Australasian Transport Research Forum 2024 Proceedings
27-29 November, Melbourne, Australia
Publication website: https://australasiantransportresearchforum.org.au/

Network free flow speed calibration in MATSim with Google API data: A Melbourne study

Sapan Tiwari[1], Chengqi Lu[2,*], Afshin Jafari[1]

[1]Centre for Urban Research, School of Global, Urban and Social Studies, RMIT University, Melbourne, Victoria, 3004, Australia

[2] Technische Universität Berlin, Chair of Transport Systems Planning and Transport Telematics, Straße des 17. Juni 135, 10623 Berlin, Germany

* Email for correspondence: lu@vsp.tu-berlin.de

1 Introduction

Simulation models are crucial for forecasting traffic flow, evaluating infrastructure changes, and optimising traffic operations in transportation planning. Among these models, Multi-Agent Transport Simulation (MATSim) is a powerful tool for analysing individual mobility behaviour within transportation networks (Horni et al., 2016). The accuracy of MATSim, however, largely depends on precise network parameters, especially the free-flow speeds of road segments. Free-flow speed represents a vehicle's maximum speed without obstructions such as traffic lights or other vehicles. These speeds play a key role in replicating real-world traffic conditions and ensuring reliable simulation outcomes for decision-making.

In practice, factors like road type and traffic density complicate this estimation of free-flow speeds. Consequently, free-flow speed correction becomes essential in simulations to account for these real-world complexities (Caliper, 2008; PTV AG, 2015). Traffic simulators typically adjust free-flow speeds on urban roads where traffic lights exist but are not explicitly modelled, simulating more realistic traffic conditions. Adopting machine learning techniques, including neural networks, has enhanced the accuracy of traffic model calibration. These methods improve travel times and queue length predictions, which are critical for realistic traffic simulations (Ištoka Otković et al., 2023). Rakow & Nagel (2024) demonstrated how integrating machine learning with microscopic simulations using data from SUMO (Lopez et al., 2018) can dynamically adjust free-flow speeds and reduce prediction errors.

This paper introduces a novel method for adjusting MATSim's free-flow speeds using real-world travel time data from the Google Distance API (Google Maps Platform, 2022). This approach also refines free-flow speed parameters extracted from OpenStreetMap (OSM) data (Contributors, 2017), enhancing the realism of network models. This approach bridges the gap between simulation parameters and empirical traffic conditions, improving MATSim's reliability. By developing an automated algorithm that refines free-flow speeds based on real-time data, this study contributes a dynamic framework that adapts to changing traffic patterns, enhancing traffic management strategies.

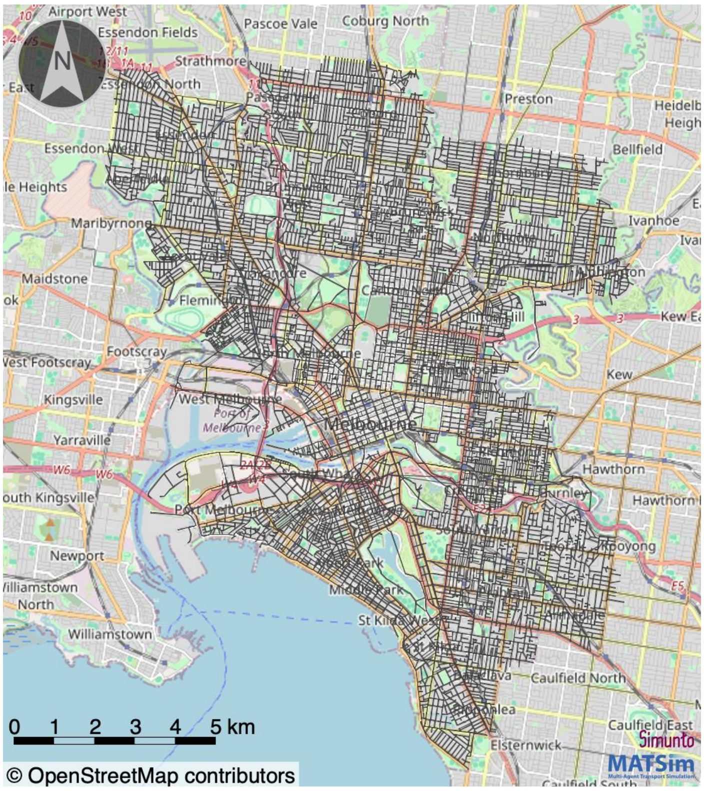
2 Methodology

This study utilises an existing agent-based MATSim model for the Greater Melbourne Network (Jafari et al., 2022), focusing on the Melbourne Inner SA4[1] area. The network was generated using OpenStreetMap (OSM) data as of 3 April 2024, as shown in Figure 1. We calibrated the model by extracting 14,670 origin-destination (OD) pairs from previous studies (Tiwari et al., 2023, 2024), which define the routes requiring calibration. Later, the OD pair samples were divided into a random 90/10 and 95/5 percent split, where 90% and 95% OD pairs were used to train the model, and the remaining OD pairs were used to test the model. This approach is chosen to check for overfitting.

[1]Statistical Areas Level 4 (SA4) are the largest sub-State regions in the Main Structure of the Australian Statistical Geography Standard (ASGS).
https://www.abs.gov.au/ausstats/abs@.nsf/Lookup/by%20Subject/1270.0.55.001~July%202016~Main%20Features~Overview~1

Real-world travel times and distances were retrieved using the Google Distance Matrix API (Google Maps Platform, 2022), with queries executed during a low-traffic period on 24 April 2024 at 1:30 AM. This timing ensured that travel times represented free-flow conditions, providing a reliable baseline for comparison with simulation results.

Figure 1: Case study map of Melbourne Inner SA4



This study employs a two-step approach to calibrate the MATSim network model for the Inner Melbourne area, incorporating both manual tuning and automated calibration methods. The initial step in our methodology involves manually adjusting the free flow speeds to refine the R-squared values compared to actual travel times observed in real-world data. Manual tuning aims to establish a more accurate speed parameters baseline, providing a solid starting point for subsequent simulations. Following this, an automated calibration algorithm is applied to iterate and adapt the speeds to enhance the model's accuracy.

The automated calibration algorithm adopts an iterative approach. For each iteration, the travel time of the OD pairs is first calculated on the network and compared against the data from API. A threshold for the discrepancy $\epsilon \in (0,1)$ is defined. This threshold determines the acceptable deviation between the simulated and actual travel times obtained via the API. For this study, ϵ was set at 0.05. This value was chosen based on preliminary experiments, which indicated that it offers an optimal balance between sensitivity and specificity of the calibration. If the travel time on the network ($t_{network}$) is too fast compared to the travel time on the API t_{API} (i.e., $t_{p,network} < t_{p,API} \cdot (1-\epsilon)$) or is too slow compared to the data from the online API (i.e., $t_{network} > t_{API} \cdot (1+\epsilon)$), then it is likely that some road segments covered by the path of the OD pair on the network are too fast or too slow.

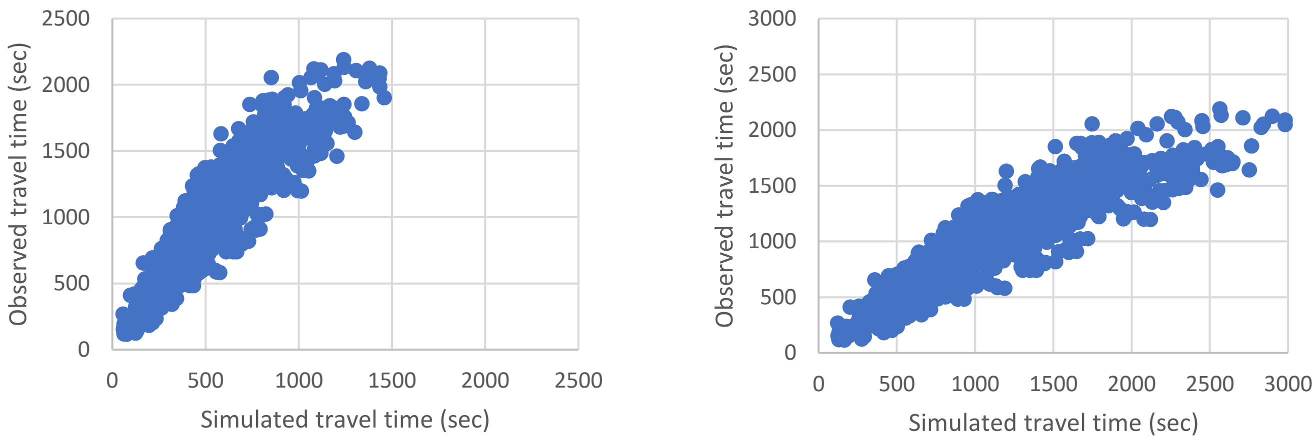
Going through the OD pairs, we can identify the links that are too fast and too slow, respectively. If most of the paths that pass through a specific road segment are too quick on the simulation network, then it is likely that the free flow speed of that road segment is too low, and vice versa. For those segments, we should adjust the free flow speed accordingly. For this, we define another threshold value for accumulated scores $c \in (0,1)$, based on which we assign each road segment l into one of the three sets: F, likely to be too fast; S, likely to be too slow; U, undetermined. The logic is shown in Equation 1, where $n_{l,\,fast}$ and $n_{l,\,slow}$ are the number of paths that cover the road segment l and are too fast or too slow, respectively. $n_{l,total}$ is the total number of paths that cover road segment l .

$$l \in \begin{cases} F, & if \ \dfrac{n_{l,fast} - n_{l,slow}}{n_{l,total}} > c \\ S, & if \ \dfrac{n_{l,fast} - n_{l,slow}}{n_{l,total}} < -c \\ U, & otherwise \end{cases} \tag{1}$$

3 Main findings

The calibration of free flow speeds using real-world data has significantly enhanced the accuracy of the studied network, as evidenced by Figures 2 and 3. Figure 2 shows the transformation in travel time validation through manual tuning. Initially, the baseline network, prepared from OSM data, shows a broad spread of travel times, indicating a divergence from actual traffic conditions, as shown in Figure 2a. Following manual adjustments to the free flow speeds, Figure 2b displays a tighter dispersion and a more linear alignment of simulated travel times, suggesting that the model's predictions are more reflective of the observed data.

Figure 2: Observed vs. Simulated Travel Times (in sec) (a) Baseline network; (b) Manually tuned network

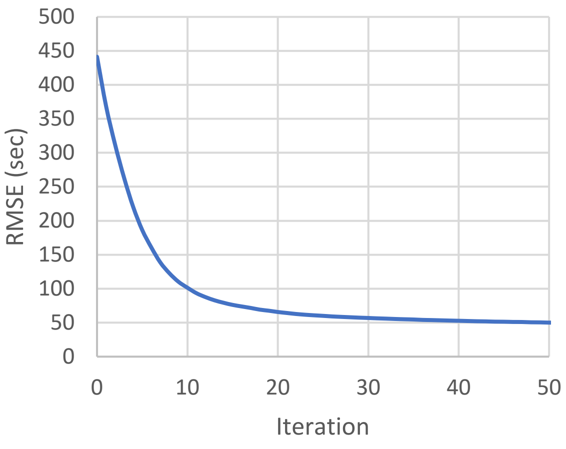


(a) (b)

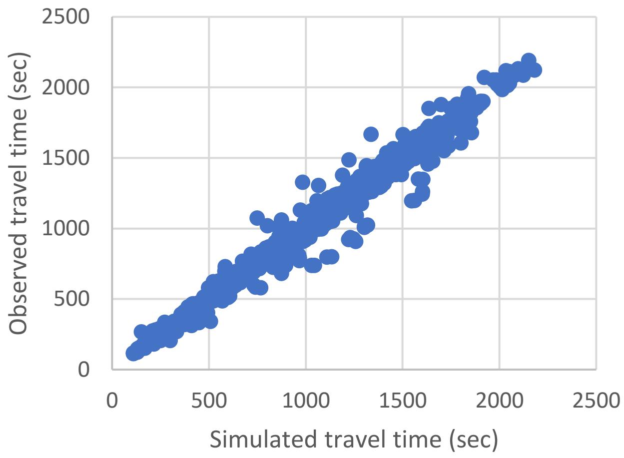
Figure 3a shows the trend of the Root Mean Squared Error (RMSE) values after the automated calibration of the manually calibrated network conducted over 50 iterations. The rapid decline in the RMSE values over calibration iterations demonstrates the precision improvements in travel time predictions. This rapid improvement stabilises, indicating that the model approaches an optimal state where further iterations do not significantly enhance accuracy. After completing the 50 iterations, the calibrated network was used to compare the travel time, as shown in Figure 3b. The final simulated travel times post-calibration demonstrate a tight alignment along a linear path, indicative of a high level of precision in replicating actual travel conditions within the network. The iterative improvement of travel times, evidenced by the reduction in RMSE and the improved alignment of simulated travel times, confirms the potential of the proposed model.

Table 1 summarises the performance of the MATSim travel times under different calibration approaches and data splits, detailing the R-Square, RMSE, Mean Absolute Percentage Error (MAPE), and the corresponding regression equations for each scenario.

Figure 3: (a) MSE values over different iterations; (b) Observed travel time (in sec) vs simulated travel time (in sec) after the automated calibration



(a)



(b)

Table 1: Performance metrics across calibration scenarios for accuracy estimation

Scenario	Calibration Approach	R-Square	RMSE (sec)	MAPE (%)	Regression Equation
100% OD Pairs	**No Calibration**	0.8267	532.6	44.66	y = 1.4842x + 195.33
	Manual Calibration	0.8222	303.18	22.83	y = 0.7093x + 199.16
	Automated Calibration	0.9787	68.34	4.74	y = 0.9877x + 12.605
90%-10% Split	**Manual Calibration**	0.8591	251.21	18.19	y = 0.7382x + 202.7
	Automated Calibration	0.9845	73.17	6.19	y = 0.9774x + 25.245
95%-5% Split	**Manual Calibration**	0.8324	259.89	18.00	y = 0.7144x + 235.45
	Automated Calibration	0.9760	69.83	5.76	y = 0.9812x + 26.309

4 Discussion

Integrating real-world travel data proves crucial for improving traffic models, providing urban planners with more reliable tools for scenario planning. This study presents a novel method for calibrating free-flow speeds in MATSim using real-time travel data from the Google API, significantly improving simulation accuracy. Applied to Inner Melbourne's network, the algorithm reduced RMSE values by 60-70% compared to manual tuning and 90% over the baseline. Similarly, MAPE values decreased from 20% (manually tuned) and 44% (baseline) to 4-5% post-calibration. These results emphasise the effectiveness of the automated calibration process in enhancing model precision.

This approach can be applied to various road networks without extensive data collection, making it practical even in areas needing more detailed traffic information. One of the critical contributions of this method is its ability to account for traffic-regulating factors, such as traffic signals and stop signs, without explicitly modelling each feature. Another benefit of this algorithm is that adjusting free-flow speeds based on real-world data simplifies the model while maintaining accuracy, significantly reducing computational complexity. The method also enhances accessibility, as cities with limited resources can improve their traffic simulations using freely available API data, avoiding costly proprietary systems. This makes the approach scalable and feasible for broader urban applications, allowing for better traffic management, even in resource-constrained settings.

However, this study has limitations. The calibration is sensitive to initial free-flow speed settings; incorrect estimates can constrain model improvements. Additionally, the current linear calibration approach may not fully capture urban traffic complexities, suggesting that future research could explore non-linear models or machine learning techniques for enhanced calibration.

5 References

Caliper. (2008). *TransModeler. Traffic Simulation Software, User's Guide. Version 2.0.*

Contributors, O. (2017). *Planet dump retrieved from https://planet.osm.org*.

Google Maps Platform. (2022). *Google Maps Platform Documentation | Distance Matrix API | Google Developers*. https://developers.google.com/maps/documentation/distance-matrix/

Horni, A., Nagel, K., & Axhausen, K. W. (2016). The Multi-Agent Transport Simulation MATSim. *The Multi-Agent Transport Simulation MATSim*. https://doi.org/10.5334/baw

Ištoka Otković, I., Tollazzi, T., Šraml, M., & Varevac, D. (2023). Calibration of the Microsimulation Traffic Model Using Different Neural Network Applications. *Future Transportation*, *3*(1), 150–168. https://doi.org/10.3390/futuretransp3010010

Jafari, A., Both, A., Singh, D., Gunn, L., & Giles-Corti, B. (2022). Building the road network for city-scale active transport simulation models. *Simulation Modelling Practice and Theory*, *114*, 102398. https://doi.org/10.1016/j.simpat.2021.102398

Lopez, P. A., Behrisch, M., Bieker-Walz, L., Erdmann, J., Flotterod, Y. P., Hilbrich, R., Lucken, L., Rummel, J., Wagner, P., & Wiebner, E. (2018). Microscopic Traffic Simulation using SUMO. *IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC*, *2018-Novem*, 2575–2582. https://doi.org/10.1109/ITSC.2018.8569938

PTV AG. (2015). *Vissim. Traffic Simulation Software, User Manual. Version 8.*

Rakow, C., & Nagel, K. (2024). Road network free flow speed estimation using microscopic simulation and point-to-point travel times. *Procedia Computer Science*, *238*, 698–705. https://doi.org/10.1016/j.procs.2024.06.080

Tiwari, S., Nassir, N., & Lavieri, P. S. (2023). A Hybrid Parallel-Sequential Insertion Heuristics for Shared Autonomous Vehicle Routing: A Melbourne Case Study. *2023 IEEE Engineering Informatics*, 1–9. https://doi.org/10.1109/IEEECONF58110.2023.10520538

Tiwari, S., Nassir, N., & Lavieri, P. S. (2024). Smart Insertion Strategies for Sustainable Operation of Shared Autonomous Vehicles. *Sustainability*, *16*(12), 5175. https://doi.org/10.3390/SU16125175