LONG TRAIN DYNAMIC SIMULATION BY MEANS OF A NEW IN-HOUSE CODE

NICOLA BOSSO, MATTEO MAGELLI & NICOLÒ ZAMPIERI
Department of Mechanical and Aerospace Engineering, Politecnico di Torino, Italy

ABSTRACT

The need for faster heavy-haul trains with higher axle-loads is responsible for the rise of large in-train forces on wagon connection systems. These forces have a significant impact on longitudinal train dynamics (LTD) from the point of view of performances, running stability and safety during braking and traction operations. Therefore, LTD simulations represent an essential tool to predict in-train forces and to design coupling and braking systems. Long trains are typically modelled as a system of several point masses linked to each other by means of non-linear elements, summarizing the characteristics of the coupling systems. In fact, wagon connection systems present a non-linear force-deflection characteristic, with a hysteresis loop due to the different behaviour in loading and unloading states. The calculation of in train forces is nowadays performed following three main strategies, namely look-up table approaches, mathematical equations based on experimental results and "white-box" models related to physical properties of the connection systems. An International benchmark of LTD simulators was recently proposed to assess the output results obtained by different simulators, following several modelling strategies, in the simulation of the same scenarios. The research group from Politecnico di Torino joined the competition using the multibody commercial software Simpack (ver.9.4). However, inefficiencies appeared because multibody software are usually intended for the simulation of a few vehicles with many degrees of freedom (d.o.fs), rather than many wagons with few d.o.fs each. Therefore, the team developed a new LTD simulator (LTDPoliTO) in MATLAB, totally based on a vector-arithmetic logic. The new simulator can perform both dynamic investigations in time domain and modal analysis in frequency domain. The validation of LTDPoliTO was carried out by performing the four benchmark simulations and a good agreement with the other participants was obtained, in terms of both numerical results and computational speeds.

Keywords: long train simulation, LTD benchmark, draft gear, dynamics modeling, train dynamics.

1 INTRODUCTION

The performances and the safety of railway systems are strongly affected by longitudinal train dynamics, which refer to both the motion of the whole train as well as to the relative motions between adjacent vehicles ("slack actions"). The latter are allowed by clearances in coupling systems [1], usually comprising a coupler and a damping element. The typical solution in Europe is the buffer-hook system, while coupler and draft gears are used in other countries, such as the USA and Australia [2].

Modern freight trains are facing a strong demand for higher axle-loads and speeds, which can eventually lead to the rise of large in-train forces, thus causing a serious threat to the safety and running stability of freight trains during traction and braking operations, with several possible drawbacks, namely wagon climbing, derailments, structural damages and even breaking of coupling elements [3], [4]. Therefore, studying the LTD of long freight trains is a major concern for the design of coupling and braking systems, but also for the optimization of driving cycles and train composition as well as for the forecast of vehicle position and speed during a specific route [5].

Due to the high computational power offered by modern computers, nowadays LTD investigations are performed by means of numerical simulations [6]. Usually, long trains are modelled as a system of several lumped point masses, with a single degree of freedom (d.o.f) in the running direction, linked to each other through nonlinear elements, accounting for the

doi:10.2495/CR200231

mechanical impedance of the coupler and draft gears. The calculation of in-train forces is often the bottleneck of these numerical models, since coupling systems are strongly nonlinear components, with a hysteretic behaviour, as shown in experimental drop-hammer tests [7]. Wheel-rail adhesion phenomena [8] as well as traction/braking control systems [9] are typically not considered in LTD models since even efficient contact codes [10], [11] usually require a high computational effort.

As surveyed by Wu et al. [12], three main strategies are used to model the nonlinear impedance of railway coupling systems, namely (a) look-up table (LUT) approaches, (b) mathematical equations fitting experimental data and (c) white-box models. The LUT approach is based on fixed tables storing the experimental data registered in drop-hammer tests, and it usually requires a logic switch criterion to smooth the transition between loading and unloading conditions. On the other hand, the second strategy relies on analytical equations fitted on experimental data. Finally, white-box models depend on the physical properties of coupling elements.

The in-train forces calculated by numerical models can vary significantly according to the specific strategy used to model the coupling systems. Therefore, an international benchmark ("the benchmark" in the rest of the paper) was established in recent years to compare the results from different LTD simulators [13], [14]. The railway research group from Politecnico di Torino developed an LTD model using the commercial multibody software Simpack (ver. 9.4) [15], [16] and joined the competition, but struggling occurred in the simulation of a system comprising many bodies, each one with few d.o.fs. Therefore, the research group developed a new in-house LTD code in MATLAB (LTDPoliTO) to obtain higher computational efficiencies and a better agreement with the other participants in the output results.

The paper concerns the mathematical description of the new model, the implementation in MATLAB environment and the validation of the proposed in-house tool performed on the same case study as suggested in the benchmark. The novel aspect of the proposed code is that it is totally developed in MATLAB, making the most of built-in high-level functions and vectorization strategies for large array management. The greatest point of merit of this new code is the capability of producing stable output results with high computational speeds.

2 DESCRIPTION OF THE MODEL

The mathematical definition of the new code followed the suggestions from the international benchmark. Each vehicle only has a single d.o.f in the running direction and the track is modelled as straight, while curves and slopes are accounted for by introducing extra retarding forces.

Fig. 1 shows all forces acting on the kth vehicle of the train configuration. Please note that the numbering rule in this section as well as in the code, starts from the tail-end wagon in ascending order. The forces on the vehicle include the inertial term (the product of the total mass M_k times the acceleration $\ddot{x}_k$), the forces due to front ($F_{c,k}$) and rear ($F_{c,k-1}$) coupling elements, the retarding forces ($F_{R,k}$) and the forces due to traction or dynamic braking efforts ($F_{T/DB,k}$).

The equilibrium of forces in the longitudinal direction x for the kth vehicle is expressed by eqn (1):

$$F_{T/DB,k} + F_{c,k} - F_{R,k} - F_{c,k-1} = M_k \ddot{x}_k. \qquad (1)$$

Of course, for the first vehicle of the composition, i.e., the tail-end wagon, the force on the back coupler is zero. Similarly, the force on the front coupler of the last vehicle, i.e., the leading loco, is equal to zero, too.

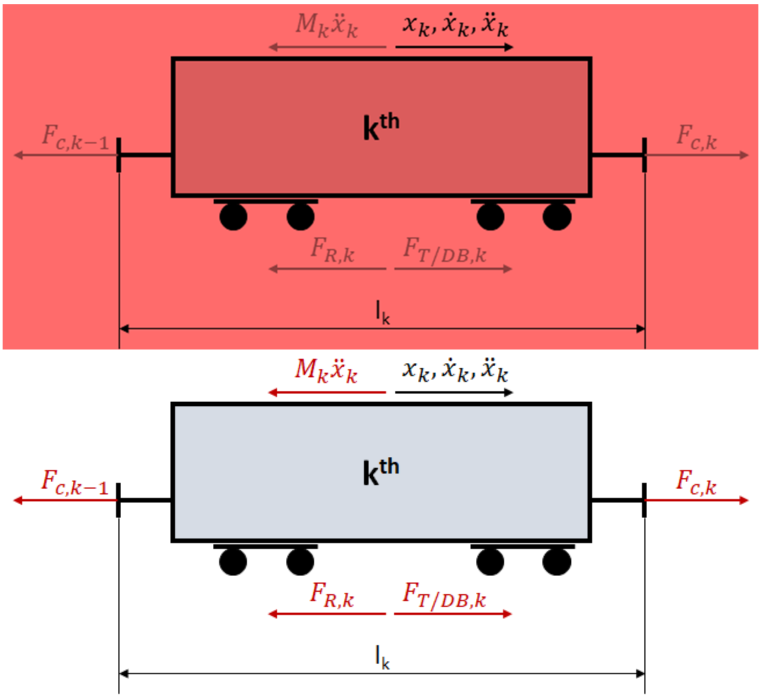

Figure 1: Forces acting on the kth vehicle.

In a vehicle with N_{AX} wheelsets, the total mass of the vehicle (M_k) is calculated as the sum of all the axle-loads (m_{ax}), expressed in kg, according to eqn (2)

$$M_k = m_{ax}N_{AX}. \tag{2}$$

The total retardation force $F_{R,k}$ is the sum of the resistant forces due to rolling resistances $F_{RR,k}$ and the forces due to track curvature $F_{RC,k}$ and grade $F_{RG,k}$. The forces due to curvature and rolling resistances always have positive sign, i.e., they always have an opposite direction with respect to the direction of the vehicle speed $\dot{x}$. On the other hand, forces due to track grade can be in the same direction as the vehicle speed, e.g., in downhill track sections. The resistant forces are modelled using the expressions suggested in the benchmark and shown in eqns (3)–(5), in which the subscript k is omitted for the sake of clarity. Please note that in eqns (3)–(5), the unit for mass is tonne, while the unit for speed is km/h. In eqn (3), Q is an aerodynamic frontal resistant factor, which is equal to 3.2 for the leading loco and equal to 1 for all other vehicles. In eqn (4), R_C is the curve radius expressed in meters and finally in eqn (5) i_G is the track grade in %. Other expressions are witnessed in the literature and are surveyed in a work by Spiryagin et al. [17]

$$F_{RR} = QM\left(2.943 + \frac{89.2}{m_{ax}} + 0.0306\dot{x} + \frac{0.122\dot{x}^2}{M}\right), \tag{3}$$

$$F_{RC} = M\frac{6116}{R_C}, \tag{4}$$

$$F_{RG} = Mgi_G. \tag{5}$$

The total retardation force could also account for a fourth contribution, i.e., the pneumatic braking force. However, at this early stage of development, the LTDPoliTO code does not include a pneumatic module, which would drastically increase the computational effort. Nevertheless, the code can still be validated with a set of benchmarking output results, since the air brake system was neglected in the international benchmark, too. In future work, the pneumatic module could be included in the code in order to calculate air brake forces, but this would probably require co-simulation and parallel computing techniques to reduce the wall time of the simulations.

The only braking effort considered in the model is therefore the dynamic braking force exerted by the locomotives. In fact, the term $F_{T/DB}$ in Fig. 1 and eqn (1) can be either positive, when traction operations are simulated, or negative, when dynamic braking manoeuvres take place. Obviously, this term is set to zero for all non-powered vehicles. On powered vehicles, the tractive/braking effort is calculated from the locomotive mechanical characteristic, depending on the current value of speed and notch level.

Finally, coupling forces are calculated as a nonlinear function of the deflection δx and relative speed δv of each coupler, obtained from eqns (6) and (7)

$$\delta x_k = x_{k+1} - x_k - \frac{l_{k+1} - l_k}{2}, \tag{6}$$

$$\delta v_k = \dot{x}_{k+1} - \dot{x}_k. \tag{7}$$

In eqns (6) and (7), l_k is the total length of the kth vehicle, which includes the car body and both semi-lengths of front and rear coupling systems, see Fig. 1, while $\dot{x}_k$ refers to the speed of the kth vehicle. The transition between loading and unloading conditions is smoothed according to a logic criterion depending on the current value of the relative speed, as suggested by Zhang et al. [18]. The force on each coupler is calculated using eqns (8)–(10), which ensure that a smoothing transition occurs when the absolute value of the relative speed is below a threshold value v_ε, with a coupling force lying between the loading and unloading curves. On the other hand, if the absolute value of the relative speed is above v_ε, the coupling force is equal to either the loading (F_{load}) or the unloading (F_{unload}) force.

$$F_{c,k}(\delta x_k, \delta v_k) = \begin{cases} F_{cM,k}(\delta x_k) + F_{cA,k}(\delta x_k)\,\mathrm{sign}(\delta x_k \cdot \delta v_k), & |\delta v_k| \geq v_\varepsilon, \\ F_{cM,k}(\delta x_k) + \frac{|F_{cA,k}(\delta x_k)|}{v_\varepsilon}\delta v_k, & |\delta v_k| < v_\varepsilon, \end{cases} \tag{8}$$

$$F_{cM,k}(\delta x_k) = \frac{F_{load,k}(\delta x_k) + F_{unload,k}(\delta x_k)}{2}, \tag{9}$$

$$F_{cA,k}(\delta x_k) = \frac{F_{load,k}(\delta x_k) - F_{unload,k}(\delta x_k)}{2}. \tag{10}$$

The next section briefly shows the implementation of the model in MATLAB and shortly describes the LTD scenarios proposed in the benchmark, which were simulated with the new code in order to validate and to assess the output results obtained with LTDPoliTO.

3 IMPLEMENTATION OF THE MODEL IN MATLAB

Eqns (1)–(10) must be written for each of the N_v vehicles in the train composition, thus obtaining a system on nonlinear ordinary differential equations (o.d.e), which needs a numerical integration scheme to be solved. It is not difficult to demonstrate that at each time-step, the LTD o.d.e system can be expressed in the form of eqn (11), where t is time, $\underline{\mathbf{y}}$ is the array storing the dependent variables of the problem, $\underline{\mathbf{y}}$' is the first derivative of $\underline{\mathbf{y}}$, $\underline{\mathbf{f}}$ is a function of t and $\underline{\mathbf{y}}$ for the calculation of the derivative $\underline{\mathbf{y}}$' and finally $\underline{\underline{\mathbf{M}}}$ is the mass matrix of the system

$$\underline{\underline{\boldsymbol{M}}}\underline{\boldsymbol{y}}' = \underline{\boldsymbol{f}}(t, \underline{\boldsymbol{y}}). \tag{11}$$

The column vector $\underline{\mathbf{y}}$ stores 2N$_v$ elements, corresponding to the positions and speeds of the N$_v$ vehicles. Therefore, $\underline{\mathbf{y}}$' collects the N$_v$ speeds and N$_v$ accelerations of the vehicles. The mass matrix is a constant (i.e., not time dependent) sparse diagonal square matrix 2N$_v$x2N$_v$ in size, with the first N$_v$ elements on the diagonal all equal to 1, and the following N$_v$ elements equal to the mass of each vehicle. Finally, $\underline{\mathbf{f}}$ is the same size as $\underline{\mathbf{y}}$ and $\underline{\mathbf{y}}$', and its elements 1-N$_v$ correspond to the speed of each vehicle, while each of the elements N$_v$+1-2N$_v$ is equal to the left-hand side of eqn (1), written for the corresponding vehicle.

Written in the form of eqn (11), the LTD o.d.e system can be easily implemented and solved in MATLAB, using a numerical integration method, once the model inputs are provided. LTDPoliTO inputs include the definition of the train arrangement, in terms of the number and types of vehicle, as well as the following input characteristics:

- Track curvature and elevation along the running direction.
- Driving cycle, i.e., the notch level during the train route, as a function of either time or position along the running direction.
- Mechanical characteristic of locomotives, i.e., the typical curve relating traction/dynamic braking force to speed and notch level.
- Definition of the mechanical impedance of coupling elements.

The train configuration is represented in the code using an array with N$_v$ integer, each one corresponding to a specific type of vehicle. Instead, the input characteristics are imported into LTDPoliTO by means of several text files, which describe the main features of the input data. Forces on coupling elements are calculated using a LUT approach, with two fixed tables storing the experimental force-deflection characteristic for both loading and unloading conditions. Before the simulation starts, the user must define the following 4 re-discretization steps in order to re-interpolate the input characteristics with a fixed step:

1. A step for the discretization of the coupling system deflections, in order to re-interpolate the force-deflection LUT.
2. A re-discretization step corresponding to the shortest track section considered, in order to re-interpolate the track curvature and elevation characteristics.
3. A fixed time-step to discretize the driving cycle input text file, which currently defines the notch level as a function of time.
4. A step for the re-discretization of the vehicle speed, which is needed to re-interpolate the locomotive mechanical curves.

The re-discretization of the input files with fixed steps is needed to allow easy indexing operations throughout the simulation. All indexing and array management operations are handled using MATLAB vectorization strategy and optimized built-in functions, thus increasing the total computational efficiency. At each time-step, the code can perform the calculation of all in-train, retardation and tractive/braking forces with no need for loop flows or Boolean statements, by simply exploiting MATLAB vector logic.

At the end of the simulation, the MATLAB solver outputs the speed and position of each vehicle throughout the simulation. Therefore, deflections, relative speeds and in-train forces on all coupling systems are calculated with a specific post-processing routine. Finally, the whole output dataset can be saved in binary files for further investigations or simply to back up the simulation results.

4 CASE STUDY

The validation of the new code was performed on the four simulation scenarios proposed in the international benchmark questions paper [13], which correspond to four different train compositions, built using 2 different types of locomotives and 2 different types of wagons and schematically illustrated in Fig. 2. As shown in Fig. 2, the number of d.o.fs increases from train 1 to train 3, while train 4 has the same number of d.o.fs as train 3 but it features a different arrangement. Table 1 briefly summarizes the main parameters of each vehicle type.

Two types of wagon connection system are considered in the benchmark, i.e., couplers, with a total 10 mm slack, and drawbars, with no slack. Starting from the front wagon, in all four train configurations, each pair of wagons is connected by means of a drawbar, and wagon pairs are linked to each other with couplers. All locomotives are connected to the other vehicles using couplers.

Among the benchmarking input was a draft gear force-deflection cross plot, representing an experimental drop-hammer test. Fig. 3 shows the loading and unloading characteristics for the series of two draft gears and a coupler/drawbar obtained from the benchmark input data and used in the simulations performed during the numerical activity described in the paper. According to the benchmark rule, tensile deflections and forces have a negative sign, while compression states have positive values. The detailed view in Fig. 3 highlights the total 10 mm coupler slack, with a 2 mm slack in compression and 8 mm slack in tensile conditions.

The benchmarking input also included the mechanical characteristics of two different types of locomotive. Both locomotives have 8 notch levels for traction and dynamic braking, but locomotive #2 guarantees higher power efforts.

Track curvature and elevations as well as driving cycles are once again extracted from the benchmark input dataset. Track features, shown in Fig. 4, are fixed in all four simulations, while the driving cycle is different for each train, and can be easily determined from the benchmarking question paper.

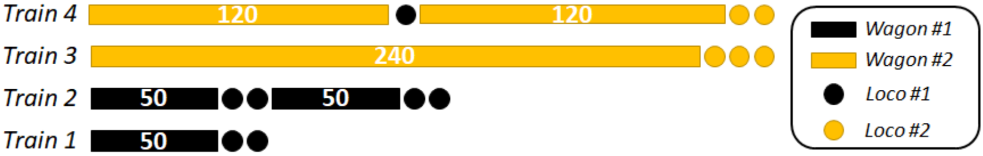

Figure 2: Schematic representation of the four benchmark train configurations. The wagon block rectangles are proportional in length to the number of wagons, which is written in the middle.

Table 1: Main parameters of the four vehicle types presented in the benchmark.

Vehicle type	Axle-load (tonne)	Axle numbers	Total mass (tonne)	Vehicle length (m)
Loco #1	22.33	6	134	22.95
Loco #2	32.50	6	195	24.35
Wagon #1	32.00	4	128	15.00
Wagon #2	40.00	4	160	11.00

As already mentioned in the previous section, all the benchmark input characteristics were summarized into text files and then re-discretized during the simulations presented in the paper using the following values. The coupler deflection was discretized with a step equal to 1e-5 m, the shortest track section was considered equal to 1 m, the fixed time-step for the notch level characteristic was set to 0.001 s and finally the locomotive speed discretization step was assumed equal to 0.001 m/s. The choice of the re-discretization steps must guarantee a good reproduction of the original data without exceeding the computer RAM memory limits.

A deep investigation was carried out to select the proper numerical solver for the case study shown in the previous lines. The best compromise in terms of accuracy of output results and computational efficiency was offered by the MATLAB variable step-size predictor-corrector stiff solver ode15s.

The constant sparsity pattern for the Jacobian matrix was passed to the solver as an input option, in order to reduce the total number of elements to be computed from the ode15s internal routine in failed time-steps.

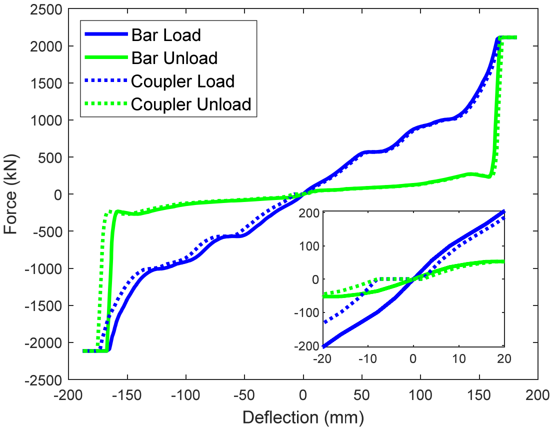

Figure 3: Force-deflection characteristics of drawbars and couplers.

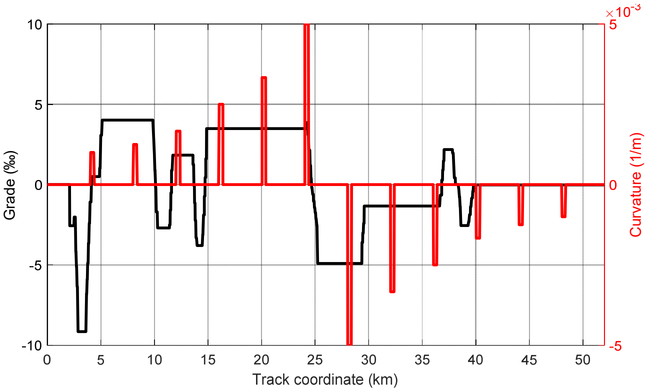

Figure 4: Track curvature and grade.

Finally, a fine tuning was also performed on the relative speed threshold value v_ε, see eqns (8)–(10). In fact, low values of this parameter proved to significantly slow down the computation, while higher values can considerably change the coupling system mechanical impedance. A good compromise was observed for a value $v_\varepsilon = 0.001$ m/s in all four configurations.

5 NUMERICAL RESULTS

This section deals with the validation and assessment of the proposed LTDPoliTO code. The validation of the output results was performed considering the assessment quantities described in the benchmark results paper [14] to compare the results produced by the various simulators, namely:

- The maximum and mean speed value among all vehicles and time steps.
- The largest in-train force and the corresponding position calculated as the maximum value among all time steps and all coupling systems, for both tensile and compressive states. Coupling system positions are numbered in this section in ascending order starting from the one on the rear of the leading loco, in accordance with the benchmark results paper.
- The mean value of the maximum force registered throughout the simulation in each coupling position, for both compressive and tensile states.
- The maximum deflection on a selected coupler position for both tensile and compressive states.

The outputs obtained in all four configurations showed a good agreement with the results from the other benchmark participants, as it can be inferred from the comparison of Table 2 with the analogous table from the international benchmark results paper.

Fig. 5 presents a direct comparison between the new LTDPoliTO code and the previous Simpack model in terms of the forces and deflections calculated on the selected coupling system for train 4, i.e., the one in position 146. As shown in Fig. 5(a), numerical divergences occurring in the previous Simpack model are not present in the new code, which produces numerically stable results. The cross-deflection characteristic in Fig. 5(a) shows the nonlinear transition path predicted by eqns (8)–(10). Discrepancies between LTDPoliTO and Simpack model outputs can be related to slight differences in the re-discretization of the loading and unloading curves as well as to the absence of numerical divergences in the new code. Similar observations are also true for the other simulation scenarios.

Table 2: LTDPoliTO results for the four benchmark train configurations.

	Speed (km/h)		Largest in-train force (kN) @ coupler pos.		Mean in-train force (kN)		Max. deflection (mm) @ selected coupler pos.		
Train	Max	Mean	Tens.	Comp.	Tens	Comp	Pos.	Tens.	Comp.
1	87.30	65.97	-565@2	338@2	-336	195	10	-64.56	45.07
2	89.40	64.55	-769@54	407@54	-444	263	61	-89.22	48.05
3	81.83	44.79	-1636@3	1008@144	-1166	620	147	-169.56	129.00
4	81.80	44.99	-1309@123	867@122	-932	501	146	-161.71	96.35

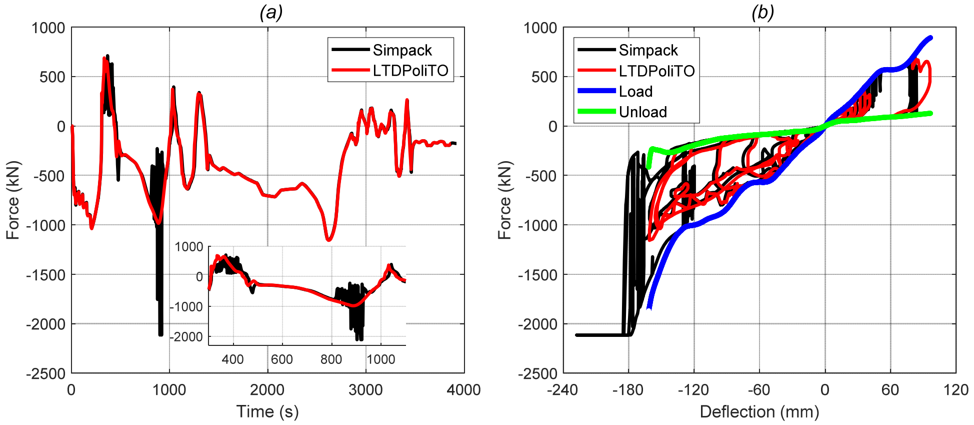

Figure 5: Comparison of output results from LTDPoliTO and previous Simpack model for the selected coupler position on train 4. (a) The in-train force time history; and (b) The force-deflection characteristic.

Shifting focus to the assessment of the computational efficiency of the new code, Table 3 presents the main quantities introduced in the paper for all four simulations, i.e., the wall time T_w, the train operation time T_{op} and their ratio Speed 1. The wall time is the time needed for the code to solve the LTD o.d.e system while the train operation time is the total simulated train route time. Low values of Speed 1 indicate that the time required for the numerical calculation is far lower than the simulated train operation time, thus highlighting a good computational efficiency. The simulations for the code validation were all performed on the same notebook with the following main features: Windows 10 Pro OS, Intel i7 CPU and 16GB RAM memory. Comparing the computational efficiencies of the simulators joining the benchmark is not an easy task since the participants did not run their models on the same computer. In fact, hardware and OS characteristics can drastically modify the computational speeds. However, the juxtaposition of Table 3 with the analogous from the benchmark results paper shows that LTDPoliTO is the fastest code in the four simulation scenarios, apart from TABLDSS. Therefore, LTDPoliTO can surely be considered as a solid and efficient tool for LTD simulations.

Table 3: LTDPoliTO computational times for the four benchmark simulation scenarios.

Train configuration	Train operation time (s)	Wall time (s)	Speed 1 (ms/s)
Train 1	2684	35	13
Train 2	2711	88	33
Train 3	3951	269	68
Train 4	3863	284	74
Average value	-	-	47

6 CONCLUSIONS

The paper deals with the implementation of a new LTD simulator, developed by the railway research group from Politecnico di Torino, in MATLAB, with great focus to its validation performed using the input dataset proposed in the international benchmarking of LTD simulators. The main goal of the work described in the paper was to develop a reliable tool for LTD simulations, able to produce numerically stable outputs with high computing performances, due to the inefficiencies of a previous model realized by the research group using the commercial multibody package Simpack.

The new code is developed in MATLAB and takes advantage of built-in high-level functions and vectorization strategies to handle large array indexing and mathematical operations. This approach guarantees extremely high computing performances and represents the main novel aspect of the proposed code. The simulations performed on the four benchmark scenarios led to a good agreement between the LTDPoliTO outputs and the results from the other simulators joining the benchmark. Moreover, the wall times obtained in the four scenarios make the new code as the second fastest simulator among the benchmark participants, since only TABLDSS was faster in the four configurations. Clearly, a fair comparison of the simulator performances would need all participants to run their codes on the same computer. Anyway, the initial goal of the activity described in the paper was achieved with success.

A lot of work is still needed to further increase LTDPoliTO computational efficiency as well as to widen the application extent of the code. In fact, the proposed simulator currently does not consider the retarding forces due to the air brake system. Therefore, a pneumatic module should be included for the calculation of air brake forces.

Moreover, at the current state of work, no explicit multi-thread computing technique is used, apart from MATLAB internal optimization processes. Future developments could investigate the possibility of parallelizing the code. Parallel computing and co-simulation strategies could be essential to maintain good computational efficiencies especially in case a pneumatic module is included in the code.

REFERENCES

[1] Colin, C., Longitudinal train dynamics. *Handbook of Railway Vehicle Dynamics*, Taylor & Francis: Boca Raton, 2006.

[2] Massa, A., Stronati, L., Aboubakr, A.K., Shabana, A.A. & Bosso, N., Numerical study of the noninertial systems: Application to train coupler systems. *Nonlinear Dynamics*, **68**(1), pp. 215–233, 2012. https://doi.org/10.1007/s11071-011-0220-2.

[3] Wei, L., Zeng, J. & Wang, Q., Investigation of in-train stability and safety assessment for railway vehicles during braking. *Journal of Mechanical Science and Technology*, **30**(4), pp. 1507–1525, 2016. https://doi.org/10.1007/s12206-016-0304-5.

[4] Zhu, T., Yang, B., Yang, C., Xiao, S., Yang, G. & Yang, B., The mechanism for the coupler and draft gear and its influence on safety during a train collision. *Vehicle System Dynamics*, **56**(9), pp. 1375–1393, 2018. https://doi.org/10.1080/00423114.2017.1413198.

[5] Wu, Q., Spiryagin, M. & Cole, C., Longitudinal train dynamics: an overview. *Vehicle System Dynamics*, **54**(12), pp. 1688–1714, 2016. https://doi.org/10.1080/00423114.2016.1228988.

[6] Cole, C., Spiryagin, M., Wu, Q. & Sun, Y.Q., Modelling, simulation and applications of longitudinal train dynamics. *Vehicle System Dynamics*, **55**(10), pp. 1498–1571, 2017. https://doi.org/10.1080/00423114.2017.1330484.

[7] Cole, C., Improvements to wagon connection modelling for longitudinal train simulation. *Conference on Railway Engineering Proceedings: Engineering Innovation For a Competitive Edge*, 1998.

[8] Bosso, N., Gugliotta, A. & Zampieri, N., Study of adhesion and evaluation of the friction forces using a scaled roller-rig C3. *5th World Tribology Congress, WTC 2013*, **3**, pp. 2640–2643, 2014.

[9] Bosso, N. & Zampieri, N., Real-time implementation of a traction control algorithm on a scaled roller rig. *Vehicle System Dynamics*, **51**(4), pp. 517–541, 2013. https://doi.org/10.1080/00423114.2012.750001.

[10] Bosso, N., Gugliotta, A. & Zampieri, N., RTCONTACT: An efficient wheel-rail contact algorithm for real-time dynamic simulations. *2012 Joint Rail Conference*, pp. 195–204, 2012. https://doi.org/10.1115/jrc2012-74044.

[11] Bosso, N. & Zampieri, N., A novel analytical method to calculate wheel-rail tangential forces and validation on a scaled roller-rig. *Advances in Tribology*, 2018. https://doi.org/10.1155/2018/7298236.

[12] Wu, Q., Cole, C., Luo, S. & Spiryagin, M., A review of dynamics modelling of friction draft gear. *Vehicle System Dynamics*, **52**(6), pp. 733–758, 2014. https://doi.org/10.1080/00423114.2014.894199.

[13] Spiryagin, M., Wu, Q. & Cole, C., International benchmarking of longitudinal train dynamics simulators: benchmarking questions. *Vehicle System Dynamics*, **55**(4), pp. 450–463, 2017. https://doi.org/10.1080/00423114.2016.1270457.

[14] Wu, Q. et al., International benchmarking of longitudinal train dynamics simulators: Results. *Vehicle System Dynamics*, **56**(3), pp. 343–365, 2018. https://doi.org/10.1080/00423114.2017.1377840.

[15] Bosso, N. & Zampieri, N., Long train simulation using a multibody code. *Vehicle System Dynamics*, **55**(4), pp. 552–570, 2017. https://doi.org/10.1080/00423114.2016.1267373.

[16] Bosso, N., Gugliotta, A. & Zampieri, N., A mixed numerical approach to evaluate the dynamic behavior of long trains. *Procedia Structural Integrity*, **12**, pp. 330–343, 2018. https://doi.org/10.1016/j.prostr.2018.11.083.

[17] Spiryagin, M., Cole, C., Quan Sun, Y., McClanachan, M., Spiryagin, V. & McSweeney, T., Longitudinal train dynamics. *Design and Simulation of Rail Vehicles*, CRC Press: Boca Raton, 2014.

[18] Zhang, Z., Li, G., Chu, G., Zu, H. & Kennedy, D., Compressed stability analysis of the coupler and buffer system of heavy-haul locomotives. *Vehicle System Dynamics*, **53**(6), pp. 833–855, 2015. https://doi.org/10.1080/00423114.2015.1023318.