A Synthetic Ground-Truth Dataset for Online Map Matching

José Vinícius Evangelista Dias de Souza[1]
Regis Pires Magalhães[1]
José Antônio Fernandes de Macêdo[1]

[1]Insight Data Science Lab – Federal University of Ceará (UFC)
Fortaleza – CE – Brazil

{jose.souza,regis,jose.macedo}@insightlab.ufc.br

Abstract. *Reliable evaluation of online map-matching algorithms is hindered by the scarcity of trajectory datasets with reusable, exact road-segment ground truth. This work presents a parameterizable pipeline that generates synthetic workloads by preserving reference routes while independently controlling GNSS-like spatial noise and temporal sampling. Its demonstration case uses the O'Hare road network, OpenStreetMap (OSM), and the Simulation of Urban MObility (SUMO) to generate trajectories, reconcile simulator edges with OSM identifiers, and derive noisy observations at 1, 5, 10, and 20 s intervals. The resulting dataset contains 1,759 trajectories ranging from 519.33 m to 8.97 km. Validation on 10 routes yielded 3 perfect, 3 minor-deviation, and 4 significant-deviation matches; downstream experiments supported 790 valid executions across four map-matching tools. The pipeline and its O'Hare instantiation provide an auditable workload for reproducible online map-matching evaluation.*

Keywords: Online map matching; Synthetic ground truth; GNSS trajectories

1. Introduction

Urban mobility, logistics, navigation, and road-monitoring systems increasingly depend on vehicular traces that can be interpreted at the level of road segments. These traces are usually collected through the Global Navigation Satellite System (GNSS), but raw coordinates are affected by blockage, multipath propagation, Non-Line-Of-Sight (NLOS) reception, and sparse sampling in dense urban environments [de Sousa Pissardini et al. 2017]. *Map matching* addresses this mismatch by associating noisy GNSS observations with a digital road network [Chao et al. 2020]. The problem is central to transport applications because downstream analyses–from travel-time estimation to fleet monitoring–depend on knowing which road segments were actually used [Quddus et al. 2007]. Figure 1 illustrates this correction of noisy observations onto the road network. Evaluation, however, is only meaningful when the correct road sequence is known; real traces rarely expose reusable segment-level *ground truth*, which motivates controlled synthetic reference routes [Erdmann and Ebendt 2014].

The proposed synthetic-data pipeline constructs this reference through microscopic traffic simulation. Starting from the OpenStreetMap (OSM) road network of O'Hare, Chicago, the pipeline extracts the vehicular graph with OSMnx [Boeing 2025], converts it to the Simulation of Urban MObility (SUMO), generates trips with a 500 m

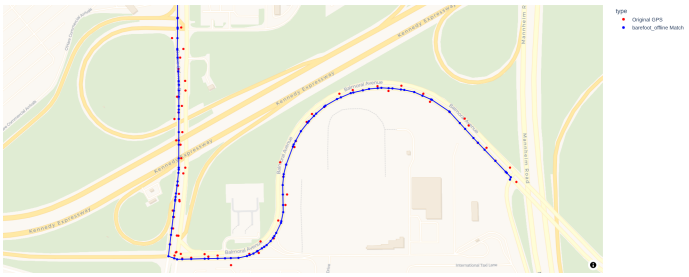


Figure 1. Example of map matching applied to a synthetic trajectory from the O'Hare dataset: noisy GNSS observations (red) are corrected by Barefoot onto the road network (blue), recovering the correct route along Balmoral Avenue and Mannheim Road despite coordinate displacement.

minimum distance, records exact vehicle states every 0.5 s through the Traffic Control Interface (TraCI), reconstructs the correspondence between SUMO edges and OSM identifiers, injects two-dimensional Gaussian noise with $\mu = 0$ and $\sigma = 5$ m, and derives 1 s, 5 s, 10 s, and 20 s sampling variants. The resulting dataset contains 1,759 trajectories with lengths up to 8.97 km.

The primary contribution is a parameterizable and auditable pipeline for constructing online map-matching workloads: a traceable workflow from road-network extraction to OSM-level ground truth, with explicit spatial-noise and temporal-sampling factors. The released O'Hare dataset is its demonstration instantiation, providing validation evidence that the resulting trajectories include both simple and failure-inducing cases. By treating the dataset as an experimental instrument rather than as a by-product of simulation, the proposed workflow makes map-matching evaluation easier to reproduce, inspect, and extend.

2. Related Work

Map-matching evaluation depends on trajectories for which the traveled road sequence can be compared against the route inferred by an algorithm, since segment-overlap metrics quantify agreement between an inferred path and a reference path [Li et al. 2014]. Real traces are valuable because they capture operational conditions, but reusable segment-level answer keys are difficult to obtain at scale, motivating synthetic benchmarks with known reference routes [Erdmann and Ebendt 2014]. Goh *et al.* evaluated online HMM map matching using four unpublished bus trajectories in Singapore [Goh et al. 2012], while Saki and Hagen studied large-scale open-source matching over a private high-frequency dataset from Frankfurt am Main [Saki and Hagen 2022]. These studies demonstrate the practical value of realistic map-matching workloads.

This tension becomes sharper in urban scenarios, where Global Navigation Satellite System (GNSS) observations may be displaced by multipath and Non-Line-Of-Sight (NLOS) reception [Zhang and Hsu 2021].

The closest methodological precedent is the use of SUMO for synthetic map-matching benchmarks. Erdmann and Ebendt [Erdmann and Ebendt 2014] demonstrated that microscopic simulation can provide known reference routes and then degrade them through temporal subsampling and two-dimensional Gaussian noise with a 5 m standard deviation. Guastella *et al.* [Guastella et al. 2025] further support SUMO as a reproducible

environment for traffic modeling, which is relevant here because the dataset construction depends on deterministic simulation, route extraction, and state logging. The pipeline follows this simulation-based line, but shifts the emphasis from using simulation only as an input generator to producing an auditable dataset artifact, with OpenStreetMap (OSM) identifiers preserved from graph extraction to validation.

Controlled datasets must also expose route-level failure modes, because map-matching errors are rarely isolated to a single coordinate. Detours caused by shifted observations and correspondence breaks caused by discontinuous snapping are two representative cases emphasized in the literature [Chao et al. 2020]. These examples motivate why the workload generated here preserves both the clean route and the degraded observations, allowing route continuity to be audited after matching.

The online setting makes dataset design more demanding because the matcher must infer the route incrementally, often with sparse observations and limited future context. Goh *et al.* [Goh et al. 2012] are important for establishing online HMM inference and the relevance of accuracy and response-time metrics. Saki and Hagen [Saki and Hagen 2022] show the engineering value of open-source tools for large GPS workloads. From the perspective of dataset construction, these works indicate that online evaluation needs both operational realism and an auditable reference route, especially when the objective is to compare behavior under controlled sparsity.

Broader benchmarks connect dataset design with comparative evaluation: Rehrl *et al.* [Rehrl et al. 2018] compare GraphiumMM and Barefoot under online and offline conditions using a real trajectory set, whereas Wöltche [Wöltche 2023] provides an extensive open-source benchmark focused on offline map matching. Taken together, prior work establishes the importance of online inference, open-source tools, simulation, and multidimensional metrics. The remaining gap is a dataset-centered contribution: a reusable synthetic workload in which exact routes, degraded GNSS-like observations, sampling rates, and OSM segment identifiers remain traceable from generation to validation.

3. Synthetic Dataset Development Process

Dataset construction is organized around four dependent stages: OSM/SUMO road-network preparation, microscopic simulation, GNSS post-processing, and benchmark validation. The design requirements are traceability, so each record links simulated vehicle state, noisy observation, and OSM road identifier, and controllability, so spatial error and sampling frequency vary without changing the underlying route. Fixed seeds, explicit sampling rates, documented noise assumptions, length strata, and preserved OSM identifiers support transparent computational experimentation [Sandve et al. 2013]. The proposed contribution is a parameterizable pipeline for generating synthetic ground-truth workloads for online map matching; O'Hare and the reported simulation, noise, and sampling settings constitute its demonstration case. The pipeline source code, SUMO simulation configuration, and the base O'Hare road network are publicly available.[1]

The choice of a synthetic dataset was deliberate. Real trajectories are valuable for ecological realism, but they rarely expose the exact sequence of road segments required for quantitative accuracy metrics. In contrast, microscopic simulation provides a known

[1] `https://github.com/JoseEdSouza/map-matching`

route before any sensor degradation is introduced. This makes the dataset closer to an experimental instrument: the clean route is the reference answer, while noise and sparsity define controlled stress conditions for the matcher.

3.1. Road Network Extraction and Preparation

The road network was selected as an experimental stress condition rather than as a neutral geographic backdrop. A suitable map-matching workload must combine dense and topologically complex areas with longer continuous stretches, because both nearby candidate roads and sparse observations can challenge route inference [Chao et al. 2020]. Community Area 76 of Chicago, Illinois, USA, corresponding to O'Hare, satisfies these requirements through its highway interchanges, ramps, local streets, arterial roads, and airport-adjacent connectors.

The region's structural diversity is important for the dataset objective. Multi-level roads, high-speed segments, ramps, and sharp turns create cases in which two candidate paths may be geometrically close but topologically different. Such cases expose whether a matcher merely snaps points to nearby roads or preserves a plausible connected path, which is central when the dataset is intended to evaluate road-segment agreement rather than only coordinate proximity.

The physical graph was extracted from the OpenStreetMap database using the OSMnx Python library [Boeing 2025], with the query "O'Hare, Chicago, Illinois, USA". A strict vehicular *drive* network filter eliminated pedestrian paths, cycleways, and private service roads to ensure the graph represented strictly navigable space for automobiles. The raw road network was retained in XML and GraphML form so that both the simulation input and the identifier-mapping process could be audited later.

Before conversion, non-relevant OSM entities were filtered so that the SUMO network retained only the *highway* elements required for vehicular movement. The SUMO-compatible network was then generated with conversion parameters aimed at preserving the original topology, limiting excessive automatic junction creation, and maintaining the correspondence between simulator edge identifiers and OSM road identifiers. This preparation matters because later evaluation compares inferred routes against OSM-level ground truth; if the conversion stage collapses junctions or loses identifiers, the benchmark can no longer distinguish algorithmic error from preprocessing ambiguity. At this stage, the artifact is a road-network graph, not yet a trajectory dataset. Point observations are generated later during TraCI simulation, where each vehicle state is sampled every 0.5 s. Figure 2 provides the corresponding SUMO network view used as the simulation substrate.

3.2. Microscopic Traffic Simulation and State Tracking

Synthetic traffic was generated by sampling random origin–destination pairs and computing feasible shortest paths. The simulation used random seed 42, a one-hour window, vehicle insertion every two seconds, a non-uniform binomial demand process, and a minimum topological travel distance of 500 m to avoid trivial routes. Routes were computed using Dijkstra's shortest path algorithm [Dijkstra 1959], which is appropriate here because the dataset requires connected, reproducible, and topologically valid reference paths rather than behaviorally calibrated driver choices.

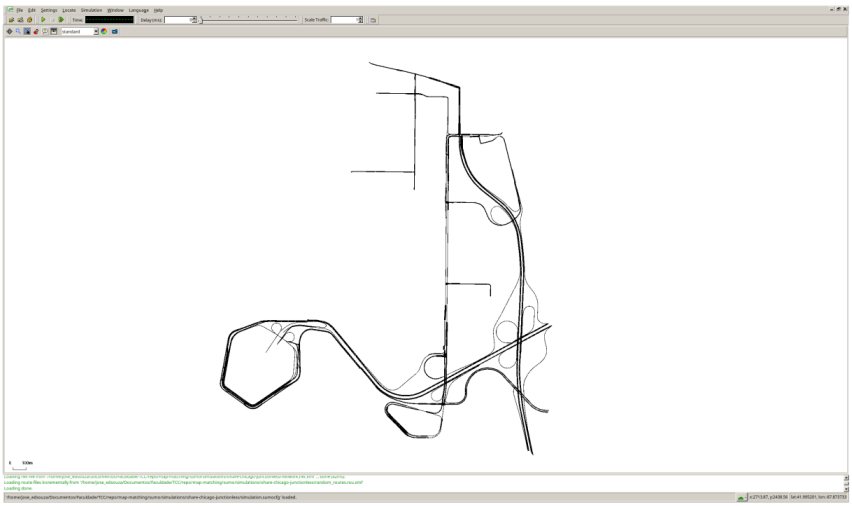


Figure 2. The O'Hare road network loaded into the SUMO microscopic traffic simulator, serving as the baseline geometry for ground truth extraction.

SUMO advanced in 0.5 s steps, and the TraCI interface [Guastella et al. 2025] recorded each active vehicle's identifier, timestamp, exact 2D coordinate (ϕ, λ), and occupied SUMO edge/lane. These records form the uncorrupted reference trajectory. This clean trace is retained as the common reference route for all subsequent noise and sampling variants. The simulation completed 1,759 distinct trips.

3.3. Topological Reconciliation

TraCI records SUMO edge identifiers from the simulator's microscopic road-network representation [Alvarez Lopez et al. 2018]. However, open-source map-matching engines, such as Barefoot, GraphHopper, and OSRM, operate on standard OSM node and way identifiers. To bridge this semantic gap, a topological reconciliation procedure reconstructed the correspondence between internal SUMO edges and OSM ways using metadata preserved during network conversion. This step is central to the dataset because it converts simulator truth into the same identifier space used by the evaluated engines.

The main difficulty is that simulator-level junction links and connector segments may not exist as explicit OSM ways. Such links are internal lanes of joined junctions, for which the network converter preserves the original OSM entry and exit nodes as lane parameters; the reconciliation procedure recovers this node pair and uses it as the lookup key into the OSM edge index, so the association is determined by provenance metadata rather than by geometric similarity. Cases in which only the reverse-oriented edge exists correspond to simplified U-turn maneuvers and are deferred to the topological repair step. If a discontinuity appeared between consecutive mapped segments, the procedure used shortest-path repair over the OSM graph to close the gap while preserving the simulated route order. In the network used here, no internal lane exhibited multiple incoming or outgoing connections, making the mapping deterministic.

This reconciliation is more than a formatting step. Without it, the simulator would provide exact positions and internal edge names, but the benchmark would still be unable to compare inferred routes against a common reference. The procedure therefore treats the OSM identifier as the stable comparison unit and uses the SUMO edge sequence as the

high-resolution movement trace. The result is an auditable edge sequence that keeps local junction detail during simulation while exposing the road identifiers needed for segment-level evaluation.

3.4. Procedural Noise Injection and Sparsity Simulation

Post-processing turns the clean simulator trace into a benchmark workload by degrading only the observations, never the reference route. Noisy GNSS telemetry is emulated by adding 2D Gaussian perturbations to exact coordinates, following SUMO-based validation literature [Erdmann and Ebendt 2014]. In the generated trace, each exact simulated coordinate $p_i = (\phi_i, \lambda_i)$ at timestamp t_i is transformed into the noisy observation $\tilde{p}_i$:

$$\tilde{p}_i = p_i + \epsilon_i, \quad \epsilon_i \sim \mathcal{N}(0, \sigma^2 I_2), \quad \sigma = 5m. \tag{1}$$

This formulation makes the spatial-error assumption explicit: each coordinate receives an independent two-dimensional perturbation whose variance is controlled by σ. With $\sigma = 5$ m, most points remain close to the simulated path while some are displaced enough to challenge the matcher near ramps, junctions, and parallel roads. Because the clean SUMO position and the reconciled OSM edge sequence are retained, the noisy point can be treated as the matcher input while the original route remains the answer key.

The second transformation emulates sparse online tracking, a central stress factor for map matching under noise and low-frequency telemetry [Newson and Krumm 2009]. Starting from the continuous 0.5 s traces, the pipeline derives four fixed-interval variants by retaining only the noisy observations aligned with each target sampling rate sr:

$$T_{sr} = \{\tilde{p}_i \in T \mid (t_i - t_0) \bmod sr = 0\}, \quad sr \in \{1, 5, 10, 20\}s. \tag{2}$$

As the sampling interval increases from 1 to 20 s, the distance between consecutive points grows, reducing the geometric context available to the map-matching algorithm and testing the predictive power of its transition model. The transformation preserves the temporal order and timestamp semantics of the original trace, so each sparsity tier remains a valid online stream rather than an unordered subset of points. Both degradation steps are applied after the route has been generated and reconciled with OSM identifiers. This ordering is intentional: all noisy and sparse variants of a trajectory inherit the same reference path, so changes in matching quality can be attributed to spatial perturbation or temporal sparsity rather than to changes in the underlying route.

The final Parquet dataset stores ground-truth edge annotations and noisy GNSS coordinates across sparsity tiers. Before release, records were inspected for missing values, invalid coordinates, temporal inconsistencies, and topology errors; detected issues triggered parameter adjustment and reruns within the workflow in Figure 3. The artifact schema in Table 1 makes explicit which fields support grouping, temporal ordering, spatial reference, matcher input, and segment-level comparison. This quality-control stage is important for reproducibility because small inconsistencies can become large evaluation errors. A missing timestamp can distort temporal sparsity, an invalid coordinate can bias spatial metrics, and a broken edge sequence can make a correct matcher appear

wrong. For this reason, the final artifact keeps clean trajectory state, degraded observations, sampling regime, and reference identifiers together, allowing each matched output to be audited against the exact route that generated it.

Table 1. Logical schema of the generated dataset artifact.

Component	Stored information	Evaluation role
Trajectory identity	Vehicle and trajectory identifiers	Groups records by trip.
Temporal state	Timestamp at 0.5 s resolution	Supports ordering and subsampling.
Exact position	Noise-free SUMO coordinate	Defines the spatial reference.
Road reference	Reconciled OSM edge or way ids	Enables segment-level comparison.
Noisy observation	GNSS-like perturbed coordinate	Represents matcher input.
Sampling variant	Target interval $sr \in \{1, 5, 10, 20\}$ s	Controls temporal sparsity.

The intermediate artifacts and validation loops in Figure 3 connect simulated road-network traces to the finalized benchmark dataset.

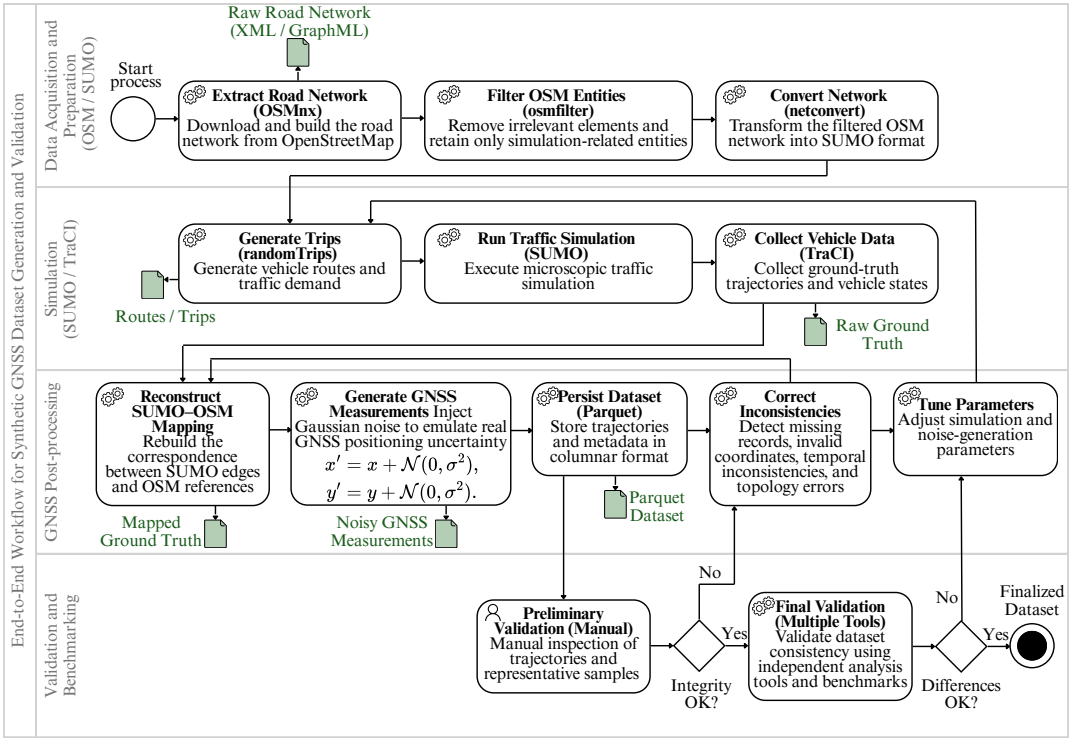


Figure 3. End-to-end workflow for synthetic GNSS dataset generation and validation, from OSM road-network extraction and SUMO simulation to SUMO–OSM mapping, noisy GNSS generation, Parquet persistence, quality control, and final benchmark validation.

4. Dataset Characterization and Validation

The generated dataset contains 1,759 vehicular trajectories ranging from 519.33 m to 8.97 km, combining short local trips and longer arterial movements. Each trajectory is available under four telemetry regimes (1, 5, 10, and 20 s), enabling controlled analysis of temporal sparsity without changing the underlying route. The full artifact preserves all

generated trajectories, whereas representative subsets support computationally bounded benchmark campaigns. Both uses rely on route-overlap metrics enabled by the stored matcher inputs and reference OSM route [Li et al. 2014]. If P_m is the route inferred by a map matcher, P_g is the ground-truth route, and $L(\cdot)$ is the total length of a road-segment sequence, precision, recall, and route-level F1-score are computed as follows:

$$F_1 = \frac{2 \cdot Prec \cdot Rec}{Prec + Rec}, \quad Prec = \frac{L(P_m \cap P_g)}{L(P_m)}, \quad Rec = \frac{L(P_m \cap P_g)}{L(P_g)}. \tag{3}$$

The shared segment length $L(P_m \cap P_g)$ is therefore the bridge between the dataset and quantitative evaluation. This is also why preserving OSM identifiers is central: the overlap can only be computed consistently when the matcher output and the reference route use the same road-segment vocabulary.

A complementary error metric proposed by Newson and Krumm [Newson and Krumm 2009] decomposes route disagreement into two directional components. Let d_- be the length of ground-truth road not covered by the inferred route, and d_+ be the length of inferred road not present in the ground truth:

$$d_- = L(P_g) - L(P_m \cap P_g), \qquad d_+ = L(P_m) - L(P_m \cap P_g). \tag{4}$$

The Newson–Krum Error (NKError) normalises their sum by the ground-truth route length:

$$\text{NKError} = \frac{d_- + d_+}{L(P_g)}. \tag{5}$$

A value of zero indicates a perfect match; values above one occur when the inferred route is substantially longer than the reference. NKError and F1-score capture complementary aspects of matching quality: F1-score rewards segment overlap symmetrically, whereas NKError penalises both missed and spurious road coverage relative to the reference length, making it sensitive to unnecessary detours and correspondence breaks.

The route-length strata in Table 2 indicate that the workload is not concentrated in a single trip profile, which matters because online map matching behaves differently on short, medium, and ramp-heavy longer routes.

The quartile distribution also supports computationally bounded experimentation. The full artifact contains all 1,759 trajectories, while downstream campaigns may select balanced subsets from each length stratum to reduce computational cost. This design keeps the workload scalable: small experiments can remain representative of short, medium, and long trips, while larger experiments can consume the complete artifact. In the benchmark campaign derived from this dataset, the subset used five trajectories per quartile, preserving route-length variation without requiring every generated trip to be processed by every tool.

Validation combined two complementary checks of structural integrity and practical utility. The first check was preliminary and focused on topological integrity: a sample

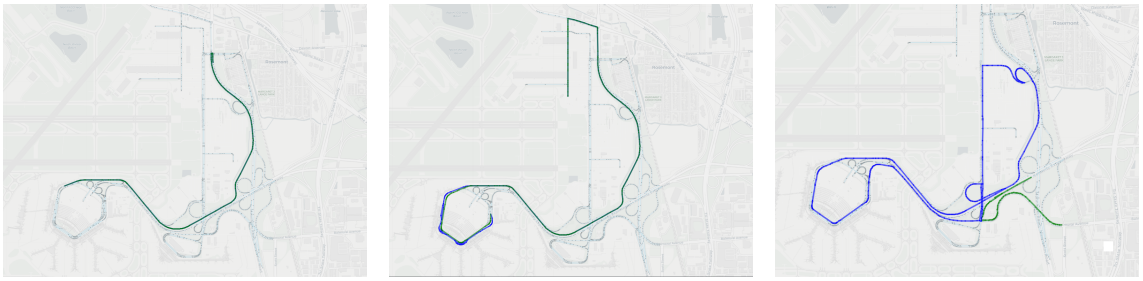
Table 2. Route-length distribution of the generated synthetic trajectories.

Stratum	Length interval (m)	Trajectories
Q1	519.33–2,239.33	440
Q2	2,241.52–3,231.63	440
Q3	3,232.84–4,636.51	439
Q4	4,638.27–8,973.01	440

of 10 trajectories was processed with GraphHopper and visually compared against the exact *ground truth*. The inspection looked for visible gaps, discontinuities, implausible detours, and inconsistencies between the simulated route and the inferred path. This preliminary step served only to verify dataset integrity and SUMO–OSM reconciliation, not to independently assess GraphHopper accuracy. The second check was conducted after the evaluated systems could consume the workload, focusing on whether the same trajectories exposed distinct matching behaviors under shared experimental conditions. The first validation sample appears in Table 3.

Table 3. Qualitative validation of map matching results on a random sample of synthetic trajectories.

Classification	Quantity	Description
Perfect Match	3	The recovered trajectory was topologically identical to the ground truth.
Minimal Deviations	3	The main route was correctly identified, suffering only minor snapping errors at complex junctions.
Significant Deviations	4	The inferred route exhibited unnecessary detours, illogical U-turns, or complete correspondence breaks due to noise.



(a) Perfect match. **(b) Minimal deviations.** **(c) Significant deviations.**

Figure 4. Representative validation outcomes from the 10-route qualitative inspection using GraphHopper: a topologically identical match, a route with minor snapping errors at junctions, and a route with significant deviations including detours or correspondence breaks.

The 40% rate of significant deviations is useful for benchmarking: it shows that the workload contains both well-conditioned cases and challenging cases near junctions, nearby roads, and multi-level structures. Figure 4 illustrates one trajectory from each outcome category, showing how the inferred path ranges from topologically identical to the ground truth (a) to a route with minor snapping errors at complex junctions (b) and

one exhibiting detours and correspondence breaks (c). In this context, deviations are not treated as dataset defects by default; they indicate that the generated observations can stress map-matching decisions while the preserved reference route remains available for auditing.

The second validation step tested whether the dataset produces meaningful topological differences across tools and sampling regimes. A stratified subset of 20 trajectories, five per quartile, was submitted to four map-matching systems. An execution denotes one successful application of a tool to a trajectory at a given sampling rate; Table 4 reports the mean and sample standard deviation across the valid executions contributing to each tool and rate result ($n = 20$ for Barefoot and approximately $n = 60$ for the other systems).

Table 4. Map-matching quality by tool and sampling rate. Values are mean $\pm$ sample standard deviation across valid executions contributing to each tool and rate result ($n = 20$ for Barefoot and $n \approx 60$ for the other systems). Variance is the square of the reported standard deviation.

Tool	Rate (s)	F1-score	NKError
Barefoot	1	0.962 ± 0.048	0.078 ± 0.098
	5	$\mathbf{0.989 \pm 0.014}$	$\mathbf{0.023 \pm 0.028}$
	10	0.985 ± 0.017	0.029 ± 0.032
	20	0.976 ± 0.022	0.045 ± 0.042
GraphiumMM	1	0.931 ± 0.075	0.139 ± 0.154
	5	0.937 ± 0.065	0.121 ± 0.128
	10	0.963 ± 0.045	0.069 ± 0.080
	20	0.957 ± 0.048	0.080 ± 0.085
OSRM	1	0.946 ± 0.049	0.114 ± 0.108
	5	0.937 ± 0.053	0.120 ± 0.101
	10	0.863 ± 0.094	0.233 ± 0.129
	20	0.819 ± 0.119	0.294 ± 0.154
GraphHopper	1	0.786 ± 0.154	0.489 ± 0.347
	5	0.864 ± 0.182	0.284 ± 0.379
	10	0.838 ± 0.173	0.314 ± 0.352
	20	0.772 ± 0.190	0.434 ± 0.375

The results in Table 4 show that the dataset distinguishes matching behavior across tools and sampling regimes. Barefoot reaches its best mean performance at 5 s (F1 = 0.989 ± 0.014, NKError = 0.023 ± 0.028), whereas OSRM and GraphHopper exhibit higher NKError at longer intervals; GraphHopper also has the largest dispersion. This variation confirms that the controlled combination of road topology, Gaussian noise, and temporal sparsity can discriminate between map-matching strategies.

Together, the two steps support the claim that the dataset is not only structurally consistent but also capable of exposing topologically simple, ambiguous, and failure-inducing cases. All systems received trajectories derived from the same reference routes, sampling intervals, and noise assumptions, allowing their results to be compared under shared workload conditions. The reported metrics are accuracy-oriented: they quantify route agreement under temporally ordered sparse streams, but they do not measure response time or end-to-end latency. In the downstream benchmark campaign, this

same workload supported 790 valid executions out of 800 planned runs across Barefoot, GraphiumMM, OSRM, and GraphHopper.

5. Discussion

The dataset should be interpreted as a controlled experimental instrument rather than as a substitute for all real-world telemetry. Its main strength is the explicit separation of road-network topology, spatial noise, and temporal sparsity: SUMO provides the exact route, OSM/SUMO reconciliation places the reference in the identifier space used by map-matching engines, and post-processing degrades only the observations. The validation results indicate that this design yields a workload that is traceable, consumable by different systems, and capable of producing meaningful differences under controlled conditions. Because temporal order is preserved and each trajectory is exposed as a sparse stream, the artifact supports online-style evaluation of incremental input availability; latency and response-time measurement remain future uses of the artifact. At the same time, the scope remains intentionally bounded to O'Hare and to a first-order Gaussian error model, making the dataset best suited for reproducible comparative evaluation and straightforward future extensions to other cities, network morphologies, sensor-noise assumptions, and explicit latency benchmarking.

6. Conclusion

The primary contribution is a reproducible, parameterizable pipeline for generating synthetic online map-matching data with exact road-segment *ground truth*; the released O'Hare dataset is its demonstration instantiation. By integrating OSMnx, SUMO, TraCI, OSM identifier reconciliation, Gaussian noise injection, and temporal subsampling, the pipeline produced 1,759 trajectories with lengths from 519.33 m to 8,973.01 m. The dataset records high-resolution simulated states, GNSS-like noisy observations, and sampling variants of 1, 5, 10, and 20 s, creating a controlled workload for evaluating how online matchers behave under increasing spatial and temporal uncertainty. The dataset and generation scripts are publicly available at `https://github.com/JoseEdSouza/map-matching/tree/main/dataset`.

The validation evidence indicates that the artifact is structurally consistent and experimentally usable. A 10-route inspection produced 3 perfect matches, 3 minor-deviation matches, and 4 significant-deviation matches, while a downstream benchmark campaign planned 800 executions and retained 790 valid runs across four map-matching systems. These results support the dataset's role as a reference workload for reproducible geospatial experimentation. Future work should combine this controlled setting with real trajectories and extend the evaluation to broader urban regions and additional sensor-error models.

References

Alvarez Lopez, P., Behrisch, M., Bieker-Walz, L., Erdmann, J., Flötteröd, Y.-P., Hilbrich, R., Lücken, L., Rummel, J., Wagner, P., and Wießner, E. (2018). Microscopic traffic simulation using sumo. In *Proceedings[...]*. IEEE International Conference on Intelligent Transportation Systems, IEEE.

Boeing, G. (2025). Modeling and Analyzing Urban Networks and Amenities With OSMnx. *Geographical Analysis*, 57(4):567–577.

Chao, P., Xu, Y., Hua, W., and Zhou, X. (2020). A Survey on Map-Matching Algorithms.

de Sousa Pissardini, R., de Oliveira, R. H., Vaz, J. A., de Almeida Filho, F. G. V., and da Fonseca Junior, E. S. (2017). O problema do posicionamento para transporte terrestre no ambiente urbano. *Revista Brasileira de Geomática*, 5(3):380–403.

Dijkstra, E. W. (1959). A note on two problems in connexion with graphs. *Numerische Mathematik*, 1(1):269–271.

Erdmann, J. and Ebendt, R. (2014). Benchmarking Trajectory Matchers with SUMO. In *Proceedings[...]*, pages 226–230, Porto, Portugal. European Simulation and Modelling Conference, Eurosis-ETI.

Goh, C., Dauwels, J., Mitrovic, N., Asif, M. T., Oran, A., and Jaillet, P. (2012). Online map-matching based on Hidden Markov model for real-time traffic sensing applications. In *Proceedings [...]*, pages 776–781. International IEEE Conference on Intelligent Transportation Systems.

Guastella, D. A., Montero-Porras, E., Morales-Hernández, A., and Bontempi, G. (2025). Traffic Modeling with SUMO: a Tutorial.

Li, H., Kulik, L., and Ramamohanarao, K. (2014). Spatio-temporal trajectory simplification for inferring travel paths. In *Proceedings[...]*, SIGSPATIAL '14, pages 63–72, New York, NY, USA. ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, Association for Computing Machinery.

Newson, P. and Krumm, J. (2009). Hidden Markov map matching through noise and sparseness. In *Proceedings [...]*, pages 336–343, New York, NY, USA. ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, Association for Computing Machinery.

Quddus, M. A., Ochieng, W. Y., and Noland, R. B. (2007). Current map-matching algorithms for transport applications: State-of-the art and future research directions. *Transportation Research Part C: Emerging Technologies*, 15(5):312–328.

Rehrl, K., Gröchenig, S., and Wimmer, M. (2018). Optimization and Evaluation of a High-Performance Open-Source Map-Matching Implementation. In Mansourian, A., Pilesjö, P., Harrie, L., and van Lammeren, R., editors, *Geospatial Technologies for All*, pages 251–270. Springer International Publishing, Cham.

Saki, S. and Hagen, T. (2022). A Practical Guide to an Open-Source Map-Matching Approach for Big GPS Data. *SN Computer Science*, 3(5):415.

Sandve, G. K., Nekrutenko, A., Taylor, J., and Hovig, E. (2013). Ten Simple Rules for Reproducible Computational Research. *PLOS Computational Biology*, 9(10):1–4.

Wöltche, A. (2023). Open source map matching with markov decision processes: A new method and a detailed benchmark with existing approaches. *Transactions in GIS*, 27(7):1959–1991.

Zhang, G. and Hsu, L.-T. (2021). Performance assessment of GNSS diffraction models in urban areas. *NAVIGATION*, 68(2):369–389.