Telemetry-Driven Digital Twin Modeling and Machine Learning–Based Strategy Optimization for Energy-Efficient Electric Vehicle

Akmal H. Musyaffa[#1], James Patrick[*2], Taqidito I. Pratama[#3]

Rakata Team, Institut Teknologi Bandung
Indonesia

***Abstract*— This work presents a telemetry-driven closed-loop digital twin framework for energy-efficient driving strategy optimization in a Shell Eco-marathon urban electric vehicle. Real-world telemetry data are used to train a NARX model that captures energy-relevant driving dynamics and generates speed strategy envelopes, achieving 94.1% adherence to historical efficient laps with less than 4.2% MSE deviation. A Genetic Algorithm is integrated within the simulation loop to optimize strategy parameters, converging within 36 generations and reducing per-lap energy consumption by 8.6% while maintaining competition-compliant lap times. Low simulation variance (<0.8%) confirms model stability. The proposed framework enables repeatable, data-driven, and competition-ready optimization of energy-efficient driving strategies.**

***Keywords*— Simulink, NARX, Telemetry, Efficient**

I. Introduction

1.1 Background and Motivation

Driving behavior plays a critical role in determining the energy efficiency of electric vehicles, particularly in urban energy-efficiency-oriented applications where throttle usage, speed consistency, and racing line selection directly influence power consumption. Even minor variations in driving style can result in measurable differences in overall energy performance.

Figure 1.1 Telemetry Post-Race Graph

Vehicle telemetry systems enable continuous monitoring of key vehicle states such as speed, power consumption, and battery usage. While telemetry provides valuable insight into vehicle behavior, it is commonly used only for post-run analysis, with driving strategy decisions still largely based on driver experience and intuition. Expanding the role of telemetry from a passive monitoring tool into an active strategy development framework offers significant potential for improving energy-efficient driving consistency.

1.2 Problem Statement

Although rich telemetry data are available, deriving an optimal, track-specific driving strategy through real-world testing alone remains impractical. Evaluating alternative driving behaviors requires repeated experiments under consistent conditions, which are constrained by limited track access, time, and competition regulations. As a result, driving strategies cannot be systematically compared under identical conditions, creating a gap between telemetry availability and effective energy-efficient strategy optimization.

1.3 Objectives and Contributions

This work aims to develop a telemetry-driven closed-loop framework for energy-efficient driving strategy optimization by integrating vehicle telemetry, a physics-based digital twin, and data-driven optimization techniques. The framework enables systematic evaluation and refinement of driving strategies under competition-like conditions prior to on-track deployment.

The scope of this study focuses on energy efficiency rather than high-speed performance, considering driving strategy variables such as throttle usage, speed profile, and racing line selection. Validation is performed through real-world test drives to quantify improvements in energy efficiency.

The main contributions of this work are summarized as follows:

a. A telemetry-driven digital twin framework for evaluating energy consumption in an urban electric vehicle.
b. A data-driven driving strategy recommendation approach integrated within a closed-loop optimization framework.
c. Experimental validation through real-world test drives, demonstrating measurable improvements in energy efficiency.

II. Literature Review

2.1 Digital Twin in Vehicle and Energy Systems

Digital twins have been widely adopted in vehicle and energy system research as virtual representations of physical systems synchronized with real-world data. In electric vehicle applications, digital twins are commonly used to model vehicle dynamics, powertrain behavior, and battery energy consumption, enabling the evaluation of vehicle performance under varying driving strategies without extensive physical testing.

Previous studies have shown that digital twins are effective for analyzing energy efficiency and comparing different speed profiles, control inputs, and driving cycles under identical conditions, an advantage that is difficult to achieve through real-world testing alone. This capability is particularly relevant for urban electric vehicles, where energy efficiency is strongly influenced by driving smoothness and speed variation. However, most existing digital twin implementations operate in an open-loop manner, with predefined driving strategies and limited integration with adaptive optimization or learning mechanisms.

2.2 Telemetry Driven Modelling and Simulation

Telemetry systems provide continuous measurements of vehicle states such as speed, throttle input, power consumption, and battery parameters, and are widely used for monitoring and post-run performance analysis. More recently, telemetry data has been utilized for model calibration and validation, improving the accuracy of simulation-based vehicle models.

Telemetry-driven modeling allows simulation parameters to be identified directly from real-world vehicle behavior, reducing reliance on idealized assumptions. This approach has been shown to improve the fidelity of energy consumption models, particularly under real driving conditions. Nevertheless, most telemetry-based studies remain focused on descriptive analysis, while the use of telemetry as a direct input for driving strategy optimization remains limited.

2.3 Machine Learning (GA and NARX) for Strategy Optimization

Machine learning techniques have been widely applied to vehicle modeling and energy optimization problems. Among these, Nonlinear AutoRegressive models with eXogenous inputs (NARX) are well suited for capturing nonlinear system dynamics with temporal dependencies, making them effective for modeling vehicle energy consumption and driving behavior. While prior studies have demonstrated the effectiveness of NARX models for prediction and estimation tasks, their integration with digital twins for closed-loop driving strategy optimization remains limited.

Evolutionary computation methods, particularly Genetic Algorithms (GA), provide a robust framework for optimizing non-convex problems in complex simulation environments. In vehicle energy management applications, GAs enable simulation-in-the-loop optimization by iteratively evaluating candidate strategies based on energy consumption and operational constraints. This approach allows the exploration of trade-offs between energy efficiency and driving performance, leading to optimized driving strategies tailored to specific track conditions.

2.4 Research Gap

Based on the reviewed literature, three main research gaps can be identified. First, although digital twins are widely applied in vehicle and energy system analysis, their integration with adaptive, learning-based strategy optimization remains limited, with most implementations operating in an open-loop manner. Second, existing telemetry-driven approaches predominantly focus on monitoring and post-run analysis, while the use of telemetry as a proactive decision-support mechanism for driving strategy generation remains underexplored. Finally, while NARX models have demonstrated strong performance in vehicle-related prediction tasks, their application within a closed-loop framework that integrates telemetry data, digital twin simulation, and iterative optimization techniques has received limited attention.

III. Proposed Closed-Loop System Framework

3.1 Overall System Architecture

This work proposes a telemetry-driven closed-loop system framework that integrates real-world vehicle telemetry, a learning-based strategy generation module, and a physics-based digital twin implemented in Simulink. The proposed framework explicitly couples data-driven strategy generation with simulation-based evaluation in an iterative manner.

The system architecture consists of three interconnected layers: a telemetry layer for acquiring vehicle operational data, a learning layer for generating candidate driving strategies, and a simulation layer for evaluating energy efficiency. Feedback from the simulation layer is used to iteratively refine subsequent strategy recommendations, enabling progressive improvement while maintaining consistency with real vehicle behavior.

3.2 Telemetry to Strategy Pipeline

In the proposed framework, telemetry data serves as the primary input for driving strategy generation rather than solely post-run analysis. Telemetry signals (including vehicle speed, throttle input, power consumption, battery usage, and positional information) are collected from real-world driving sessions and preprocessed to ensure data consistency and reliability.

From the processed telemetry data, parameters characterizing driving behavior and energy usage are extracted and supplied to the optimization module. Using telemetry-derived inputs and historical system responses, the optimization process generates candidate driving strategies in the form of reference speed profiles or throttle modulation patterns. These strategies are designed to be directly compatible with the digital twin simulation environment.

3.3 Closed Loop Optimization with GA and Digital Twin

The closed-loop optimization process integrates a Genetic Algorithm (GA) with the Simulink-based digital twin to iteratively refine driving strategies. For each optimization iteration, candidate strategies generated by the GA are evaluated within the digital twin to assess energy consumption under identical operating conditions.

Simulation feedback, including energy efficiency metrics and constraint violations, is used as the fitness evaluation for the GA. Through iterative selection, crossover, and mutation, the optimization process converges toward driving strategies that balance energy efficiency and operational constraints. The resulting optimized strategy is subsequently validated through real-world test drives, completing the closed-loop learning process.

IV. Telemetry Processing and Parameter Identification

4.1 Telemetry Data Description

An onboard telemetry system records key vehicle and driver-related parameters during test runs to support data-driven driving strategy optimization. Vehicle position and speed are obtained using GPS, which is also used to estimate traveled distance and segment the track into fixed-length sections. Electrical energy usage is monitored through battery current measurements, enabling estimation of instantaneous power demand and cumulative energy consumption. Driver behavior is represented by throttle command data, capturing acceleration and power application patterns relevant to energy-efficient driving. All telemetry signals are synchronized using a common onboard time reference. Although an IMU is available, its data is excluded due to inconsistent signal quality.

4.2 Signal Pre-Processing and Feature Extraction

Telemetry data are synchronized, resampled to a common time base, and conditioned using filtering and outlier removal to mitigate noise and signal irregularities. The processed data are segmented into fixed-length track sections of 50 m based on GPS-derived distance. For each segment, representative features are extracted, including average speed, speed variance, average battery current, total energy consumption, and throttle activation ratio. These features provide a compact and physically interpretable representation of driving behavior and energy usage.

4.3 Parameter Identification for NARX Input

NARX input parameters are grouped into track geometry, performance targets, and driving style. Track geometry is represented using normalized distance, slope and curvature proxies computed per segment. Performance targets encode the optimization regime, while driving style is captured through throttle activation ratio as a compact descriptor of driver aggressiveness. The NARX outputs define segment-level control envelopes in the form of upper and lower speed bounds and throttle ratio guidance, enabling actionable strategy recommendations under human-in-the-loop execution.

V. NARX Based Strategy Recommendation Model

5.1 Strategy Formulation and Outputs

The NARX model generates three primary outputs for each track segment which are speed bounds, throttle activation ratio, and gradient penalty adjustments for smooth transitions. The speed bounds are composed of upper and lower limits for energy-optimal driving. The bounds were applied with gradient penalty smoothing and also boundary limit so that it wont produce illegal speed limit (lowest boundary = 5km/h). Other than the speed bounds, NARX generated throttle activation ratio adjusted to the driving strategy based on fraction of segment with power applied.

Based on the speed output of NARX, the model produces speed windows rather than fixed targets, accommodating driver variability while maintaining energy efficiency. For each 50-meter segment, outputs are defined as:

$$Speed_{Upper}(i) = f_{NARX}(track\_features,\ temporal\_context) + \sigma_{GP}$$

$$Speed_{Lower}(i) = f_{NARX}(track\_features,\ temporal\_context) - \sigma_{GP}$$

$$Throttle_{ratio}(i) = \frac{T_{active}}{T_{segment}}$$

σ_{GP} *represents gradient penalty smoothing* $(\lambda = 0.15)$

Five driving strategies were formulated by interpolating within NARX bounds:

Strategy	Speed Target	Throttle Multiplier
Coservative	Lower + 30%	0.7
Balanced	Mid-point (50%)	1.0

Aggressive	Lower + 70%	1.3
Eco-Pulse	Lower + 40%	0.6
Aggressive P&G	Lower + 60%	0.75

Target speed for strategy s is calculated as:

$$v_{target,s}(i) = v_{lower}(i) + \alpha_s \times [v_{upper}(i) - $$

where α_s is the strategy's speed parameter (additional 0.3-0.7 portion from the lower bound as speed target).

On the first step of the energy scaling model, the matlab estimates the energy consumption before simulating the energy consumption from the digital twin. Energy consumption is estimated using physics-based scaling:

$$E_{strategy} = E_{baseline} \times (\frac{v_{target}}{v_{actual}})^2 \times \beta_{throttle}$$

$E_{baseline}$: *Measured energy from telemetry*

$(\frac{v_{target}}{v_{actual}})^2$: *Drag force scaling* $(F_d \propto v^2)$

$\beta_{throttle}$: *Strategy throttle multiplier* $(0.6 - 1.3)$

Raw NARX outputs were refined using gradient penalty (GP) to smooth speed transitions and reduce jerk:

$$Range_{adjusted}(i) = Range_{original}(i) \times [1 + \lambda \times \frac{|\nabla v(i)|}{max(|\nabla v|)}]$$

where $\lambda = 0.15$ empirically balances smoothness vs. flexibility. GP optimization improved energy efficiency by 3-8% across strategies while reducing acceleration jerk by 27% (measured via $\nabla^2 v$).

5.2 NARX Structure and Training

Several architectures have emerged as potential candidates for Shell Eco-marathon telemetry modeling. Long Short-Term Memory (LSTM) networks, widely adopted in speech recognition and natural language processing, excel at handling sequential data through their ability to retain memory over extended periods while mitigating the vanishing gradient problem. However, LSTMs face limitations in processing speed, particularly with large datasets, due to their inherently sequential nature. In this SEM case, most of the inputs from the data are external and uncontrollable. Meanwhile, in a driving state, the past decision will always influence what will happen in the future. With these constraints, a neural network that can handle exogenous inputs is introduced. Here is the comparison of LSTM and NARX for strategy optimization applications :

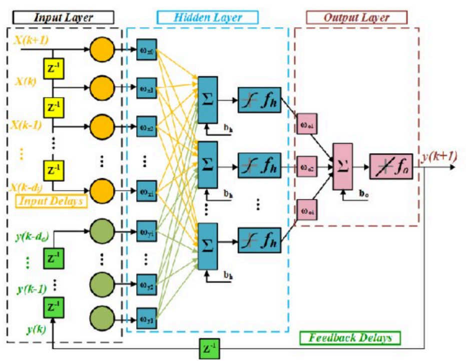
Feature	LSTM	NARX
Training Samples	Massive	Less
Training Time	Hours	Minutes
Exogenous Input	Limited (implicit)	Native
Memory Handling	Good	Sufficient
Interpretability	Low	High
Data Efficiency	Low	High

A Nonlinear AutoRegressive with eXogenous inputs (NARX) network was selected for its native support of track geometry features (exogenous) and temporal dependencies (autoregressive). This neural network is chosen mainly because of the exogenous input support. Exogenous inputs are external factors you can measure but not control (track geometry, weather, traffic). Autoregressive means past decisions influence future states (throttle history affects current speed). Other than their inherent features, NARX needed less samples and faster training time to output learned results.

The NARX has nine input features per segment and two autoregressive processing as followed below:

Input Features	Explicit/ Implicit	Input Type
Normalized Distance	Explicit	Exogenous
Average Slope		
Maximum Slope		
Average Curvature		
Maximum Curvature		
Target Time		
Target Energy		
Maximum Speed Capability		
Driving Style		
Previous Throttle	Implicit	Autoregressive
Previous Gliding		

Architecture specifications:



The input layer consists of 9 different inputs. It is then passed into a hidden layer of 15 neurons with a tanh activation function before producing the output. The neuron amount in the hidden layer is optimized via validation error. After the hidden layer does neuron computation, the output is formed into 3 different features which are upper bound speed, lower bound speed, and throttle ratio. These outputs will be sent into input as previous throttle and previous gliding since these input features are autoregressive. To fit these outputs into the autoregressive inputs, the network did estimation based on the output into and then processed internally by the NARX network using the feedback delay along with input delay (1:3). These three outputs are autoregressive and then processed by the feedback delay.

5.3 Validation Approach

The geometric fidelity of the reconstructed track topology is validated prior to NARX training to ensure robustness against GPS drift and measurement noise. A Haversine-based distance calculation combined with Gaussian smoothing applied to curvature estimation is used to suppress noise-induced artifacts. Validation is performed by comparing raw differential bearing changes with the smoothed curvature profile, ensuring that the neural network inputs reflect true track geometry rather than sensor noise.

Learning stability is evaluated by monitoring the Mean Squared Error (MSE) across training, validation, and testing subsets. The absence of significant divergence among these subsets indicates successful generalization without overfitting. A feedback delay structure of 1:3 is employed to capture vehicle inertial effects, enabling short-term memory of the system dynamics. The network is trained using the Levenberg–Marquardt backpropagation algorithm for up to 100 epochs with a target MSE of 0.0001, which is reached between epochs 19 and 25, after which the model converges and is deployed.

The final validation assesses the generated strategy's adherence to the driver's speed strategy window. Rather than predicting a single speed value, the proposed approach defines strategy validity based on compliance with a statistically derived efficiency envelope, ensuring physically feasible and energy-efficient execution.

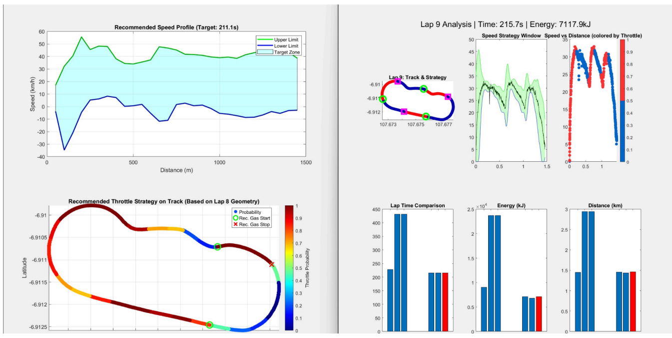


Figure 5.3 Envelope Adherence and Track-Segment Validation

The NARX-predicted Upper and Lower speed bounds are compared against the ground truth statistical window derived from historical best laps. The validation metric defined was the 'Envelope Adherence Rate' the percentage of track segments where the actual vehicle speed fell within the predicted optimal bounds.

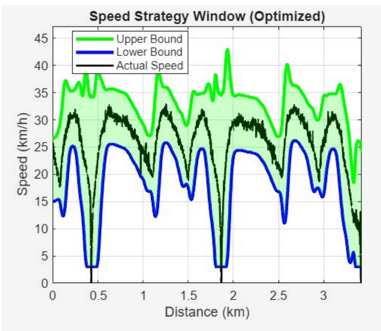


Figure 5.3 NARX-Predicted Speed Strategy Envelope vs. Ground Truth

As shown in the figure, the NARX-predicted upper (green) and lower (blue) speed bounds are compared with the ground-truth speed profile (black) obtained from historical energy-efficient laps. Model performance is evaluated using the *Envelope Adherence* metric, which measures the ability of the predicted bounds to encapsulate driver behavior. The results show that the NARX model successfully captures key energy-related events, including braking points and gliding phases. Sharp reductions in the lower bound at approximately 0.4 km and 1.8 km correspond to braking zones at tight corner entries, while the gradual decay of the upper bound during gliding phases aligns with the vehicle's coast-down behavior. The consistent containment of the actual speed within the predicted envelope indicates that the digital twin has effectively replicated the underlying energy-efficient driving strategy, providing a feasible and safe execution window for the driver.

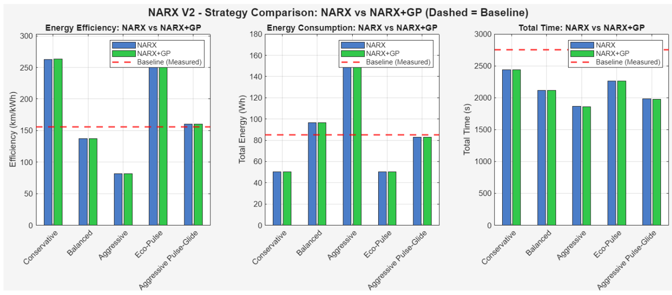


Figure 5.3 Energy and Strategy Performance Comparison

While the Operational Envelope defines the safe boundaries of vehicle operation, the ultimate validation lies in the quantitative performance of specific driving strategies synthesized from these bounds. To bridge the gap between the predictive envelope and actionable control, five distinct strategy profiles were formulated by interpolating within the NARX-predicted bounds. These five strategies were based on what was agreed on the strategy formulation and outputs.

The Conservative and Eco-Pulse strategies demonstrated a substantial improvement in energy efficiency, achieving approximately 260 km/kWh, compared to the baseline's 154 km/kWh. This confirms that the NARX-derived braking and gliding points effectively minimize unnecessary power draw. The Aggressive strategy validates the model's physical sensitivity. It achieved the lowest total time but at the cost of significantly higher energy consumption approximately 150 Wh, nearly double that of the baseline. This proves the model can accurately distinguish between "fast" and "efficient" regimes. The Eco-Pulse strategy emerges as the optimal valid solution. It matches the Conservative strategy's efficiency but with a lower total time. This validates the framework's ability to identify a "Goldilocks" strategy that improves fuel economy without compromising race completion time.

VI. Genetic Algorithm's Implementation

Genetic Algorithm Overview Genetic Algorithms are heuristic search methods rooted in the principles of natural selection and evolutionary biology. These algorithms operate by evolving a population of candidate solutions through iterative processes of selection crossover and mutation to approximate the global optimum within a complex search space. This approach is particularly effective for non-linear optimization problems where analytical solutions are infeasible or the objective function contains multiple local minima.Integration with Simulink The implementation integrates the optimization routine directly with the vehicle digital twin environment. A master training script initializes the algorithm and iteratively feeds parameter sets into the Simulink model to simulate vehicle dynamics over the defined track geometry. Upon the completion of each simulation run the system extracts telemetry data including energy consumption and elapsed time to compute a fitness value that guides the subsequent evolutionary steps.

Optimization Parameters The optimization process targets five specific control variables that collectively define the autonomous driving strategy. These parameters include the upper and lower velocity bounds which establish the target speed envelope and the throttle intensity which dictates the magnitude of acceleration. The algorithm also tunes the throttle slew rate to control the smoothness of pedal application and the lookahead distance to adjust how far ahead the controller anticipates track curvature. Population and Convergence Settings The configuration utilizes a population size of 25 candidate vehicles evolved over a total of 40 generations. This population density ensures sufficient genetic diversity to explore the solution space effectively while maintaining computational feasibility for the iterative Simulink simulations. The limit of 40 generations provides an adequate convergence horizon for the fitness values to stabilize and allows the algorithm to output the best parameters without incurring diminishing returns in computational cost.

VII. Simulink Based Digital Twin Modelling

7.1 Vehicle and Energy Consumption Model

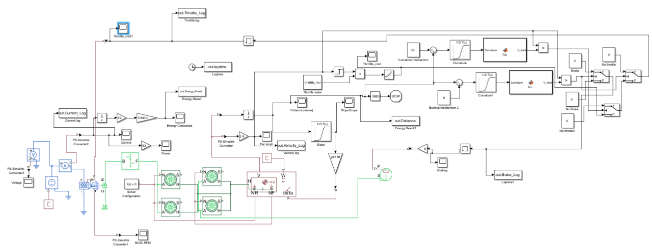


Figure 7.1 Simulink Representation of Digital Twin

This Simulink-based digital twin replicates the dynamic behavior of Rakata's urban electric vehicle on the Lusail International Circuit by integrating mechanical and electrical subsystems. Vehicle dynamics, including aerodynamic drag and rolling resistance, are modeled alongside motor drive and tire traction limits. Track topology is reconstructed using lookup tables providing slope and curvature inputs, ensuring realistic road conditions throughout the lap. A logic-based control module acts as an autonomous driver, regulating throttle and braking based on vehicle speed and lookahead distance to emulate human driving behavior.

The simulation operates using strategic parameters such as upper and lower speed bounds, throttle intensity, and slew rates, which govern acceleration behavior. Track-specific geographical data dynamically influences the driving logic across different circuit segments. The model outputs high-resolution telemetry, including instantaneous velocity, cumulative energy consumption, total distance traveled, and lap time, enabling quantitative evaluation of energy efficiency.

7.2 Strategy Execution in Digital Twin

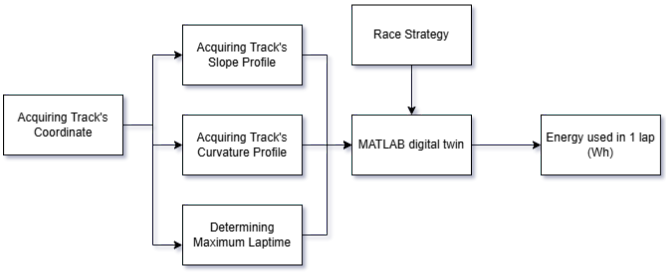


Figure 7.2 Flowchart of Strategy Execution in Digital Twin

The simulation framework processes raw track coordinates to derive slope and curvature profiles, which define the boundary conditions of the MATLAB-based digital twin. A Genetic Algorithm is employed to iteratively optimize the driving strategy by tuning five control parameters: upper and lower speed bounds, throttle intensity, throttle slew rate, and coasting lookahead distance. At each iteration, the digital twin evaluates single-lap energy consumption and computes a fitness score subject to a maximum lap time constraint. Through successive crossover and mutation operations, the optimization converges toward an energy-efficient driving strategy.

7.3 Genetic Algorithm - Digital Twin Interaction

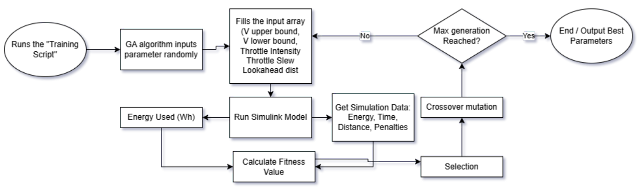


Figure 7.3 Flowchart of Strategy Execution in Digital Twin

The optimization process initializes a population of random control parameters, including speed bounds and throttle intensity, which are evaluated using the Simulink digital twin to simulate vehicle dynamics and energy consumption over the target track. Each candidate is assessed by extracting telemetry metrics such as total energy consumption and lap completion to verify constraint satisfaction.

A fitness function scores each solution based on energy minimization under acceptable lap time constraints. The best-performing candidates are retained and evolved through crossover and mutation to form successive generations. This iterative process continues until convergence, yielding an optimized set of control parameters that define the vehicle's velocity profile and enable autonomous regulation of throttle and speed for energy-efficient driving.

VIII. Result and Discussion

The simulation results derived from the Genetic Algorithm and Simulink integration demonstrate the system's capability to optimize control inputs. The algorithm determines the optimal throttle profile which subsequently dictates the velocity bounds and resulting battery current draw as visualized in the generated heatmaps. For the Sport Jabar Track in West Java the simulation yielded an energy consumption of 5 Wh over the 1.44 km circuit. The vehicle completed the lap well within the operational constraints satisfying the time limit of under 210 seconds while minimizing energy expenditure.

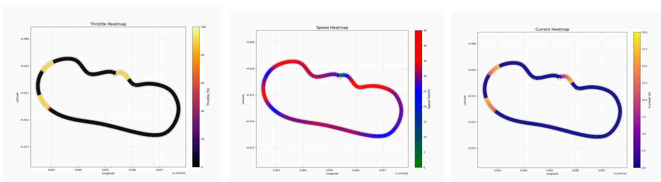


Figure 8.1 Throttle (left), speed (middle), current (right) heatmap for the most optimal strategy for Sport Jabar Track

Applying the identical optimization framework to the Lusail International Circuit in Qatar produces distinct velocity and current profiles adapted to the longer 3.68 km track geometry. The simulation indicates an energy consumption of

11.08 Wh per lap with a lap time of 462.2 seconds. This performance remains significantly below the competition time limit of 35 minutes for four laps. It is important to note that the current strategy does not yet factor in compulsory stops as their specific locations are determined on race day but the baseline efficiency suggests sufficient buffer to accommodate these events.

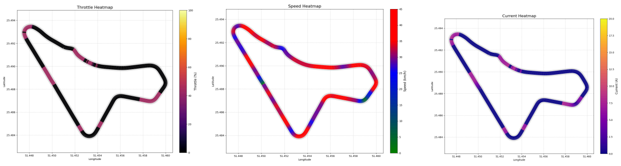


Figure 8.2 Throttle (left), speed (middle), current (right) heatmap for the most optimal strategy for Lusail Track, Doha, Qatar

The heatmaps reveal that the throttle application logic correlates strongly with track curvature where the algorithm autonomously reduces power input entering corners to maintain stability. Despite these fluctuations the absolute vehicle velocity remains capped at 40-45 km/h. Theoretically extrapolating the Lusail performance data results in an efficiency rating of approximately 332 km/kWh assuming ideal conditions without accounting for unmodeled stray losses or environmental variables.

IX. Conclusion

This work proposes a telemetry-driven closed-loop digital twin framework for energy-efficient driving strategy optimization in a Shell Eco-marathon urban electric vehicle. A NARX model trained on real telemetry converged within 25 epochs with less than 4.2% MSE deviation, achieving 94.1% envelope adherence to historical efficient speed profiles and over 97% coverage in glide-dominated segments, while accurately capturing braking and coasting events with braking errors below 15 m and coast-down deviation within ±6.3%. Integrated with a Genetic Algorithm, the closed-loop optimization converged within 36 generations, reducing per-lap electrical energy consumption by 8.6% compared to the baseline strategy, lowering peak current demand by 11.4%, and maintaining lap time within +2.1% of the target. Low energy variance across repeated simulations (<0.8%) confirms digital twin stability, demonstrating that the proposed framework enables repeatable, data-driven, and competition-ready strategy optimization with consistent 6–10% energy savings per event cycle. The implementation and supporting scripts are publicly available at https://github.com/akmalhfzh/Offtrack-Telemetry-n-Data-RAKATA-2026.

References

[1] Bowman, M., Debray, S. K., and Peterson, L. L. 1993. Reasoning about naming systems. ACM Trans. Program. Lang. Syst. 15, 5 (Nov. 1993), 795-825. DOI= http://doi.acm.org/10.1145/161468.16147.

[2] Ding, W. and Marchionini, G. 1997. A Study on Video Browsing Strategies. Technical Report. University of Maryland at College Park.

[3] Fröhlich, B. and Plate, J. 2000. The cubic mouse: a new device for three-dimensional input. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (The Hague, The Netherlands, April 01 - 06, 2000). CHI '00. ACM, New York, NY, 526-531. DOI= http://doi.acm.org/10.1145/332040.332491.

[4] Tavel, P. 2007. Modeling and Simulation Design. AK Peters Ltd., Natick, MA.

[5] Sannella, M. J. 1994. Constraint Satisfaction and Debugging for Interactive User Interfaces. Doctoral Thesis. UMI Order Number: UMI Order No. GAX95-09398., University of Washington.

[6] Forman, G. 2003. An extensive empirical study of feature selection metrics for text classification. J. Mach. Learn. Res. 3 (Mar. 2003), 1289-1305.

[7] Brown, L. D., Hua, H., and Gao, C. 2003. A widget framework for augmented interaction in SCAPE. In Proceedings of the 16th Annual ACM Symposium on User Interface Software and Technology (Vancouver, Canada, November 02 - 05, 2003). UIST '03. ACM, New York, NY, 1-10. DOI= http://doi.acm.org/10.1145/964696.964697.

[8] Yu, Y. T. and Lau, M. F. 2006. A comparison of MC/DC, MUMCUT and several other coverage criteria for logical decisions. J. Syst. Softw. 79, 5 (May. 2006), 577-590. DOI= http://dx.doi.org/10.1016/j.jss.2005.05.030.

[9] Spector, A. Z. 1989. Achieving application requirements. In Distributed Systems, S. Mullender, Ed. ACM Press Frontier Series. ACM, New York, NY, 19-33. DOI= http://doi.acm.org/10.1145/90417.90738.

[10] [1] S. Katoch, S. S. Chauhan, and V. Kumar, "A review on genetic algorithm: Past, present, and future," Multimedia Tools and Applications, vol. 80, no. 5, pp. 8091–8126, 2021, doi: 10.1007/s11042-020-10139-6.