PERFORMANCE ASSESSMENT OF MULTI-ARTICULATED FLEXIBLE VEHICLE PLATFORMS IN REALISTIC ROAD INFRASTRUCTURE MODELS




N. MUTHAKANA, MSc, Researcher at HAN University of Applied Sciences.

D. DEVASIA, MSc, Researcher at HAN University of Applied Sciences.

S. AJAYKUMAR, MSc, Researcher at HAN University of Applied Sciences.

K. KURAL, MSc, PhD., Senior Researcher at HAN University of Applied Sciences.

Abstract

Digital twinning of logistics vehicles can enhance decision making, to determine the right vehicle to use for the right duty (e.g. an EMS vehicle's ability to negotiate the roundabouts in a route) and predict future tyre wear using vehicle telemetry (i.e. how much rubber is worn out and how will it wear based on current usage pattern). Multibody models of heavy-duty vehicles are suitable for such digital twinning of vehicle performance and operation. This paper presents an automated workflow to simulate heavy-duty vehicle performance by modelling real infrastructure and using realistic simulation inputs. Roundabouts in the planned missions are identified using map data features and their road width is extracted from satellite images using computer vision techniques. FMS data does not contain tyre forces to predict tyre wear, these are calculated by the digital twin models using MF Tyre models for individual tyres and FMS driver inputs like steer and throttle. The simulators used for this purpose are packaged and deployed into FMU models and Docker containers.

Keywords: Digital Twin, Zero Emission, Heavy-duty Vehicles, Roundabouts, Computer Vision, Model Predictive Control

1. Introduction

As part of the Green Deal, Europe aims for carbon-neutrality by 2050, with a 55% CO_2 reduction by 2030. The road transport sector targets a 30% reduction in CO_2. The European Union has been revising legislation to allow for cross-border longer and heavier vehicle combination usage. Heavier vehicles allow for an easier transition from internal combustion to cleaner yet heavier Battery and Fuel Cell Electric vehicles (e-tractors, e-trucks, e-dollys and e-trailers, a.k.a Zero Emission Vehicles or ZEVs). Longer vehicle combinations as shown in Figure 1 have a big impact in reducing emissions, due to the various advantages of transporting more cargo with a single towing vehicle. In combination with zero emission powertrains, such vehicles will pave the way for a cleaner road transport sector. The ZEFES (Zero Emissions flexible vehicle platforms with modular powertrains serving the long-haul Freight Eco System) project, funded by the European Union's Horizon program, explores real-world adoption of these longer and heavier vehicles. The zero-emission powertrains are out of scope of this paper.

One of the objectives of the ZEFES project is to develop a digital twin platform consisting of various digital tools with user interfaces (ZEFES, 2024). The HAN University of Applied Sciences contributes to two of the tools in ZEFES, namely the right vehicle for the right duty (RVRD) and the predictive maintenance tools. Determining the RVRD is necessary since long vehicle combinations like the EMS (European Modular System) 1 & 2, as shown in Figure 1 can be up to 25.5 meters and 34.5 meters long respectively. For simplicity, we refer to these vehicles as LHVs (Longer and Heavier Vehicles). Given the long length of these vehicles, an operator needs to know whether the vehicle combination that they intend to use can negotiate the route that needs to be driven. This introduces a route planning process that accounts for various factors. Different vehicle combinations have different turning circle radii, which depends on their length and pivot points. Meanwhile, ZEVs need to be routed based on charging point locations. Another layer in decision making is the amount of payload that needs to be dropped off at various points. The RVRD tool takes all the relevant factors into account to provide information about the ideal vehicle combination to use for the mission.

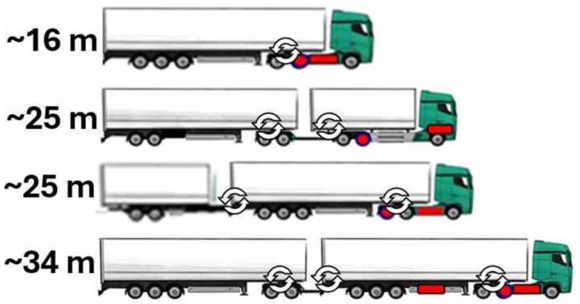


Figure 1 – Various vehicle combinations used in ZEFES

Although LHVs provide significant advantages for transport efficiency, they face several issues when navigating roundabouts and sharp turns due to their size and manoeuvrability capabilities. Matuszkova and others (2018) and Dijkstra (2021) identified for instance, the following key risks:

1. Increased space requirement in bends: The large turning radius can cause overshooting or encroachment on opposing lanes leaving little to no room for other road users in some cases.
2. Manoeuvrability in roundabouts: Regardless of roundabout exit, LHVs have to perform multiple consecutive manoeuvres to complete the turn, increasing the likelihood of colliding with the boundaries or the driver losing focus on other road users, including cyclists and pedestrians.

These risks have led to accidents as well, which have been documented (Aarts et al., 2011). Given that Europe's road infrastructure is increasingly relying on roundabouts, current estimates put the number of roundabouts in Europe to be greater than 100000 (The Mayor, 2024). Although several countries have legal restrictions regarding the parts of infrastructure that can be accessed by LHVs, it is not guaranteed that the swept path of the LHVs fits the infrastructure. This can be due to the fact that infrastructure designed before the introduction of such vehicles were not designed with LHVs in mind. The RVRD tool is meant to identify parts of the infrastructure that might pose such problems for the desired LHVs such that a route planner can efficiently assign the right vehicle combination for the missions to be performed.

Another tool of interest in this paper is Predictive Maintenance, where real-time data from fleets will enable tyre wear and battery health estimations (ZEFES, 2024). The demonstrator LHVs operating daily for a period of a few months will provide the data necessary for the tools to function. Since each tyre in the vehicle combination experiences different forces given its location, the forces used as input to any tyre wear estimator also needs to be calculated individually based on the vehicle's trajectory. Hence, the digital twins of the vehicles need to be augmented with such data signals. Further reading on the workflows of each tool and the architecture to facilitate the simulations are elaborated in the deliverables of the ZEFES project (ZEFES, 2024).

1.1 Vehicle-infrastructure interaction simulation

The focus of this paper is the creation and deployment of LHV simulators developed in MATLAB. The simulators take for example the desired route/mission as an input and outputs whether a desired LHV can fit in that route or not. This involves modelling of the vehicles and the real-world infrastructure, packaging the models into a simulator tool for interaction within a workflow and packaging the whole application as a standalone tool.

Vehicle modelling is done using multibody formulation of vehicle dynamics. HAN University of Applied Sciences and the Technical University of Eindhoven developed the commercial vehicle library using the Simscape Toolbox of MATLAB (MathWorks, 2024a), consisting of modular vehicle units (tractors, trailers, dollys, trucks, etc.) that can be combined into any desired heavy vehicle combination as shown in Figure 2. Rigid bodies and their inertias are linked to each other through joints with varying degrees of freedom. The physics of chassis flexibility, suspension dynamics and roll behaviour are modelled in the units along with drivelines. These models were validated with LHV tests by Kural and others (2013) and updated to the new Simscape Toolbox by Ajaykumar (2022). The Magic Formula Tyre Force and Torque equations (for combined slip and turn slip effects) are implemented to simulate the non-linear behaviour of the tyre-road interactions (MathWorks, 2024b).

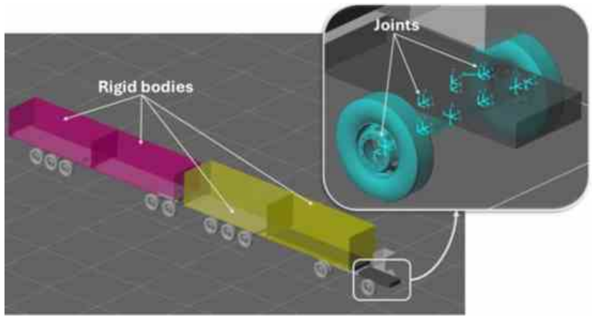


Figure 2 – Visualisation of a Simscape Multibody model

The fidelity of these models cannot be captured by traditional swept path tools as used in Gkountzini and others (2020) and Godavarthy and others (2016) for similar swept path analysis. Furthermore, the swept path tools available require the user to define the road infrastructure manually for each analysis. Vehicle kinematics and their driving trajectories are handled by the software but analysing a route, finding the roundabouts and their dimensions as well as determining the feasibility of manoeuvrability, in an automated way in real-time is not possible.

In ZEFES, we first tackle roundabouts in a route. The evaluation of LHV manoeuvrability at roundabouts present in any arbitrary route, with an automated workflow requires the identification of actual roundabout sizes and their modelling. A survey of roundabout design comparison among various countries in Europe shows that the specifications are largely similar, but with varying radii of entry and exit lanes (Kennedy, 2008). Particularly, The Netherlands has the smallest roundabout lane width compared to its other European country equivalents.

1.2 Research objectives

Existing swept path analysis methodologies are manual and hence not deployable to any automated digital twin tool that can compute results within minutes. Furthermore, predictive maintenance of tyres requires tyre forces data, which is typically not available in the FMS (Fleet Management System) data streams. Hence, the digital twin of LHVs require additional computation of such data signals. In view of these issues, this paper explores the following:

- How can the boundaries of real roundabouts be identified and modelled to represent actual infrastructure?
- How can an LHV be simulated to represent truck driver behaviour such that the available road width is used efficiently and realistically in limited time?
- How can tyre forces be computed from commonly available FMS data?

Section 2 describes the detection of roundabout locations and modelling of roundabout boundaries. Section 3 explores various techniques of reference path generation to negotiate a roundabout, and swept path analysis. Section 4 describes the workflow used to package the simulators and showcases their usage to calculate tyre forces.

2. Critical section detection and modelling

The detection of locations of roundabouts along a route is straightforward, where open source map data providers like OpenStreetMap have geotagged information of road segments. The GPS (Global Positioning System) coordinates of a route to be assessed are put through a simple heading angle variation check, which groups waypoints that have a significant change in heading; these groups are likely to be a sharp turn or a roundabout. The road segment data from providers like OpenStreetMap is checked for each of the identified groups to tag the groups that represent a roundabout (taginfo, 2024). Once a roundabout's location has been identified, the satellite image of the roundabout can be obtained, also from similar sources. This paper sources the images from MATLAB's Mapping Toolbox. This following subsections describe how the boundaries of the roundabout are modelled based on the satellite images obtained.

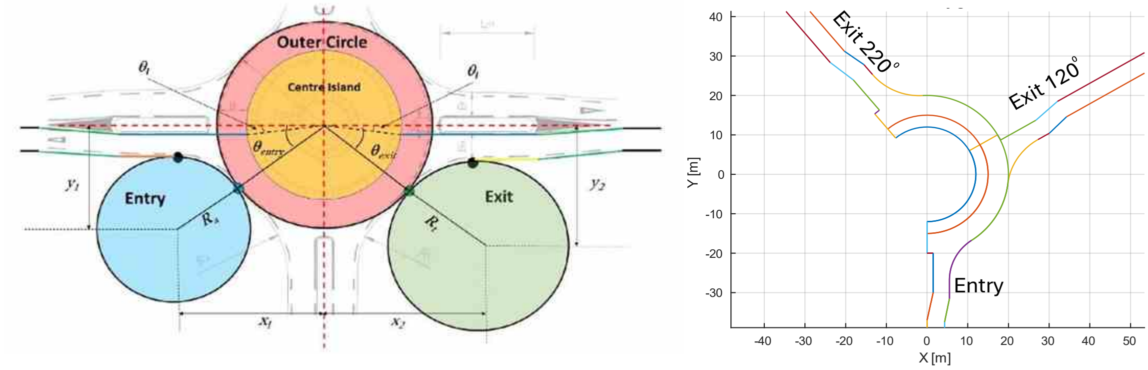
2.1 Satellite image gathering

Figure 3 shows examples of satellite images obtained through the Mapping Toolbox. The size of the images queried using the *readBasemapImage()* function are determined by the identified waypoints grouping, making sure that all the grouped waypoints are within the image. This guarantees that the whole roundabout or at least a significant portion of it lies within the queried image. A notable feature of the images is the difference in image resolution. Western European countries tend to have sharper images than northern and southern Europe.

Figure 3 – Roundabout images from different countries

2.2 Roundabout modelling

Roundabout designs vary per country, but they generally follow the same boundary structure (Kennedy, 2008). The design guidelines of roundabouts describe that the boundaries can be parametrically linked to the radius of the central island (see Figure 4 (a)). The lane widths leading into the roundabout, the curvature of the lanes, and the width of the splitter islands (which separate incoming and outgoing traffic) are all influenced by the radius of the roundabout. Figure 4 (b) shows the individual boundary elements that make up the infrastructure for various exit angles. Hence, the parameterisation of the boundary elements allows the creation of any roundabout configuration. This fact was verified by measuring various roundabouts in The Netherlands by Google Earth and also physically measuring all the arcs by means of a roller measuring wheel. Furthermore, by detecting the centre island, the overrun area (ring around the centre island) is also known from the guidelines and taken as drivable space in the algorithm.



(a) (b)

Figure 4 – (a) Parameterisation of roundabout boundaries, (b) Examples of generated roundabout boundaries

2.3 Roundabout size detection

The most identifiable characteristic of roundabouts in satellite imagery are the centre islands. Therefore, accurately detecting the radius of the central island enables the modelling of the roundabout boundaries for simulation, since the rest of the roundabout features can be deduced from the centre island size. The images are queried with the coordinates of the corners of the images; thus, the scale of the images is determined by finding the pixels per meter for the downloaded image.

$$scale = mean\left(\frac{N_{x\_pixels}}{Length_x}, \frac{N_{y\_pixels}}{Length_y}\right) \quad (1)$$

In Equation 1, N_{x_pixels} and N_{y_pixels} represent the number of pixels in the horizontal and vertical directions of the images, respectively, while $Length_x$ and $Length_y$ denote the actual lengths of the picture frame in meters.

Once the scale of the images is determined, the next step is to prepare the image for circle detection. The Figure 5 shows the image processing pipeline where, the raw image is first blurred, cropped to a general area of interest, greyed, its edges extracted and finally masked with a more refined area of interest. These steps are detailed below:

- A 2D Gaussian filter is applied to blur the image to reduce noise by smoothing out local regions of higher intensity pixels while preserving the edges of important details.
- Based on the GPS coordinates passing through the roundabout, the image is cropped to focus the area of interest.

- The image is converted to grayscale, reducing the three-channel colour information to a single intensity channel. This simplifies the data and speeds up the following edge detection process.
- The Canny edge detection algorithm (using MATLAB's *edge( )* function) is then employed to identify edges within the image. The edges will help distinguish the centre island circle from the road. The threshold values for the edge detection were determined by testing various images from across Europe. The thresholds for Canny edge detection of [0.01,0.2] were found to have a good balance, ensuring that edges were detected effectively across all roundabout images without unnecessary noise.
- The unimportant edges are then further removed by applying a mask on the image. This mask is meant to isolate the centre island, which is done by using the GPS coordinates of the driving route along with the reverse of it as shown in the Figure 3, 'Masked edges' image. For instance, a first exit roundabout path along with the reverse of it (third exit) can generate the mask boundaries. In the instance of Figure 3, the route shows a second exit, both in forward and reverse direction, thus isolating the centre island.

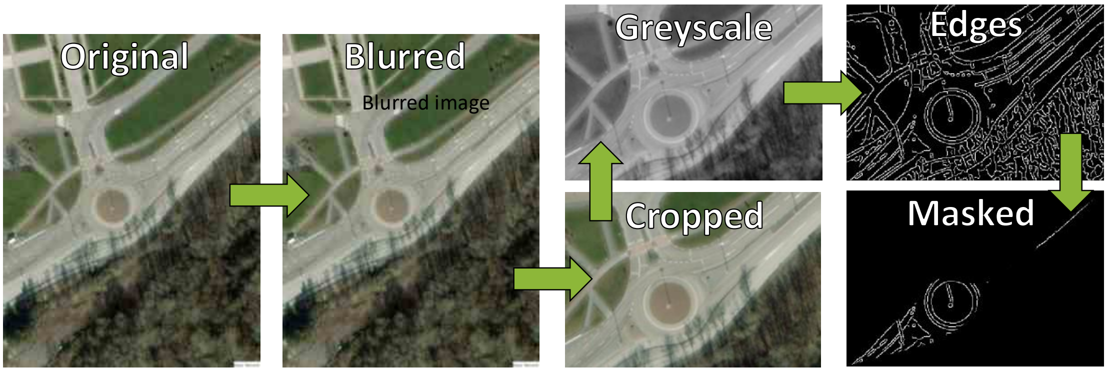


Figure 5 - Image processing pipeline

This procedure results in an image which is ready for the circle detection algorithm. The circle detection algorithm utilizes a convolutional approach to identify circles within an image as used by Atherton (1999). Convolution is a mathematical operation used to extract features from data, in this case black and white images.

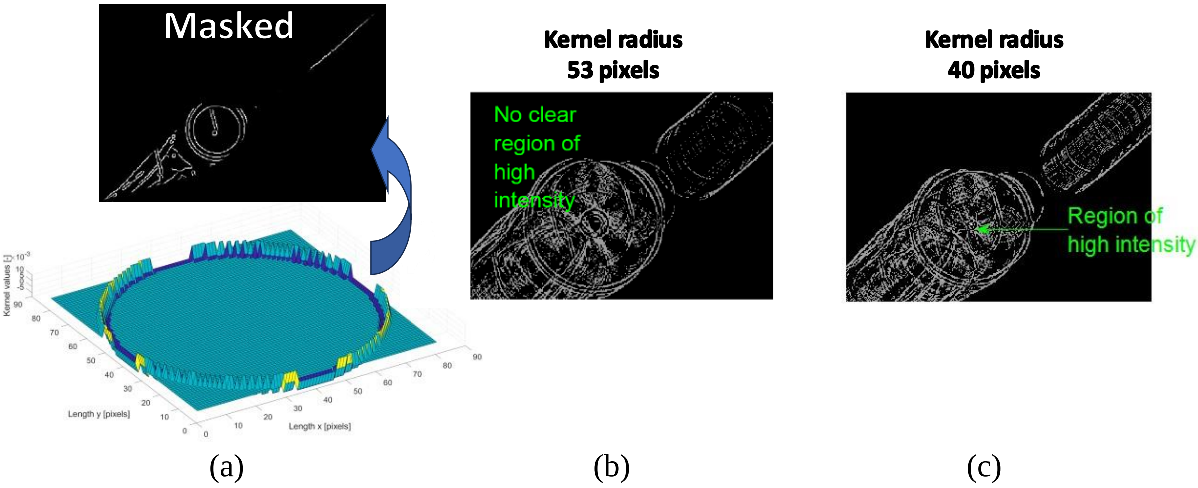


Figure 6 – (a) Applying kernel on masked image, (b) convoluted image with 53 pixel kernel, (c) convoluted image with 40 pixel kernel

It involves sliding a kernel (a small matrix that resembles a circle) over the input data and at each position as shown in Figure 6 (a), multiplying the values of the kernel with the corresponding values of the input. The sum of these products is recorded to create a new feature map called convoluted image as shown in Figure 6 (b). This operation captures spatial relationships, allowing the algorithm to detect circles in the image.

In the convoluted images, each white pixel represents the potential centre of a circle with a radius matching that of the applied kernel. A series of kernels, corresponding to radii ranging from 8 m to 40 m (range of possible radii), are used on the image. The kernel with the best match to the image is selected. The intensity of each pixel in the convoluted image reflects the number of edge-detected points contributing to the detection, thereby indicating the strength and likelihood of the circle's presence as shown in Figure 6 (c) where the brightest point shows the location of the centre. The kernel size of 40 pixels corresponds to a radius of 10.4 meters in this case. Taking the images of Figure 3 as examples, the identified circles, along with their radii, are overlaid in Figure 7. These images show the original image alongside the edges of the image and the circle. The convolution algorithm is able to pick up the circles in a wide variety of image resolutions due to the tuning of edge detection.

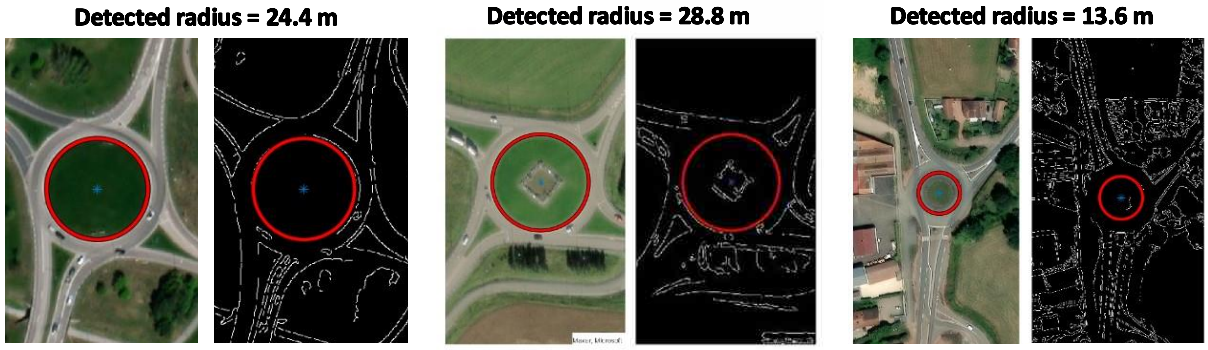


Figure 7 - Final results of circle detection

The roundabout entry and exit lanes can be of varying angles to the centre as seen from the examples in Figure 7. Hence, to be able to reconstruct the rest of the roundabout boundaries after detecting the centre island radius, the entry and exit angles of the roundabout have to be estimated. Using the GPS coordinates of the route, the angles can be identified based on the difference between the entry and exit lane heading angles. The result is shown in Figure 8 where the inner and outer boundaries in red match the 2nd exit roundabout manoeuvre. The circle detected in this example was 9.7 meters, and the entry and exit lane radii were 12 and 15 meters respectively.

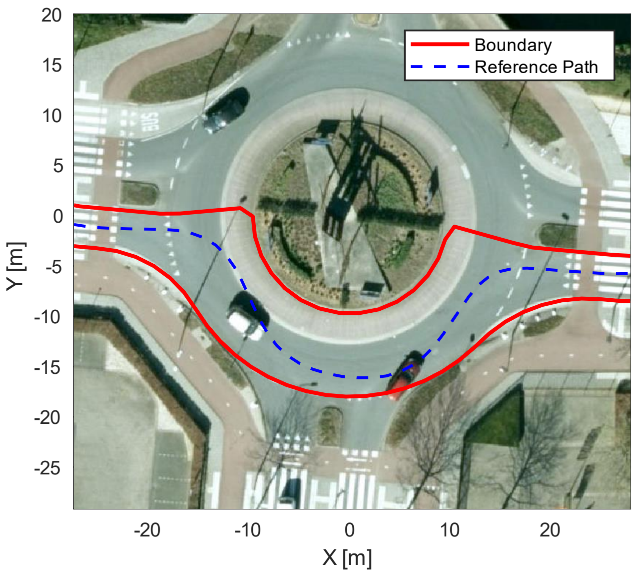


Figure 8 – Roundabout boundary reconstruction

3. Swept path analysis

To simulate the manoeuvrability of any LHV at a roundabout, it is necessary to replicate the real behaviour of a driver. This is achieved by generating a realistic reference path for the vehicle models to follow. Figure 8 shows an example of a reference path (blue dotted line) meant for the steer axle trajectory, which utilizes the available road width in the roundabout. We explore two key methods to generate such a path, namely, controller optimization and boundary based manual definition. The reference has to realistic in the context of a driver's ability to apply the steering wheel inputs to negotiate the roundabout. A highly optimized reference path that can theoretically use as much road width as possible could result in a successful manoeuvre, but does not guarantee practical feasibility (e.g. driver has a certain amount of rate of steer angle application). Hence, the methods to generate the path needs to take into account the practical limitations of driver error.

3.1 Comparison of reference path generation

The Figure 9 pictorially shows the process of generating a reference path for a roundabout. Figures 9 (a) and (c) depict the process of path optimization by using a Model Predictive Controller (MPC). An MPC uses a mathematical model of a system (in this case LHVs) to predict the future behaviour of the system and optimize control actions over a specified time horizon. By solving an optimization problem at each time step (for the present state and future possible states), the MPC adjust control inputs to minimize errors while respecting system constraints. An MPC is chosen for this comparison since the system constraints that can be specified make sure that the controller acts as close as possible to a real driver.

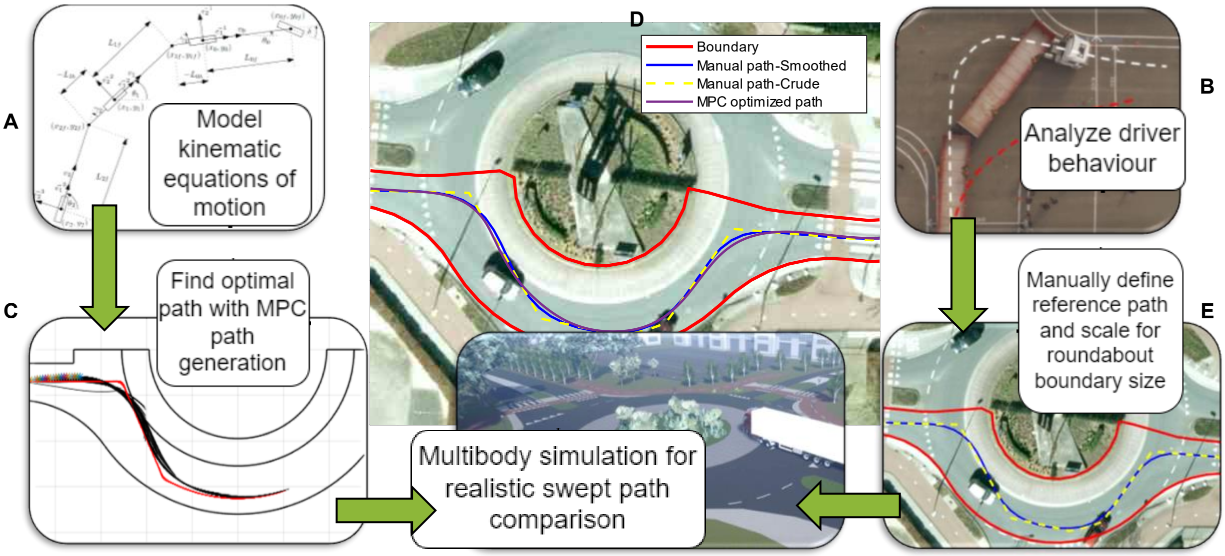


Figure 9 – Reference path generation methods

The main building block of an MPC is the cost function as shown in Equation 2.

$$J\big(x(k),u(k)\big) = J_y\big(x(k)\big) + J_u\big(u(k)\big) + J_{\Delta u}\big(u(k)\big) \qquad \forall k \in \{1,2,\dots N\} \qquad (2)$$

Where J is the overall cost; J_y, J_u, and $J_{\Delta u}$ are costs related to output states, system inputs and system input rate tracking respectively. The states of the vehicle such as axle positions, yaw angles, and articulation angles are denoted in the state vector x which in turn define the mathematical model of the system, the inputs u to the mathematical model are the steer angle and speed for every time step k. The time steps are related to the prediction horizon of N steps.

We define constraints to the states of the system, such as an articulation angle limit to ensure that the vehicle model does not jack-knife. The inputs to the system are constrained to ensure a logical speed and realistic range of steer angle is applied. The rate of change of the inputs are constrained to capture a realistic application of speed and steer angle. These constraints ensure that the MPC controls the vehicle trajectory in a realistic manner. The red line shown in Figure 9 (c), is the provided ‘initial guess’ path. This is the reference input to the steer axle position that the MPC tries to generally follow, which is the programmer’s way to hinting to the controller what to do. The black lines in the image are the computed possible trajectories at every time step considering the initial guess and the set constraints. These lines are generated to find the optimal trajectory while making sure the vehicle envelope does not cross the roundabout boundary.

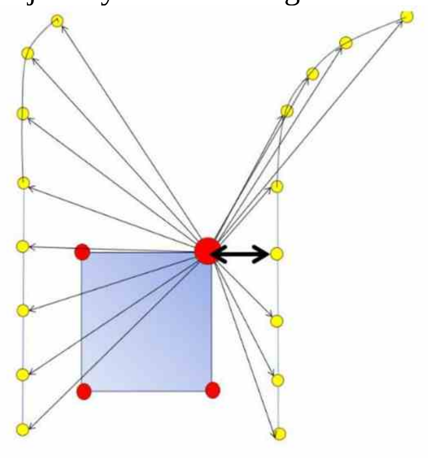


Figure 10 – Distances from vehicle envelope to boundaries

To achieve this, at every time step, an inequality condition is computed simultaneous to the cost function.

$$c_{ineq}(-d) \leq 0 \qquad (3)$$

Where c_{ineq} is a vector of distances d from the vehicle extremities to the lane boundaries as shown in Figure 10. The distance from every corner of the vehicle envelope of every vehicle unit in the LHV combination is to the boundary is calculated and used in the inequality function. The computation of Equations 2 and 3 for the entire prediction horizon results in the trajectories of the prediction in Figure 9 (c).

Conversely, Figures 9 (b) and (e) follow a more qualitative approach. By manually defining a general reference trajectory as shown by the dotted yellow line in Figure 9 (d), we attempt to use as much of the road width as possible. This manual definition happens to be quite similar to the initial guess provided to the MPC in Figure 9 (c). Hence, by smoothing the manually defined line, we get the blue reference path in Figure 9 (d). This approach is simple, yet can be scaled to different roundabout sizes by parameterizing the manually defined points with the roundabout boundary features.

By analysing the swept paths generated by multibody simulations of both reference path sources, we note that the MPC has better flexibility across various LHV types since the state function $x(k)$ is defined for every LHV type, and hence the MPC optimizes the path for the envelope of the vehicle type and possibly also trailer steered-axles. With the manually defined path, this flexibility is not possible without added effort to manually tune the path for every LHV type. The noteworthy disadvantage of using the MPC is the computational effort and time, which can take up to 3 minutes for the computation of one roundabout path. Considering that routes can have tens of roundabouts, the computation time is a bottleneck.

The comparison shows that a fusion of both approaches is required to be able to deploy such an algorithm for efficient operation. A manually defined generic path that is further optimized by a leaner MPC to account for individual vehicle combination’s dynamics to generate a solution that is kinematically viable and respects the physical system constraints could reduce the current computational effort of the MPC and allow for easier scaling up. This improvement is foreseen since the MPC will be closer to the optimal solution by using a better informed initial guess.

3.2 Swept path compliance

Although the MPC provides an optimized reference path based on the vehicle envelope, it does not take into account tyre slip, road friction, roll, etc., since the state function $x(k)$ is a kinematic representation of the vehicle movement. Hence, the MPC generated paths are validated with multibody simulations. Multibody simulation models offer the ability to measure and log any vehicle states of interest, so the simulators are designed to output the tyre outer wall positions during roundabout manoeuvring.

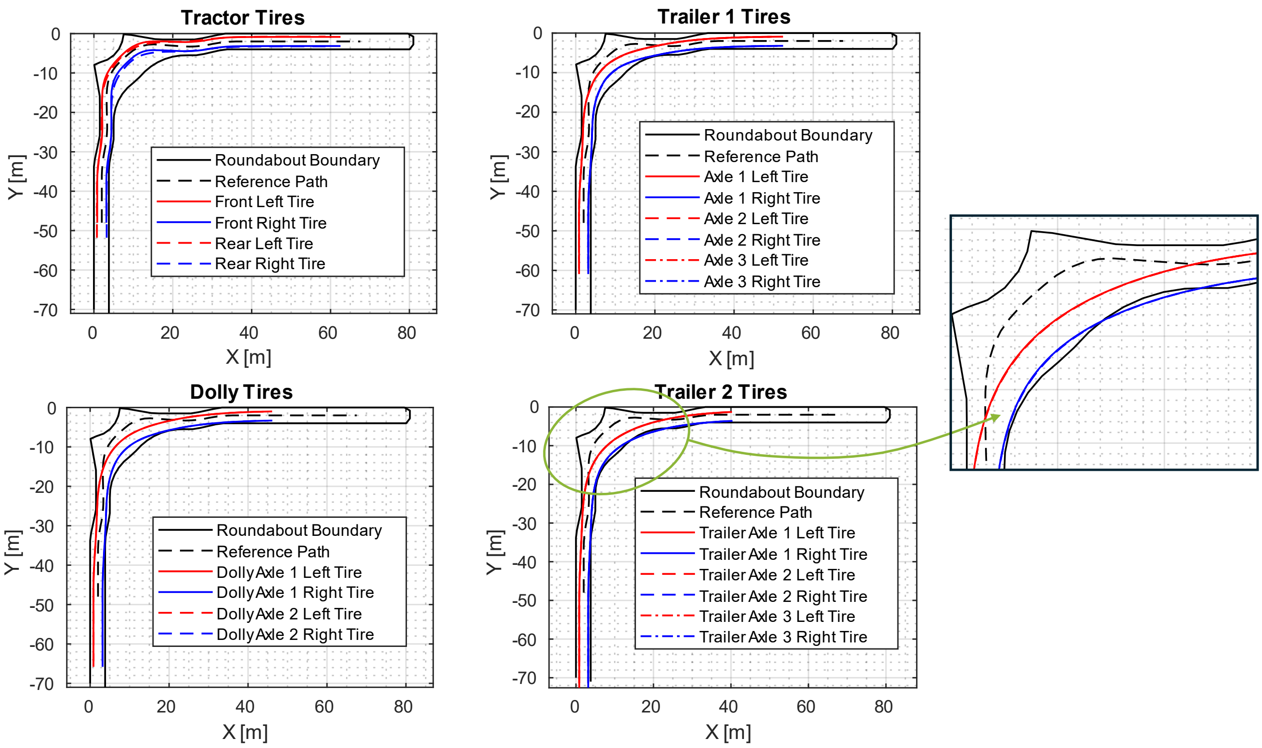


Figure 11 – Swept path of tyres of an EMS 2 vehicle at a roundabout

Based on the tyre positions, and the boundary positions, it is possible to compute the distance of any tyre trajectory to the boundary. If there are intersections between the trajectories and the boundary as shown in Figure 11 where the second trailer's tyres intersect the boundary, that particular roundabout size (16 meters outer radius) and exit (90º) is unsuitable for the simulated LHV type. The result of this analysis is a flag- Red, Amber or Green to indicate the feasibility of manoeuvring this roundabout. Since Figure 11 shows a clear intersection, the flag for this roundabout is Red. In cases of easy passage, the flag will be Green, and if the swept path is close to the boundaries (~30 cm) but still within the boundary, we indicate Amber. These flags are depicted to the user on a map to be able to make choices for rerouting if necessary.

4. Simulator packaging

MATLAB and Simulink increasingly offer tools to export models and scripts to generic formats to be able to be used in other environments. The Functional Mock-Up Interface (FMI or FMU) is one such standard of packaging and sharing simulation models. The FMI standard enables the development and exchange of simulation models across different platforms (FMI Standard, 2024). The conversion of a Simulink model to an FMU model makes sure that the Simscape multibody model library used for the LHV simulations at the HAN University of Applied Sciences, can function elsewhere without the need for a MathWorks license or any Simulink installation.

However, packaging a Simscape multibody model into an FMU does not allow for model parameters to be updateable during simulation (MathWorks, 2024c). This has the following implications:

- The inertia parameters like chassis mass, dimensions, etc., are hard-coded during the conversion and hence the vehicle specification cannot be modified.
- The payload in the trailers can vary drastically in reality, but the load mass cannot be changed in the simulation.
- Initial conditions of the simulation are also hard-coded, so the simulation will always start with a certain scenario.

Considering the limitations, a workflow to create simulation models needs to be created that can capture the required tuneability of the models. By generating various FMUs with varying payload and chassis parameters that reflect realistic ranges, the right FMU can be selected during simulation. For instance, regulations dictate the possible dimensions of trailers and payload. Hence, the number of FMUs that reflect these ranges can be limited. Furthermore, the initial conditions of all FMUs can be the static condition of standstill. Consider that a roundabout manoeuvre needs to be simulated with an initial speed of 10 km/hr. The simulation can first take into account the acceleration from standstill to 10 km/hr and then perform the actual manoeuvre.

In order to be able to automate this process of choosing the right FMU and managing the Simulation, a script of MATLAB is made. This script can also interact with other APIs (Application Programming Interface) to receive simulation requests and output results. This MATLAB script can be compiled into a standalone application which removes the need for any MathWorks licenses for operation. To be able to use the application without needing to install the application on a host computer, we package the application into a container environment called Docker. A Docker container is a lightweight, standalone, executable package that includes everything needed to run a piece of software - code, runtime, system tools, libraries, and settings. Containers are isolated from each other and from the host system (any operating system and hardware), allowing developers to run multiple applications consistently across different environments without interference .

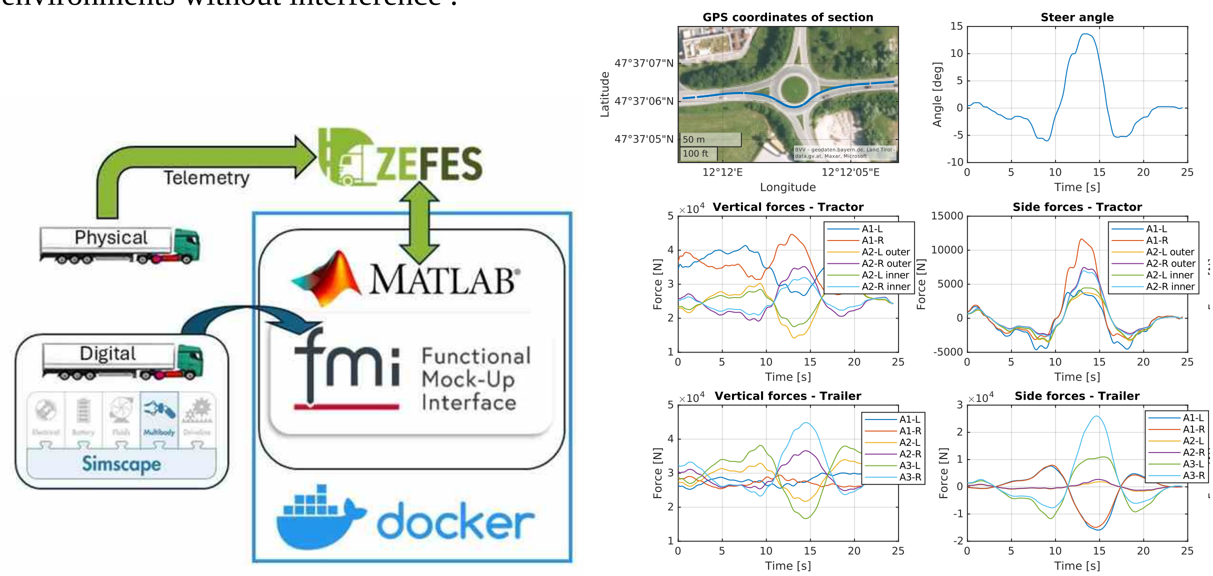


Figure 12 – Packaging an LHV simulator (left), example of logged data and simulated tyre forces (right)

As shown in Figure 12 (left), the multibody models are packaged into FMUs, which are in turn run in a Docker environment. The MATLAB script intercepts the inputs (planned routes, logged FMS telemetry, etc.) from the ZEFES Digital Platform and outputs the results of the simulations back to the platform. The Figure 12 (right) shows an example of logged GPS and steer input data from FMS telemetry along with the simulated tyre forces for those inputs. This workflow ensures that the simulator application can be run without the need for licenses or paid software from the user side.

5. Conclusions

This paper presented automated workflows to detect road features, focusing on roundabouts. Using OpenStreetMaps to determine the locations of roundabouts in a route was found to be not ideal. In remote regions of Scandinavia, multiple roundabouts visible in satellite imagery are missing in the API's response. Hence, in ZEFES, the RVRD tool uses PTV's xServer API, which is a commercial product (PTV being a partner in ZEFES), which has a more updated dataset. The inference of a roundabout's outer boundary dimensions from the size of the centre island was validated by measuring the boundaries of actual roundabouts in The Netherlands. For the algorithm to be valid pan-Europe, the roundabout design guides of each country need to be taken into account in the workflow. The workflow described in section 2.3 to detect the centre island radius resulted in a 77.4% accuracy from a total of 93 roundabouts (verified by manual measurements in Google Earth) of the various ZEFES use case routes. The dimensions of 4 roundabouts were incorrect, due to poor image quality, 13 were special roundabouts (huge roundabouts exceeding 50 m radii and turbo roundabouts) and 4 had an incorrectly placed mask.

Using an MPC to generate a realistic reference path to drive a roundabout is a promising solution, where a separate MPC per LHV type offers realistic results of manoeuvring albeit with the disadvantage of computation time. The packing of the simulators in a Docker environment is a working solution for packaging Simscape models that works around the limitations of MATLAB and Simulink licensing. FMU models are also efficient to run compared to their original Simscape counterparts. With the logic of choosing the right FMU, processing inputs and outputs in the MATLAB scripts, the Simulator application is highly automated.

6. Future work

The detection of the roundabout's centre island radius enables the inference of the other roundabout features. However, there is a need to identify special roundabouts, like turbo roundabouts and ones with shallow entry approach and exit departure angles. The presented workflow potentially overestimates the difficulty of manoeuvring in these special cases, since these special cases are not identified as such. To combat the significant computation time of the MPC, the setup of the cost function and better initial guess has to be worked on. This requires further research. The fusion of manually generating a path that is close to the final solution and optimizing it with the MPC for realistic trajectory generation could speed up the MPC since it will just have to tweak the initial guess rather than deviate heavily from it. The swept paths predicted by the simulators are to be validated with the use case demonstrator vehicles in the project. Also, other possible critical sections like right-hand turns (swept path analysis) and sharp inclines (gradability analysis) can be identified and added to the simulators.

7. Acknowledgement

Funded by the European Union

The ZEFES project has received funding from the European Union's Horizon Europe Research and Innovation programme under the grant agreement number 101084046. Views and opinions expressed are however those of the author only and do not necessarily reflect those of the European Union or the European Commission. Neither the European Union nor the granting authority can be held responsible for them.

8. References

- Dijkstra, A. (2021), Risico's bij de interactie tussen een Super EcoCombi en medeweggebruikers, SWOV – Instituut voor Wetenschappelijk Onderzoek Verkeersveiligheid.
- Matuszkova, R., Heczko, M., Cepil, J. and Radimsky, M. (2018), "Issues of using Longer Heavier Vehicles on Roads" in Proceedings of the 4th International Conference on Advanced Engineering and Technology, IOP Publishing, IOP Conference Series: Materials Science and Engineering, 317, 012040, 1-6.
- Aarts, L., Honer, M., Rakic, B., Stegeman, J., and Kind, M. (2011), Monitoring Traffic Safety for Longer and Heavier Vehicles, Rijkswaterstaat, Ministry of Infrastructure and Environment.
- The Mayor (2024), "Europe: The Champion of Roundabouts – Which Country Has the Most?" [online] Available: https://www.themayor.eu/en/a/view/europe-the-champion-of-roundabouts-which-country-has-the-most-11322.
- ZEFES (2024), "Deliverables", ZEFES Project, European Union Horizon Programme, [online] Available: https://zefes.eu/deliverables/.
- MathWorks (2024a), "Simscape Multibody", [online] Available: https://www.mathworks.com/products/simscape-multibody.html
- Ajaykumar, S. (2022), Digital twin approach for articulated vehicle performance analysis, Master Thesis, HAN University of Applied Sciences.
- Kural, K., Prati, A., Besselink, I.J.M., Pauwelussen, J.P., and Nijmeijer, H. (2013), "Validation of Longer and Heavier Vehicle Combination Simulation Models" in SAE International Journal of Commercial Vehicles, 340-352.
- MathWorks (2024b), "Magic Formula Tyre Force and Torque", MathWorks Documentation, [online] Available: https://www.mathworks.com/help/sm/ref/magicformulatyreforceandtorque.html.
- Gkountzini, A., Lemonakis, P., Kaliabetsos, G., and Eliou, N. (2020), "Investigation of Vehicle Swept Path in Roundabouts," in Proceedings of the Advances in Mobility-as-a-Service Systems, CETRA, 1154, 1-12.
- Godavarthy, R.P., Russell, E., and Landman, D. (2016), "Using vehicle simulations to understand strategies for accommodating oversize, overweight vehicles at roundabouts," Transportation Research Part A: Policy and Practice, 87, 41-50.
- Kennedy, J., (2008), International comparison of roundabout design guidelines, TRL Limited, [online] Available: https://www.trl.co.uk/publications/ppr206
- taginfo (2024), TAGS, [online] Available: https://taginfo.openstreetmap.org/tags
- Atherton, T.J., Kerbyson, D.J. (1999), "Size invariant circle detection," Image and Vision Computing 17, 795-803.
- FMI Standard (2024), "Functional Mock-up Interface," [online] Available: https://fmi-standard.org/
- MathWorks (2024c), "Limitations," [online] Available: https://www.mathworks.com/help/simscape/ug/limitations.html
- Docker (2024), "What is Docker," [online] Available: https://www.docker.com/