The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume XLIX-B4-2026
XXV ISPRS Congress 2026 “From Imagery to Understanding”, Commission IV, 4–11 July 2026, Toronto, Canada

Topological Analysis of OpenDRIVE Models for Advanced Autonomous Vehicle Simulations

János Máté Lógó[1], Viktor Győző Horváth[1], Vivien Potó[1], Árpád Barsi[1]

[1] BME, Department of Photogrammetry and Geoinformatics, H-1111 Budapest, Hungary
(logo.janos.mate, poto.vivien, horvath.viktor.gyozo, barsi.arpad)@emk.bme.hu

Keywords: OpenDRIVE, Topological analysis, Autonomous vehicle simulation, Road network modeling, High-Definition maps (HD maps)

Abstract

Simulation-based testing is essential for autonomous vehicle development and depends on high-fidelity digital road network models. Although prior research focuses on geometric accuracy and semantic completeness of HD maps, topology remains underexplored. Topological inconsistencies, such as disconnected lane segments, invalid predecessor-successor links, or incomplete junction definitions, can invalidate simulations by preventing realistic vehicle navigation and traffic flow.
This study presents a formal topological analysis framework for OpenDRIVE models, representing road networks as directed graphs and applying graph-based methods to analyze connectivity and verify consistency. We define connectivity relationships at three hierarchical levels: road-level predecessor-successor relations, lane-level adjacency, and junction-level merging-diverging structures. The framework introduces explicit consistency predicates for topological validity and enables algorithmic verification through adjacency matrix analysis, reachability computation, and connected component decomposition.
A Python-based verification system processes OpenDRIVE XML files and evaluates consistency predicates across the network hierarchy. The approach is validated on four datasets: two synthetic scenarios and two real-world maps from Budapest and Karlsruhe. Results reveal widespread topological defects, with error-to-road ratios between 2.68 and 3.94. Lane-level connectivity violations account for 56-61% of errors in production datasets, while the professionally generated Karlsruhe dataset fragments into 467 isolated components despite containing 886 lanes and centimeter-level geometric accuracy.
These findings show that topological verification is essential for HD map quality assurance revealing critical structural defects invisible to conventional geometric validation and simulation-based autonomous vehicle testing.

1. Introduction

Simulation-based testing has become essential for autonomous vehicle (AV) development, enabling reproducible validation of perception, localization, and decision-making systems under controlled conditions (Poggenhans et al., 2018). The effectiveness of virtual testing environments depends critically on the structural integrity of digital road representations. While considerable research has focused on geometric accuracy and semantic completeness of HD maps (Liu et al., 2020, Soorchaei et al., 2022, Elghazaly et al., 2023, Křehlík et al., 2023), the topological dimension of OpenDRIVE-based road networks remains underexplored.

Topology captures the logical connectivity and structural relationships between network elements: which lanes connect to which, how traffic flows through junctions, and whether the network forms a consistent, traversable whole. Topological inconsistencies such as disconnected lane segments, circular dependency chains, or incomplete junction definitions can lead to unrealistic vehicle behaviors and invalid simulation results (Lógó and Barsi, 2023). Despite its critical importance, topological validation is often treated as secondary to geometric validation, yet real-world OpenDRIVE datasets frequently contain subtle topological errors that severely compromise simulation fidelity.

This study introduces a formal topological analysis framework for OpenDRIVE models. We interpret road networks as directed graphs and apply mathematical techniques to analyze connectivity, detect structural anomalies, and verify consistency. The framework operates at three hierarchical levels: road connectivity through predecessor-successor relationships, lane adjacency within sections and across boundaries, and junction merging-diverging patterns. We validate the approach using both synthetic test cases and real-world datasets from Budapest (Hungary), Karlsruhe (Germany), demonstrating that topological verification identifies significant structural defects that would otherwise invalidate simulation outcomes.

2. Methodology

2.1 Topology in OpenDRIVE

The ASAM OpenDRIVE specification defines road networks through a hierarchical data model capturing geometric, semantic, and structural information (ASAM e.V., 2026). Topologically, OpenDRIVE provides explicit mechanisms for describing connectivity at multiple abstraction levels.

At the road level, connectivity is established through predecessor and successor links encoded in the `<link>` element. Roads may reference other roads directly or indirectly through junction mediators. Within each road, lane-level topology is organized into lane sections-longitudinal segments where lane configuration remains constant. Lanes are numbered relative to a center line (positive left, negative right), and each lane specifies predecessor and successor links to lanes in adjacent sections or roads, enabling representation of merges, splits, and continuations.

Junction entities represent topologically complex intersection areas through connection objects. Each connection describes a directed path from an incoming lane to an outgoing lane via a connecting road, decoupling geometric complexity from topological structure. This mechanism clearly specifies permitted traffic movements independent of underlying geometry.

Despite these explicit mechanisms, OpenDRIVE's topological treatment exhibits characteristics that complicate automated verification. First, the specification permits multiple valid representations of identical connectivity patterns, introducing interpretation ambiguity. Second, hierarchical structure creates implicit cross-level dependencies: road connectivity must align with lane links, and junction definitions must match constituent road relationships. These dependencies are not always validated by standard parsers. Third, OpenDRIVE's model is declarative rather than constructive-it describes intended connectivity through attributes but provides no procedural validation rules. Consequently, semantic inconsistencies such as non-reciprocal predecessor-successor links or incomplete junction connection sets can arise without violating XML schema constraints.

In simulation environments, these topological ambiguities have direct operational consequences. Autonomous vehicle simulators construct internal graph representations from OpenDRIVE data for path planning and traffic flow simulation. Topological errors can produce disconnected graph components preventing valid routing, circular dependencies causing algorithmic failures, or inconsistent junction configurations leading to unrealistic traffic conflicts. Current validation practices emphasize geometric conformance and semantic completeness but provide only basic topological checks-insufficient to guarantee network-level coherence, particularly for complex junctions or irregular connectivity patterns.

2.2 Formal Topological Definition

To establish a rigorous foundation for topological analysis of OpenDRIVE models, we introduce a formal grammar that captures the essential connectivity relationships and structural constraints of road networks. This formalism provides a mathematical language for expressing topological properties, enabling both symbolic reasoning about network structure and algorithmic implementation of consistency verification procedures.

Let us define a road network N as a tuple $N = (R, L, J, E_R, E_L)$, where:

- $\mathrm{R} = \{r_1, r_2, \ldots, r_n\}$ is a finite set of road entities
- $\mathrm{L} = \{l_1, l_2, \ldots, l_m\}$ is a finite set of lane entities
- $\mathrm{J} = \{j_1, j_2, \ldots, j_k\}$ is a finite set of junction entities
- $\mathrm{E_R} \subseteq R \times R$ is the road connectivity relation
- $\mathrm{E_L} \subseteq L \times L$ is the lane connectivity relation

Each road $r \in R$ possesses a unique identifier $id(r)$ and is associated with a set of lane sections $S(r) = \{s_1, s_2, \ldots, s_p\}$. Each lane section s contains a set of lanes $L(s) \subseteq L$, numbered according to the OpenDRIVE convention with indices $i \in \mathbb{Z} \setminus \{0\}$. We denote the i-th lane in section s as $l_{s,i}$.

The road connectivity relation E_R is derived from the explicit predecessor-successor links in the OpenDRIVE model. For any road r, we define

- $\mathrm{pred(r)} \subseteq R$ as the set of predecessor roads
- $\mathrm{succ(r)} \subseteq R$ as the set of successor roads

The relation E_R is then defined as $E_R = \{(r, r') \mid r' \in succ(r)\}$. A fundamental consistency requirement is that predecessor-successor relationships should be reciprocal, formalized as:

$$\forall r, r' \in R : (r, r') \in E_R \Rightarrow (r', r) \in E_R^{-1}, \tag{1}$$

where E_R^{-1} denotes the inverse relation. Violations of this property indicate topological inconsistencies in the road-level connectivity graph.

At the lane level, connectivity is more intricate due to the possibility of lane merges, splits, and reconfigurations across section boundaries. For each lane $l_{s,i}$ in section s of road r, the OpenDRIVE model specifies:

- $\mathrm{lpred}(l_{s,i}) = \{l'_1, l'_2, \ldots\} \subseteq L$: the set of predecessor lanes
- $\mathrm{lsucc}(l_{s,i}) = \{l''_1, l''_2, \ldots\} \subseteq L$: the set of successor lanes

The lane connectivity relation E_L is constructed as:

$$E_L = \{(l, l') \mid l' \in \mathrm{lsucc}(l)\}. \tag{2}$$

For lane-level topological consistency, we require:

$$\forall l, l' \in L : (l, l') \in E_L \Rightarrow l \in \mathrm{lpred}(l'). \tag{3}$$

Additionally, lane connectivity must be compatible with road connectivity. Specifically, if lane l on road r connects to lane l' on road r', then either $r' \in succ(r)$, or both r and r' are connected through a junction $j \in J$.

Junction entities introduce a mediated connectivity pattern. A junction j is characterized by a set of connections $C(j) = \{c_1, c_2, \ldots, c_q\}$, where each connection c specifies:

- $\mathrm{in}(c) = (r_{in}, l_{in})$: the incoming road and lane
- $\mathrm{out}(c) = (r_{out}, l_{out})$: the outgoing road and lane
- $\mathrm{path}(c) = r_c$: the connecting road within the junction

The junction topology induces additional constraints on the network graph. For every connection $c \in C(j)$, we require:

1. $r_{in} \in \mathrm{pred}_j(j)$: the incoming road must reference j as its successor
2. $r_{out} \in \mathrm{succ}_j(j)$: the outgoing road must reference j as its predecessor
3. The lane l_{in} on r_{in} must have a successor link to the first lane of r_c
4. The last lane of r_c must have a successor link to l_{out} on r_{out}

To formalize the overall network connectivity, we define the network graph $G(N) = (V, E)$ as a directed graph where vertices represent lanes and edges represent navigable connections. Specifically:

- $V = L$
- $E = E_L$

A critical topological property is strong connectivity, which requires that for any two lanes $l, l' \in L$, there exists a directed path from l to l' in $G(N)$. While realistic road networks are not necessarily strongly connected due to one-way streets and access restrictions, we can partition L into weakly connected components and verify connectivity within each component.

We introduce the concept of topological closure to capture the requirement that all declared connections must be internally consistent. For a road r with lane sections $S(r) =$

$\{s_1, s_2, \ldots, s_p\}$, the internal lane topology must satisfy:

$$\forall i \in \{1, \ldots, p-1\}, \forall l \in L(s_i) : \\ \text{lsucc}(l) \cap L(s_{i+1}) \neq \emptyset \\ \vee\ \text{lsucc}(l) \cap \left(L \setminus \bigcup_{s \in S(r)} L(s) \right) \neq \emptyset \tag{4}$$

This condition ensures that lanes either continue to the next section within the same road or exit to a different road; lanes should not terminate arbitrarily within a road's interior.

To detect topological anomalies, we define several consistency predicates (road, lane and junction levels, respectively):

- RC_1: $\forall r \in R, \forall r' \in \text{succ}(r) : r \in \text{pred}(r')$
- RC_2: $\forall r \in R : |\text{pred}(r)| \leq K_p \wedge |\text{succ}(r)| \leq K_s$ (bounded fan-in/fan-out)
- LC_1: $\forall l \in L, \forall l' \in \text{lsucc}(l) : l \in \text{lpred}(l')$
- LC_2: $\forall l \in L : |\text{lsucc}(l)| \geq 1 \vee$ terminal (l) (non-terminal lanes have successors)
- LC_3: $\forall l \in L, \forall l' \in \text{lsucc}(l) : \text{road}(l) \in \text{pred}(\text{road}(l')) \vee \exists j \in J : \text{connects}(j, l, l')$
- JC_1: $\forall j \in J, \forall c \in C(j) : j \in \text{succ}(\text{in}(c).\text{road}) \wedge j \in \text{pred}(\text{out}(c).\text{road})$
- JC_2: $\forall j \in J, \forall r \in \text{pred}_j(j) : \exists c \in C(j) : \text{in}(c).\text{road} = r$ (all junction approaches have connections)

A network N is topologically valid if and only if all consistency predicates $RC_1, RC_2, LC_1, LC_2, LC_3, JC_1$, and JC_2 are satisfied.

Furthermore, we can extend this formalism to capture higher-order topological properties. The adjacency matrix A of the lane graph $G(N)$ is defined as:

$$A[i, j] = \begin{cases} 1 & \text{if } (l_i, l_j) \in E_L, \\ 0 & \text{otherwise.} \end{cases} \tag{5}$$

The matrix A^k (the k-th power of A) gives the number of directed paths of length k between lanes. The reachability matrix R can be computed as:

$$R = I + A + A^2 + \cdots + A^{m-1}, \tag{6}$$

where $m = |L|$ and I is the identity matrix. The entry $R[i, j] > 0$ indicates that lane l_j is reachable from lane l_i.

This formal grammar provides both a declarative specification of topological correctness and a computational foundation for implementing validation algorithms. By encoding these relationships in matrix form and applying graph-theoretic techniques, we can efficiently verify consistency across large-scale road networks and systematically identify the location and nature of topological defects.

2.3 Algorithm Implementation

The practical realization of the formal topological framework requires software infrastructure capable of parsing OpenDRIVE XML files, extracting topological relationships, and evaluating consistency predicates. Our implementation approach follows the Self-Describing Data Model (SDDM) paradigm (Dailey et al., 2002, Oaks, 2003), which establishes a direct mapping between XML schema elements and native data structures, preserving hierarchical organization while enabling efficient computational processing.

In previous development work, we created a MATLAB-based OpenDRIVE parser that implements the SDDM concept by mapping XML elements to MATLAB structure arrays (Lógó and Barsi, 2022). This parser has been deployed in various road network analysis contexts and provides a reference implementation for validating XML parsing correctness. For the present topological analysis framework, however, we utilized the MATLAB parser primarily for verification purposes, comparing its structural interpretation against our new implementation.

The topological verification system itself was developed in Python as a standalone analysis tool. The implementation employs object-oriented design with entity classes (Road, Lane, Junction) directly corresponding to the formal model elements R, L, and J defined in Section 2.2. Python dataclasses provide compact entity definitions with automatic initialization and hashing mechanisms, enabling efficient set-based connectivity operations essential for topological analysis.

XML parsing utilizes Python's ElementTree API through a multi-pass extraction strategy. The first pass instantiates road and junction entities from `<road>` and `<junction>` XML elements, populating basic attributes and building initial identifier mappings. The second pass traverses the `<laneSection>` hierarchies within each road to construct lane entities and extract predecessor-successor link declarations from nested `<link>` sub-elements. The third pass resolves symbolic ID references into direct object pointers, thereby building the explicit connectivity relations E_R and E_L as sets of entity tuples. This incremental parsing approach decouples entity instantiation from relationship resolution, enabling forward references to be correctly handled after all entities are loaded. Dictionary-based symbol tables maintain mappings from string identifiers to object instances, facilitating $O(1)$ cross-reference resolution complexity.

The verification module implements each consistency predicate as a dedicated validation function. Road-level checks iterate over the road collection, verifying reciprocal predecessor-successor relationships through set membership testing. Lane-level verification employs nested iteration over the network hierarchy-roads, lane sections, and individual lanes-validating that successor declarations are reciprocated in target lane predecessor sets and that lane connectivity aligns with road-level topology or is properly mediated through junctions. Junction verification correlates connection definitions with road network structure, ensuring that incoming roads reference the junction in their successor sets and that all junction-approaching roads have corresponding connection entries. Detected violations are logged to a structured report object with categorical error classification and entity-specific diagnostic information, including road IDs, lane indices, and the specific predicate violated.

For global connectivity assessment, the system constructs the lane-level adjacency matrix A using NumPy sparse matrix representations. The matrix is populated through iteration over the EL connectivity relation, marking entries $A[i, j] = 1$ where directed lane connections exist. The reachability matrix R is computed via the series expansion $R = \sum A^k$ for $k = 0$ to $m - 1$, where $m = |L|$, implemented through optimized matrix exponentiation routines. The resulting reachability matrix is analyzed to identify weakly connected components using a graph traversal algorithm that marks visited nodes and counts distinct component clusters. Networks containing multiple disconnected components trigger warnings indicating potential routing discontinuities that could affect simulation validity.

The verification system generates comprehensive structured text reports containing network statistics (entity counts, connectivity edge counts), categorized error listings with sequential indexing and descriptive messages, and an overall topological validity assessment. Report output is formatted for both human readability-with fixed-width section separators and hierarchical organization-and machine parseability, enabling integration into automated continuous quality assurance pipelines. Each detected inconsistency is documented with sufficient detail to facilitate targeted correction in source OpenDRIVE files, including specific entity identifiers and the nature of the topological violation.

3. Synthetic and Real-World Test Scenarios

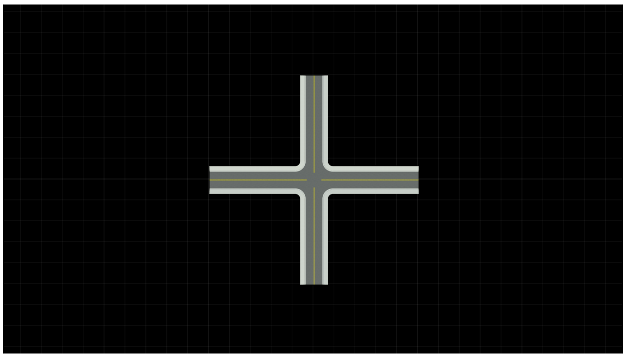
To validate the topological analysis framework and demonstrate its practical applicability, we applied the verification system to four distinct test cases: two synthetic scenarios generated specifically for controlled testing, and two real-world road networks derived from production-grade HD map datasets. This dual approach enables both systematic evaluation of the formal grammar under known conditions and empirical assessment of topological integrity in operational mapping contexts.

3.1 Synthetic Test Scenarios

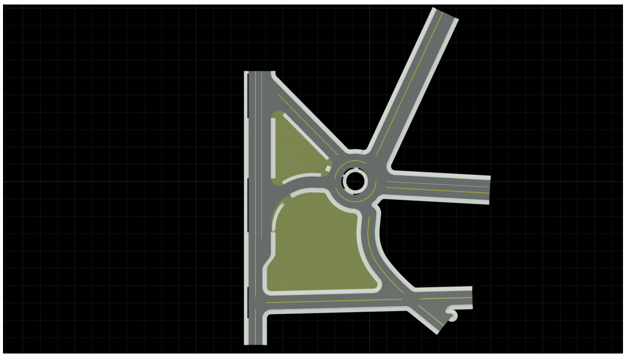
The synthetic test cases were generated using MathWorks RoadRunner, a professional road modeling and 3D visualization software that enables the creation of realistic road networks, junction configurations, and traffic infrastructure elements (Fig. 1). RoadRunner is particularly suitable for autonomous vehicle testing environments as it allows direct export to OpenDRIVE format while maintaining precise control over topological structure. The synthetic scenarios provide known ground truth, facilitating direct validation of the verification algorithms.

TestCase1 - Four-Way Intersection: This scenario models a regular, right-angled four-way intersection where each direction contains a lane pair consisting of one incoming and one outgoing lane. The geometric configuration is symmetric and straightforward, representing a fundamental junction type commonly encountered in urban environments. The primary verification objective for this case is to confirm correct path connections, including entry and exit points, and proper lane-to-lane association across the junction. This scenario serves as a baseline test for the most elementary topological configuration, where consistency should be readily achievable.

TestCase4 – Complex Multi-Branch Roundabout: The most complex synthetic scenario features an irregular roundabout geometry with connecting roads of varying directions and lengths. This configuration deliberately introduces structural challenges including asymmetric approach angles, unequal branch spacing, and multiple lane transitions. The topological verification focuses on detecting redundant connections, identifying cyclic paths, and ensuring consistent successor-predecessor relationships throughout the circular structure and its branches. This case emphasizes the practical necessity of the topological rule system by exposing potential inconsistencies that arise in non-trivial geometric configurations.



(a) TestCase1

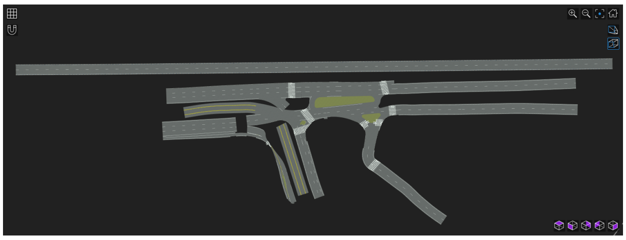


(b) TestCase4

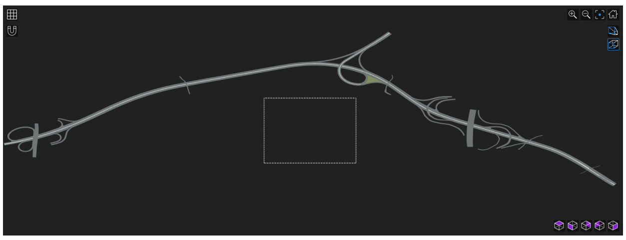
Figure 1. Synthetic test scenarios.

3.2 Real-World Test Scenarios

Real-world test cases provide empirical evidence of topological challenges encountered in production HD map generation workflows. These datasets represent actual locations and were produced using professional mapping tools and methodologies (Fig. 2).



(a) Szent Gellért Square (Budapest)



(b) Südtangente (Karlsruhe)

Figure 2. Real-world test scenarios.

Szent Gellért Square (Budapest): This dataset models a complex urban intersection in central Budapest, Hungary. The

https://doi.org/10.5194/isprs-archives-XLIX-B4-2026-417-2026

OpenDRIVE model was created through a collaborative process involving MSc students working under supervision, with partial models subsequently merged into a unified representation. This development methodology introduces potential for topological inconsistencies at model boundaries and connection points. The area encompasses multiple converging roads, tramway infrastructure, and pedestrian crossings, creating a topologically dense network. The dataset provides insight into topological challenges arising from manual editing, model composition, and representation of multi-modal transportation infrastructure.

Karlsruhe Südtangente (Germany): This model represents a section of the southern bypass road around Karlsruhe, produced by Atlatec, a professional HD mapping company specializing in automated map generation from sensor data. The Sudtangente ¨ dataset features highway-grade infrastructure including acceleration lanes, deceleration zones, grade-separated interchanges, and complex merging configurations. As a professionally generated model from an established mapping provider, this case enables assessment of topological quality achievable through automated production workflows versus potential residual inconsistencies requiring formal verification.

3.3 Test Case Characteristics

The four test scenarios exhibit significant variation in scale and complexity (Table 1). The synthetic cases range from a minimal 20-road network (TestCase1) to an 83-road configuration (TestCase4), with corresponding variations in lane counts and junction complexity. Real-world datasets demonstrate substantially greater structural detail, particularly the Karlsruhe Südtangente model with 114 roads, 886 lanes, and 189 junctions. The file sizes and line counts in the XML representation reflect this complexity, with the Südtangente model requiring over 22,000 lines of structured data.

These test cases collectively span the range from minimal controlled configurations to large-scale operational networks, enabling comprehensive evaluation of the topological verification framework across diverse structural conditions and data sources.

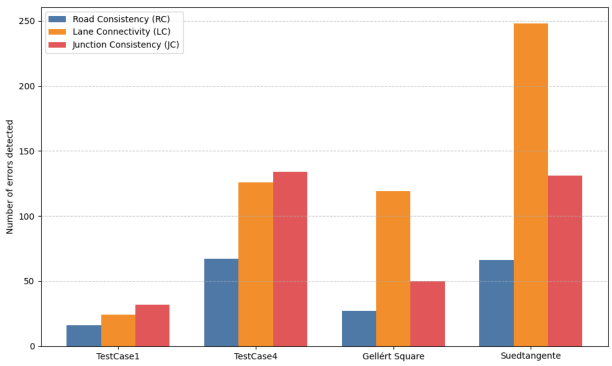


Figure 3. Topological error distribution across the examined networks.

4. Results

The Python-based topological verification system was applied to all four test scenarios, revealing significant structural inconsistencies across both synthetic and real-world datasets. The analysis provides empirical evidence of widespread topological defects in OpenDRIVE models and demonstrates the critical need for formal verification beyond visual inspection and geometric validation.

4.1 Overall Network Metrics and Error Density

The analysis categorizes detected inconsistencies according to the formal consistency predicates defined in Section 2.2: road-level constraints (RC), lane-level constraints (LC), and junction-level constraints (JC).

Table 2 demonstrates that topological inconsistencies are pervasive across all network types, with error-to-road ratios ranging from 2.68 to 3.94. Notably, the professionally generated Südtangente dataset exhibits an error density comparable to synthetic test cases, indicating that automated HD map generation workflows do not inherently guarantee topological correctness despite achieving high geometric accuracy.

4.2 Error Distribution by Category

The topological errors distribute across three primary categories, with distinct patterns emerging between synthetic and real-world datasets (Table 2 and Fig. 3)

Lane-level inconsistencies (LC violations) constitute the dominant error category in real-world datasets, accounting for 56–61% of detected defects. This pattern reflects the fundamental challenge of maintaining lane continuity across road section boundaries in production mapping workflows. Junction-level errors predominate in synthetic cases, particularly TestCase1, where the manually constructed intersection logic exhibits incomplete connection definitions.

4.3 Road-Level Inconsistencies (RC Violations)

Road connectivity violations manifest primarily as broken predecessor-successor relationships, violating the fundamental consistency predicate RC_1. In the Südtangente dataset, 66 such violations were identified. A representative example: "RC_1 violation: Road ’3212’ $\rightarrow$ ’7223’ but ’7223’ does not list ’3212’ as predecessor.”

This error pattern indicates asymmetric link definitions where a road correctly identifies its successor, but the successor road fails to reciprocate with the appropriate predecessor reference. The consequence for simulation systems is severe: global path planning algorithms operating on the road-level graph encounter incomplete edge definitions, potentially resulting in routing failures or suboptimal path generation. The OpenDRIVE specification implicitly assumes bidirectional consistency in predecessor-successor pairs; violations of this assumption compromise the semantic validity of the network representation regardless of geometric correctness.

4.4 Lane-Level Inconsistencies (LC Violations)

Lane-level defects represent the most critical category of topological errors from a simulation perspective. The verification analysis revealed that the majority of detected inconsistencies in real-world datasets involve lane continuity violations, particularly LC_2 violations where non-terminal lanes lack successor definitions.

In the Südtangente dataset, 248 lane-level errors were identified, with the dominant pattern being “Non-terminal lane has

Characteristic	TestCase1	TestCase4	Gellért Square	Südtangente
Total roads	20	83	73	114
Total lanes	56	258	243	886
Total junctions	1	7	11	22
File size (kB)	168	1003	524	1663
XML lines	2015	11053	7337	22127

Table 1. Characteristics of synthetic and real-world test scenarios. Connected roads refers to the number of primary approach roads to the junction; XML line counts derived from exported OpenDRIVE files.

Category	TestCase1	TestCase4	Gellért Square	Südtangente
Total topological errors	72	327	196	396
Errors/Road Ratio	3.60	3.94	2.68	3.47
RC (Road)	16 (22%)	67 (20%)	27 (14%)	66 (15%)
LC (Lane)	24 (33%)	126 (39%)	119 (61%)	248 (56%)
JC (Junction)	32 (45%)	134 (41%)	50 (25%)	131 (29%)
Dominant Issue	Junction logic	Mixed	Lane continuity	Lane continuity

Table 2. Distribution of topological errors. Percentages indicate proportion of total errors for each test case.

no successors.” This error occurs when a lane terminates at a road section boundary without explicit connection to lanes in the subsequent road section, despite the geometric continuation of the roadway. From a simulation standpoint, this creates a “logical discontinuity”: although the road visually continues, the lane graph contains a terminal node, causing simulation engines to interpret the lane as ending.

Modern autonomous vehicle simulation platforms (IPG CarMaker, VTD, CARLA) rely on explicit lane-to-lane connectivity for agent navigation. When a simulated vehicle reaches a lane with no defined successor, the simulation typically generates an error condition or removes the vehicle from the scenario. This behavior renders the OpenDRIVE model unsuitable for dynamic simulation without manual correction or preprocessing to reconstruct missing lane links.

The Gellért Square dataset exhibits similar patterns, with 119 lane-level errors (61% of total defects). The high proportion of LC violations in this manually edited and merged dataset suggests that topological consistency is particularly vulnerable at model composition boundaries, where independently authored road sections are joined.

4.5 Junction Inconsistencies (JC Violations)

Junction-level errors arise from misalignment between junction definitions and the road network structure. The Südtangente verification identified 131 junction-related violations across 189 junctions, indicating that nearly 70% of junctions contain at least one topological defect.

Typical violations include incomplete incoming road coverage, where a junction object expects specific approach roads that do not correctly reference the junction in their link definitions, and missing connection definitions for roads that geometrically intersect the junction area. These errors violate predicates JC_1 and JC_2, which require consistent junction membership declarations and complete connection coverage for all approaching roads.

The practical implication is that intersection behavior becomes undefined: simulation agents cannot correctly determine right-of-way, permissible turning movements, or lane-to-lane routing through the junction. In the Südtangente case, with 22 junctions exhibiting systematic inconsistencies, the majority of complex maneuvering scenarios would fail under simulation.

4.6 Network Fragmentation Analysis

The connectivity analysis component of the verification system quantifies network fragmentation through computation of the reachability matrix R and identification of weakly connected components. The results reveal dramatic fragmentation in real-world datasets.

In the Gellért Square case, 241 connected components have been found with 243 lanes. The Südtangente network, despite containing 886 lanes, fragments into 467 isolated components with an average component size of 1.90 lanes. This fragmentation pattern indicates that the OpenDRIVE file does not represent a coherent network but rather a collection of hundreds of logically disconnected short road segments. While the geometric representation appears continuous, the topological structure is discrete and fragmented.

The lane connectivity ratio-measuring the proportion of theoretically possible lane-to-lane connections that are actually defined-approaches zero in both real-world cases. This metric provides quantitative evidence that the models fundamentally lack the graph structure necessary for network-level routing and long-range simulation.

4.7 The Geometric-Topological Paradox

The Südtangente case study reveals a fundamental discrepancy between geometric precision and topological validity, which we term the “Geometric-Topological Paradox.” This professionally generated HD map achieves centimeter-level geometric accuracy in road center lines, lane boundaries, and curvature representation. Visual inspection in standard viewing software (RoadRunner, odrViewer) reveals no apparent defects; the road network appears complete and well-formed.

However, the formal topological analysis exposes severe structural deficiencies invisible to geometric validation. The network’s fragmentation into 467 components with an average size of 1.9 lanes demonstrates that every second lane transition contains a logical discontinuity. This pattern renders the dataset unsuitable for trajectory simulation or path planning without extensive manual correction.

The paradox arises from the optimization priorities of automated HD map generation pipelines. Computer vision algorithms excel at detecting geometric features-lane markings, road edges, surface boundaries-from sensor data, achieving high spatial accuracy. However, these algorithms do not inherently understand functional relationships: which lanes logically connect, how traffic flows through junctions, or what constitutes a valid lane transition. Geometric reconstruction succeeds while topological inference fails.

This finding has significant implications for HD map production workflows. It demonstrates that geometric validation alone is insufficient to ensure simulation readiness. The formal topological verification framework developed in this research addresses this gap by providing automated detection of logical inconsistencies that evade geometric quality control.

5. Conclusions

This research establishes a comprehensive framework for formal topological analysis of OpenDRIVE-based road networks, addressing a critical gap in validation methodologies for high-definition maps used in autonomous vehicle simulation. While existing quality assurance practices emphasize geometric accuracy and semantic completeness, our findings demonstrate that topological consistency represents an equally essential-yet systematically undervalidated-dimension of HD map quality.

The formal grammar introduced in Section 2.2 provides a rigorous mathematical foundation for expressing connectivity relationships and structural constraints inherent in OpenDRIVE models. By interpreting road networks as directed graphs with explicit consistency predicates at road, lane, and junction levels, we establish a declarative specification of topological correctness that enables both symbolic reasoning about network structure and algorithmic verification through graph-theoretic techniques.

The comparative analysis across four diverse test cases establishes several key findings. First, topological inconsistencies are ubiquitous in OpenDRIVE models regardless of generation method, network scale, or intended application domain. Error-to-road ratios ranging from 2.68 to 3.94 across all test scenarios indicate that multiple topological defects per road represent the norm rather than the exception. Notably, professionally generated datasets exhibit error densities comparable to manually constructed synthetic cases, demonstrating that neither automated sensor-based mapping workflows nor careful manual modeling inherently guarantee topological correctness.

Second, lane-level connectivity defects constitute the dominant error category in production datasets, directly impacting simulation validity. The prevalence of non-terminal lanes lacking successor definitions creates logical discontinuities in otherwise geometrically continuous roadways, rendering affected network sections unsuitable for agent-based traffic simulation. Third, network fragmentation metrics reveal that many ostensibly complete HD maps are topologically incoherent despite geometric correctness. The Südtangente dataset's fragmentation into 467 isolated components from 886 total lanes exemplifies this geometric-topological paradox: centimeter-level geometric precision coexisting with fundamentally broken logical connectivity structure.

These results validate the necessity of incorporating topological verification into HD map production workflows as a standard quality assurance step. The formal grammar and verification algorithms developed in this research provide the computational foundation for such integration, enabling automated detection of structural defects prior to deployment in simulation or autonomous vehicle testing environments. Beyond error detection, the framework establishes a quantitative basis for assessing topological quality through metrics such as connectivity ratios, component fragmentation indices, and predicate violation counts.

For simulation platform developers, this research demonstrates that assuming topological coherence in externally sourced OpenDRIVE files will frequently lead to runtime failures. Implementing topological validation as a dataset import step, with automated correction or manual review workflows for detected inconsistencies, becomes essential for maintaining simulation reliability. As the autonomous vehicle industry continues its transition from prototype testing to large-scale deployment, ensuring the logical coherence of digital road representations becomes not merely a quality improvement but a prerequisite for trustworthy simulation-based validation.

References

ASAM e.V., 2026. ASAM OpenDRIVE Specification. https://asam.net/standards/detail/opendrive. Last accessed: 2026-02-12.

Dailey, D. J., Maclean, S., Cathey, F. W., Meyers, D., 2002. A Self-Describing Data Transfer Model for ITS Applications. *IEEE Transactions on Intelligent Transportation Systems*, 3, 293-299. https://doi.org/10.1109/TITS.2002.806805.

Elghazaly, G., Frank, R., Harvey, S., Safko, S., 2023. High-Definition Maps: Comprehensive Survey, Challenges, and Future Perspectives. *IEEE Open Journal of Intelligent Transportation Systems*, 4, 527-550. https://doi.org/10.1109/OJITS.2023.3295502.

Křehlík, S., Vanžura, M., Skokan, A., 2023. Minimum required accuracy for HD maps. *The Journal of Navigation*, 76, 238-254. https://doi.org/10.1017/S0373463323000140.

Liu, R., Wang, J., Zhang, B., 2020. High Definition Map for Automated Driving: Overview and Analysis. *The Journal of Navigation*, 73, 324-341. https://doi.org/10.1017/S0373463319000638.

Lógó, J. M., Barsi, A., 2022. THE ROLE OF TOPOLOGY IN HIGH-DEFINITION MAPS FOR AUTONOMOUS DRIVING. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLIII-B4-2022, 383-388. http://doi.org/10.5194/ISPRS-ARCHIVES-XLIII-B4-2022-383-2022.

Lógó, J. M., Barsi, A., 2023. Evaluation of Topology Description Models in Road Network Formats. *Acta Polytechnica Hungarica*, 20, 143-156. https://doi.org/10.12700/APH.20.1.2023.20.10.

Oaks, P., 2003. Towards self-describing web services. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2722, 476-485. https://doi.org/10.1007/3-540-45068-8$_9$0.

Poggenhans, F., Pauls, J. H., Janosovits, J., Orf, S., Naumann, M., Kuhnt, F., Mayr, M., 2018. Lanelet2: A high-definition map framework for the future of automated driving. *IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC*, 2018-November, 1672-1679. https://doi.org/10.1109/ITSC.2018.8569929.

Soorchaei, B. E., Razzaghpour, M., Valiente, R., Raftari, A., Fallah, Y. P., 2022. High-Definition Map Representation Techniques for Automated Vehicles. *Electronics 2022, Vol. 11, Page 3374*, 11, 3374. https://doi.org/10.3390/ELECTRONICS11203374.