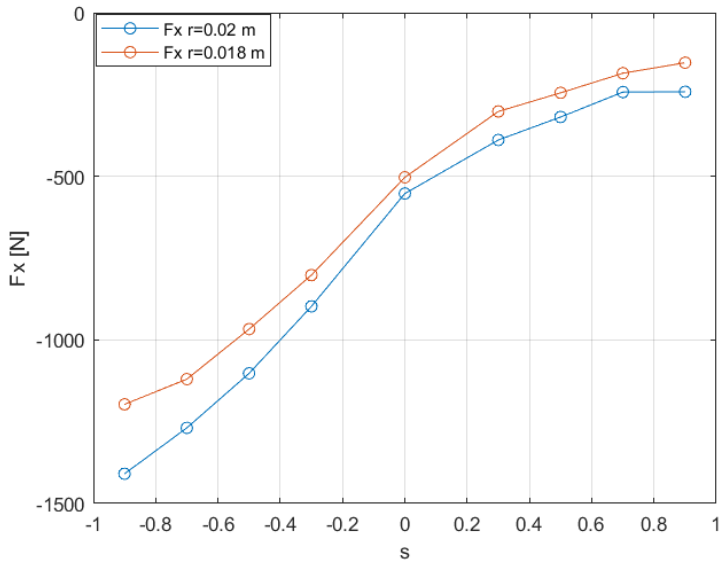


Figure 15: Fx as function of particle radius reduction.

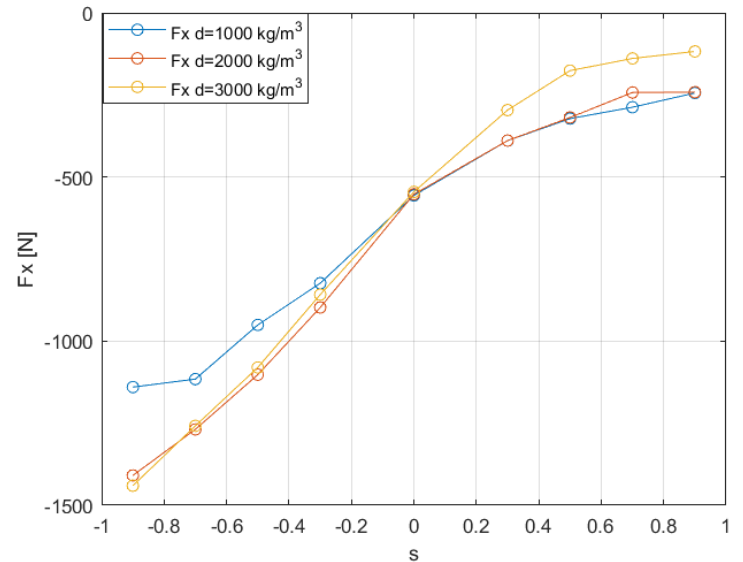


Figure 14: Fx as function of density variation.

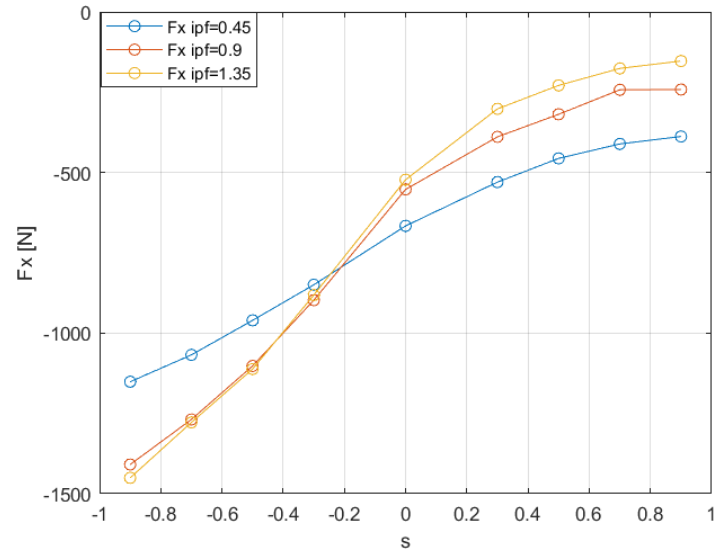


Figure 16:  $F_x$  as function of  $ipf$  variation.

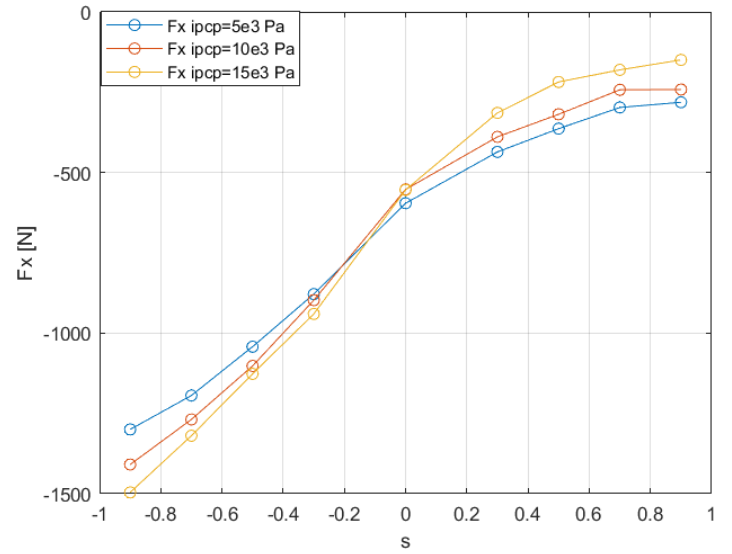


Figure 17:  $F_x$  as function of  $ipcp$  variation.

The general behaviour of  $F_x$  meets expectations as it increases with slip, but the curve seems to be shifted downward for negative values. The parameter that appears to influence  $F_x$  most significantly at high slip values seems to be the inter-particle cohesion pressure. Given that the wheel is completely rigid, the results obtained appear somewhat unusual. The tire force is negative for both negative and positive values of longitudinal slip, indicating that a braking force is generated even in traction cases. Additionally, when the slip is zero, the tire force is not zero as expected. While some computational errors may arise from approximations made by MATLAB during post-processing, this alone is not sufficient to explain this behavior.

Soil parameters	Soft terrain
Particle radius [m]	0.018
Number of layers	8
Density [kg/m <sup>3</sup> ]	3000
Inter-particle friction	0.9
Inter-particle cohesion pressure [Pa]	15e3
Young Modulus [Pa]	1e7

Table 4: Soil parameters

Using sand as a reference for soft soil, a new type of soil has been defined based on the plots, with its parameters reported in Table 2. Finally, by compiling all the results, the behaviour of  $F_x$  was compared across three different terrains, as shown in Figure 16:

- Soil parameters shown in table 1, defined as “Reference”;
- Soil parameters shown in table 4;
- Soil parameters shown in table 6, but  $ipcp$  has been modified according to the previous assumption.

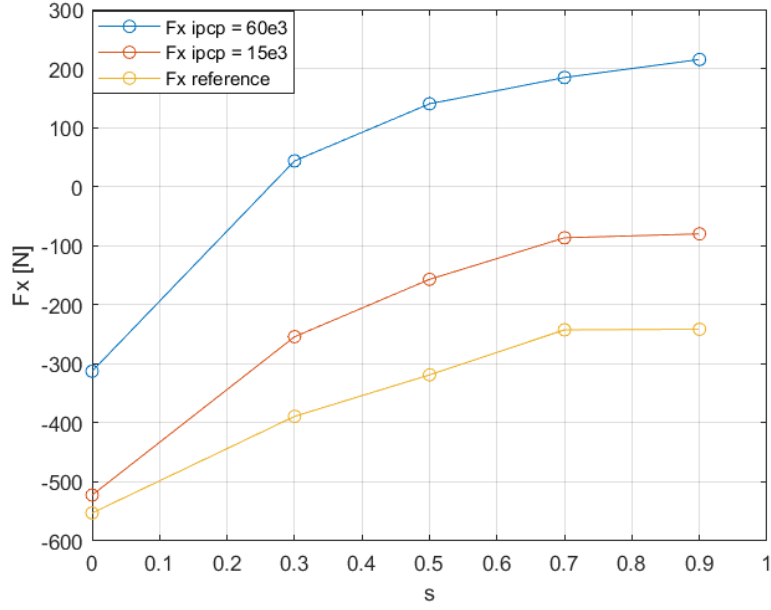


Figure 18: comparison among three different terrains.

s	Fx reference [N]	Fx ipcp 15e3 [N]	Fx ipcp 60e3 [N]
0	-552.6	-522.3	-312.7
0.3	-389.1	-254.1	43.6
0.5	-318.7	-157.1	140.2
0.7	-242.5	-86.6	184.8
0.9	-241.6	-80.1	215.3

Table 5:  $F_x$  mean values for positive values of longitudinal slip

Even if the terrain properties do not represent a proper sand soil due to the large particle radius, the result obtained seems to be more consistent, as also reported in table 5. For sake of simplicity there are reported only  $F_x$  values for positive values of long slip. Another possible limitation is that Chrono:Multicore does not allow to create a terrain with different particle dimensions as in real sand soil. So, possible improvements could be simulated with smaller particle, despite of the larger computational effort. In table 6 is reported the terrain with the ipcp value modified.

Soil parameters	Soft terrain
Particle radius [m]	0.018
Number of layers	8
Density [kg/m3]	3000
Inter-particle friction	0.9
Inter-particle cohesion pressure [Pa]	60e3
Young Modulus [Pa]	1e7

Table 6: ipcp parameter modified.

## 7.2 Torque analysis

Through the function “rig.ReportTireForce()”, it is possible to compute the moment applied to the tire, as mentioned in paragraph 5. Being a driven wheel and no external motor which applies a torque to the wheel, the rotational equilibrium should be as follow:

$$F_x * r = I * \dot{\omega}^2 \leftrightarrow T(F_x) = T_y \quad 9$$

Being  $F_x$  and the tire radius already available from the measurements, it is reported in figure 19 the moment generated by the tyre indicated as  $T(F_x)$  and the moment obtained through the function mentioned before. Their mean values are reported in table 7 and the terrain properties are the same of table 6.

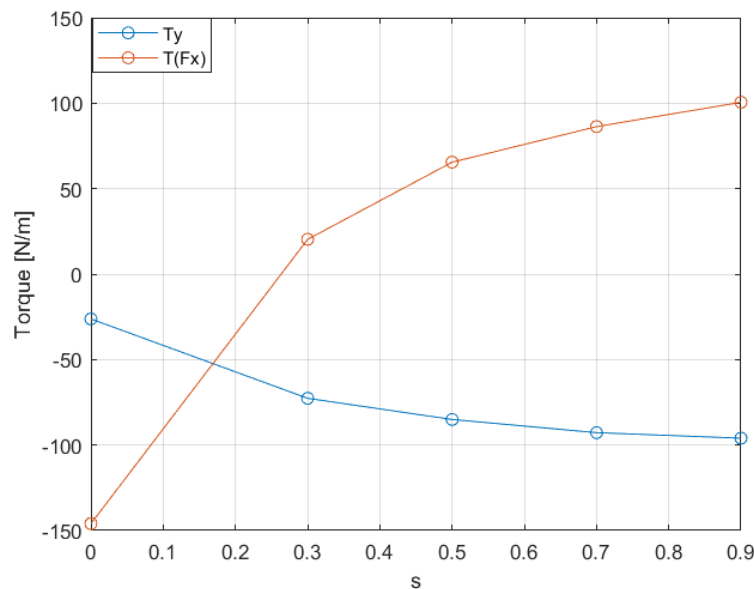


Figure 19: Torque as function of long slip.

$T_y$  should be at least equal and opposite to  $T(F_x)$ , but this is not the case. A possible explanation could be the inaccurate representation of the terrain parameters, as previously mentioned in paragraph 6.1. However, for high longitudinal slip values, the two forces come very close to each other in magnitude.

Slip	$T_y$ [Nm]	$T(F_x)$ [Nm]
0	-26.2	-146.1
0.3	-72.6	20.3
0.5	-85.1	65.5
0.7	-92.7	86.3
0.9	-95.9	100.5

Table 7: slip,  $T_y$  and  $T(F_x)$

## 7.3 Rigid Terrain

To the same test rig it is possible to change the soft terrain in rigid terrain. It has been created through the function “rig.SetTerrainRigid” which takes as inputs the parameters shown in table 8. These values were already implemented in the code and have not been changed.

Friction coefficient	0.8
Coefficient of restitution	2e7
Contact material Young modulus	0
Length of terrain patch [m]	10

Table 8: Rigid terrain parameters

This kind of terrain represents a classic Coulomb model in which there is a single contact point and the tire force  $F_x$  follows the formula:

$$F_x = \mu * F_z$$

16

In figure 20 it is shown the behaviour of the tire force as function of the longitudinal slip. As reported in table 9, the positive and negative values are almost equal and opposite in sign, with a  $F_x$  close to zero in correspondence of zero slip.

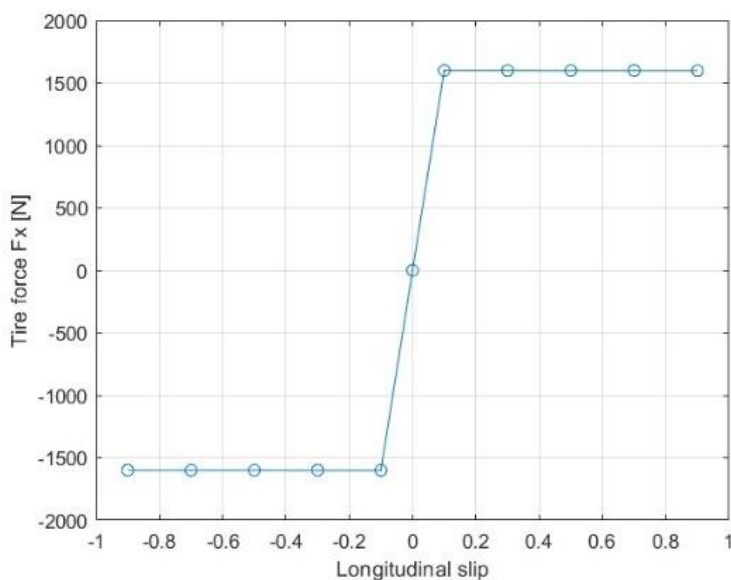


Figure 20:  $F_x$  as function of longitudinal slip.

Slip	$F_x$ mean
-0.9	-1599
-0.7	-1598
-0.5	-1599
-0.3	-1599
-0.1	-1600
0	-6.81e-06
0.1	1600
0.3	1599
0.5	1599
0.7	1598
0.9	1597

Table 9:  $F_x$  mean values.

As expected,  $F_x$  shows a step-like behavior, showing negative values for negative longitudinal slip and positive values for positive slip.



Figure 21 shows the torque  $T_y$  and  $T(F_x)$ , as previously discussed in paragraph 6.2, plotted against constant longitudinal slip. The various  $T_y$  curves overlap, as do the  $T(F_x)$  curves. Their mean values, presented in Table 9, are very close to each other but opposite in sign.

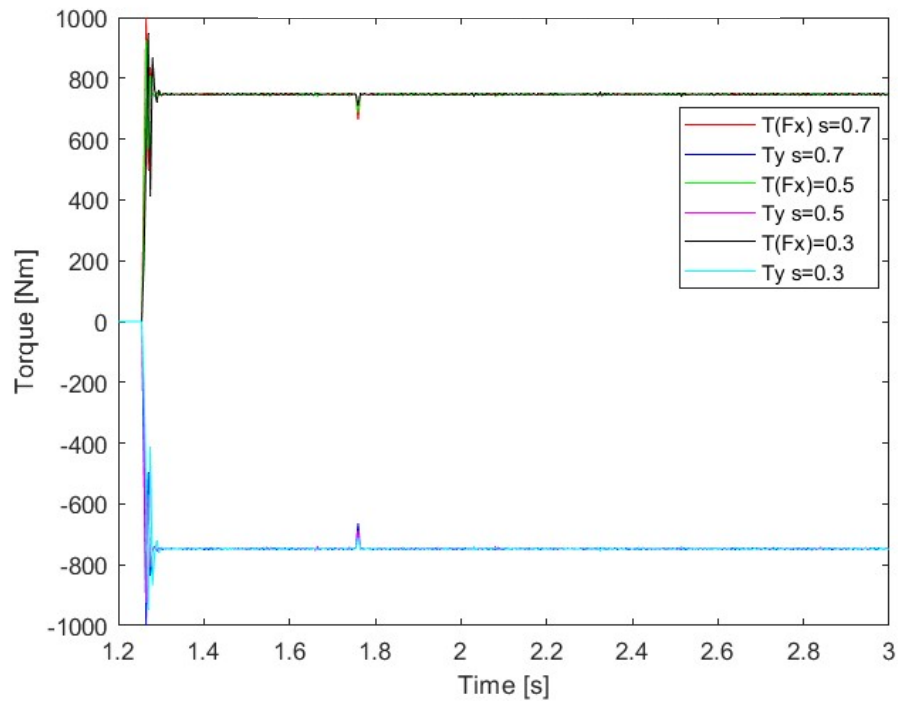


Figure 21: Torque behaviour considering a rigid terrain

The assessment time changes with respect to the granular terrain from 1 s to 1.3 s. For this reason, there is a noisy behaviour close to that time. After this transitory time, the two torques are equal and opposite, as expected. In table 10 are reported their mean values.

Slip	$T_y$	$T(F_x)$
0.3	-433.83 Nm	433.83 Nm
0.5	-434.46 Nm	434.46 Nm
0.7	-434.68 Nm	434.68 Nm

Table 10: slip,  $T_y$  and  $T(F_x)$  mean values.

## 8. SCM terrain model

This work introduces a deformable soil model capable of real-time simulation, based on the Soil Contact Model (SCM) originally developed by the DLR (Deutsches Zentrum für Luft und Raumfahrt - German Aerospace Centre). SCM generalizes the semi-empirical Bekker-Wong and Janosi-Hanamoto soil interaction models to handle arbitrary 3D shapes and contact patches. This implementation is available in Chrono, with the opportunity to perform simulations on more computationally intensive deformable soil models. The original Bekker model has been expanded by Wong and Reece [4] to incorporate realistic shear effects, which are crucial for estimating forces exchanged between tyre and ground. A practical shear model developed by Janosi-Hanamoto [2] is also utilized in this project. This shear model is based on the Mohr-Coulomb failure criterion, with the contribution of an additional parameter  $k$  known as the Janosi-Hanamoto shear modulus. The SCM in Chrono builds upon DLR's model including the Janosi-Hanamoto formulation, which relates normal pressure  $p$  to sinkage  $z$  for a wheel of width  $b$ , using parameters  $k_c, k_\phi$ , and  $n$  derived from experimental calibration. SCM considers the soil as a height-map, thus it represents terrain with a regular Cartesian grid, whose deformation is modelled through vertical displacement of grid nodes. This semi-empirical approach, effectively generalizes the Bekker-Wong model, providing a relationship between pressure and vertical deformation of the soil as:

$$p = \left( \frac{k_c}{b} + k_\phi \right) * y^n \quad 13$$

Where:

- $p$  is the contact patch pressure;
- $y$  is wheel sinkage
- $k_c$  is an empirical coefficient representing the cohesive effect of the soil;
- $k_\phi$  is an empirical coefficient representing the stiffness of the soil;
- $n$  is an exponent expressing the hardening effect, which increases with the compaction of the soil in a non-linear fashion;
- $b$  is the length of the shorter side of the rectangular contact footprint (since the original Bekker theory assumes a cylindrical tire rolling over flat terrain).

In the original Bekker model, the term  $b$  refers to a simple measurement across a wheel or track's contact patch on the ground, typically the width of the wheel or track. However, in more complex contact shapes, like irregular footprints seen with arbitrary 3D surfaces, the length  $b$  isn't straightforward to define. To handle this, the model uses a method called a flooding algorithm to identify all areas where the vehicle's surface is in contact with the terrain. The flooding algorithm groups connected contact points, essentially "filling" the contact area to find the complete boundary of the contact footprint. Once this connected area is identified, an effective length  $b$  is then estimated based on the shape and size of this footprint, giving a more accurate representation for complex, non-uniform contact areas.

$$b \approx \frac{2A}{L} \quad 14$$

where  $A$  is the area of such a contact patch and  $L$  its perimeter. Fig 22 illustrates an example of how the tire footprint is discretized according to the types of nodes. There are the contact nodes (located inside the footprint, shown in red) and contour nodes (located along the footprint boundary, shown in blue) for an arbitrary footprint shape [12].

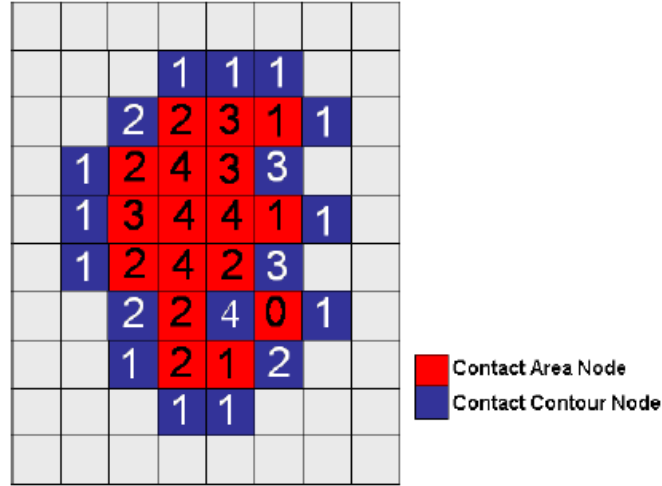


Figure 22: Footprint weightin factors [12]

The weighting factors  $c_{A,Contact}$  and  $c_{A,Contour}$  account for the specific location type of the grid nodes within the footprint and along its contour. The classification of node types is determined by the number of contact nodes in their immediate neighborhood.

$$A = \left( \sum_{i=1}^n c_{A,Contact,i} + \sum_{j=1}^n c_{A,Contour,j} \right) dA \quad (14.1)$$

The same principle applies to pressure distribution within a footprint, which is generally non-uniform, even for flat ones. Specifically, pressure decreases from the central regions toward the edges of the footprint, scaled by a weighting factor  $\gamma$  that is applied to the contact pressure. In Chrono are implemented formulas (14.1-15) which account for damping and the weighting factors.

$$\sigma = \gamma_i \left( \frac{k_c}{b} + k_\varphi \right) z_i^n + D_{soil} v_{\vartheta,i} \quad (15)$$

Where:

- $v_{\vartheta,i}$ : contact velocity normal to the footprint surface;
- $D_{soil}$ : damping parameter.

Fig 23 provides an overview of the tire footprint, illustrating grid deflection as well as the direction of the pressure and shear stress vectors.

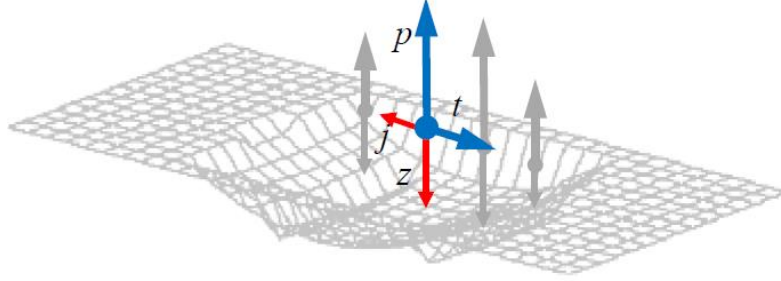


Figure 23: SCM terrain modellization

The Bekker-Wong formula is enhanced by incorporating shear stress calculations based on the Janosi-Hanamoto method:

$$\tau = \tau_{max} \left( 1 - e^{-\frac{j}{k}} \right) \quad 15a$$

$$\tau_{max} = c + p * \tan(\varphi) \quad 15b$$

Where:

- $j$  is the accumulated shear;
- $c$  cohesion coefficient;
- $\varphi$  internal friction angle;
- $k$  Janosi parameter.

Equations 15a and 15b can be used together to enable the application of normal and tangential contact forces on the impacting object, with soil deformation tracked exclusively along the normal direction defined by the SCM. Once the shear and normal stresses are computed, the discrete contact force  $\Delta F_i$  can be calculated by:

$$\Delta F_i = (\sigma_i \mathbf{n}_i + \tau_i \mathbf{t}_i) \Delta A \quad 16$$

Where  $\mathbf{t}_i = \frac{\mathbf{v}_{\tau,i}}{|\mathbf{v}_{\tau,i}|}$  denoting the local tangent vector,  $\mathbf{n}_i$  the local normal vector and  $\Delta A$  the DEM grid size. Finally, the total contact force  $\mathbf{F}$  and torque  $\mathbf{T}$  vectors applied to the contact body are obtained by integrals of  $\Delta F_i$  over all  $N$  contact nodes as illustrated in eq 17:

$$\mathbf{F} = \sum_{i=1}^N \Delta F_i \text{ and } \mathbf{T} = \sum_{i=1}^N (\mathbf{r}_i \times \Delta F_i) \quad 17$$

where  $\mathbf{r}_i$  is the distance vector from the selected reference frame to the contact node  $i$  [12].

In Chrono could be implemented some other features, such as:

- the initial undeformed grid mesh as:
  - a regular tiled mesh (filling a flat rectangle). it's a grid made up of repeating, regularly shaped elements (like squares or rectangles) arranged to cover an entire rectangular surface
  - a height-map (provided as a gray-scale BMP image). Different shades of grey represent different heights when the image is processed, so this method can be used to generate a more complex, uneven surface.
  - In a programmatic way. The mesh is created directly using code, which allows for a high degree of customization. This method can be used to create unique or complex shapes by specifying the position, shape, and size of each element in the grid through programming.
- support for arbitrary orientation of the terrain reference plane; by default, the terrain is defined as the (x,y) plane of a z-up ISO frame;
- support for specifying location-dependent soil parameters; this can be achieved by providing a custom callback class which implements a method that returns all soil parameters at a given (x,y) point specified in the terrain's reference plane.

The SCM model relies on the semi-empirical Bekker formulas, which introduce some limitations. It is unable to simulate soil flow beneath the tyre and relies on experimentally measured data, which are difficult to acquire due to specialized equipment required, and can vary significantly. To enhance realism, Chrono's SCM module includes an optional bulldozing feature, specifically designed for soft soils like sand. This mechanism conserves displaced soil volume around contact areas by redistributing it to nodes along the perimeter of the contact patch rather than removing it from the simulation domain. At each simulation step, the vertically displaced soil is calculated and added to these perimeter nodes. To ensure a natural spread, the accumulated soil volume undergoes multiple iterations of an erosion algorithm. The SCM model employs an isochoric operator on a triangular mesh allowing for flexible grid topologies and ensuring realistic vertical displacements in nodes surrounding the contact area without affecting those directly beneath the tire. Additionally, users can extend the erosion boundary beyond the contact patch, diffusing displaced soil into the surrounding region. This diffusion operation represents the only part of the bulldozing algorithm where parallelization could potentially be implemented using standard Laplace equation techniques. However, parallelization is not currently applied, as only a few smoothing iterations are typically sufficient. Future versions of Chrono's SCM module may incorporate this feature to further optimize performance.

## 7.1 Test scenario

The model under consideration employs a regular grid mesh and a purely plastic approach to soil behaviour, emphasizing deformation characteristics. Since terrain interaction is managed through the Chrono contact system, it supports both rigid and finite element tire models, but it is analysed only rigid tyre. In order to study how change the tire-ground interaction with respect to the granular terrain, the demo “demo\_VEH\_SCM\_TireTestRig” has been chosen to analyse. The class “SCMTerrain” implements a deformable terrain based on the Soil Contact Model. Unlike rigid terrain, the vertical coordinates of this terrain mesh can be deformed due to interaction with ground vehicles or other collision shapes. The system implemented in this demo is a SMC system in which the contacts are modelled using a smooth (penalty-based) method already explained in paragraph 1. The test scenario is made up of the same components of the multicore case, as it could be noticed in fig 26.

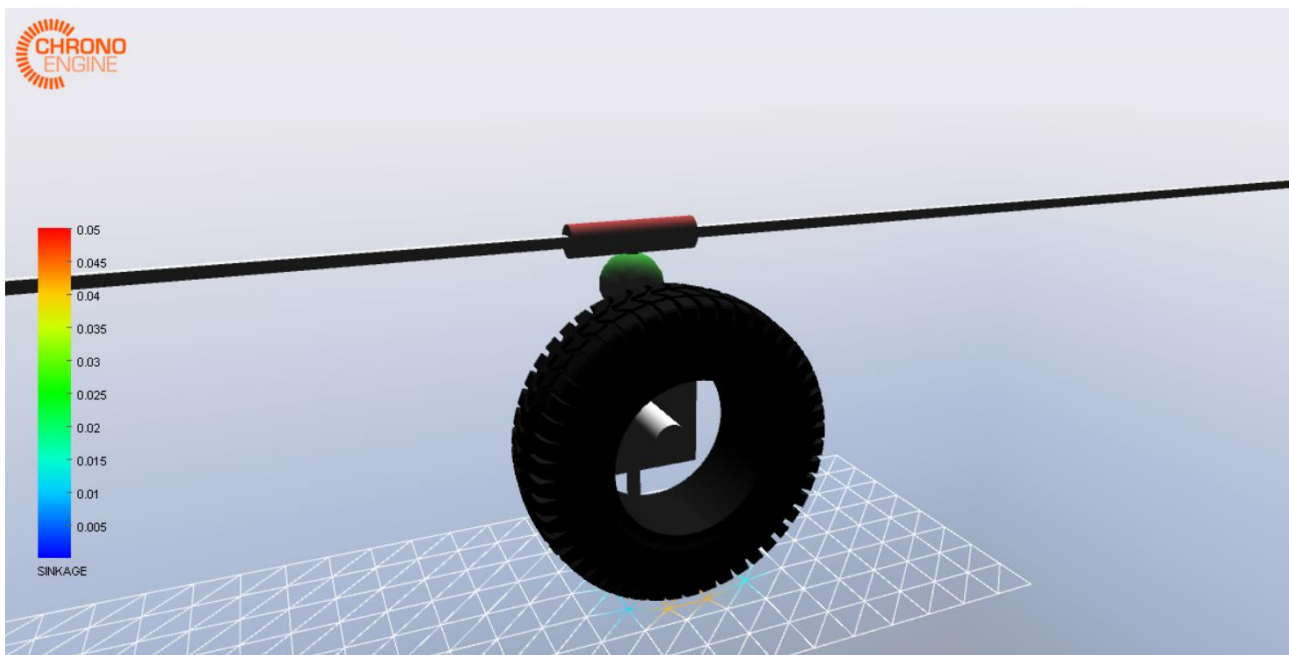


Figure 24: SCM tire test rig

Through the function “rig.SetTerrainSCM ()” it is possible to characterized the SCM terrain using the parameters reported in table 11.

SCM parameters	Baseline values
Bekker $k_\phi$	2e6
Bekker $K_c$	0
Bekker $n$ exponent	1.1
Mohr cohesive limit (Pa)	0
Mohr friction limit (degrees)	30
Janosi shear coefficient (m)	0.01
Terrain length (m)	20

Table 11: SCM parameters

Frictional resistance in the model is defined through the internal friction angle, an input parameter, along with cohesion, which represents the attractive forces between soil particles. These parameters establish the soil's shear strength, allowing the Bekker-Wong model to incorporate shear stresses effectively as already explained in chapter 7. Moreover, the mesh resolution is of 0.125 m and the terrain is 20 meters length and 1 meter width.

## 7.2 Simulation processing

The simulation lasts a total of six seconds, comprising 1200 iterations with a fixed time step of 0.005 s. At the beginning, the tire is dropped onto the terrain and settles for a fixed interval: no forward motion takes place for the first second, giving the system time to stabilize before the wheel starts moving. Like the multicore test rig setup, the carrier's speed increases gradually (following a ramp profile) and reaches a steady 2 m/s by around 1.4 s. In this particular case, a longitudinal slip of 0.2 is imposed on the wheel to approximate a realistic driving condition. Fig 27 depicts the system at this steady-running point, with the tire in motion

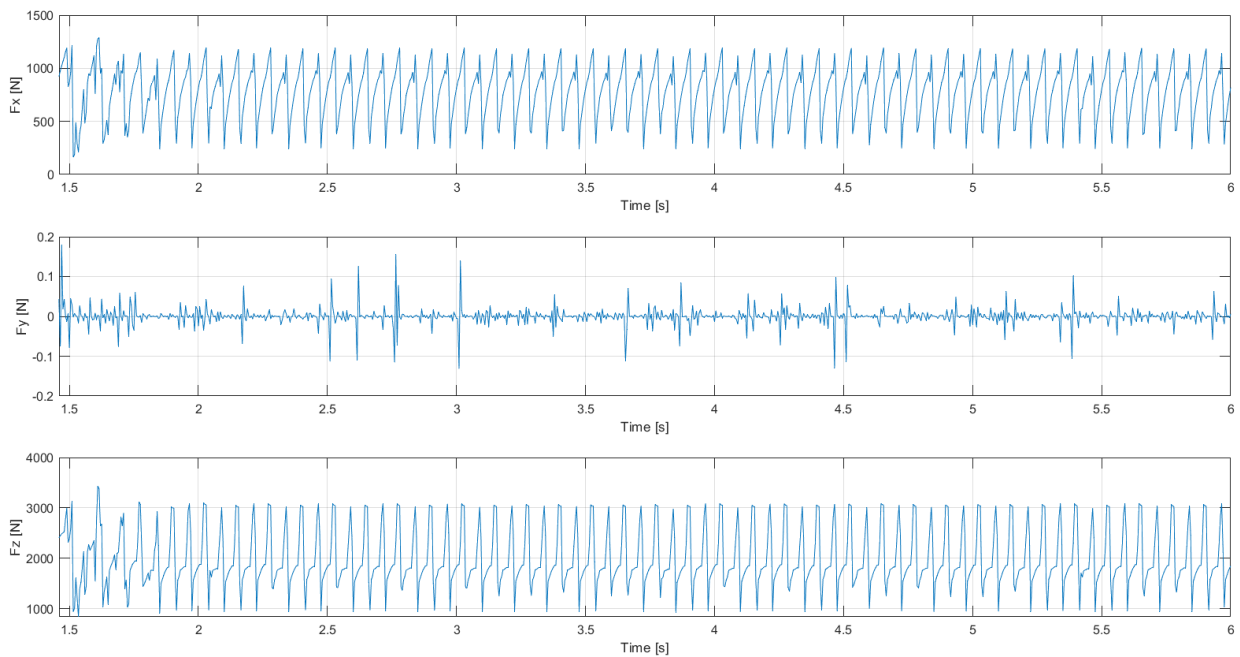


Figure 25: Tire forces behaviour

under those specified speed and slip parameters.

Fx mean	780.64 N
Fy mean	-0.00016 N
Fz mean	1997 N

Table 12: Mean force values

Compared to the granular case, the behaviour of  $F_x$  appears more stable and realistic, maintaining positive values, as reflected in the mean value reported in Table 12. The lateral force  $F_y$  exhibits greater variability but remains close to zero on average, while the vertical force  $F_z$  has a mean value slightly below the nominal 2000 N. The model, classified as a penalty-based approach, computes contact forces based on small interpenetrations or sinkage between the wheel and the soil. Even though the wheel is modelled as a rigid cylinder without treads, its contact patch continuously come across to different terrain cells as it moves forward. Each soil cell keeps in mind its own deformation history, meaning that some regions may already be compacted while others remain relatively undeformed. As the wheel advances, the solver reassigns the load to the new engaged cells, causing fluctuations in both the vertical force  $F_z$  and the longitudinal force  $F_x$ . Areas in which there is a more concentrated load distribution result in deeper sinkage, leading to sharper force peaks, whereas transitions to less compacted regions may cause temporary reductions in force. The magnitude and frequency of these fluctuations depend on several factors, including mesh resolution, time step, and soil properties such as stiffness and cohesion. A finer mesh may result in smoother force variations, while coarser discretization or increased stiffness tends to amplify these oscillations. Ultimately, this force behaviour, in both normal and tractive components, come out from the discretized nature of the SCM terrain, where each soil cell deforms independently. Consequently, during the wheel motion, the continuous redistribution of load leads to dynamic variations in force.

In fig 29 it is analysed only the tire force  $F_x$  in function of longitudinal slip for different  $F_z$  values. Each curve corresponds to a specific normal load value on the wheel, 2000 N, 3000 N, and 4000 N, and reveals how both braking and driving forces increase in magnitude under higher load. In the negative slip region, the wheel is braking, and a greater vertical load leads to larger negative forces because the soil is more heavily compressed and offers stronger resistance. At the contrary, in the positive slip region, the wheel is driving, and heavier loads allow the wheel to generate higher tractive forces, as illustrated by the upward shift of the 3000 N and 4000 N curves relative to the 2000 N curve.

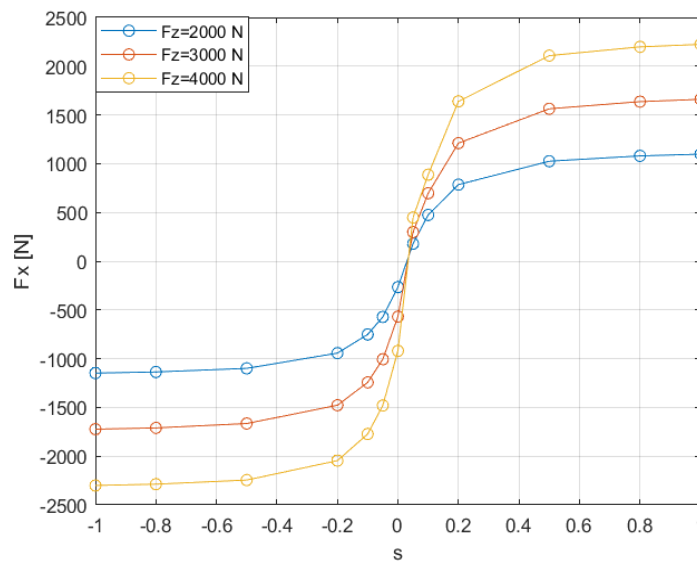


Figure 26:  $F_x$  behaviour as function of vertical load increase



Around  $s = 0$ , where there is no relative sliding of the wheel against the terrain, the longitudinal force does not return to zero but remains slightly negative. This behavior arises from the internal stresses in the soil, which create a rolling resistance that opposes motion even at zero slip, and it becomes more pronounced as the normal load increases because deeper compression leads to higher residual stresses. As slip moves from negative to positive, each curve crosses from a braking regime to a driving regime, saturating at higher slip where additional rotation does not significantly improve tractive force. In table 13 are reported the  $F_x$  mean values as function of longitudinal slip and vertical load variations.

$s$	$F_z = 2000 \text{ N}$	$F_z = 3000 \text{ N}$	$F_z = 4000 \text{ N}$
-1	-1146.9	-1721.41	-2299.31
-0.8	-1135.3	-1708.3	-2285.58
-0.5	-1098.4	-1663.3	-2241.35
-0.2	-941.7	-1474.09	-2037.21
-0.1	-751.8	-1240.8	-1764.7
-0.05	-573.7	-1005.1	-1476.8
0	-272.9	-574.87	-927.41
0.05	176.79	293.09	436.56
0.1	466.01	682.13	869.64
0.2	780.64	1201.14	1622.24
0.5	1024.98	1561.24	2104.53
0.8	1079.92	1636.34	2196.92
1	1097.84	1660.22	2223.47

Table 13:  $F_x$  mean values in function of  $F_z$  increase.

The results obtained seems consistent with respect to real world application, showing that  $F_x$  increases as the load  $F_z$  increases.

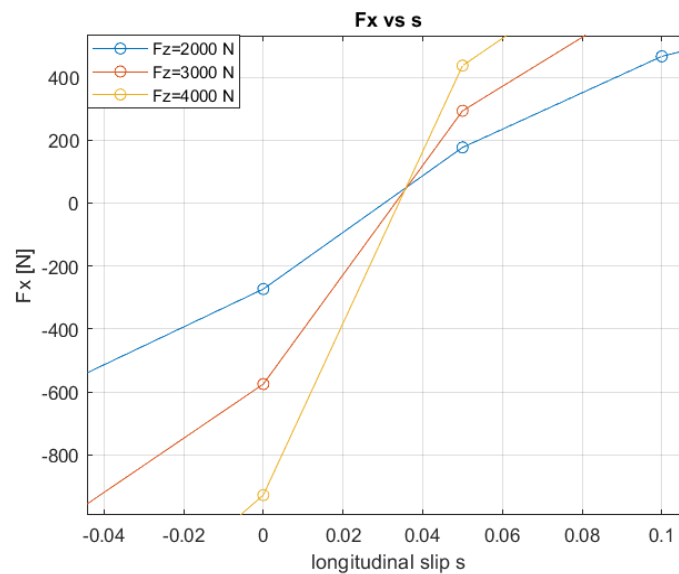


Figure 27: Close-up of the behaviour at zero slip

The plot examines the behaviour of  $F_x$  near zero slip, showing that instead of being zero, the longitudinal force takes a negative value. Specifically, there is an angular asymmetry between the entry and exit points of the contact area. As the wheel interacts with the deformable terrain, the forces generated at the leading edge (where the wheel first get in contact with the ground) and the trailing edge (where it leaves the ground) are not evenly distributed. This imbalance results in a net tangential force opposing motion, even in the absence of slip. Unlike rigid surfaces, where zero slip would correspond to null longitudinal force, deformable soil introduces internal resistance due to plastic deformation. This effect intensifies, as the normal load increases, leading to a deeper compressed region the wheel and further amplifying the opposing force. This explains why the magnitude of negative  $F_x$  grows with higher loads. Additionally, a small amount of slip is needed before the wheel can generate positive traction, as the terrain must undergo additional deformation to provide the necessary resistance for forward motion. Even when rolling without apparent slip, energy is continuously lost due to permanent changes in the soil's structure, which translates into a net resistance force acting against the wheel.

## 8. Sensitivity analysis

A sensitivity analysis of crucial SCM parameters has been performed in order to evaluate their impact on wheel-terrain interaction. Varying parameters such as the Bekker stiffness modulus  $k_\phi$ , the exponent  $n$ , and the internal friction angle  $\phi$ , the analysis explores how changes in terrain properties influence tire sinkage and longitudinal force  $F_x$ .

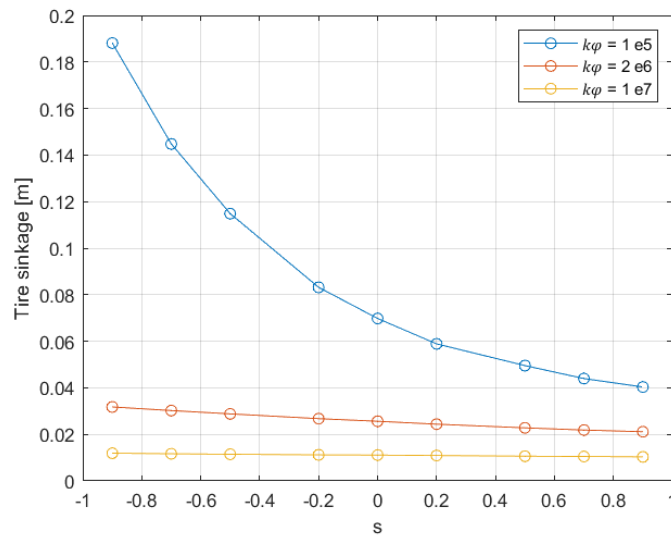


Figure 28: Tire sinkage as function of different  $k_\phi$

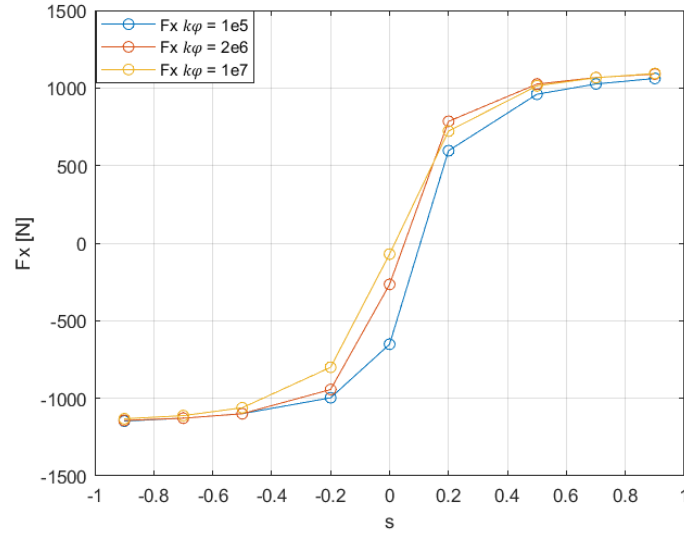


Figure 29: Longitudinal force  $F_x$  as function of different  $k_\varphi$

As shown in fig 28, a lower  $k_\varphi$  value corresponds to softer terrain that lead to larger sinkage providing more resistance to wheel motion, while a higher  $k_\varphi$  results in a more compact surface that minimizes sinkage. The correlation between sinkage and  $F_x$  varies with slip conditions as illustrated in fig 29. At low slip values, longitudinal force is strongly influenced by terrain stiffness. When  $k_\varphi$  is low, the wheel generates less grip, leading to lower  $F_x$ . Conversely, when at higher  $k_\varphi$ , the wheel remains closer to the surface, enabling more efficient force transfer and better traction. Between -0.2 and 0.2 slip, the longitudinal force takes lower values for softer soils while higher values for harder one. At high slip values ( $s > 0.5$ ), the longitudinal force stabilizes regardless of terrain stiffness, as the soil reaches its maximum shear resistance. The sinkage trends further support these findings: for  $k_\varphi = 1e^5$ , the tire can penetrate up to 0.1–0.2 m under heavy braking, while for  $k_\varphi = 1e^7$ , penetration remains minimal across all slip values. In tables 14 and 15 are reported the tire sinkage values and the longitudinal force values as function of  $k_\varphi$  variation.

s	Tire sinkage $k_\varphi = 1 \text{ e}5$	Tire sinkage $k_\varphi = 2 \text{ e}6$	Tire sinkage $k_\varphi = 1 \text{ e}7$
-0.9	0.1881 m	0.318 m	0.0119 m
-0.7	0.1448 m	0.0302 m	0.0117 m
-0.5	0.1148 m	0.0288 m	0.0115 m
-0.2	0.0832 m	0.0267 m	0.0112 m
0	0.0698 m	0.0256 m	0.0111 m
0.2	0.0589 m	0.0244 m	0.0109 m
0.5	0.0495 m	0.0228 m	0.0106 m
0.7	0.044 m	0.0219 m	0.0105 m
0.9	0.0404 m	0.0211 m	0.0103 m

Table 14: tire sinkage mean values

s	Fx $k_\varphi = 1e5$	Fx $k_\varphi = 2e6$	Fx $k_\varphi = 1e7$
-0.9	0.1881 m	0.318 m	0.0119 m
-0.7	0.1448 m	0.0302 m	0.0117 m
-0.5	0.1148 m	0.0288 m	0.0115 m
-0.2	0.0832 m	0.0267 m	0.0112 m
0	0.0698 m	0.0256 m	0.0111 m
0.2	0.0589 m	0.0244 m	0.0109 m
0.5	0.0495 m	0.0228 m	0.0106 m
0.7	0.044 m	0.0219 m	0.0105 m
0.9	0.0404 m	0.0211 m	0.0103 m

Table 15: Fx mean values

Similarly, the exponent  $n$ , in the Bekker-Wong formulation, defines how terrain stiffness evolves under increasing pressure, impacting both sinkage and longitudinal force transmission.

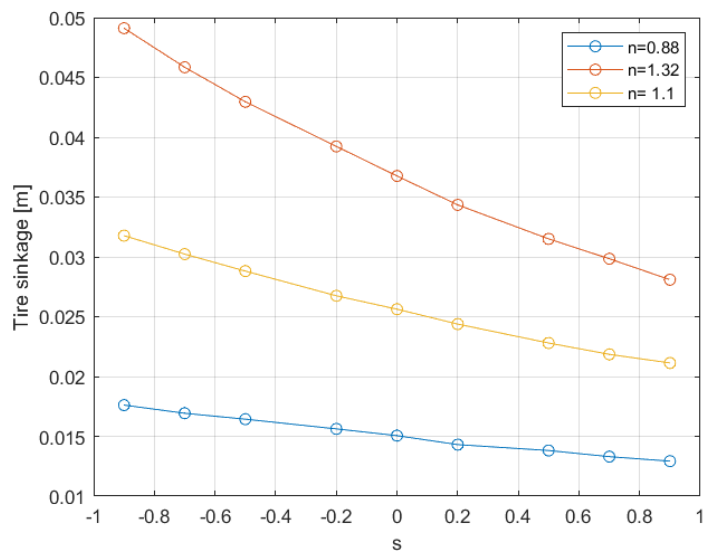


Figure 30: Tire sinkage as function of  $n$

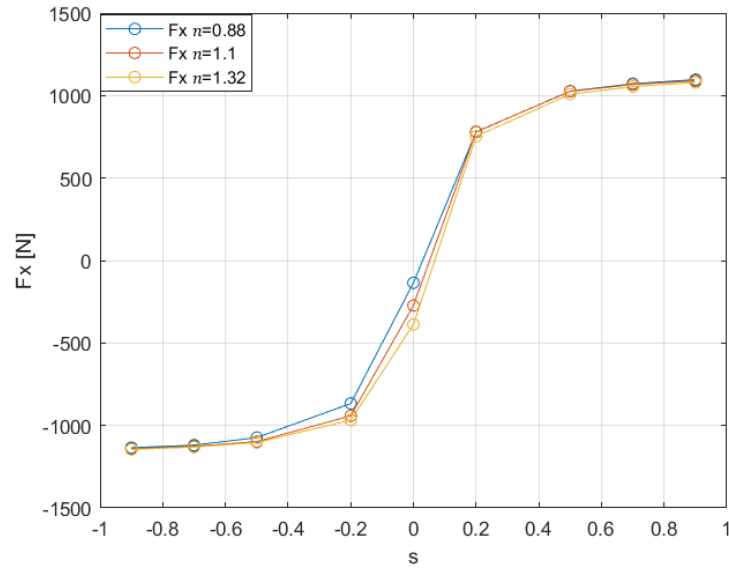


Figure 31: Longitudinal force as function of  $n$

From the fig 30, it is evident that higher values of  $n$  lead to greater sinkage across the entire slip range. The lowest sinkage is observed for  $n=0.88$ , followed by  $n=1.1$ , while  $n=1.32$  results in the deepest penetration. This suggests that as  $n$  increases, the terrain becomes more susceptible to local deformation under the wheel's load, allowing for greater soil deformation. Fig 31, which illustrates the longitudinal force  $F_x$  as a function of slip, reveals that variations in  $n$  have a more subtle effect on force generation compared to their impact on sinkage. While at low slip values  $F_x$  is slightly lower for higher  $n$ , this difference diminishes as slip increases. At moderate slip levels, the variations between curves become less marked, and at high slip values, they converge almost entirely, indicating that the force response is no longer significantly affected by  $n$ , as the terrain's capacity to generate traction becomes more dependent on shear resistance rather than normal deformation. In tables 16 and 17 are reported the tire sinkage values and the longitudinal force values as function of  $n$  variation.

s	Tire sinkage n=0.88	Tire sinkage n=1.1	Tire sinkage n=1.32
-0.9	0.0176 m	0.318 m	0.0491 m
-0.7	0.0169 m	0.0302 m	0.0458 m
-0.5	0.0164 m	0.0288 m	0.0430 m
-0.2	0.0156 m	0.0267 m	0.0392 m
0	0.0151 m	0.0256 m	0.0368 m
0.2	0.0143 m	0.0244 m	0.0343 m
0.5	0.0138 m	0.0228 m	0.0315 m
0.7	0.0133 m	0.0219 m	0.0298 m
0.9	0.0129 m	0.0211 m	0.0281 m

Table 16: Tire sinkage as function of  $n$

s	Fx n = 0.88	Fx n=1.1	Fx n=1.32
-0.9	-1134.4 N	-1141.1 N	-1145.2 N
-0.7	-1119.4 N	-1127.1 N	-1130.8 N
-0.5	-1073.6 N	-1097.7 N	-1103.3 N
-0.2	-866.7 N	-941.4 N	-967.4 N
0	-135.1 N	-273 N	-387.5 N
0.2	779.1 N	780.6 N	753.1 N
0.5	1027 N	1025 N	1008 N
0.7	1072 N	1067.1 N	1054 N
0.9	1096.5 N	1089.5 N	1079.8 N

Table 17: Fx as function of  $n$  variations

In addition to normal stiffness effects, the internal friction angle  $\varphi$  plays an important role in defining the soil's shear strength and its ability to resist deformation.

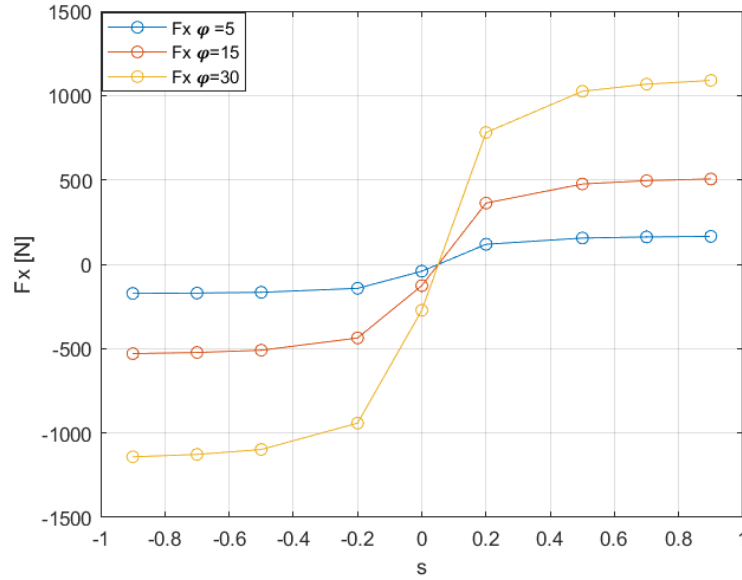


Figure 32:  $F_x$  as function of  $\varphi$  variation

In fig 32, the effect of varying  $\varphi$  is evident in the behaviour of the longitudinal force  $F_x$  as a function of slip  $s$ . When  $\varphi$  is low, as in the case of  $\varphi = 5^\circ$ , the soil has a lower shear resistance, meaning it deforms more easily under the tire's forces, leading to lower traction and braking forces. As a result, the  $F_x$  curve remains relatively flat, with minimal force generation in both the braking ( $s < 0$ ) and driving ( $s > 0$ ) regions. In contrast, for  $\varphi = 30^\circ$ , the terrain offers much higher shear strength, allowing the wheel to generate significantly larger tractive and braking forces. The transition from braking to traction is more distinct, and the curve rises sharply as slip increases, indicating a more efficient force transfer from the wheel to the ground. At negative slip values, where the wheel is braking, higher values of  $\varphi$  lead to greater resistance against motion, resulting in stronger negative  $F_x$ . For  $\varphi = 30^\circ$ , the braking force reaches much larger magnitudes compared to  $\varphi = 5^\circ$ , where the soil deforms more easily, offering little resistance. Similarly, in the driving phase ( $s > 0$ ), the ability to transmit force improves significantly as  $\varphi$  increases, with the force saturating at a much higher value. When  $\varphi$  is small, the wheel experiences excessive shearing of the terrain, meaning that much of the rotational motion is dissipated rather than converted into traction. Summing up, for higher friction angles lead to a sharper and steeper force increase, while lower friction angles result in a more gradual force buildup. This behaviour is aligned with expectations, because the friction is embedded in the Mohr friction angle (passed in degrees to `SCMDeformableTerrain::SetSoilParameters`) used in the Janosi-Hanamoto formula for tangential forces. In table 18 are reported the  $F_x$  mean values as function of longitudinal slip and  $\varphi$  variations.

s	Fx $\varphi = 5$ deg	Fx $\varphi=15$ deg	Fx $\varphi=30$ deg
-0.9	-172.84 N	-529.38 N	1141.1 N
-0.7	-170.67 N	-522.78 N	-1127.1 N
-0.5	-166.34 N	-509.52 N	-1097.7 N
-0.2	-142.63 N	-436.77 N	-941.4 N
0	-41.38 N	-126.54 N	-273 N
0.2	118.26 N	362.33 N	780.6 N
0.5	155.32 N	475.72 N	1025 N
0.7	161.69 N	495.38 N	1067.1 N
0.9	165.09 N	505.6 N	1089.5 N

Table 18: Fx as function of  $\varphi$  variation

## 8.1 LETE Sand Wong terrain

In chapter 8.1, 8.2 and 8.3 three different types of terrain will be analysed, each characterized by real parameters obtained from experimental tests. This approach allows for a more accurate estimation of how the forces exchanged with the terrain are influenced, as well as the resulting wheel sinkage. The first terrain analyzed is characterized by parameters derived from experimental data on LETE sand, representing a deformable soil with moderate stiffness and significant shear strength. The parameters used are reported in table 19.

SCM parameters	Lete sand soil
Bekker $k_{\varphi}$	5.058 e5
Bekker $K_c$	6.94 e3
Bekker $n$ exponent	0.71
Mohr cohesive limit (Pa)	1.15 e3
Mohr friction limit (degrees)	31.5012
Janosi shear coefficient (m)	0.01
Terrain length (m)	20

Table 19: LETE sand parameters

In the upcoming plots, two curves will be shown: one representing the analysed terrain and the other corresponding to a terrain labeled as "Reference." The latter refers to the default terrain implemented in the simulation, with its parameters listed in Table 11.



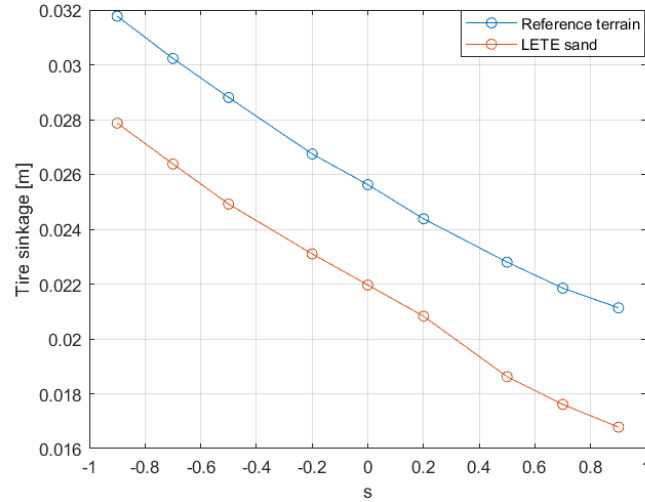


Figure 33: Tire sinkage on Lete sand terrain

Fig 33, reveals a more pronounced difference between the two terrains. Across the entire slip range, the wheel exhibits less penetration in LETE sand than in the reference terrain, confirming that LETE sand offers greater support due to its higher stiffness parameters. The Bekker modulus  $k_\phi$  plays a crucial role in this behavior, as a higher value implies greater resistance to vertical compression, limiting the extent to which the wheel sinks into the soil. This effect is most noticeable at negative slip values, where braking forces tend to increase vertical loading on the front part of the contact patch. The reference terrain, having lower stiffness, allows for greater soil displacement, leading to deeper sinkage and increased rolling resistance.

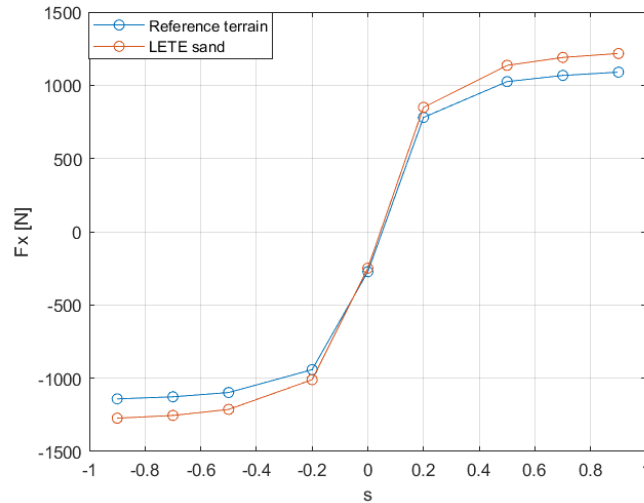


Figure 34: Longitudinal force exchanged with LETE sand terrain

Observing fig 34, both terrains follow a similar overall trend, with  $F_x$  increasing in magnitude as slip moves away from zero. However, LETE sand provides slightly higher traction and braking forces, particularly at moderate and high slip values. This behavior is primarily attributed to its higher internal friction angle, which enhances the soil's shear resistance and allows for more efficient force transmission between the wheel and

the ground. As a result, the wheel can generate greater tractive and braking forces before reaching a saturation point where additional slip no longer contributes significantly to force development.

## 8.2 Road North Gower Clayey Loam

The second terrain analyzed, referred to as Road North, is characterized by experimental parameters that indicate a stiffer and more cohesive soil compared to LETE sand. The parameters are reported in

SCM parameters	Road clayey loam 46 %
Bekker $k_\phi$	2.471 e6
Bekker $K_c$	4.16 e3
Bekker $n$ exponent	0.73
Mohr cohesive limit (Pa)	6.1 e3
Mohr friction limit (degrees)	26.6024
Janosi shear coefficient (m)	0.01
Terrain length (m)	20

Table 20: Clayey loam terrain parameters

The comparison between Road North Gower Clayey Loam 46% and the reference terrain reveals some differences in both tire sinkage and longitudinal force  $F_x$ , reflecting the impact of the terrain's higher stiffness and cohesion on wheel-soil interaction.

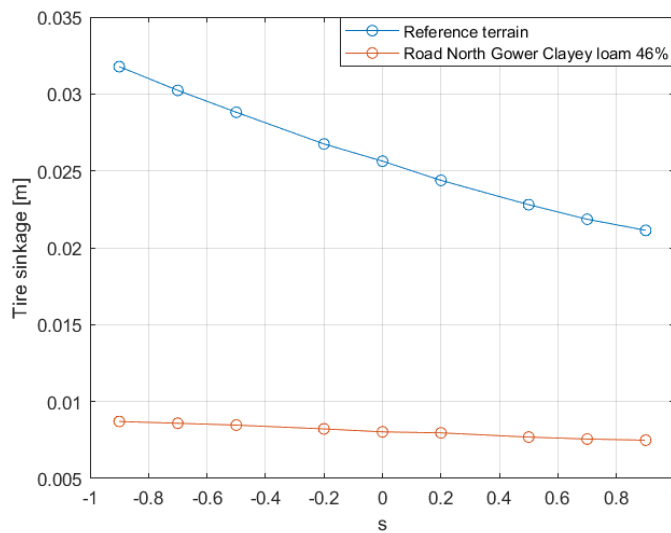


Figure 35: Tire sinkage in Clayey loam terrain

In fig 35, the Road North terrain exhibits significantly lower penetration across all slip values with respect to the reference terrain. This behaviour aligns with its higher Bekker modulus  $k_\phi$ , which provides greater

resistance to terrain vertical deformation, effectively supporting the wheel and preventing deep penetration. The reduction in sinkage is evident at negative slip values, where braking forces tend to increase vertical load transfer. The reference terrain which is characterized by lower stiffness, makes the wheel to sink deeper into the softer soil. For positive slip, both terrains follow a decreasing sinkage trend because the load is transferred more backward.

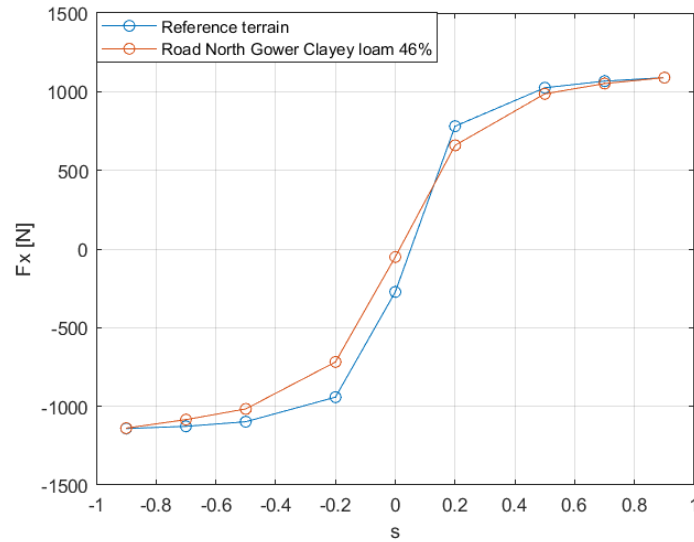


Figure 36: Longitudinal force exchanged with clayey loam terrain

In fig 36, both terrains exhibit a similar overall trend, with  $F_x$  increasing in magnitude for positive slip values. For low slip (i.e.  $s \approx 0.2$ ) the force exerted by the tire tyre on the clayey road is slightly lower than in the reference one, mainly due to the lower friction angle which has great influence as highlighted in the sensitivity analysis. However, at high slip, the two forces are really close each other due to  $k_\phi$  similarity with the reference terrain. At negative slip, where braking forces dominate, the difference between the two terrains is less pronounced, but Road North still maintains a slightly higher braking force, converging both at the same values for low slip values.

### 8.3 Snow road

The third terrain analysed, Snow Road, represents a significantly softer and less cohesive surface compared to the previously examined terrains. The terrain parameters are reported in table 21

SCM parameters	Road snow
Bekker $k_\phi$	1.967 e5
Bekker $K_c$	4.37 e3
Bekker $n$ exponent	1.6
Mohr cohesive limit (Pa)	1.15 e3
Mohr friction limit (degrees)	19.704
Janosi shear coefficient (m)	0.01
Terrain length (m)	20

Table 21: Snow Road parameters

Fig 37 shows a drastic increase in penetration for the Road Snow surface, confirming its highly deformable nature. Across the entire slip range, the wheel sinks much more in Road Snow than in the reference terrain, with the largest difference occurring at negative slip values, where braking increases vertical loading. The exponential nature of the sinkage curve further reflects the nonlinear stiffening response of the terrain, as indicated by the high  $n$  exponent. At high slip values, sinkage decreases as force distribution shifts within the contact patch, but the difference between the two terrains remains significant.

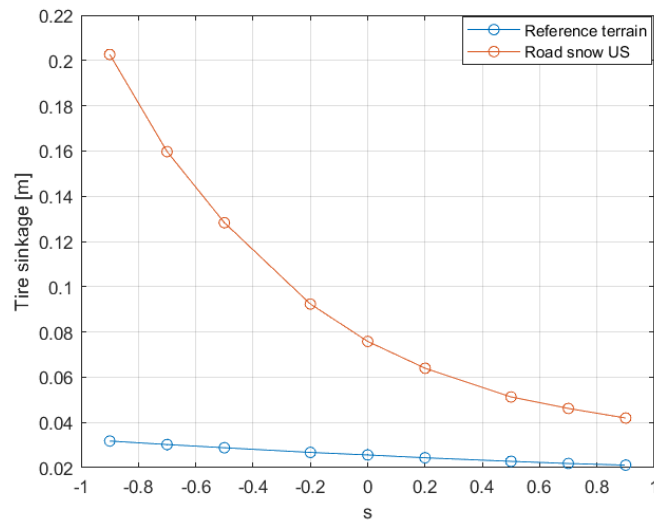


Figure 37: Tire sinkage in a road snow

Fig 38 further illustrates the limitations of Road Snow in terms of traction and braking performance. Across all slip values, the longitudinal force  $F_x$  remains lower than that one of the reference terrain. At negative slip, the braking force is smaller, mainly due to the internal friction angle meaning that the wheel generates less grip making the braking effort less incisive. In the positive slip region, traction develops more slowly, and peak force values are significantly lower, suggesting that the wheel is unable to efficiently transmit force to the terrain. This behavior is consistent with the lower friction angle and higher  $n$  exponent of road snow, which contribute to an overall reduction in force transmission efficiency.

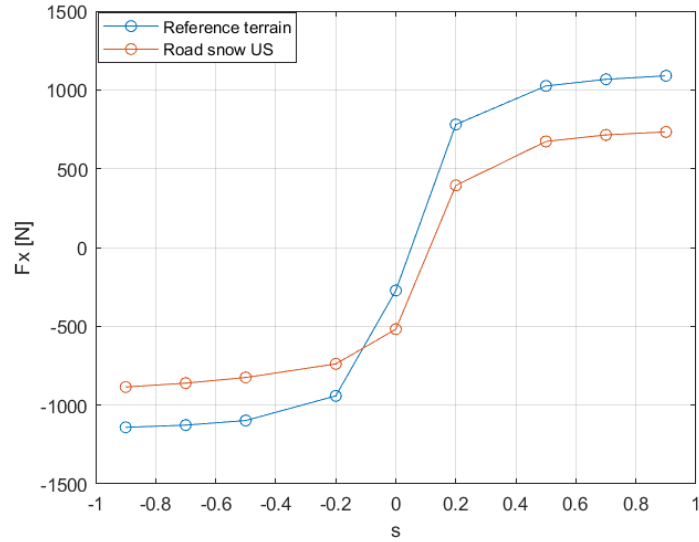


Figure 38: Longitudinal force exerted on a snow road

The analysis of these terrains reveals that the key parameters influencing wheel-terrain interaction are  $k_\phi$ ,  $n$ , and  $\phi$ . Among these,  $k_\phi$  and  $n$  primarily affect sinkage: an increase in  $n$  leads to a non-linear sinkage behaviour, while higher terrain stiffness results in reduced sinkage. Conversely,  $\phi$  has a greater impact on the forces exchanged with the terrain, generating high traction and braking force.

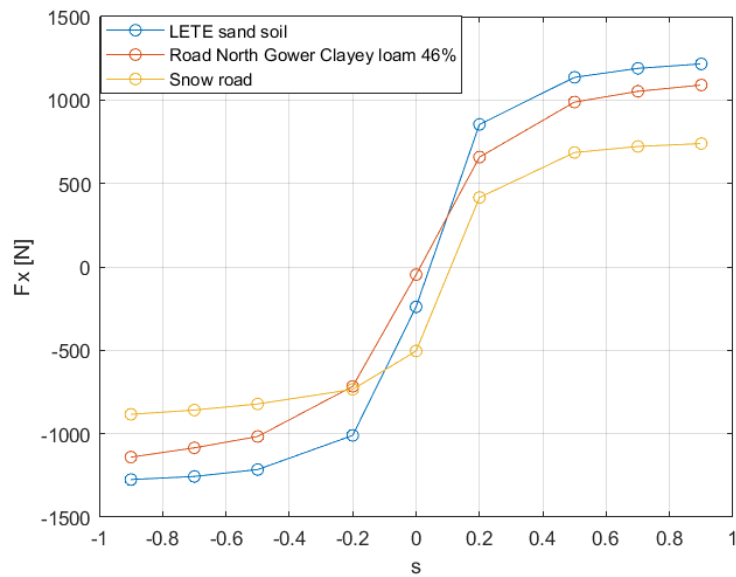


Figure 39: Comparison among the terrains

A common trend emerges across all terrains, where the force varies non-linearly with slip. In the negative slip region, representing braking, all surfaces exhibit increasing resistance as slip grows. In the positive slip region, where traction is applied, the force rises progressively, but at different rates. The LETÉ sand soil demonstrates the highest traction force at positive slip values, with a steep increase in force, indicating strong resistance to deformation. It also exhibits the most pronounced resisting force during braking. This behaviour is explained by its high pressure-sinkaged parameters and significant friction angle, which enhance grip. In contrast, the North Gower Clayey Loam produces a more moderate response, with a force curve positioned between sand and snow. Its slightly higher cohesion and stiffness make it stiffer than sand, yet more compliant than snow, allowing for a balanced interaction between traction and resistance. The snow road, however, provides the weakest traction, with force saturating early, suggesting limited grip compared to the other surfaces. The high compressibility of snow, reflected in its elevated exponent value, coupled with low stiffness and a reduced friction angle, results in a surface that struggles to generate significant force, making it the least effective for maintaining traction. These results reflect real-world behaviour, where sand provides the highest resistance to movement, clayey loam exhibits a balanced response, and snow remains the most deformable surface. The  $F_x$  mean values are reported in table 22.

Fx LETÉ sand	Fx clayey loam	Fx snow soil
-1274.5 N	-1140.1 N	-883.5 N
-1256.2 N	-1084.5 N	-858.06 N
-1214.6 N	-1016.4 N	-821.5 N
-1009.8 N	-716 N	-734.04 N
-239.4 N	-46.6 N	-504.24 N
854.1 N	658 N	415.8 N
1136.2 N	987.9 N	685.7 N
1189.9 N	1052.2 N	721.8 N
1216.7 N	1090 N	739.04 N

*Table 22:  $F_x$  mean values of different soils*

## 9. Lateral slip

The analysis of lateral dynamics is essential for understanding how a wheel behaves on a deformable terrain when subjected to varying slip angles. This study aims to evaluate how different slip angles influence the wheel's lateral motion and the corresponding lateral force  $F_y$ , which plays a key role on stability and manoeuvrability. To understand better this effect, a sinusoidal slip angle is imposed, following the equation

$$\alpha = A \sin(2\pi f t + \theta)$$

(17)

Where:

- $A$  represents the amplitude of the slip angle
- $f$  is the frequency
- $\theta$  is the phase offset.

The selected amplitudes range from  $2^\circ$  to  $20^\circ$ , while the frequency is fixed at 0.1 Hz with no phase shift. This approach allows the observation of periodic lateral variations and their impact on both the trajectory and lateral force response. By comparing different amplitudes, the study aims to evaluate the system's sensitivity to increasing lateral slip.

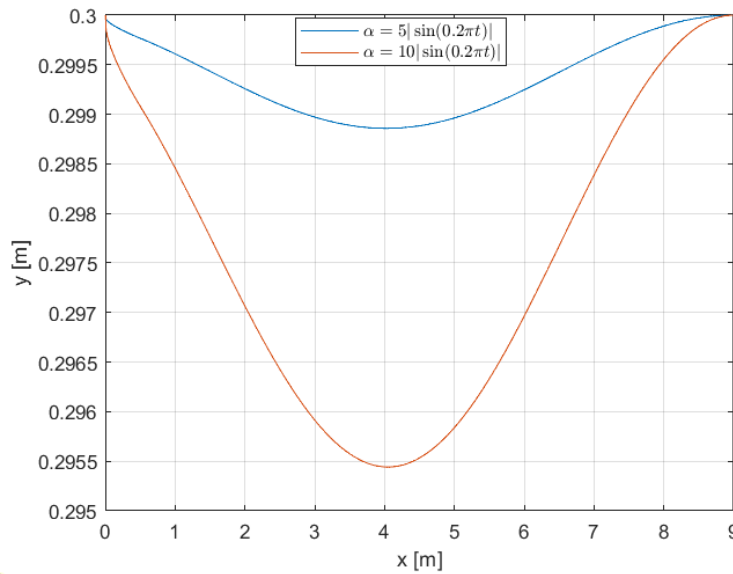


Figure 40: Wheel's trajectory

Fig 40 shows the effect of periodic slip angle variations on the wheel's lateral movement. As expected, a higher slip angle amplitude leads to more significant deviations from the straight trajectory, while a lower amplitude keeps the wheel's movement more close to it. This demonstrates that increasing the slip angle amplitude directly enhances lateral motion, causing larger deviations in the wheel's path.

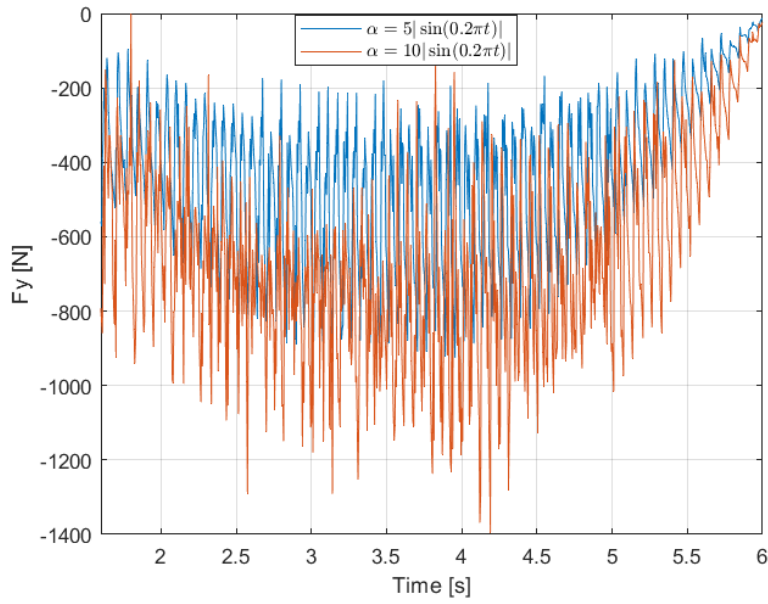


Figure 41: Lateral force evolution over time

In fig 41 is represented the oscillatory behaviour of  $F_y$  as a result of sinusoidal slip angles which continuously alters the lateral interaction between the wheel and the terrain. The wheel experiences a constantly changing lateral force as it shifts its direction of movement relative to the soil. This cyclic behavior causes the force to fluctuate between states of increasing and decreasing resistance, with higher oscillation amplitudes observed for larger slip angle variations. The discretized nature of the terrain in the SCM model further amplifies these fluctuations, as the wheel moves across soil cells with different deformation histories. Each transition forces the solver to reassign contact forces, introducing additional variations in  $F_y$  that are particularly noticeable when the lateral displacement is significant.



## 9.1 Combined slip

In real-world conditions, wheels rarely experience pure longitudinal or pure lateral slip. Instead, vehicles often operate under a combination of both, where traction and lateral forces interact dynamically. This phenomenon, known as combined slip, plays a crucial role in determining overall mobility, stability, and force transmission with the terrain. The interaction between longitudinal and lateral slip is particularly relevant when evaluating vehicle manoeuvrability on deformable terrain, as the soil response varies based on both the applied slip ratio and the slip angle. The results provide insights into the trade-off between traction and lateral control.

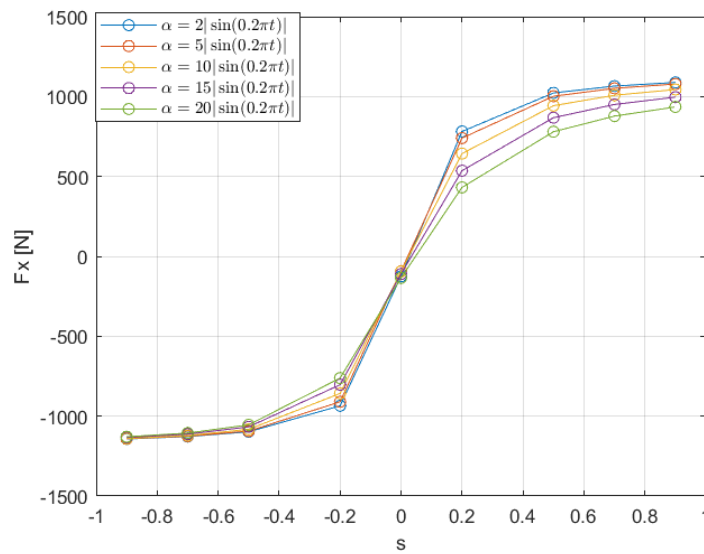


Figure 42:  $F_x$  as function of longitudinal slip

Fig 42 illustrates the variation of longitudinal force  $F_x$  as a function of slip ratio  $s$  for different slip angle amplitudes. The general behaviour follows the characteristic pattern where  $F_x$  increases in magnitude as slip increases in positive or negative way, reaching a saturation point. At low slip values, the longitudinal force response remains nearly identical across all the range of slip angles, indicating that at minimal slip, lateral variations do not significantly alter the wheel's ability to generate traction. As slip increases ( $s \approx 0.2-0.5$ ), differences between the curves begin to become more evident. While smaller slip angles ( $\alpha=2^\circ$  and  $\alpha=5^\circ$ ) maintain a similar behaviour to the pure longitudinal case, larger slip angles ( $\alpha=15^\circ$  and  $\alpha=20^\circ$ ) exhibit a slight reduction in  $F_x$ . This reduction is due to the cosine dependency on the side slip angle, which limits the maximum achievable longitudinal force. As slip continues to increase beyond  $s > 0.5$ , the differences between the curves become more noticeable. Higher slip angles result in a slight reduction in peak  $F_x$ , consistent with the expectation that as lateral oscillations increase, the soil response is modified, leading to a redistribution of forces within the contact patch. The deviation of  $F_x$  at greater slip values indicates that at large slip angles, the terrain yields more in the lateral direction, reducing its ability to effectively resist on longitudinal motion.

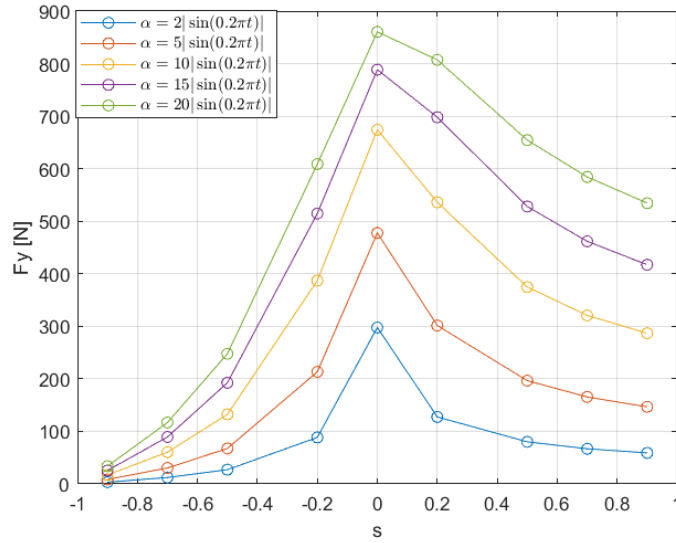


Figure 43: Lateral force as function of longitudinal slip

Fig 43 presents the variation of lateral force  $F_y$  as a function of longitudinal slip  $s$  for different slip angle amplitudes. Unlike the longitudinal force  $F_x$ , which follows a more monotonic saturation trend,  $F_y$  exhibits a distinct peak before decreasing at higher slip values. At negative low slip values,  $F_y$  remains small across all cases, indicating that minimal lateral slip does not generate significant lateral forces. As slip increases ( $s=-0.5$  and  $s=-0.2$ ),  $F_y$  rises rapidly, with higher slip angles producing larger peak forces. This behaviour is expected, as a higher lateral slip amplitude redirects more of the wheel's contact force toward resisting lateral motion, resulting in increased lateral force magnitude. Beyond the peak,  $F_y$  begins to decrease as slip continues to rise. This suggests that at excessive slip values, the terrain's ability to resist lateral displacement diminishes, likely due to soil shearing and reduced effective contact between the wheel and the ground. This is particularly evident for the largest slip angles ( $\alpha=15^\circ$  and  $20^\circ$ ), where  $F_y$  initially reaches high values but drops more significantly after the peak. The decrease in lateral force at high slip values highlights the trade-off between maintaining lateral stability and excessive lateral slip, as too much lateral motion can reduce the terrain's effectiveness in sustaining lateral resistance.

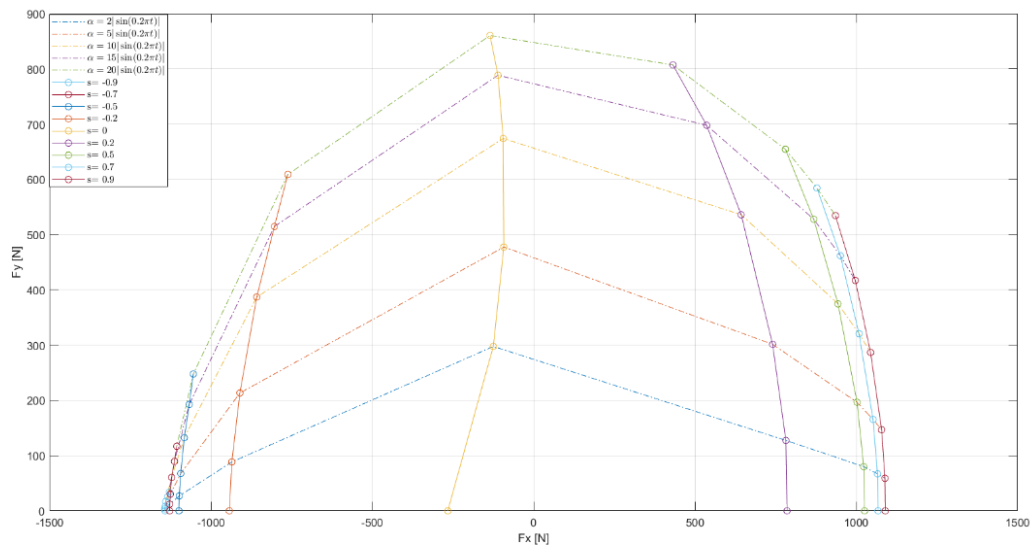


Figure 44:  $F_y$  as function of  $F_x$

Fig 44 represents the combined force response of the wheel by plotting the relationship between the longitudinal force  $F_x$  and the lateral force  $F_y$  under different slip conditions. This plot is derived from the previous two graphs, where  $F_x$  and  $F_y$  were analysed separately as functions of slip ratio. By merging these two, this representation provides a comprehensive view of how the wheel distributes forces between traction and lateral stability. The curves make a force ellipse-like distribution, which is a well-known characteristic in combined slip analysis. It is immediately evident that the maximum lateral force does not coincide with the maximum longitudinal force. This phenomenon is linked to the distribution of tangential forces, which develop either in pure longitudinal or pure lateral motion. The way a tire generates forces while cornering differs from how it behaves under pure longitudinal slip, resulting in adhesion and slip zones with distinct shapes and extents. Consequently, the combination of forces arising from both the plastic deformation of the terrain and frictional effects can lead to different peak force values. Furthermore, footprint deformations under braking and acceleration differ, causing a slight variation between the maximum force achievable in braking and that in acceleration. As a result, the overall force limit curve does not form a perfect circle but is better approximated by an ellipse, reflecting the redistribution of available grip between longitudinal and lateral directions. When a tire generates force in one direction, the limit for the other direction is reduced. The presence of a slip angle combined with longitudinal slip gradually compresses the force limit curve downward. As the tire generates lateral force, which increases with the slip angle, the maximum achievable longitudinal force decreases. Each increase in slip angle results in a reduction of the longitudinal force that can be attained for a given slip level, reflecting the inherent trade-off in tire force distribution.

## 10. Full vehicle simulation

After analysing single-wheel dynamics in various conditions, a full vehicle simulation was conducted on different test conditions to evaluate its dynamic response on SCM deformable soil. This study provides a more realistic representation of vehicle-terrain interaction. Unlike single-wheel tests, a complete vehicle model introduces additional complexities to analyse, such as load transfer, and traction distribution among multiple wheels. The simulation was performed using the Chrono::Vehicle module, which allows for the modelling of wheeled vehicles with various subsystems, including chassis, suspension, powertrain, and tires. The HMMWV model, an off-road military vehicle, was already implemented in the chosen demo, for this reason has been analysed as off-road vehicle. The terrain was defined using the SCM model, in particular the Sand Lete-Wong terrain which parameters are reported in tab 22. The objective was to observe how traction forces develop across all four wheels, analyse longitudinal and lateral slip distribution, and evaluate vehicle stability under acceleration. Data extracted from the simulation are: tire forces, slip ratios, vertical load variations, and vehicle states such as longitudinal acceleration, lateral acceleration and pitch angles. The results from this simulation provide insight into how a multi-wheel system behaves on soft terrain, extending the results from previous single-wheel studies, accounting for additional real-world effects such as weight transfer.

SCM parameters	Values
Bekker $k_\varphi$	2 e6
Bekker $K_c$	6.94 e3
Bekker $n$ exponent	0.71
Mohr cohesive limit $c$ (Pa)	1.15 e3
Mohr friction limit $\varphi$ (deg)	30
Janosi shear coefficient $j$ (m)	0.01
Elastic stiffness (Pa/m)	2 e8
Damping (Pa s/m)	3 e4

Table 23: SCM terrain parameters used for full vehicle analysis

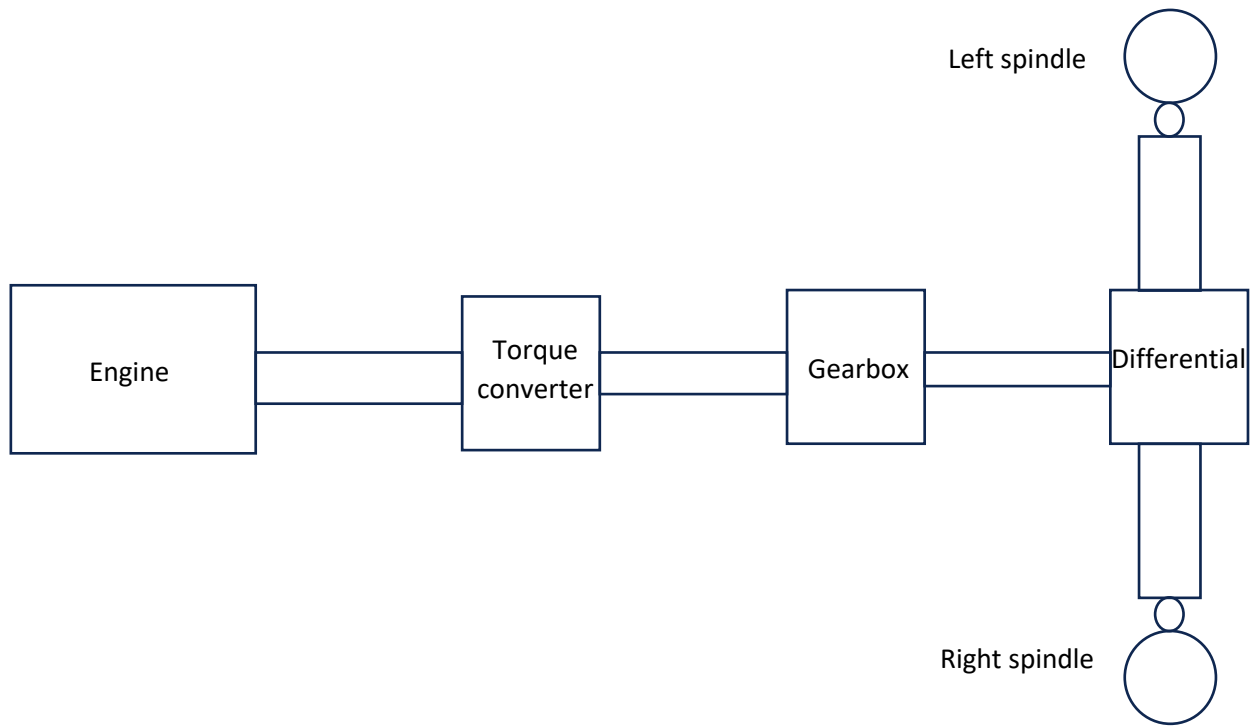


Figure 45: Vehicle schematization

In fig 45 it is represented a vehicle driveline. The vehicle features a front-wheel-drive (FWD) layout, integrating a rack-and-pinion steering system, and the powertrain is composed by an internal combustion engine, a torque converter, a gearbox, and a differential with conical gear connected each other using rigid shafts. The engine torque model is based on speed-torque curves following the formulation:

$$T_e = T_e(\omega_e)$$

Where  $T_e$  is the engine torque and  $\omega_e$  is the engine speed. The torque converter is modelled using two fundamental curves. The first is the capacity factor curve, which relates the speed ratio between the turbine and impeller to the overall capacity factor as follow:

$$K = K(R_\omega) \quad (19)$$

Where:

- $K = \frac{\omega_I}{\sqrt{T_I}}$  which is the capacity factor defined s the ratio between the impeller angular speed and the square of impeller torque;
- $R_\omega = \frac{\omega_T}{\omega_I}$  which is the speed ratio between turbine and impeller.

The second curve is the torque ratio, which relates the torque as function of speed ratio:

$$R_T = R_T(R_\omega) \quad (20)$$

Where:

- $R_T = \frac{T_T}{T_I}$  which is the torque ratio;
- $T_T$  and  $T_I$  are the torque of the turbine and the impeller respectively.

These models are directly implemented as API, which replicate gear shifting dynamics and power delivery across different driving conditions. The gear box is parameterized by a set of forward gear ratios and a single reverse gear ratio. The upshifting occurs around 2500 rpm while the downshifting around 1200 rpm. Moreover, the suspension system consists of independent double-wishbone on both front and rear axles, but it is not implemented an antiroll bar system. The Bullet physics engine is employed for collision detection which efficiently manages contact interactions between rigid vehicle components and soft soil. Moreover, the Smooth Contact Model (SMC) is used as the contact method, applying a penalty-based force formulation, already explained in Chapter 2. In table 23 are reported the vehicle characteristics.

Vehicle parameters	Values
Mass (kg)	2086.52
Mass moment inertia Jxx (kg/m <sup>2</sup> )	1078.52
Mass moment inertia Jyy (kg/m <sup>2</sup> )	2955.66
Mass moment inertia Jzz (kg/m <sup>2</sup> )	3570.20
Tire radius	0.467
Wheelbase (m)	3.378
trackwidth (m)	1.82
COG height (m)	0.6
Front stiffness suspension (N/mm)	224.8 N/mm
Rear stiffness suspension (N/mm)	496.7 N/mm

Gear ratio 1 <sup>st</sup>	0.2
Gear ratio 2 <sup>nd</sup>	0.4
Gear ratio 3 <sup>rd</sup>	0.8
Reverse gear ratio	-0.1
Differential gear ratio	0.2

*Table 24: Vehicle parameters*

In the simulation, the vehicle is controlled through three primary inputs normalized within a range from -1 to 1: throttle, brake, and steering, each of them influencing different aspects of vehicle dynamics. The throttle input is responsible for regulating the engine's power output, which directly affects the vehicle's acceleration and traction. A value of 1 represents full throttle, delivering maximum power to the wheels, while a value of 0 means no throttle application. The brake input applies a deceleration force to slow down or stop the vehicle. A value of 1 corresponds to maximum braking force, while 0 means no braking is applied. It is implemented so that the input which is given is a percentage of a maximum braking torque which can be delivered, and then evenly distributed across the four wheels. The braking force influences both longitudinal dynamics and lateral stability, as excessive braking can lead to wheel lock-up and loss of traction on soft terrain. The steering input determines the angular displacement of the front wheels, controlling the vehicle motion direction. A positive steering value corresponds to turn to left direction, while negative ones represent right direction, while a null value keeps the wheels aligned straight. By modifying these three inputs, different driving scenarios can be replicated, in particular in this project have been analysed step and ramp acceleration, braking, and steering maneuvers. For the braking and acceleration cases, the results obtained from ramp and step maneuvers are really close, with nearly overlapping force and motion responses. For this reason, only the step case is presented. However, for the steering analysis, both ramp and step cases are included.

## 10.1 Step acceleration manoeuvre

The step acceleration test is designed to evaluate the vehicle's longitudinal dynamics when subjected to a sudden throttle input. In this simulation, the vehicle starts from rest, and after a 1 second delay, a step throttle input of 0.4 is applied, rapidly increasing engine power.

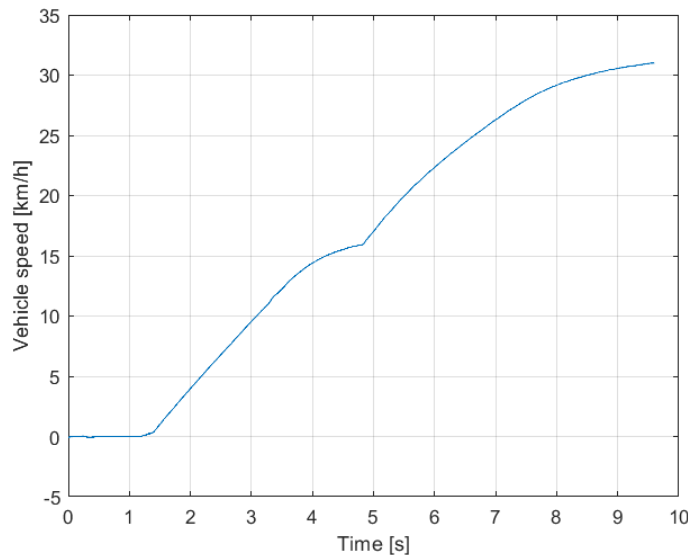
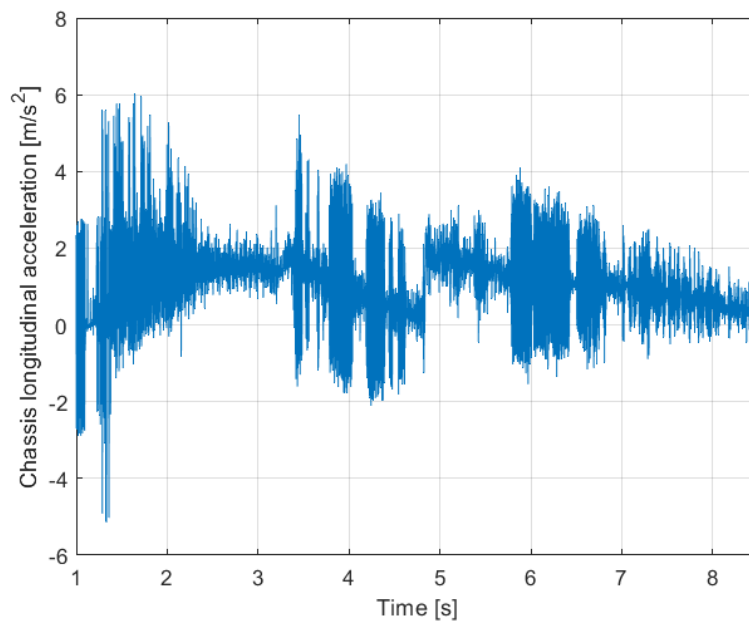


Figure 476: Vehicle speed profile

In fig 46 it illustrated the vehicle speed evolution over time. As it can be noticed, the vehicle remains stationary, but after a 1-second delay, the throttle is engaged at 0.4, starting the acceleration. Due to the terrain resistance and drivetrain inertia, the speed does not increase instantaneously but follows a gradual rise, similar to the ramp acceleration case which it's more similar to the real-world behaviour in which the final speed is not reached instantaneously. Between 2 and 4 seconds, the acceleration follows a steady, nearly linear trend, while around 5 seconds, a bump appears in the velocity curve. This corresponds to the gear shift from first to second gear, which occurs when the engine reaches 2500 rpm. This transition in power transmission briefly alters the speed profile as the engine torque delivered to the wheels decreases. Beyond this point, the speed continues to rise up to 9 seconds in which reaches vehicle is approaching a quasi-stationary state just before another gear shifting. Since the velocity was stabilizing and reaching its final value, the simulation time was chosen accordingly, ensuring that all relevant dynamic behaviours were captured without the need for an extended simulation duration.

The longitudinal acceleration profile in fig 47 provides a deeper view of the vehicle's dynamic response during the step acceleration. Initially, as the throttle is applied after the 1-second delay, a sharp increase in acceleration can be observed. This phase exhibits relatively high fluctuations, which can be attributed to the interaction with terrain deformation, as the vehicle overcomes the initial resistance of the soft soil. Between 2 and 4 seconds, the acceleration stabilizes, reflecting the nearly linear speed increase seen in the velocity graph. However, around 5 seconds, a distinct increase in acceleration fluctuations occurs, which aligns with the gear shift from first to second gear. This transition briefly interrupts the smooth torque delivery, causing the step increase in acceleration. Beyond this point, the acceleration magnitude gradually decreases, consistently with the flattening slope of the speed curve. The oscillations persist but become progressively smaller, indicating that the vehicle is reaching a more stationary state as highlighted in fig 44.



*Figure 47: Vehicle longitudinal acceleration*

In the front axle force graph, fig 48, an initial spike in  $F_x$  occurs immediately after the throttle is engaged at  $t=1$  s, reflecting the sudden demand for traction. The force rises sharply, reaching peak values above 3000 N, demonstrating that the front wheels are effectively gripping the terrain to drive the vehicle forward. As the vehicle begins to accelerate and stabilizes into this phase, the force shows a lower fluctuation amplitude, up to a sudden decrease due to the gear shift. This occurs around 5 seconds where a spike appears, corresponding to the gear shift into second gear at 2500 rpm. Beyond this point,  $F_x$  continues to decrease gradually, indicating that as the vehicle gains speed and less tractive effort is required to sustain motion due to reduced net resistance.



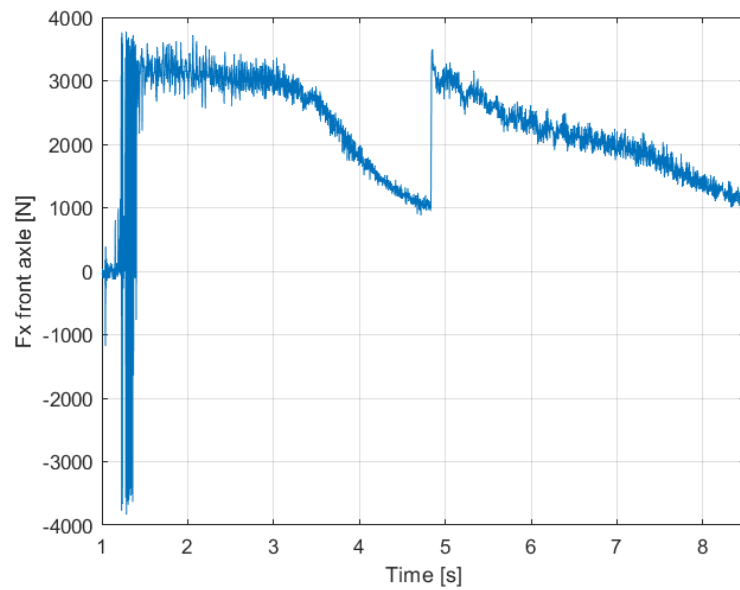


Figure 48: Longitudinal force at the front axle

The rear axle force plot, fig 49, on the other hand, exhibits a different behaviour. Unlike the front axle, which is actively generating traction, the rear axle primarily reacts to terrain deformations and load transfer effects. The large amplitude observed during the initial acceleration phase reflects the effects of load transfer. Despite the high fluctuations, a negative mean  $F_x$  value can be identified, resulting from residual tensions caused by terrain deformation due to the front wheels and the fact that these are dragged wheel, meaning they do not generate traction. Around the 5-second mark, the impact of gear shifting becomes less pronounced, while at higher time values, the amplitude progressively decreases, following the acceleration behavior.

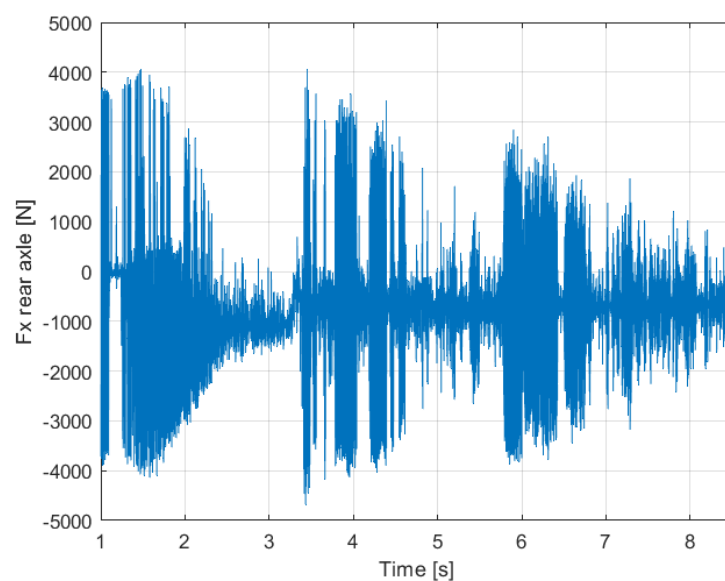


Figure 49: Longitudinal force on rear axle

During the initial phase when the vehicle is stationary, the front axle sustains approximately 6500 N. As the throttle is engaged and acceleration begins,  $F_z$  gradually decreases, reaching a minimum of around 5500 N. This reduction in normal load is caused by the rearward load transfer of the vehicle. This occurs as acceleration decreases, as shown in Figure 47. The gear shifting event momentarily increases acceleration, further reducing the front axle load. Once the vehicle speed stabilizes, the load is gradually redistributed to the front again. The vertical force on the rear axle follows an opposite trend to that of the front axle, increasing as acceleration progresses. Initially, the rear axle supports approximately 6000 N, which gradually rises as the vehicle gains speed. This behavior is consistent with the expected rearward weight shift due to acceleration on soft terrain. The increase in  $F_z$  enhances traction on the rear wheels, contributing to overall stability despite the vehicle being front-wheel drive. Toward the end of the simulation, the average  $F_z$  value begins to decrease, which is consistent with the acceleration plot, indicating a progressive redistribution of the load as the vehicle stabilizes.

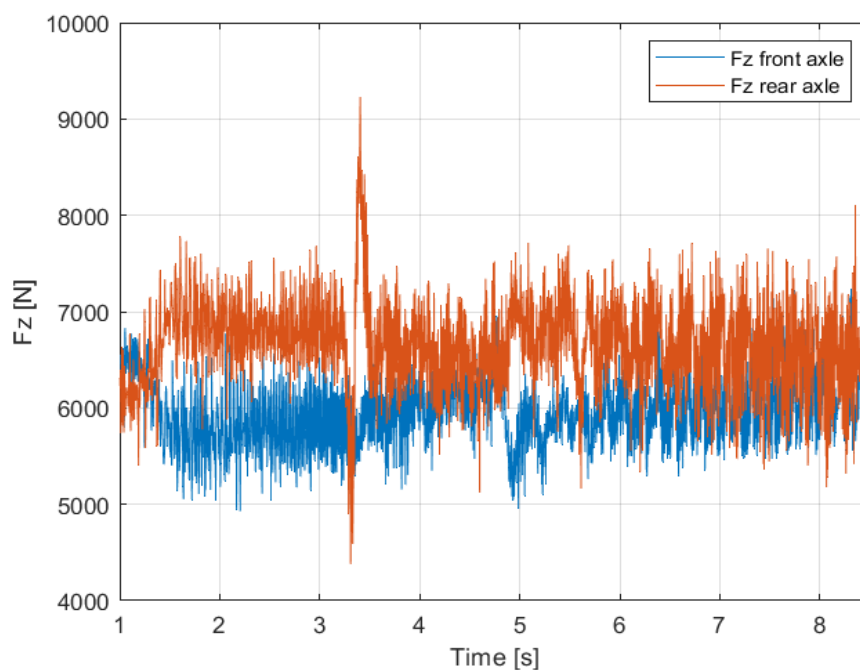


Figure 50: vertical load on the front axle

Fig 51 illustrates vehicle's pitch angle evolution over time. Before throttle application, the vehicle remains stationary, as confirmed by the speed and acceleration profiles. However, due to the way the code is implemented, the delay affects only the equations of motion and not the torque delivery. As a result, torque is applied to the front axle even while the vehicle is still at rest, causing a slight compression in the front suspension and making the vehicle dip forward. This results in a positive pitch angle, aligning with the expected load distribution. Once the throttle is engaged at 1 second, the vehicle begins accelerating, shifting weight toward the rear axle. This transition causes the pitch angle to turn negative as the rear suspension compresses while the front extends. This effect follows the ISO frame convention, where rearward tilt is represented as negative pitch. Between 2 and 4 seconds, the pitch angle progressively stabilizes at a lower value, reflecting a steady load transfer as the vehicle gains speed. Around the 5-second mark, a noticeable fluctuation appears, corresponding to the gear shift from first to second gear. Beyond this point, the pitch angle continues to show minor fluctuations but remains predominantly negative, due to non-null longitudinal acceleration as shown in fig 49.

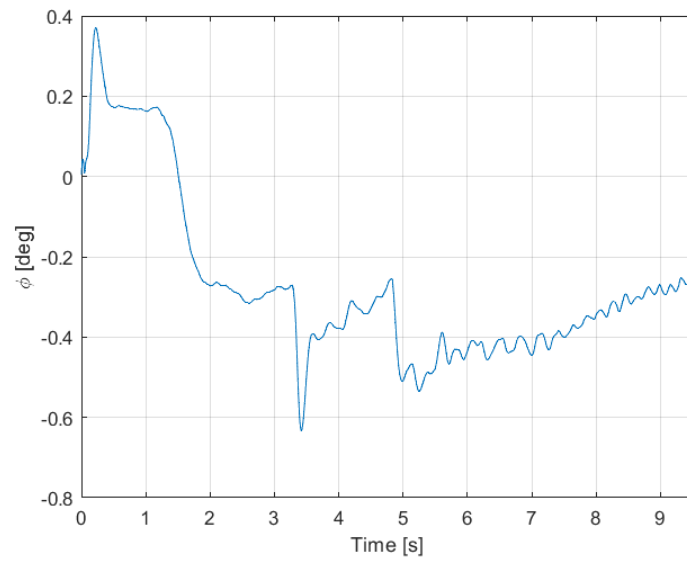


Figure 51: Pitch angle evolution

## 10.2 Step braking manoeuvre

The step braking test is designed to evaluate the vehicle's deceleration dynamics when a sudden braking input is applied while moving on a deformable terrain. In this simulation, the vehicle initially accelerates from rest with a throttle input of 0.4, allowing it to gain speed and establish a near steady motion phase. After approximately 5 seconds, a braking input of 0.4 is applied, generating a rapid deceleration. The braking system follows a torque-based approach, where the applied brake input proportionally translates into braking torque at the wheels. Moreover, as braking induces a forward weight transfer, the suspension system reacts by compressing at the front axle and unloading the rear, affecting the vertical load at each wheel and the overall braking performance.

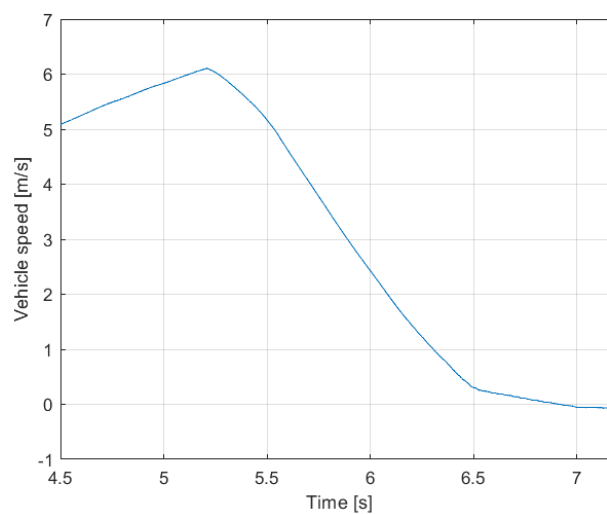


Figure 52: vehicle speed characteristic

Fig 52 illustrates the vehicle speed from  $t=4.5\text{s}$  up to the end of the simulation. This is done because the initial accelerating condition is the same as describe for the step acceleration case. The speed increase up to a peak velocity value of approximately 6 m/s, coinciding with the moment the braking input is applied. At this point, a rapid deceleration phase begins, evident from the sharp downward slope of the velocity curve. The braking force significantly reduces the vehicle's speed, with a steep gradient indicating a strong braking effect on the deformable terrain. As braking continues, the velocity rapidly decreases until it approaches zero around  $t=7$  seconds.

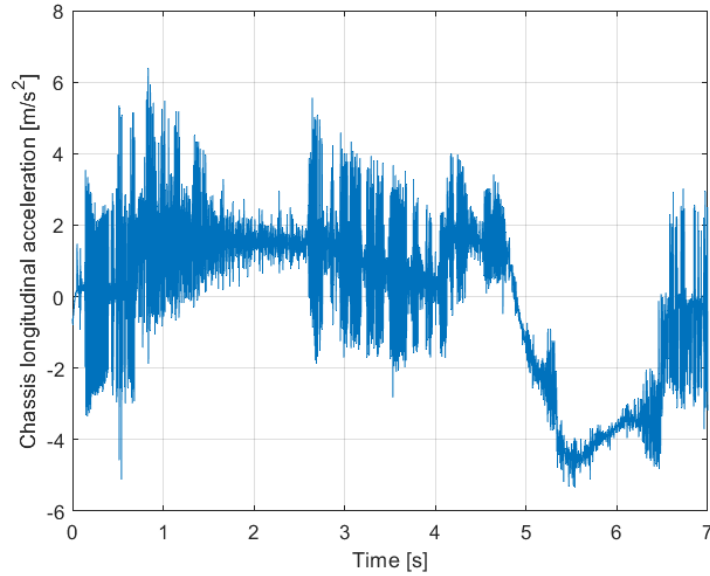


Figure 53: Vehicle longitudinal acceleration

At the beginning of the simulation, the acceleration before  $t=1$  second is not considered, as the vehicle speed plot confirms that the vehicle is still stationary. Once the throttle input is applied, a rapid increase in acceleration is observed, indicating the initial force buildup as the powertrain delivers torque to the driven wheels. All the observations made for the step acceleration case also apply to this scenario. At  $t=5$  seconds, the braking input is engaged, leading to a sharp decline in acceleration, with a temporary stable fluctuation related to the down shifting. As the vehicle continues to decelerate, the acceleration profile gradually stabilizes around  $t=6.5$  seconds, approaching zero as the vehicle comes to a complete stop.

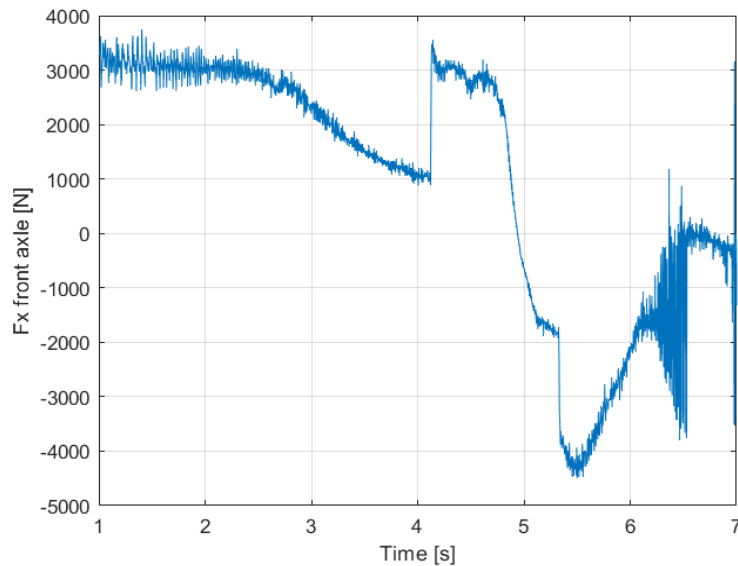


Figure 54: Longitudinal force at front axle

Fig 54 shows the longitudinal fore exerted by the front wheels. During the initial acceleration phase, the front axle sustains a high positive longitudinal force, reaching approximately 3000 N, which remains relatively

stable apart from minor oscillations due to terrain interaction. This force is consistent with the vehicle's front-wheel-drive configuration, where the engine delivers power to the front wheels, ensuring traction. Following the application of braking at approximately 5 seconds, the front axle force undergoes a sharp transition from positive to negative values. This indicates that the front wheels experience a significant load transfer, as evidenced by the fact that the peak braking force exceeds the maximum traction force. The magnitude of the negative force progressively decreases in absolute value, stabilizing close to zero.

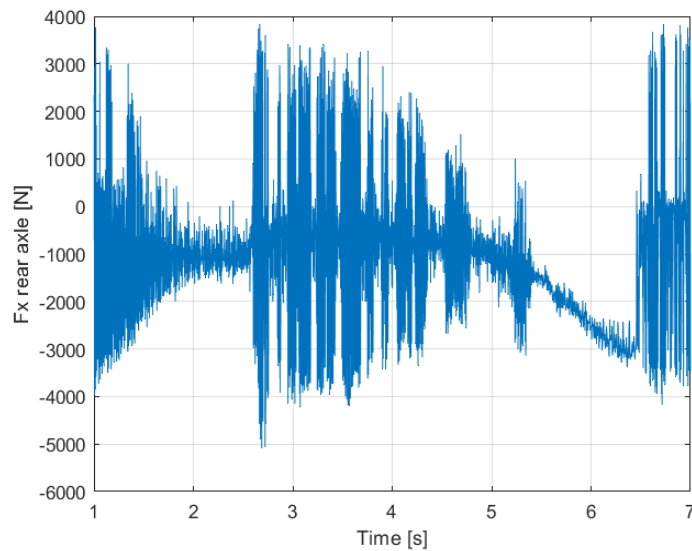


Figure 55: Longitudinal force at rear axle

The longitudinal force behaviour on the rear axle is shown in fig 55. Up to 5 second, the trend is the same of that already analysed in the step acceleration case and no significative variation occurs. However, after 5 seconds when braking is applied, the rear axle presents a decreasing trend for a small amount of time and then, towards the end of the simulation, an oscillatory behaviour. This is due to the load transfer at the front which causes the rear wheels locking clearly seen during the simulation. Therefore, rear tires do not contribute in an effective way to slow down the vehicle.

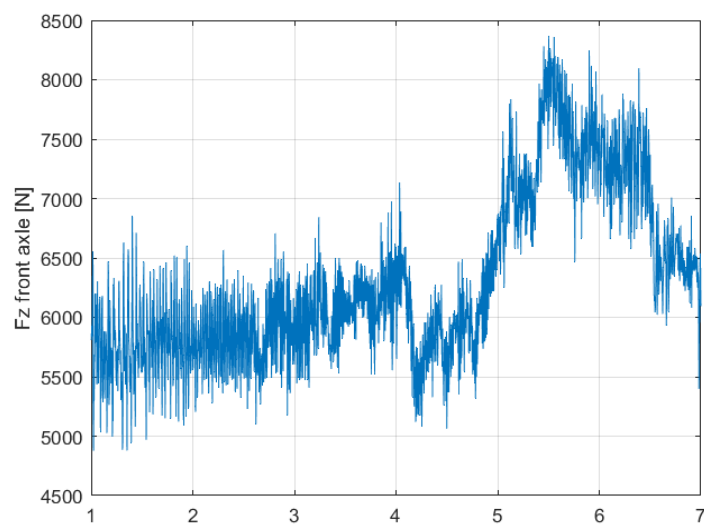


Figure 56: vertical load on front axle

The vertical force acting on the front axle exhibits variations corresponding to the different phases of vehicle motion, particularly during acceleration and braking. Initially, the front axle supports a stable load of approximately 6000 N, with minor oscillations. As the vehicle accelerates, a reduction in  $F_z$  occurs due to load transfer toward the rear axle. Instead, a significant increase in  $F_z$  is observed around 5 seconds, coinciding with the application of braking. As the deceleration force is exerted primarily through the front wheels, weight shifts forward, increasing the vertical load on the front axle. This results in a peak force reaching 8000 N, highlighting the effect of load redistribution under braking conditions. Once the braking force stabilizes, the front axle force gradually decreases.

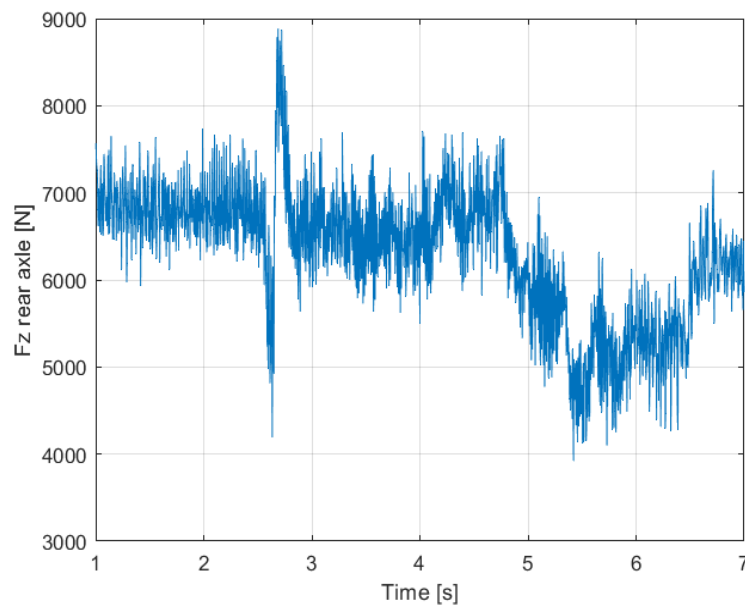
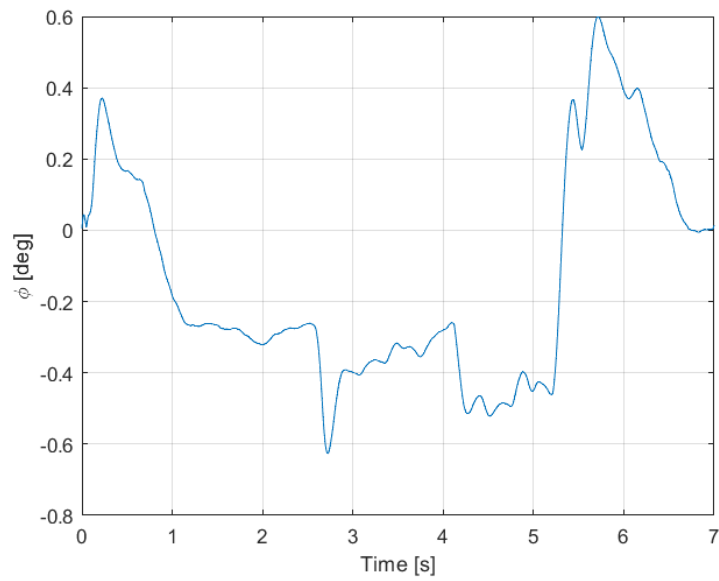


Figure 57: Vertical load at the rear axle

The rear axle vertical force follows an inverse trend to that of the front axle due to the dynamic load transfer during acceleration and braking. Initially, the rear axle supports a relatively steady load of around 7000 N, showing moderate oscillations similar to the front axle. As the vehicle accelerates, the normal force on the rear axle increases, reflecting the weight shift toward the rear. Around 5 seconds, when braking is applied, a sharp decrease in  $F_z$  is evident, indicating a strong forward load transfer. The vertical force at the rear axle drops significantly, reaching values close to 4000 N, demonstrating the unloading of the rear wheels. This reduction in normal force reduces the rear wheels' ability to generate braking forces, which aligns with the previous observations of rear axle longitudinal force oscillations and wheel lock-up.



*Figure 58: vehicle pitch motion*

The pitch angle behavior in the step braking scenario closely follows the trends observed in step acceleration, with notable differences due to braking-induced load transfer. Around the 5-second mark, when braking is applied, the pitch angle shifts from negative to positive, indicating a substantial forward weight transfer. This contrasts with the acceleration case, where the pitch angle remained negative as speed increased. The peak in the pitch curve results from sudden deceleration, shifting the load toward the front axle. These fluctuations are more pronounced than in the acceleration case, suggesting that braking introduces additional dynamic instability.



## 10.3 Step steer manoeuvre

The step steer manoeuvre is used to evaluate the vehicle's lateral dynamics and handling response when subjected to a sudden steering input. In this simulation, the vehicle begins moving in a straight line with a throttle input of 0.4, as in the step acceleration case, ensuring sufficient speed before the steering action is introduced. When it is almost reached a constant speed of 31 km/h, the steering input is applied. Considering deg that the vehicle can achieve a maximum steering angle of 28 deg, two cases has been considered: a steering step input of 0.3 corresponding to 8.4 deg and steering step input of 0.6 corresponding to 16.8 deg.

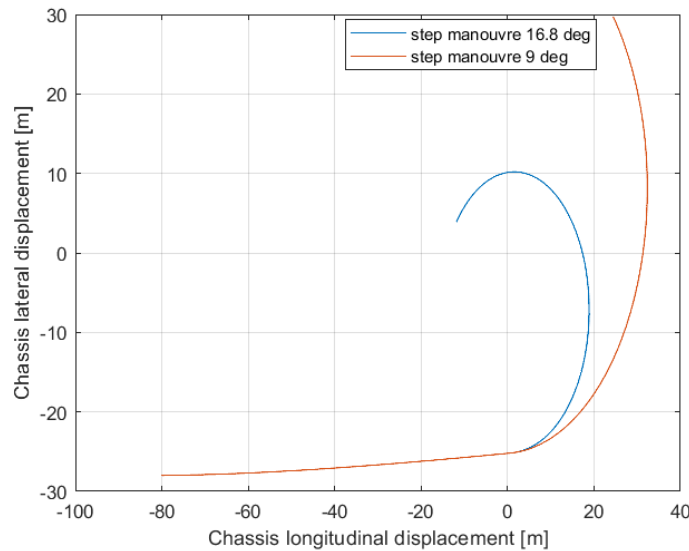


Figure 59: vehicle's trajectory

The vehicle trajectory plot illustrates the curved path followed by the vehicle from the applied step steering input. Initially, the vehicle moves forward, as no steering action is present, but doesn't follow a straight path, but shows a small lateral deviation. However, after the imposed delay, the step input is applied causing the vehicle to gradually deviate from the straight trajectory and follow a continuously increasing lateral path.

The vehicle speed profile is shown in fig 60. Between 12 and 13 seconds, the vehicle reaches a constant speed of 31 km/h. After 13 seconds, the vehicle begins the turning phase, but the speed doesn't remain constant, but decreases. This behaviour is due to the implementation of the demo. Generally, this vehicle takes as input only three parameters, which are the percentage of throttle, brake and steering. In this case, only throttle and steering are relevant. However, once the steering input is applied, the throttle remains fixed at its initial value of 0.4, causing the vehicle speed to decrease. Since there is no direct access to the vehicle speed as an input parameter, this approach makes it particularly difficult to study properly the manoeuvre characteristics.

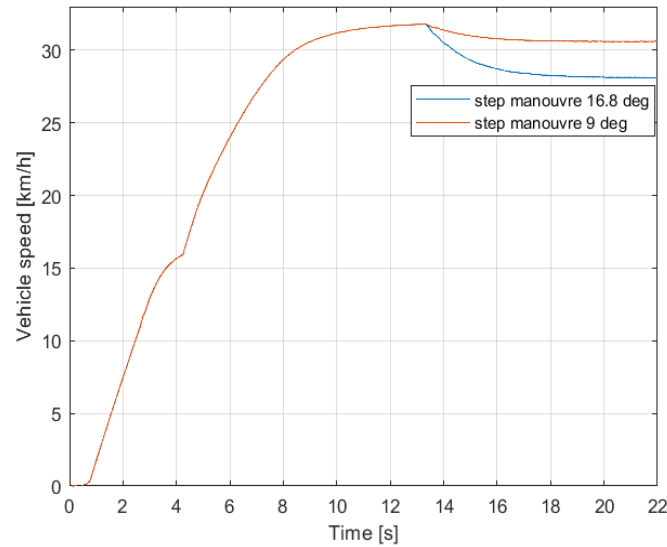


Figure 60: vehicle speed

The curve represented in fig 61 depicts the steering angle response following a step input. Initially, the angle remains close to zero, but as soon as the inputs are introduced, a sharp spike occurs before the system stabilizes around their nominal values of 9.4 and 16.8 degrees. This initial overshoot is likely caused by the dynamic response of the steering system, where the sudden application of the input leads to a brief transient phase due to system inertia. After this brief overshoot, the angle oscillates around their constant mean values.

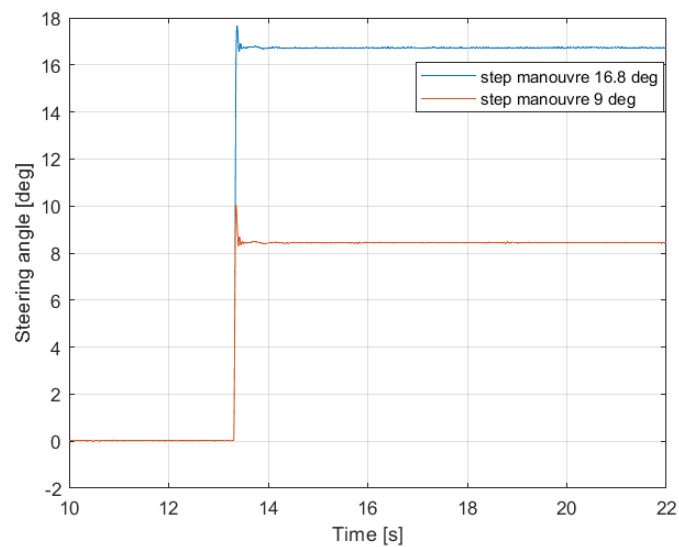


Figure 61: steering angle

In Fig 62 displays it is shown the vehicle lateral acceleration as function of steering angle. As it can be noticed, up to 6 degrees, the lateral acceleration increases as expected, but then decreases increasing the steering angle. This is due to the fact that the lateral acceleration is referred to the global reference frame solidal with the terrain instead of referring to the vehicle reference frame.

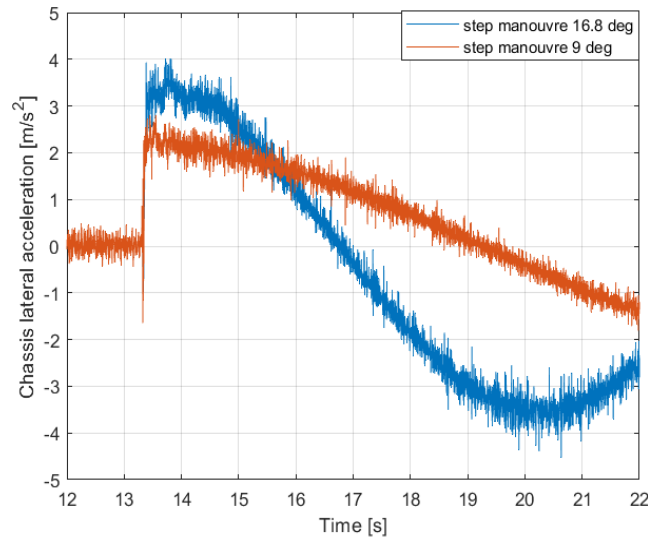


Figure 62: Vehicle lateral acceleration

At approximately 13 seconds, when the steering angle is applied, there is a sharp peak in lateral acceleration, indicating an immediate lateral force acting on the chassis due to the abrupt change in direction. This initial spike is expected as the tires exerted lateral force in response to the steering input. Following this peak, the lateral acceleration decreases due to the vehicle speed decrease

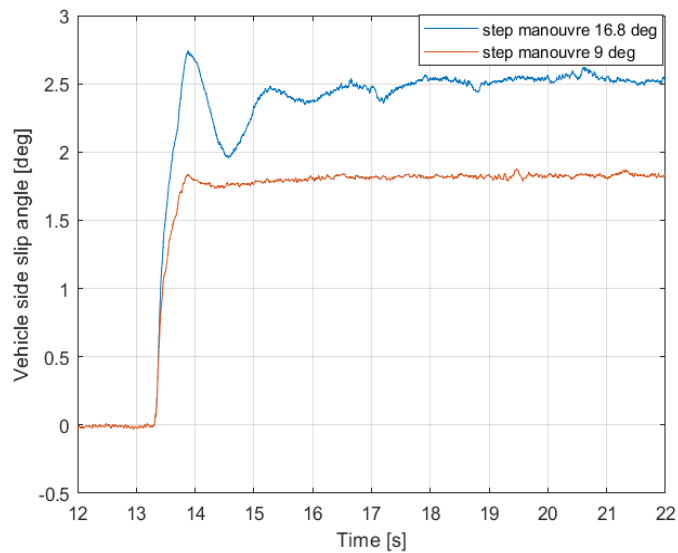


Figure 63: Vehicle side slip angle

The fig 63 represents the evolution of the vehicle's sideslip angle over time. Initially, before the steering input is applied, the sideslip angle remains near zero, indicating that the vehicle is moving straight without significant lateral motion. As expected, once the step input is introduced, there is a sharp increase in  $\beta$ , as far the 0.6 input is regarded, while regarding the 0.3 manoeuvre, the peak is smaller. This rapid rise is a direct result of the vehicle's sudden lateral force generation, causing a lateral slip as the tires do not exert enough

lateral force to handle the vehicle longitudinal speed. Following the peak, the sideslip angle begins to gradually decrease, stabilizing their mean values. This suggests that the vehicle reaches a more controlled turning state where the lateral and longitudinal forces balance out. The slow reduction of  $\beta$  over time indicates that the tires continue to adapt to the soft terrain conditions, likely due to terrain deformation.

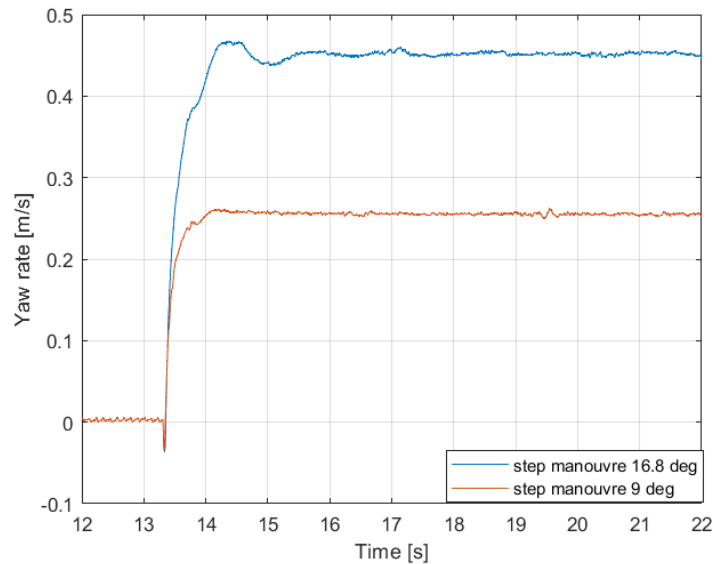


Figure 64: Yaw rate comparison

The curves in fig 64 compare the yaw rate response of the vehicle, one with a 16.8-degree steering input and the other with 9 degrees. The yaw rate represents the vehicle's rotational velocity around its vertical axis and is a key indicator of its stability during steering actions. In both cases, the yaw rate quickly rises after the step input, showing an initial transient phase where the vehicle reacts to the sudden steering command. The higher steering angle (16.8°) results in a greater yaw rate. The curve briefly overshoots before stabilizing, indicating some dynamic oscillations in the response. The lower steering input (9°) follows the same pattern but with a reduced magnitude, suggesting a more moderate rotation. The steady-state values in both cases demonstrate how yaw rate is directly proportional to steering input, provided the vehicle operates within linear handling limits.

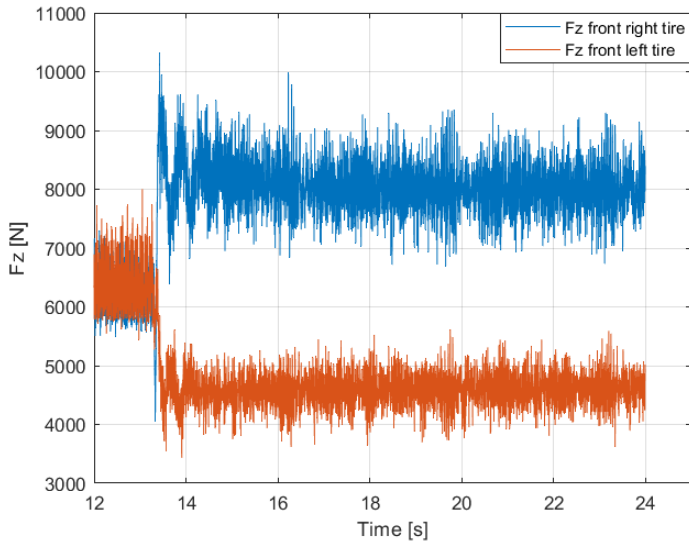


Figure 65: Vertical load on front tyres for steer input equal to 9 deg

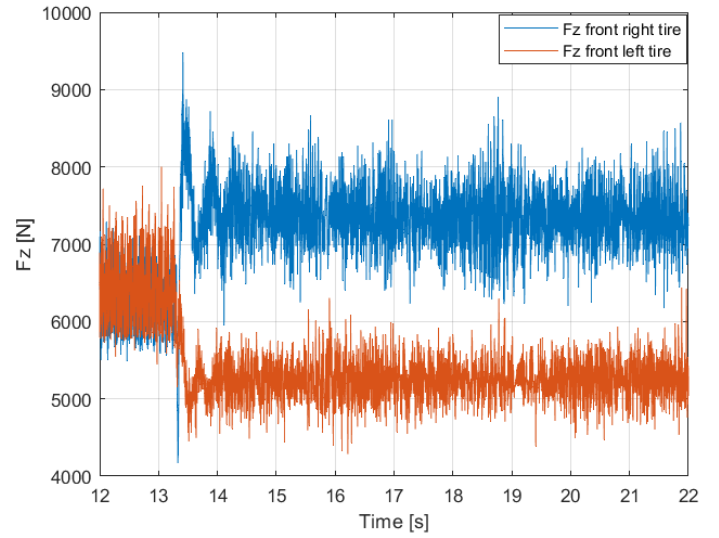


Figure 66: Vertical load on front tyres for steer input equal to 16.8 deg

The graphs illustrate the vertical load distribution on the front axle during the two step steer maneuvers. In both cases, a sudden load transfer occurs immediately after the steering input, causing an increase in the front right tire's load and a corresponding reduction on the front left tire. This behavior results from the lateral weight shift induced by the maneuver, as the vehicle leans towards the outer side of the turn. For the 9-degree steering input, the load variation is less pronounced, with the front right tire stabilizing around 7,500 N, while the front left tire drops to approximately 5,000 N. In contrast, the 16.8-degree step steer produces a more significant load transfer, as seen by the higher peak loads on the front right tire, reaching 9,000 N, while the front left tire unloads to around 4,500 N. Overall, the comparison highlights how higher steering angles amplify weight transfer, influencing tire grip and potentially affecting vehicle stability.

## 10.4 Ramp steer manoeuvre

The ramp steer maneuver is conducted to assess the vehicle's quasi-steady-state behaviour, whereas the step steer is used to analyse its dynamic response. In this simulation, the initial conditions are the same as in the step steer case, with the vehicle maintaining a constant throttle while the steering angle increases gradually at a controlled rate of around 1 deg/s. Generally, this kind of test is carried on not up to a fixed steering angle, but until axle saturation is reached. However, in this case the simulation time was kept the same as in the step steer test, even though full saturation was not achieved within this duration.

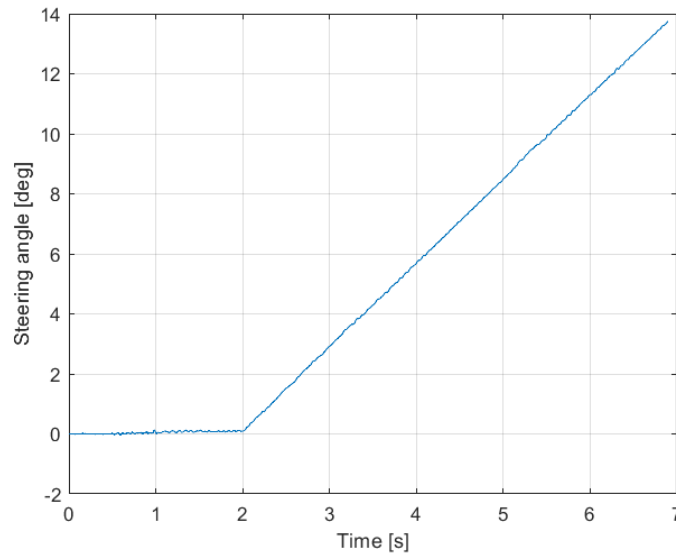


Figure 67: Steering angle

Compared to the step steer case, in the ramp steer case the vehicle begins to steer gradually after a short amount of time delay. With respect to step case, it is evident that the vehicle reaches a lower lateral displacement, travelling the same longitudinal distance. This is due to the low steer rate given as input.

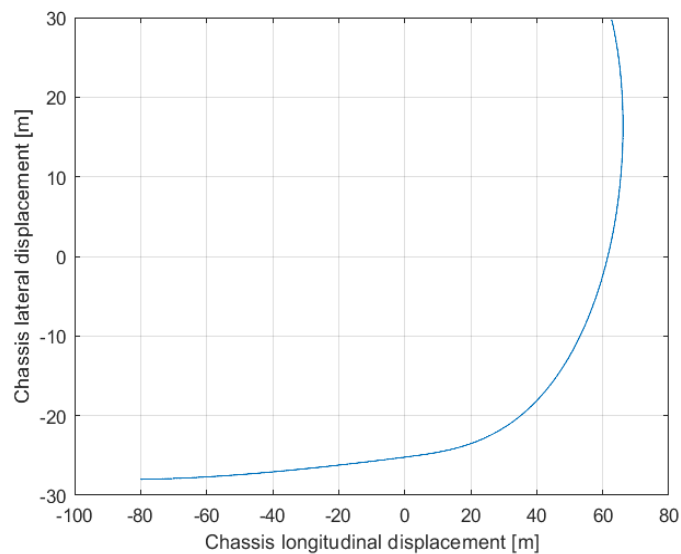


Figure 68: vehicle's trajectory.

Fig 69 displays the vehicle speed profiles for ramp steer. The steering input has been given at 13 seconds, as in the step case. Between 13 and 15 seconds, the velocity remains almost constant, due to the low steering angle, but after that time, the speed begins to decrease. The simulation stops at that time because the vehicle achieves the edge of the terrain. Several attempts were made to increase the terrain dimension, but the limit was already met.

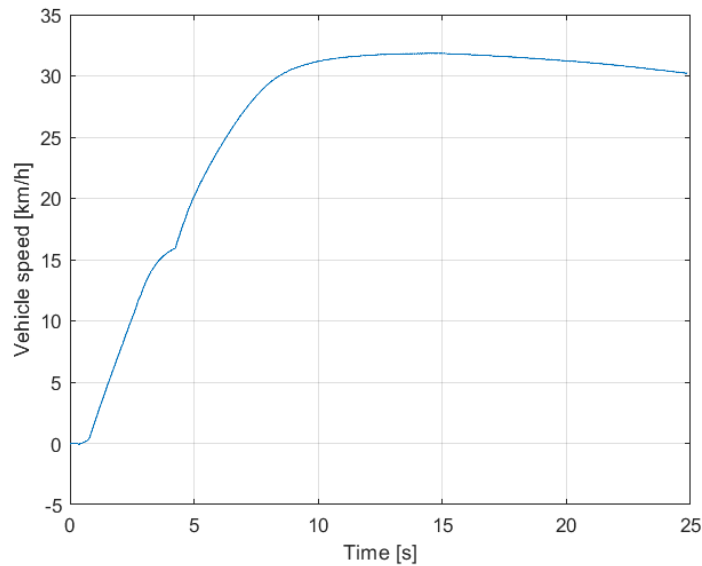


Figure 69: Vehicle speed

The ramp steer maneuver applies the input gradually, enabling lateral forces to build up progressively. This smooth transition minimizes excessive slip, leading to a controlled and gradual increase in response.

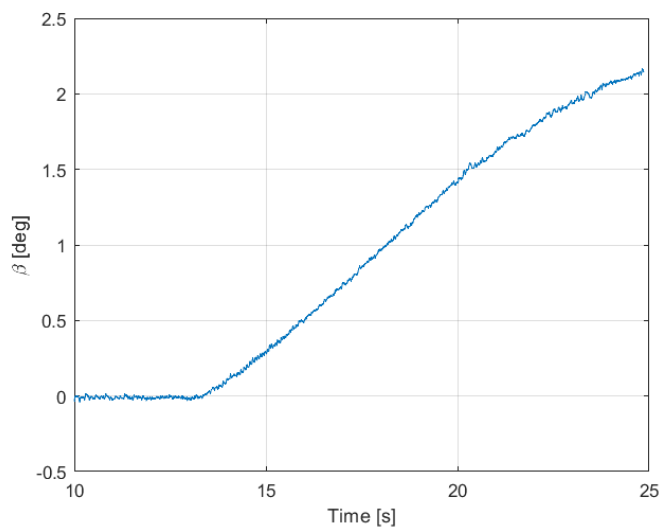


Figure 70: Vehicle side slip angle.

The graph represents the vehicle's roll angle, showing a progressive increase over time. This trend indicates that as the steering input gradually increases, the vehicle experiences a corresponding lateral acceleration, causing it to lean. The overall roll remains within a small range, suggesting that the maneuver is performed under controlled conditions without excessive instability

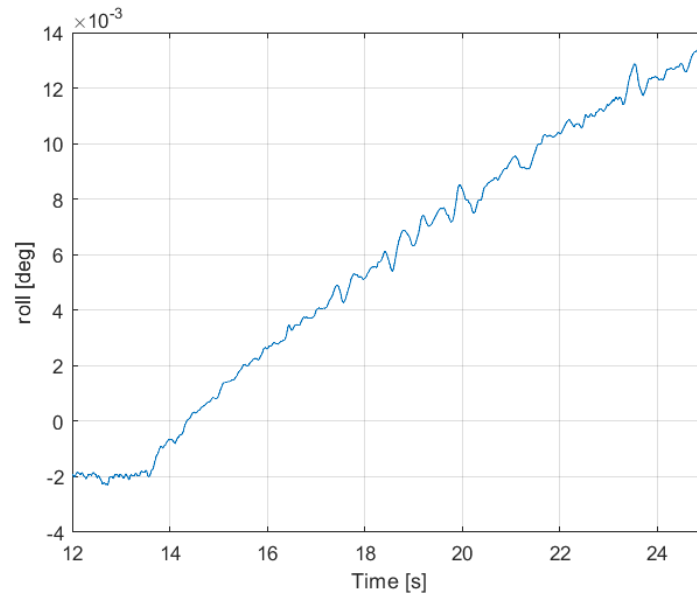


Figure 71: Vehicle's roll angle

The graph illustrates the vertical on the front left and front right tires during the ramp maneuver. As the steering input gradually increases, a noticeable load transfer occurs between the two tires. The front right tire experiences a progressive increase in vertical load, while the front left tire sees a gradual reduction. This behavior is characteristic of a vehicle undergoing lateral acceleration. As the vehicle steers to the left, weight shifts towards the right side due to the roll dynamics, increasing the load on the right front tire while unloading the left. The overall trend confirms a clear weight transfer, which is expected in a controlled steer ramp maneuver.

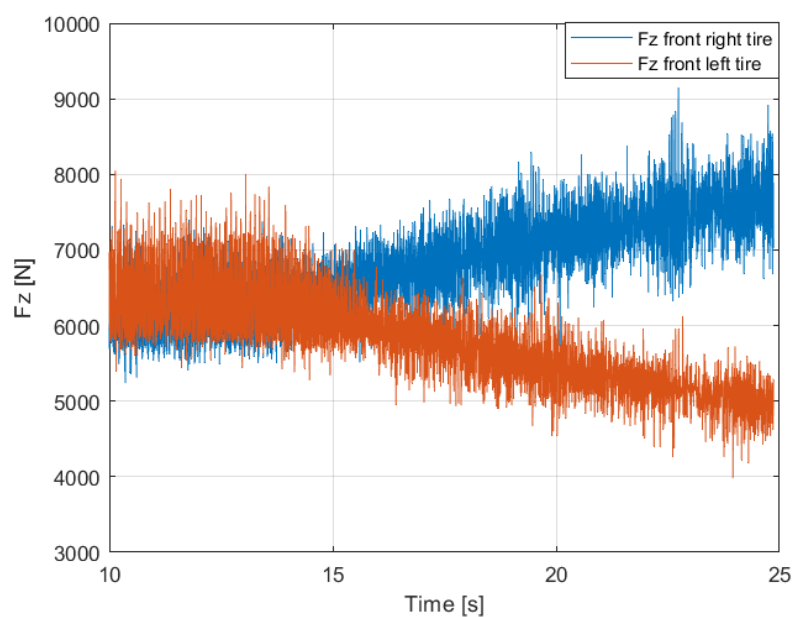


Figure 72: Fz front tires



## 11. Conclusion

The DEM simulation of the tire test rig did not produce the expected results due to some limitations. The factors which mostly affected the accuracy of the simulation are the following:

- The project was based on an existing demo, modified to extract relevant data, but the core code remained unchanged due to its complexity and the challenges of programming in C++.
- Many API-based functions lacked proper documentation, making it difficult to fully understand the internal workings of the simulation.
- The terrain model was simplified, using rigid, uniform particles in relatively small numbers (around 13,000), which is significantly lower than in comparable studies.

The SCM terrain model demonstrated significant improvements over the DEM approach, providing a more reliable and well-defined framework for analysis. Despite these advantages, some uncertainties remain regarding the sensitivity analysis and the influence of specific parameters.

- The empirical terrain model has well-characterized properties, leading to more consistent and interpretable results compared to the DEM approach.
- The sensitivity analysis gives good results, particularly the dependency on friction angle. Regarding  $k_\phi$  and  $n$ , a higher vertical load would mark better their influence at zero slip.
- The analysis of three terrains produced results that aligned well with theoretical expectations.

The simulations of the HMMWV on deformable terrain provided valuable insights into the vehicle's dynamic response under different driving conditions, including step acceleration, step braking, step steer, and ramp steer. These tests highlighted how terrain deformability influences vehicle performance, particularly in terms of traction, braking, and lateral stability.

- In the step acceleration test, the speed profile exhibited non-linear behaviour due to the gradual increase in engine torque and the gear shift occurring around five seconds. This was confirmed by the longitudinal acceleration graph, where oscillations reflected the interaction between the powertrain and terrain resistance. The normal load distribution showed a clear weight shift, with an increased load on the rear axle due to acceleration-induced weight transfer.
- In the step braking test, the vehicle experienced a rapid speed reduction after braking was applied at around five seconds. The normal load distribution behaved oppositely to the acceleration case, with the front axle carrying a greater vertical load. Braking also led to oscillatory longitudinal forces on the rear axle, eventually causing wheel lock-up toward the end of the simulation.
- The step steer test demonstrated the vehicle's lateral dynamics in response to a sudden steering input. The trajectory followed a curved path, with lateral acceleration increasing in correspondence with the applied steering input. The side slip angle peaked before stabilizing, following the ISO frame convention. The lateral forces on the front wheels showed a significant difference between the left and right tires, with the outer wheel generating greater forces due to higher normal loading.
- The ramp steer test, in contrast to the step steer, produced a smoother trajectory with a progressive increase in lateral acceleration. The side slip angle developed more gradually, and force distributions across the front tires reflected the steady nature of the manoeuvre. Compared to the step steer case, the ramp steer exhibited a more controlled response with reduced transient effects.

Through these analyses, it was evident that terrain deformability plays a crucial role in influencing vehicle behaviour. The forces exchanged between the tires and the soft soil, along with the variations in normal loads, significantly affected traction, braking efficiency, and lateral stability. Future work could focus on refining the terrain model by implementing more complete codes, extending the study to different vehicle configurations. Additionally, experimental validation of the numerical results would further enhance the credibility of the simulation framework.

## 12. Bibliography

- [1] Tasora, A., Mangoni, D., Negrut, D., Serban, R., and Jayakumar, P., 2019, "Deformable Soil With Adaptive Level of Detail for Tracked and Wheeled Vehicles," *Int. J. Veh. Performance*, 5(1), pp. 60–76.
- [2] Z Janosi and B Hanamoto. The analytical determination of drawbar pull as a function of slip for tracked vehicles in deformable soils. In *Proc of the 1st int conf mech soil-vehicle systems. Turin, Italy*, 1961.
- [3] Tasora, A., Mangoni, D., Negrut, D., Serban, R., and Jayakumar, P., 2019, "Deformable Soil With Adaptive Level of Detail for Tracked and Wheeled Vehicles,"
- [4] Bekker, M. G., 1956, *Theory of Land Locomotion; the Mechanics of Vehicle*
- [5] D. Negrut, R. Serban, and A. Tasora, *ASME Journal of Computational and Nonlinear Dynamics* 0, 0 (2017)
- [6] Zhang, R., Heuvel, C.V., Schepelmann, A., Rogg, A., Apostolopoulos, D., Chandler, S., Serban, R., Negrut, D.: A GPU-accelerated Simulator for the DEM Analysis of Granular Systems Composed of Clump-shaped Elements.
- [7] Mobility, University of Michigan Pres, Ann Arbor, MI [SCM]R. Krenn and A. Gibbesch. Soft soil contact modeling technique for multi-body system simulation. In *Trends in computational contact mechanics*, pages 135–155. Springer, 2011.
- [8] R. Serban, J. Taves, and Z. Zhou. Real time simulation of ground vehicles on deformable terrain. In *Proceedings of ASME IDETC/CIE 2022*, St. Louis, MO, 2022.
- [9] Role of inter-particle friction in granular materials under three dimensional conditions — PARTICLES 2019 Shuhan. Yang, Wei. Zhou, Jiaying. Liu, Tianqi. Qi , Gang. Ma
- [10] Fleischmann, J., Serban, R., Negrut, D., Jayakumar, P.: On the importance of displacement history in soft-body contact models. *Journal of Computational and Nonlinear Dynamics* 11(4), 044502 (2016)
- [11] "Parameter Identification of a Planetary Rover Wheel-Soil Contact Model via a Bayesian Approach", A.Gallina, R. Krenn et al.
- [12] Krenn, Rainer, and Gerd Hirzinger. "SCM—A soil contact model for multi-body system simulations." (2009).
- [13] Zhang, Yuemin, et al. "Using high fidelity discrete element simulation to calibrate an expeditious terramechanics model in a multibody dynamics framework." *Multibody System Dynamics* (2025): 1-40.
- [14] Smith, William, et al. "Comparison of discrete element method and traditional modeling methods for steady-state wheel-terrain interaction of small vehicles." *Journal of Terramechanics* 56 (2014): 61-75.

## Appendix

The logic behind the MATLAB code shown below is quite straightforward: after opening the txt file called “TireForce\_Torque\_history”, each row of the file has been read and the values extracted have been saved in empty variables. The advantage of using MATLAB is that it doesn’t need to set the size of the variables beforehand, making data extraction from the file easier. Finally, all the vectors are saved in a .MAT file and then loaded in a new script for the creation and the analysis of the graphs.

```
clear all
close all
clc

% It opens the text file in read mode
fid = fopen('TireForce_Torque_history.txt', 'r');

% It initializes empty vectors for the values to be extracted
Time_t1200 = [];
TireForce_baseline_ym20_s02pos = [];
Torque_baseline_ym20_s02pos = [];
wheel_angular_speed_baseline_ym20_s02pos=[];
com_position_baseline_ym20_s02pos=[];
speed_baseline_ym20_s02pos=[];
longitudinal_slip_baseline_ym20_s02pos=[];
pull_force_baseline_ym20_s02pos=[];
% alpha_angle=[];
%camber_angle = [];

% it reads the file line by line
tline = fgetl(fid);

% It begins a while loop in order to scan the line and to extract the desired values
while ischar(tline)

data = textscan(tline, ['Time %f: TireForce(%f, %f, %f),' Torque(%f, %f, %f),COM(%f, %f, %f),'Longitudinal slip (%f),
Carrier long speed(%f),' Angular speed (%f), Pull force (%f)']); % camber(%f), alpha (%f)

% It extracts the values and adds them to the vectors
Time_t1200 = [Time_t1200; data{1}];
TireForce_baseline_ym20_s02pos = [TireForce_baseline_ym20_s02pos; data{2}, data{3}, data{4}];
Torque_baseline_ym20_s02pos = [Torque_baseline_ym20_s02pos; data{5}, data{6}, data{7}];
com_position_baseline_ym20_s02pos = [com_position_baseline_ym20_s02pos; data{8},data{9},data{10}];
longitudinal_slip_baseline_ym20_s02pos = [longitudinal_slip_baseline_ym20_s02pos;data{11}];
speed_baseline_ym20_s02pos = [speed_baseline_ym20_s02pos;data{12}];
wheel_angular_speed_baseline_ym20_s02pos = [wheel_angular_speed_baseline_ym20_s02pos;data{13}];
pull_force_baseline_ym20_s02pos =[pull_force_baseline_ym20_s02pos;data{14}];
% camber_angle = [camber_angle;data{12}];
% alpha_angle = [alpha_angle;data{14}];

% Reads the next line
tline = fgetl(fid);
end
% It closes the file
fclose(fid);
```

```
% It saves the vectors in a MAT file
```

```
save ('baseline_ym20_s02pos.mat', 'Time_t1200', 'TireForce_baseline_ym20_s02pos',  
'Torque_baseline_ym20_s02pos','com_position_baseline_ym20_s02pos','longitudinal_slip_baseline_ym20_s02pos','sp  
eed_baseline_ym20_s02pos','wheel_angular_speed_baseline_ym20_s02pos','pull_force_baseline_ym20_s02pos');  
% 'camber_angle','alpha_angle'
```

“Camber\_angle” and “alpha\_angle” variables are commented out because they were not considered in that specific case.