

A 3D Visualization Tool for Energy Management in the Jerusalem Green Line LRT

Maoz Eitan
CAF-SHAPIR Via Advance Engineering
Jerusalem, Israel

Sebastián de la Torre
Electrical Engineering Department
Universidad de Málaga
Málaga, Spain
storre@uma.es

Abstract— This paper presents a 3D interactive visualization platform, developed in Unity 6, for the energy management simulation of the Jerusalem Green Line Light Rail Transit (LRT) system. The tool renders complete simulation results across 2496 operational scenarios (156 operational modes combined with 16 binary scenario configurations) derived from a GAMS/MATLAB optimization pipeline applied to the real 16.5 km, 33-station, 750 V DC network. Motivated by the large volume of multi-scenario results generated in previous works, the platform was designed to make these results intuitively accessible through an immersive 3D geo-referenced environment. It provides real-time visualization of inter-train energy exchanges, timetables, and financial results, supporting operational planning and decision-making for the CAF-Shapir consortium. The optimal configuration achieves a total daily energy cost of €34.47 with 186.62 kWh battery capacity and a round-trip travel time of 119.21 minutes.

Keywords— *Light Rail Transit (LRT), Visualization, Energy Management, Unity.*

I. INTRODUCTION

Urban Light Rail Transit (LRT) systems powered by 750 V DC overhead catenary supply represent a cornerstone of sustainable urban mobility, offering high passenger capacity with low local emissions. Yet their energy efficiency, particularly the recovery and redistribution of kinetic energy during train deceleration (regenerative braking) remains a central operational and economic challenge. When a train brakes, its traction motors act as generators, feeding electricity back into the DC catenary. This energy can be absorbed by nearby accelerating trains, stored in on-board or wayside Energy Storage Systems (ESS), or returned to the AC grid through reversible substations [1]. If none of these receptors is available, the energy is dissipated as heat, a loss that can represent 15 - 40% of total traction energy consumption [2].

The effectiveness of regenerative energy recovery depends critically on operational parameters such as train headway, maximum speed, and substation configuration; simulation models for regenerative braking energy in DC rail systems have been developed to characterize energy flows between trains and substations [3], and Matlab/Simulink models of reversible substations have demonstrated the feasibility of feeding surplus regenerated energy back to the AC grid [4]. Multi-train energy consumption simulators have also been proposed for transit planning, producing numerical energy profiles for given operating scenarios [5].

Several works have proposed simulation platforms with visual output for railway energy and operational systems. Arboleya et al. [5] presented a web-based software suite for DC railway network analysis incorporating visual analytic tools. Zhang et al. [6] developed a real-time coupling simulation platform unifying traction power supply, train dynamics, and dispatching with visualization capabilities. In the domain of digital twins, Ahmadi et al. [7] proposed

adapting digital twin technology for electric railway power systems, while Li and Liu [8] introduced a hybrid data-driven and physics-based digital twin approach for traction power systems with graphical state visualization.

Unity has emerged as a well-established platform for railway simulation across a broad spectrum of applications. Zhao and Wang [9] applied Unity3D to build a virtual training simulator for railway signal equipment, establishing the viability of game engines for spatially-referenced railway applications. Hu et al. [10] designed an integrated VR training simulation sand table for rail systems combining train dispatching, driving simulation, and UAV-assisted inspection into a single multi-modal Unity-based platform. Nikitin et al. [11] implemented a VR maintenance training simulator for railway wagons using HTC Vive and Unity, demonstrating ergonomic adequacy for reproducing complex inspection procedures in a 3D environment. Liu et al. [12] constructed a cloud-based virtual reality rail transit simulation system enabling accessible 3D model sharing and scenario replication through Unity. Beyond training applications, Unity has been applied as the visualization and co-simulation layer in railway vehicle dynamics frameworks: Nie et al. [13] proposed a general-purpose multibody simulation framework using Unity3D for longitudinal train dynamics, and Łukaszewski et al. [14] directly compared monorail vehicle dynamic behavior simulated in Unity against the specialized Universal Mechanism software, validating the fidelity of game-engine-based dynamics. D'Amico et al. [15] further leveraged Unity for AI perception dataset generation for autonomous train functions through the TrainSim framework. Additional applications of the Unity framework can be found in the literature; i.e. [16] and [17]. Together, all these works confirm that Unity is a relevant and useful platform for simulation and visualization for transportation applications.

The optimization framework developed and validated in our two previous works on the Jerusalem Green Line [18], [19] generates thousands of operational scenarios whose interpretation through conventional numerical outputs is cumbersome. This work addresses that gap by introducing a dedicated visualization environment that enables efficient exploration and interpretation of those results for engineering decision support. Based on these studies, we have generated 2496 pre-computed scenarios covering 156 operational modes and 16 binary configuration combinations, each yielding full timetables, energy exchange profiles, battery sizing requirements, and economic cost summaries for both directions. Presenting these results in tabular form proved insufficient for operational insight: it was difficult to grasp intuitively how energy flowed between trains in real time, or how headway choices affected the spatial distribution of exchange events. A spatially-referenced 3D animated platform was therefore identified as the most effective medium for communicating these results to both technical and non-technical stakeholders.

A well-established approach to maximize regenerative energy recovery is timetable optimization: by coordinating departure times and dwell intervals, an accelerating train can be ensured to be present in the same electrical section whenever a braking event occurs. Yang et al. [20] proposed a cooperative scheduling model for subway timetable optimization that maximizes the overlap time between braking and traction phases, demonstrating energy savings of up to 15% on the Beijing Yizhuang line. Su et al. [21] extended this concept by jointly optimizing train speed profiles and timetables in a multi-train subway system, achieving a 14.5% energy reduction. Nasri et al. [22] applied a genetic algorithm to find optimal reserve times and headways that minimize total energy consumption in an electrical railway system. These works collectively confirm that service headway and schedule structure are critical parameters governing regenerative energy exchange and directly motivating the multi-headway scenario exploration in the present work.

Among available visualization technologies, Unity (a commercial cross-platform game engine) offers distinctive advantages for engineering simulation. Its real-time 3D rendering pipeline supports geo-referenced environments with complex topographies, animated objects, and dynamic data overlays at interactive frame rates. The C# scripting layer allows structured data files produced by external optimization pipelines to be loaded at runtime and mapped onto visual scene elements. Its architecture, which fully decouples the data model from the rendering engine, makes it ideal for a pre-computed scenario explorer where hundreds of optimization results must be navigable without re-running any computation.

Despite these advances, none of the existing tools combines: (i) a complete network-level 3D geo-referenced environment; (ii) real-time color-coded visualization of inter-train regenerative energy exchanges at the individual train level; (iii) a full operational scenario explorer spanning 2496 pre-optimized configurations; and (iv) integrated timetable browsing and financial result displays, all within a single interactive platform. The present paper introduces a Unity 6-based 3D visualization platform rendering all 2496 scenarios derived from the GAMS/MATLAB optimization pipeline, making them fully accessible for operational exploration by the CAF-Shapir consortium (TransJerusalem J-Net Ltd.).

Unlike our previous publications, which focused on the development, optimization, and validation of the underlying energy management model for the Jerusalem Green Line LRT, the objective of this paper is the design of an interactive visualization environment that enables intuitive exploration of the resulting operational scenarios. Rather than introducing a new optimization algorithm, the proposed Unity-based platform transforms a large collection of validated simulation results into an operational decision-support tool, allowing users to analyze train movements, regenerative energy exchanges, timetables, and economic indicators within a single geo-referenced 3D environment.

The remainder of this paper is organized as follows: Section II summarizes the energy management model. Section III describes the Unity 3D visualization platform in detail. Section IV presents conclusions.

II. ENERGY MANAGEMENT MODEL

The energy management framework builds on two complementary works. In [18], the authors introduced a

detailed NLP model for the Jerusalem Green Line, a 16.5 km, 33-station, 750 V DC network, covering 16 binary operational scenarios with reversible and non-reversible substation configurations. In [19], that model was implemented in GAMS and validated against real track data from and establishing the core optimization pipeline for the present work. The complete mathematical formulation, optimization procedure, and validation of the energy management model are presented in detail in [19].

A. Network and Rolling Stock

The Jerusalem Green Line connects Gilo (south) with Mount Scopus (north) over 16.5 km with 33 stations and 13 traction substations at 750 V DC, with a real elevation profile with a maximum gradient of 9%. The rolling stock is the CAF Urbos-100: a 5-car, 100% low-floor articulated tram of 124 tons, maximum speed 70 km/h, and capacity of 218 passengers. Track geometry is read from a real measured dataset of 2,154 spatial segments covering both directions (1,036 segments for Track 1: Gilo-Mount Scopus and 1,118 segments for Track 2: Mount Scopus-Gilo).

B. Train Dynamics

Train motion is governed by Newton's second law with Davis resistance forces incorporating rolling, aerodynamic, and gradient components [19]. For each inter-station segment the model includes the equations of movement, constrained by maximum speed, maximum acceleration (1.2 m/s^2), and maximum deceleration (1.3 m/s^2).

C. Energy Exchange

Energy exchange between trains occurs when regenerative braking energy from one train is directly consumed by other trains operating within the same electrical section or any section electrically connected to it. This phenomenon is inherently dependent on the spatiotemporal alignment of train operations and the electrical characteristics of the traction network. For such exchange a temporal overlap must exist between trains in braking and acceleration phases. From a physical perspective, a braking train operates as a generator, injecting power into the DC traction network and nearby accelerating trains can absorb this energy if they are simultaneously demanding traction power. In this work, these energy exchanges are computed from the underlying simulation model and represented within the 3D environment as dynamic interactions between trains, enabling the analysis of their spatial and temporal distribution.

D. Simulation Scenarios

Four binary decision variables define each of the 16 configuration scenarios; i) maximum speed: either 50 km/h or 70 km/h; ii) regenerative braking: either disabled or enabled; iii) optimization objective: minimize cost vs minimize time and iv) ESS usage: no battery available vs battery installed. In [20] it was shown that the configuration featuring regenerative braking, ESS, cost minimization and 50 km/h speed provided a balanced solution characterized by a total round-trip cost of €34.47, round-trip travel time of 119.21 min, battery capacity of 186.62 kWh, and total savings of €270.42 per day.

Beyond the 16 binary scenarios, additional variability in the simulations comes from substation type, energy exchange scope, and service schedule, as summarized in Table I. The three energy exchange configurations are: (i) same section only; (ii) adjacent sections; (iii) all sections + grid sale via reversible substations. Modes 1–9 use preconfigured

schedules proposed in the real project, namely: A means 5 minutes headway in the intervals 5:00 - 10:00 and 16:00 - 22:00 and 15 minutes otherwise; B means 5 minutes headway from 5:00 to 22:00, and 15 minutes otherwise; C means 5 minutes headway all day 5:00 - 1:00. Modes 10–12 sweep tentative headway intervals from 120s to 600s in 10 second intervals, hence producing 49 sub-scenarios each. Note that the total number of modes employed equals 156, which results from multiplying 3 options in the exchanges possibilities (all/same/adjacent) by 52 timetable options: 3 prefixed (A/B/C) plus 49 with variable headways.

TABLE I. OPERATION MODES

Mode	Substation	Energy Exchange	Schedule
1-3	Reversible	All sections	A/B/C
4-6	Non Rev.	Same section only	A/B/C
7-9	Non Rev.	Same and adjacent sections	A/B/C
10	Reversible	All sections	Variable
11	Non Rev.	Same section only	Variable
12	Non Rev.	Same and adjacent sections	Variable

III. VISUALIZATION TOOL FOR ENERGY EXCHANGES

The visualization platform is developed in Unity 6 with C# scripting and Visual Studio. Its architecture fully decouples optimization from rendering: all 2496 scenarios are pre-computed offline by the GAMS/MATLAB pipeline and stored as structured data files. Unity loads these files at runtime and maps them onto a geo-referenced 3D scene of the Jerusalem Green Line. The 3D engine performs no mathematical optimization, it functions as an interactive results explorer and decision-support interface.

Fig. 1 presents the overall architecture of the simulation and visualization pipeline. Real case-study data comprising 2,154 track segments encoding the geometry, gradient, and speed limits of the Jerusalem Green Line, together with the CAF Urbos-100 rolling stock parameters and timetable preferences serve as input to the system. A GAMS optimization module solves a single-train round-trip NLP problem across 16 binary operational scenarios, producing energy and kinematic profiles for each configuration. Concurrently, a set of operating options as substation type (reversible or non-reversible), energy exchange scope, and service headway defines the operational modes fed into the MATLAB post-processing stage. This stage generates up to 2,496 pre-computed scenario files in structured Excel format, comprising: train position and speed updated at 10-second intervals; cumulative energy consumption per train; inter-train energy exchange values; station-by-station timetables for all 33 stops; and a final results summary covering travel time, energy cost, and daily savings for both directions and the round trip. In parallel, the 3D scene comprising the geo-referenced track, traction substations, and station buildings is built once in Unity and remains fixed across all scenarios. At runtime, the user selects the desired operating mode, headway configuration, and binary scenario toggles through the Unity front-end. The Unity environment loads the corresponding pre-computed Excel files and renders the full multi-train simulation in real time, enabling interactive exploration of energy exchange patterns and train movement across the complete network.

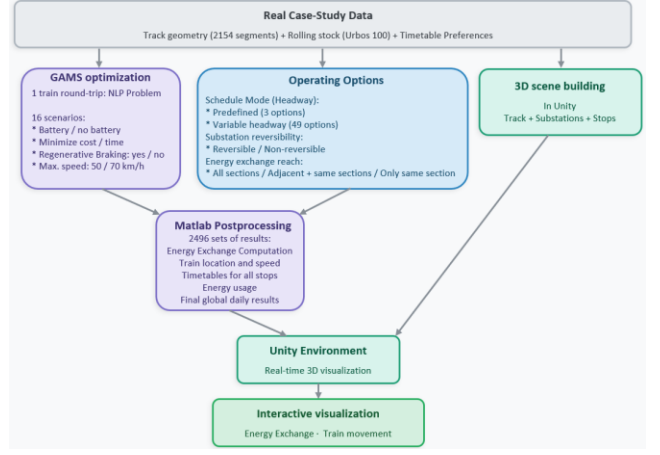


Fig. 1. Process diagram of results visualization.

A. Data Processing (Back-End)

The geometric and operational dataset underpinning the Unity simulation was extracted from the official construction drawings of the Jerusalem Green Line, Route 03, comprising 19 track sections issued for construction in January 2023 by the Jerusalem Transportation Master Plan Team (EGIS/Ingerop) and provided by the CAF-Shapir consortium. These drawings define, for each point in the track, vertical and horizontal alignment radii, gradient, and maximum speed. From these drawings, the dataset was compiled, encoding 2,154 track segments: 1,036 segments for Track 1 (Gilo Terminus to Mount Scopus) and 1,118 segments for Track 2 (Mount Scopus to Gilo). Each segment specifies its initial and final point, length, gradient, and maximum allowed speed.

The official J-Net network map was cropped to display only the Green Line corridor. The 13 traction substations, clearly identified in the dataset, were overlaid on the map as red rectangles, following the standard notation of the construction drawings. This annotated map was then imported into Unity as the geo-referenced background of the 3D scene.

The integration of the dataset into Unity proceeded in three stages. First, a uniform-motion pass was run to verify the geometric alignment of the track: the train model traversed the full route without reading any speed or position file, confirming the visual fidelity of the imported map. Second, the previously mentioned dataset was loaded via C# scripting: segments where velocity = 0 are automatically identified as passenger stops, while marked substation locations trigger the rendering of a traction substation marker. Third, for each of the 2496 operational scenarios, a set of structured Excel files produced by the MATLAB pipeline is loaded at runtime: one schedule file per station (33 files) containing arrival times, and five network-wide tables updated at 10-second intervals related to position, speed, energy state per train, inter-train energy exchange, and aggregate energy figures. The C# controller synchronizes train position, speed, and energy state at each timestep, enabling fully coordinated multi-train animation across both tracks.

B. Graphical User Interface (Front-End)

1) Start screen and scenario configuration

The application opens with a start screen displaying the project title, institutional logos and a scenario configuration panel. The operator selects: (i) operational mode (1–12) via a labeled dropdown menu; (ii) four toggle switches: Use Battery, Minimize Cost or Time, Regenerative Braking, and

Maximum Speed (50/70 km/h); (iii) for Modes 10–12, a slider for headway interval in 10 s increments appears (120s–600 s). Three action buttons are available: Start 3D simulation, Final Result and Time Table (timetable viewer), see Fig. 2.



Fig. 2. Start screen.

2) 3D simulation view

The main simulation view renders a geo-referenced 3D model of the complete Jerusalem Green Line on a white topographic surface derived from the real elevation profile, see Fig. 3. Key visual elements include: i) Station buildings: all 33 stations are represented as 3D structures with rooftops and information screens; ii) Traction substations: all 13 DC substations appear as detailed 3D buildings with electrical panel models, marked by blue spheres on the network map; iii) Train models: CAF Urbos-100 vehicles travel along both tracks with authentic CAF and Shapir logos, entering service progressively according to the active timetable iv) Color-coded energy markers: a marker hovering above each train indicates its instantaneous energy state, green when the train has surplus energy available to share with other trains (regenerative braking phase) and red when the train requires energy from the network (acceleration phase); v) Real-time clock: a simulation clock advances from 05:00 to 01:00 in accelerated time, synchronized with the computed timetable; and vi) live power graph: a real-time power consumption curve is rendered above active trains, updating at each simulation step.

The camera supports free zoom, rotation, and track-following mode. When focused on an individual train, a label displays its identifier, current speed (km/h), and cumulative energy (kWh).

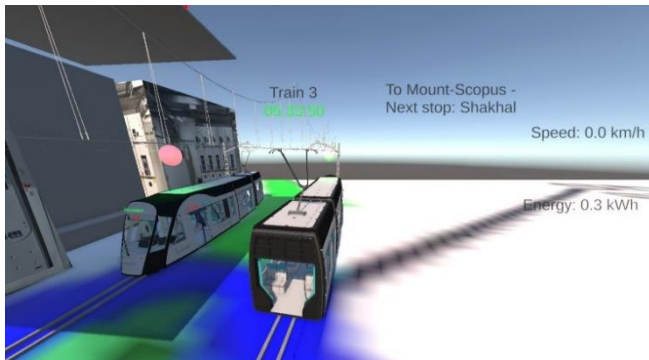


Fig. 3. Simulation view tracking one train.

3) Dual Real-Time Information Panels

During simulation, the display alternates between two information panels overlaid on the 3D view. The first panel (See Fig. 4), “Energy Consumption For Each Train [kWh]”, lists all active trains with instantaneous energy values, positive values in green and negative values in red. The second panel

(See Fig. 5), “Exchange Energy Between Trains [kWh]”, explicitly shows inter-train energy transfers. This dual-panel design is the core operational contribution of the visualization tool: it makes the regenerative energy exchange mechanism directly observable at the individual train level, in a way that no prior simulation tool in the literature achieves.

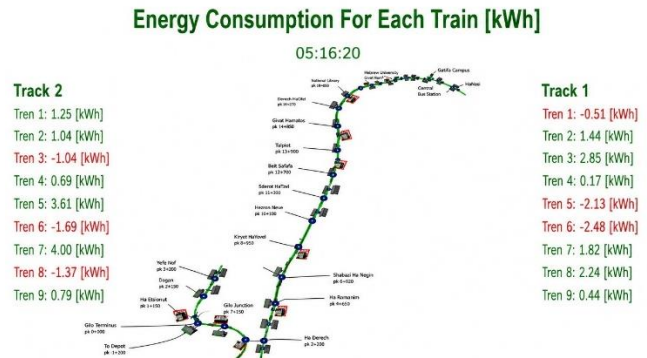


Fig. 4. Simulation view with energy consumption information.

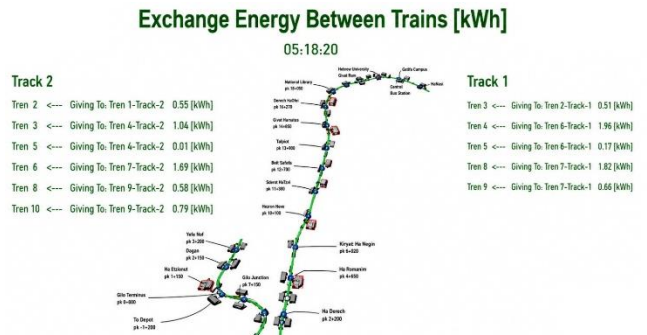


Fig. 5. Simulation view with energy exchange information.

4) Final Results Screen

After simulation, the Final Results screen presents a structured table with columns for Track 1, Track 2, and Round Trip, covering, among other results: travel time, number of trains in peak and off-peak service, battery capacity (kWh), network purchase price (€/kWh), total operating cost (€), total energy shed (kWh), cost saved by inter-train exchange, income from grid energy sale, total savings (€), and commercial speed (km/h). For scenario 0101, all displayed values match €34.47, 119.21 min, and 186.62 kWh exactly.

Detailed simulation and energy management of the trains in the new proposed Green Line in Jerusalem

Simulation Results	Track_1	Track_2	Round_Trip
Time in minutes	59.2562	59.9577	119.2139
Beta	0.3	0.3	0.3
Epsilon	0.8	0.8	0.8
Number of trains at peak hour	11	11	22
Number of trains at off-peak hour	186.6248	186.6248	186.6248
Battery capacity	0.125	0.125	0.125
Network purchase price	15.8468	18.6265	34.4733
Total cost	28232.7	21489.7	49722.4
Sum of energy poured	0.47945	0.64149	1.12094
Cost saved by exchange (peak)	0.47945	0.64149	1.12094
Cost saved by exchange (off-peak)	0.1456	0.1264	0.272
Total cost saved (peak + off-peak)	115.6042	154.72	270.419
Income from grid sale (peak)	16.532	16.4006	16.4663
Income from grid sale (off-peak)			
Total savings (exchange + grid)			
Commercial speed			

Fig. 6. Final Results Screen.

5) Timetable Viewer

A dedicated Time Table screen provides full timetable navigation for all 33 stations on both tracks. The operator selects any station from a dropdown menu and switches between Track 1 and Track 2 tabs to view all scheduled departure times from 05:00 to 01:00. Times are computed

from the GAMS solution applied to the real track profile. This screen allows direct verification of the service schedules implied by each energy management scenario.

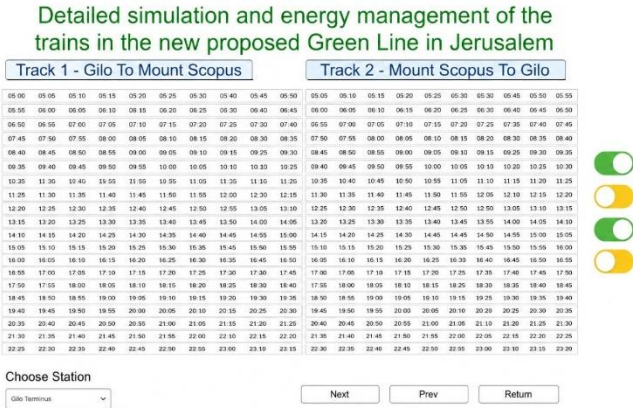


Fig. 7. Timetable viewer.

IV. CONCLUSIONS

This paper has presented a Unity 6-based 3D interactive visualization platform for the energy management simulation of the Jerusalem Green Line LRT featuring a 16.5 km, 33-station, 750 V DC network operated with CAF Urbos-100 vehicles. The main motivation is to visualize in an intuitively accessible form the large volume of results derived from [18] and [19] after considering many different operative scenarios. The tool covers 2496 pre-computed operational scenarios and provides immersive, real-time rendering of inter-train energy exchanges, timetables, and economic results for the complete LRT system.

The key contribution is the integration of spatially-referenced 3D visualization with rigorous energy optimization results, a combination not previously reported in the LRT energy management literature. The color-coded energy exchange system and dual real-time information panels allow operators and planners to observe, at a glance, how regenerative braking energy flows between trains across all simulations. Future work will extend the platform to include real-time SCADA data integration from the operational network and operator-in-the-loop scenario testing, enabling the tool to function as a live decision-support system once the Jerusalem Green Line enters service.

ACKNOWLEDGMENT

The authors would like to thank Mr. Vitali Hanin, CEO of Advance Engineering Ltd, for enabling real-time experience in the project, and for access to technical documentation; the authors would also like to thank Mr. Alon Shoa, from Advanced Reality Lab, for his help with the use of Unity 6. This work was supported by funding from project TED2021-132339B-C42, titled "Business and Market Tools for Renewable Energy Communities", from project PID2022-142372OB-C22, titled "Operation, Business Models and Markets for EV Charging Hubs", both from the Ministerio de Ciencia, Innovación y Universidades and from project DGP_PIDI_2024_01700 from the Consejería de Universidad, Investigación e Innovación - Junta de Andalucía.

REFERENCES

[1] M. Khodaparastan, A. A. Mohamed, and W. Brandauer, "Recuperation of regenerative braking energy in electric rail transit systems," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 8, pp. 2831–2847, Aug. 2019.

[2] A. González-Gil, R. Palacin, and P. Batty, "Sustainable urban rail systems: Strategies and technologies for optimal management of regenerative braking energy," *Energy Convers. Manag.*, vol. 75, pp. 374–388, Nov. 2013.

[3] M. Khodaparastan and A. A. Mohamed, "Modeling and simulation of a reversible substation for recuperation of regenerative braking energy in rail transit systems," in *Proc. IEEE ITEC*, Detroit, MI, USA, 2019, pp. 1–5.

[4] D. Gallo, C. Landi, M. Luiso, and N. Pasquino, "Train-simulator for energy consumption evaluation in mass rapid transit applications," in *Proc. IEEE ICCEP*, Otranto, Italy, 2019, pp. 1–6.

[5] El-Sayed, B. Mohamed, and P. Arbolea, "Web-based software-suite for DC railway simulation and analysis: Architecture, data management and visualization," in *Proc. IEEE VPPC*, Gijón, Spain, 2020, pp. 1–6.

[6] Y. Zhang et al., "Coupling real-time simulation of power supply, trains and dispatching for urban rail transit," *Transp. Saf. Environ.*, vol. 3, no. 3, tdb010, Sep. 2021.

[7] M. Ahmadi, H. J. Kaleybar, M. Brenna et al., "Adapting digital twin technology in electric railway power systems," in *Proc. IEEE PEDSTC*, Tehran, Iran, 2021, pp. 630–635.

[8] X. Li and Y. Liu, "A hybrid data and mechanism model-driven digital twin modelling approach for novel traction power systems," *Urban Rail Transit*, vol. 11, pp. 221–233, 2025.

[9] Y. Zhao and X. Wang, "Virtual simulation system of railway signal equipment based on Unity3D," *Proc. SPIE 12160, CMSDA 2021*, 121600W, Mar. 2022.

[10] H. Hu, J. Chen, J. Zhu, Y. Yang, and H. Zheng, "Design and development of an integrated virtual-reality training simulation sand table for rail systems," *Information*, vol. 15, no. 3, p. 141, Mar. 2024.

[11] A. Nikitin et al., "VR training for railway wagons maintenance: Architecture and implementation," *Procedia Comput. Sci.*, vol. 176, pp. 622–631, 2020.

[12] J. Liu, C. Liu, L. Zheng, and H. Li, "Construction of virtual reality rail transit simulation system," *J. Electr. Syst.*, vol. 20, no. 2, Apr. 2024, doi: 10.52783/jes.1102.

[13] Y. Nie et al., "Implementing railway vehicle dynamics simulation in general-purpose multibody simulation software packages," *Adv. Eng. Softw.*, vol. 130, pp. 1–15, 2019, doi: 10.1016/j.advengsoft.2018.12.003.

[14] K. Łukaszewski, P. Buń, and A. Karwasz, "Comparison of monorail vehicle dynamic behavior in Unity and Universal Mechanism," in *Lecture Notes in Mechanical Engineering*, Advances in Manufacturing IV, Springer Nature Switzerland, 2024, pp. 48–59, doi: 10.1007/978-3-031-56463-5_4.

[15] G. D'Amico, et al. "TrainSim: A railway simulation framework for LiDAR and camera dataset generation," *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 12, pp. 15006–15017, 2023.

[16] X. Shi, S. Yang, and Z. Ye, "Development of a Unity–VISSIM co-simulation platform to study interactive driving behavior," *Systems*, vol. 11, no. 6, p. 269, May 2023.

[17] A. Mohammadi, et al. "An open-source virtual reality traffic co-simulation for enhanced traffic safety assessment," *Appl. Sci.*, vol. 15, no. 17, p. 9351, Aug. 2025.

[18] S. de la Torre and M. Eitan, "Detailed simulation and energy management of the trains in the new proposed Green Line in Jerusalem," in *Proc. IEEE VPPC*, Washington DC, USA, 2024.

[19] M. Eitan and S. de la Torre, "Energy Management and Optimal Operation of Light Rail Trains on Jerusalem's Proposed Green Line", *IEEE Open J. of Vehic. Tech.*, vol. 7, pp. 1137–1150, 2026, doi: 10.1109/OJVT.2026.3676227.

[20] X. Yang, X. Li, Z. Gao, H. Wang, and T. Tang, "A cooperative scheduling model for timetable optimization in subway systems," *IEEE Trans. Intell. Transp. Syst.*, vol. 14, no. 1, pp. 438–447, Mar. 2013.

[21] S. Su, X. Li, T. Tang, and Z. Gao, "A subway train timetable optimization approach based on energy-efficient operation strategy," *IEEE Trans. Intell. Transp. Syst.*, vol. 14, no. 2, pp. 883–893, Jun. 2013.

[22] A. Nasri, M. F. Moghadam, and H. Mokhtari, "Timetable optimization for maximum usage of regenerative energy of braking in electrical railway systems," in *Proc. IEEE SPEEDAM*, Pisa, Italy, 2010, pp. 1218–1221.