Graph Grammar based on Similarity and Conformity Graphs

Artur Basiura[1 [0000−0002−1034−6078]], Adam Sędziwy[1 [0000−0002−2489−3726]]

AGH University of Science and Technology, Department of Applied Computer Science, Al.Mickiewicza 30, 30-059 Kraków, Poland
{abasiura,sedziwy}@agh.edu.pl

Abstract. Public lighting retrofit planning requires computationally efficient yet standards-aware methods. This paper extends previous work on similarity and conformity graphs by introducing a formal graph-grammar framework for estimating the minimum achievable installed power in outdoor lighting networks under CEN 13201. The grammar defines transformations from a base infrastructure graph to derived similarity and conformity graphs. The similarity measure is computed in the SimilarityGraphCreation operation, stored on similarity edges, and then used for filtering, grouping, and result propagation, enabling fast estimation and what-if analysis. The framework is validated on a real retrofit project in Kraków, Poland, covering 3,741 lamps. The results show that the number of explicitly calculated segments can be reduced from 602 to 9–235, while richer similarity measures produce estimates close to the reference solution.

Keywords: graph · graph methods · LED lighting · roadway lighting · similarity graph · conformity graph · similarity measure

1 Introduction

Modern decision-making in urban infrastructure increasingly requires fast, data driven estimates rather than time consuming. At a city wide scale, complete input data are often unavailable on demand, so early investment choices are made under time pressure and based on partial indicators or expert judgement, which may lead to suboptimal prioritization and unexpected costs.

This challenge is particularly evident in public lighting. Retrofits affect safety and quality of life [8] and directly impact municipal budgets, light pollution, and electricity related greenhouse gas emissions [22]. While LED technology can significantly reduce installed power, achievable savings depend on design and configuration choices [18, 10] yet preparing full photometric designs for large networks remains costly and slow, limiting their use for early stage screening. Hence, municipalities need a scalable intermediate methodology to estimate CAPEX, energy savings, and ROI early enough to support planning and identify areas that should undergo detailed photometric analysis first.

This paper targets rapid estimation of the lowest achievable street-lighting power under prescribed constraints derived from lighting standards (e.g.,

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29909-3_13

luminance/illuminance, uniformity, glare) and context dependent objectives such as light pollution mitigation (e.g., limiting upward light output and spill light, and matching distributions to roadway geometry)

The proposed approach builds on and extends the use of similarity and conformity graphs [2] as a mechanism to reduce computational complexity. Instead of exhaustively exploring the entire design space, elements of infrastructure with comparable geometric, functional, and contextual characteristics are grouped by similarity relations. Then, computations are performed for representative instances (e.g., selected road segments), and the resulting parameters and outputs are propagated to other elements within conformity defined groups, yielding actionable estimates at substantially lower computational cost.

The main contribution of this paper is a formal graph-grammar definition that specifies admissible transformations from a base graph to derived graph structures: a similarity graph and a conformity graph. The proposed grammar includes operations for constructing similarity relations, rule and threshold controlled conformity filtering (enabling, among others, the incorporation of user preferences). This formalization supports the implementation and interchangeability of different similarity measures functions and conformity filters without altering the overall processing pipeline, and it enables comparative evaluation with respect to the estimation accuracy and computational cost.

The paper is organized as follows. Section 2 reviews related work. Section 3 introduces and systematizes the proposed graph grammar based on operations on similarity and conformity graphs. Section 4 presents the case study on a real-world urban lighting network and compares alternative similarity functions. Section 5 concludes the paper and outlines the direction of future research.

2 State of the art

Lighting design and assessment are traditionally based on high accuracy photometric computations to ensure compliance with lighting standards, particularly for road lighting under CEN 13201 [3–6] or RP-8 [11]. In practice, specialized tools such as DIALux and Relux are used to verify whether a given luminaire layout and operating regime meet normative criteria (e.g. luminance/illuminance targets, uniformity, glare limits) [15–17]. Although indispensable for final design and acceptance, this accuracy becomes a bottleneck for city scale "what if" analyzes that require repeating evaluations across thousands of segments and multiple retrofit scenarios.

To address scalability, a well-established line of work represents large lighting networks as graph-based structures [19]. In lighting retrofit planning, Sędziwy and Kotulski propose decomposing these large-scale graphs into distributed representations that allow parallel processing. More broadly, graph models are widely used to represent, analyze and solve complex problems in many domains, not only in lighting and infrastructure engineering, but also in other areas [14, 1, 12, 7], where graphs are applied to model transaction networks,

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29909-3_13

exposures between entities and systemic dependencies. This framing aligns with a general trend in infrastructure analytics: graphs provide a natural formalism for representing network topology, adjacency, shared assets, and boundary constraints (e.g., continuity across segments or constraints at intersections), thereby enabling accurate computations to be performed on selected subareas of the network while preserving essential inter-segment relationships.

Related work further formulates improving roadway lighting as an optimization problem constrained by standards, where computation is used to identify over-lighting, reduce the supplied power, and thus improve energy efficiency [20]. In these formulations, graph based modeling supports structuring the search space and coordinating decisions across network elements, but feasibility checks and objective evaluation typically remain tied to high fidelity photometric computation. As a result, throughput improves with parallelism, yet early stage screening and rapid budgeting at the city scale remain challenging.

Beyond design-time computations, graph grammar has also been adopted for intelligent and adaptive lighting control. Wojnicki et al. propose formal graph representations and dual graph transformations to generate control patterns for intelligent lighting systems, emphasizing automation and scalability [23]. Closely related work introduces graph grammar concepts for efficient adjustment of outdoor lighting (e.g., traffic aware dimming), highlighting flexibility in specifying permissible modifications and the potential for computationally efficient rule execution. This line of research is particularly relevant from a methodological standpoint: it shows that graph transformations can serve not only as an implementation technique but also as a formal specification layer that makes processing pipelines explicit, modular, and reproducible.

From a formal perspective, graph grammars provide a principled mechanism to define and constrain graph transformations by means of rewriting rules. In infrastructure analytics, this is attractive because it allows one to specify admissible modifications of a modeled network, encode preconditions and invariants (e.g., type constraints, boundary consistency), and ensure that complex processing workflows can be composed from a small set of well-defined primitive operations. In the context of lighting infrastructure, a grammar-based formulation can make explicit how a base infrastructure graph is transformed into derived structures used for computation (e.g., similarity relations, conformity-based grouping, partitions) and how intermediate results are merged back into a consistent representation.

A promising approach in this context is to leverage the similarity between infrastructure fragments to enable reuse of previously analyzed configurations and reduce the number of expensive evaluations. Basiura, Sędziwy, and Komnata formalize this idea through Similarity and conformity graphs, which represent infrastructure fragments and their similarity relations to support fast comparison and controlled transfer over comparable cases [2, 13]. In this paradigm, the computational burden is reduced by identifying groups of mutually similar instances, evaluating representative instances, and propagating parameters or

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29909-3_13

outcomes to conforming instances under explicitly defined rules. This approach is used later in the paper to define a new graph grammar.

3 Similarity graph and conformity graph

The concept of Similarity and Conformity graphs was introduced in the publication [2] as a mechanism for obtaining a rough assessment of a lighting infrastructure. The publication did not provide a formal specification that would enable an unambiguous, fully formalized solution. The description focused on the conceptual introduction of graph structures, without a systematic analysis of how the choice of similarity measure influences the algorithm's outcomes. In particular, it did not examine the role and impact of the conformity filter and conformity sorting functions on the behavior, stability, and overall performance of the transformation and retrieval pipeline.

To make the transformation pipeline auditable and well-defined, we introduce a graph grammar Φ. Φ over the base graph family (BG). Its production rules encode the admissible rewriting operations and structural invariants required by the proposed agent-based information system.

Definition 1 (Graph grammar over BG). *Let BG be a family of base graphs. A graph grammar (set of graph-transformation rules) over BG is a triple*

$$\Phi = (G, \Pi, R), \tag{1}$$

where:

- *$G \in BG$ is the initial graph instance subject to transformation,*
- *Π is a finite set of graph-transformation operators of the form*

$$f : BG^n \times (R \cup \{\lambda\}) \to BG^k, \tag{2}$$

  with $n, k > 0$,
- *R is a set of structural rewriting rules that modify the graph topology and/or attributes (e.g., by adding/removing vertices or edges, changing labels/types, or updating attribute values). λ refers to the case of no explicit rule for an operator f.*

Transformation rules modify the base graphs BG from one representation to another, according to receipts included in R.

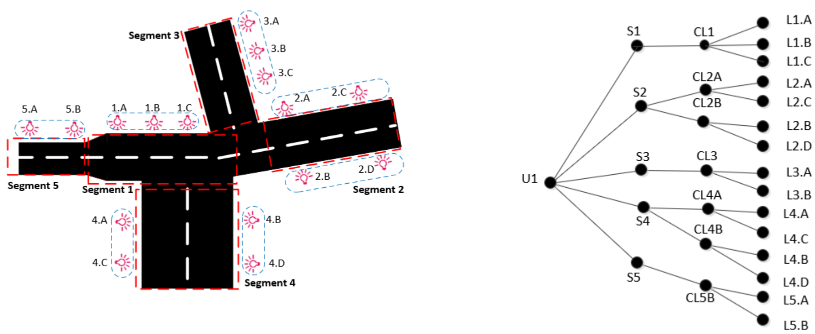
The following operators constitute special cases of Π in our pipeline:

1. `SimilarityGraphCreation`$(BG, f_{\mathrm{SIM}} \in R)$ – creates the graph SG with similarity edges according to the similarity metric f_{SIM}.
2. `ConformityGraphCreation`$(SG, f_{\mathrm{CON}} \in R, g_{\mathrm{CON}} \in R)$ – constructs a graph of type conformity graph (the definition is introduced later in this work) that captures mutual relationships among nodes. The function f_{CON} specifies the rules governing graph construction (e.g., restricting the graph to edges whose

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29909-3_13

weights exceed a prescribed threshold). The function $g_{\mathrm{CON}} \in R$ determines the processing sequence (e.g., their order of insertion in the underlying data structure and based on the weight/value of the parameters of a selected vertex).

3. `EstimationOnConformity`$(BG, CG, f_{\mathrm{EST}} \in R)$ – creates vertices and edges in the graph BG based on similarity relations encoded in the graph CG and f_{EST}. The function f_{EST} specifies how the graph enrichment operation is to be performed (e.g., the rules determining which parameters are propagated and to which vertices they are copied in the graph BG)



(a) (b)

Fig. 1: Modeling a street (a) as a graph (b)

By separating the representation of the physical infrastructure (base graphs) from similarity-based retrieval structures (Similarity and conformity graphs), the grammar Φ supports a computational strategy that is broadly applicable and straightforward to integrate into diverse processing pipelines. Similarity relations can be constructed locally, conformity structures can be updated incrementally, and solutions can be reused within conforming components. As a result, the number of expensive, high fidelity evaluations required for early stage of the city-scale lighting planning can be substantially reduced.

An example of modeling the lighting infrastructure as a graph is shown in Fig. 1. This example is used throughout the remainder of the paper to illustrate the operation of the proposed transformations.

3.1 SimilarityGraphCreation rule

The SimilarityGraphCreation rule constructs the similarity layer of the model. Its role is to add similarity edges between comparable objects in the base graph (BG) according to a predefined similarity-measure function. An example of the operation of the rule and the resulting similarity edges is presented in Fig. 2

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29909-3_13

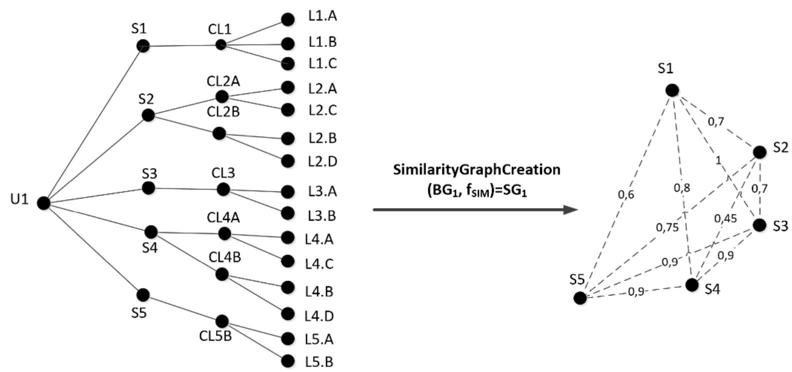


Fig. 2: `The SimilarityGraphCreation` example

The operation `SimilarityGraphCreation`$(G_{\mathrm{WE}}, f_{\mathrm{SIM}})$ is executed according to the following algorithm:

Step 0 Initialize two empty sets, $\mathcal{E}_{\mathrm{SIM}}$ and $V_{\mathrm{SIM,T}}$. The set $\mathcal{E}_{\mathrm{SIM}}$ stores the identified similarity edges. The set $V_{\mathrm{SIM,T}}$ stores similar candidate vertices of the same vertex type T (e.g., vertices of the type *Segment*).

Step 1 Determine the set $V_{\mathrm{SIM,T}}$ of vertices v_i. This set contains all vertices of the same type T that belong to G_{WE} and for which a similarity measure will be computed. For example, for the graph BG_1, the set for type *Segment* is

$$V_{\mathrm{SIM,T}} = \{S_1, S_2, S_3, S_4, S_5\}.$$

Step 2 For each vertex $v_1 \in V_{\mathrm{SIM,T}}$ (e.g., S_1), create edges to all other vertices contained in $V_{\mathrm{SIM,T}}$. Each such edge is added to $\mathcal{E}_{\mathrm{SIM}}$. For instance, for S_1 we obtain

$$\mathcal{E}_{\mathrm{SIM}} = \mathcal{E}_{\mathrm{SIM}} \cup \{(S_1, S_2), (S_1, S_3), (S_1, S_4), (S_1, S_5)\}.$$

Step 3 Remove v_1 from $V_{\mathrm{SIM,T}}$. Repeat Step 2 until $V_{\mathrm{SIM,T}}$ becomes empty.

Step 4 For each edge e_i in the set $\mathcal{E}_{\mathrm{SIM}}$, compute the similarity degree between the two incident vertices using the similarity measure f_{SIM}. The resulting value is stored as an edge attribute.

The output is a graph constructed from $V_{\mathrm{SIM,T}}$ and $\mathcal{E}_{\mathrm{SIM}}$ is *similarity graph* (SG).

3.2 Similarity measure (f_{SIM})

The similarity measure f_{SIM} is used in the SimilarityGraphCreation operation to quantify how similar two vertices (representing infrastructure objects) are. The resulting score is stored as an attribute of a similarity edge and is subsequently used by filtering and grouping procedures (e.g., during conformity graph construction) and by reuse/propagation operations.

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29909-3_13

Definition 2 (Similarity measure f_{SIM}). *Formally, f_{SIM} is a type aware function that maps a pair of comparable vertices to a bounded similarity score:*

$$f_{\mathrm{SIM}} : V_T \times V_T \to [0,1], \tag{3}$$

where:

- *$V_T \subseteq V$ denotes the set of vertices of the analyzed type T (e.g., Segment, Pole, Luminaire). The value $f_{\mathrm{SIM}}(u,v) = 1$ indicates that u and v are effectively identical with respect to the selected descriptors, while $f_{\mathrm{SIM}}(u,v) = 0$ indicates maximal dissimilarity (or non-compatibility).*
- *comparisons are restricted to vertices of the same type T, cross-type pairs are treated as non-comparable.*

Each vertex $u \in V_T$ is represented by a feature vector $\mathbf{x}(u)$ composed of descriptors relevant to the type T. For example, for vertices of type *Segment* the descriptors may include geometric attributes (e.g., segment length, roadway width, curvature), functional attributes (e.g., road class), and contextual attributes (e.g., surrounding environment class). For vertices representing equipment (e.g., *Pole*, *Luminaire*), descriptors may include mounting height, setback, tilt, or photometric/distribution classes. Continuous features are normalized to ensure commensurability across dimensions.

A common instantiation of f_{SIM} aggregates per feature similarities using a weighted scheme:

$$f_{\mathrm{SIM}}(u,v) \;=\; \frac{\sum_{i=1}^{m} w_i \, s_i(x_i(u), x_i(v))}{\sum_{i=1}^{m} w_i},$$

where $x_i(u)$ is the i-th feature of u, $s_i(p,q) \in [0,1]$ is a per feature similarity function (e.g., distance based for continuous features and match based for categorical features), and $w_i \geq 0$ are feature weights. Missing attributes can be handled by excluding the corresponding terms and renormalizing the remaining weights.

In typical use, f_{SIM} is designed to be symmetric, $f_{\mathrm{SIM}}(u,v) = f_{\mathrm{SIM}}(v,u)$, bounded in $[0,1]$, and configurable per vertex type T through the choice of descriptors, normalization, and weights. This allows the similarity relation to reflect the aspects of infrastructure that are most relevant to the planning objective and constraints (including normative requirements and user defined criteria). These criteria may depend on the type of situation being analyzed.

3.3 ConformityGraphCreation rule

The `ConformityGraphCreation` rule constructs a *conformity graph* based on the *similarity graph*. The resulting graph contains subgraphs that represent groups of similar objects with comparable properties.

The operation `ConformityGraphCreation`($SG_{\mathrm{WE}}, f_{\mathrm{CON}}, g_{\mathrm{CON}}$) is executed according to the following algorithm:

To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29909-3_13

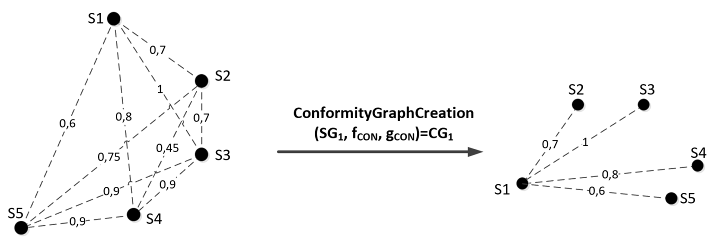


Fig. 3: The `ConformityGraphCreation` example

Step 1 Construct CG from the input graph SG_{WE}. The graph CG contains only the vertices (V_{CG}) of the analyzed type (T) and the corresponding similarity edges ($\mathcal{E}_{\mathrm{CG}}$).

Step 2 Evaluate the filter f_{CON} on each edge e_i of $\mathcal{E}_{\mathrm{CG}}$ set. If $f_{\mathrm{CON}}(e_i)$ returns a negative value, the edge is removed. In the considered implementation, f_{CON} verifies whether the similarity value is below the threshold $TSH_{f_{\mathrm{CON}}}$. If so, it returns -1; otherwise, it returns the similarity value.

Step 3 Let the vertices V_{CG} form the set V_A.

Step 4 Based on a predefined ranking function g_{CON}, each vertex v_i is assigned a score $g_{\mathrm{CON}}(v_i)$. The vertex set is then sorted in ascending or descending order with respect to $g_{\mathrm{CON}}(v_i)$, and this ordering determines the priority of subsequent processing (e.g., based on the degree of the vertex, the vertex index in a predefined ordering, or the value of a selected parameter or a weighted combination of multiple parameters). In the considered implementation we based on the degree of the vertex.

Step 5 For the first vertex $v_1 \in V_A$, retrieve the set of incident edges (v_1, v_n) then for each adjacent vertex v_n, identify the neighboring edges (v_n, v_m) and remove them from $\mathcal{E}_{\mathrm{CG}}$ and create edges (v_1, v_m) if it does not exist and filter f_{CON} gives positive value.

Step 6 Remove v_1 from V_A. If V_A contains more than one element, repeat Step 5.

The output $CG = (V_{\mathrm{CG}}, \mathcal{E}_{\mathrm{CG}})$ is the conformity graph. As shown in Fig. 3.

Example The steps and the resulting *conformity graph* construction for the base graph BG_1 are presented below.

(a) $TSH = 0$ (b) $TSH = 0.9$

Fig. 4: $CG_1 = \texttt{ConformityGraphCreation}(SG_1, f_{\mathrm{CON}}(TSH))$.

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29909-3_13

For `ConformityGraphCreation` $(BG, f_{\mathrm{CON}}\,(TSH\ = 0))$, the conformity threshold is set to 0. In practice, this corresponds to the case in which all vertices are treated as similar (i.e., no similarity relation is rejected solely due to the threshold). The newly constructed graph CG_1 is shown in Fig. 4a.

As the threshold value increases, the similarity edges are progressively filtered out. For $CG_1 = \texttt{ConformityGraphCreation}(SG_1, f_{\mathrm{CON}}(TSH\ = 0.9))$, two subgraphs (connected components) are already distinguished. The resulting graph CG_1 is shown in Fig. 4b

3.4 EstimationOnConformity rule

The primary objective of this operation is to propagate a set of parameters from one vertex to other vertices that exhibit similar properties but do not yet contain the corresponding parameters.

The graph transformation involves two graph types: base graph, which encodes the modeled infrastructure, and conformity graph, which captures similarity relations among vertices. Using the conformity graph (CG), similar vertices (or vertex induced subgraphs) are augmented by imputing missing parameter values.

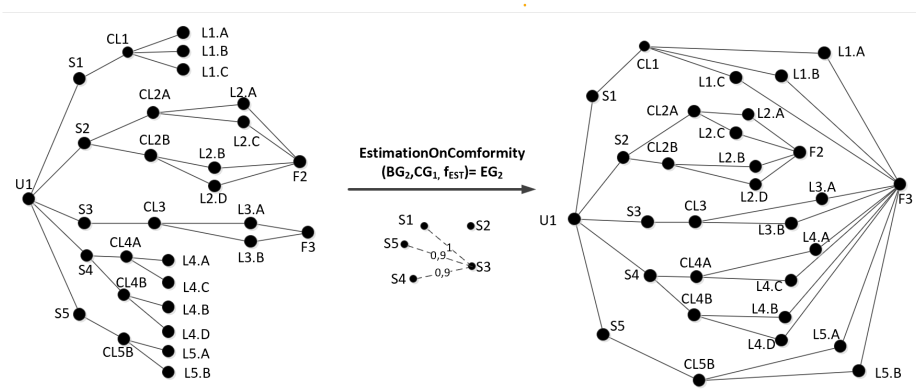


Fig. 5: The `EstimationOnConformity` example

The algorithm for the operation `EstimationOnConformity` $(G_{\mathrm{WE}}, CG_{\mathrm{WE}}, f_{\mathrm{EST}})$ is as follows:

Step 1 Create the list of vertex V_A that contains the *root* vertices of all subgraphs (connected components) that belong to CG_{WE}. And $\mathcal{E}_{\mathrm{CG}}$ that are in CG_{WE}

Step 2 For a vertex $v_1 \in V_A$, retrieve the set of vertices V_B such that an edge $(v_1, v_i) \in \mathcal{E}_{\mathrm{CG}}$ and $v_i \in V_B$.

DOI: 10.1007/978-3-032-29909-3_13

Step 3 For each vertex $v_i \in V_B$ and v_1 apply the function f_{EST}.The function operates on the vertex v_i or on the vertex induced subgraph represented by v_i, and augments the structure of v_i by transferring (imputing) missing parameters from the reference vertex v_1 or from the subgraph associated with v_1. In the considered implementation, is duplicating a specific vertex type (luminaire configuration vertex F) that occurs in the subgraph induced by v_1 and propagating it to the subgraph induced by v_i.

Step 4 Remove the vertex v_1 from the set V_A .

Step 5 If V_A is non empty, repeat Step 2.

The output is a graph that embeds a generalized solution across its subnodes by propagating representative parameter sets within each conformity-defined group.

Overall, the computational procedure reduces to identifying similar elements, performing the required computations for group representatives, and propagating the results to all nodes. This approach enables agent-based modeling of complex configurations and supports rapid estimation at the scale of the entire system.

4 Case Study

In this section, we use a real municipal retrofit as a benchmark to evaluate the proposed graph-grammar–based algorithm and to compare alternative similarity measures.

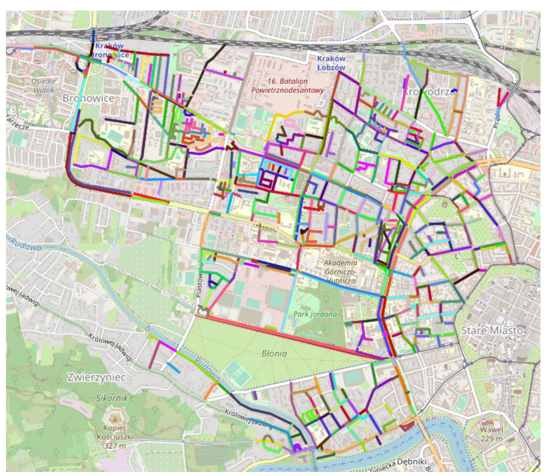


Fig. 6: The study area: the investment scope of the ISE municipal project project (Kraków, Poland)

The case study focuses on the public lighting retrofit covering 3,741 sodium lamps in Kraków, Poland. This municipal project was carried out for luminaires

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29909-3_13

Table 1: Optimization parameters

	From	To	Step	Number
Luminous flux dimming	1%	100%	1%	100
Fixture mounting angle	0°	30°	1°	31
Arm length	0 m	2 m	0.5 m	5
Fixtures types	n/a	n/a	n/a	625

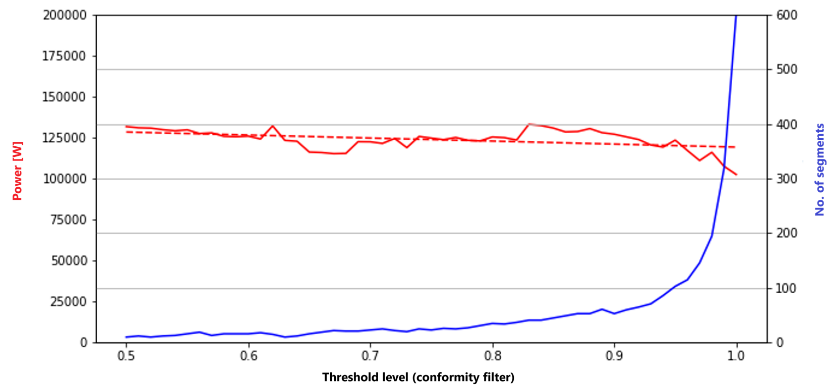


Fig. 7: Similarity measure based on road width (f_{SIM1})

that illuminate 662 road segments (lighting situations). To ensure comparability with the results reported in [2], the same input data set was used. The study was carried out using the proposed formalisms to assess how changes in the similarity measure affect the final estimation results. At the coarsest level used in this case study, the base graph contains 662 vertices corresponding to particular segments (one per lighting segment) connected to one or more streets. If the model is extended to explicitly represent technical details, a natural refinement is to add one vertex per luminaire (here: 3,741). Under that common modeling choice, the graph already reaches more than five thousand vertices before adding any optional objects (e.g., poles as distinct vertices, cabinets, feeders, intersections, etc.). The similarity graph has 218,791 edges. Within the data set analyzed, 60 segments exhibited complete similarity; consequently, the final analysis covers 602 segments.

The modernization area is shown in Fig. 6. The objective is to select lighting infrastructure parameters that minimize total installed power while satisfying the constraints imposed by CEN 13201. The explored design space reflects a practical retrofit scenario limited to replacing existing lighting fixtures and arms (i.e., without relocating poles or redesigning the network geometry). The resulting parameter space comprises approximately 9,7 million candidate configurations, with the set of parameters summarized in Table 1.

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29909-3_13

A key question, therefore, is not only which similarity measure should be used to obtain estimates that most closely match the final high fidelity solution but also how this similarity measure should be formulated in practice.

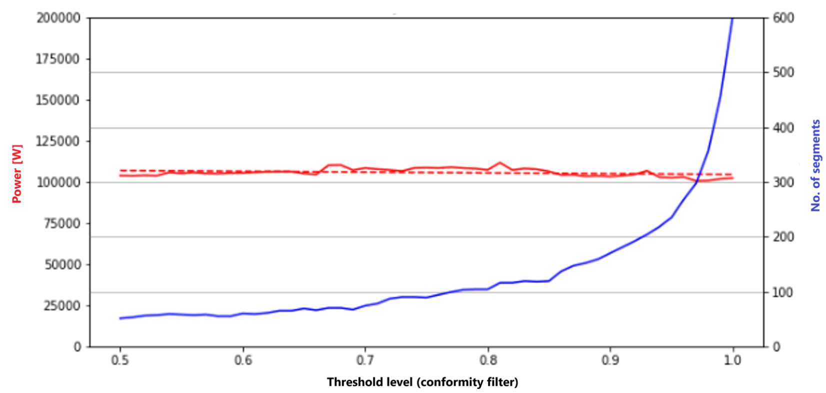


Fig. 8: Similarity measure based on road width and lighting class (f_{SIM2})

In particular, its definition must specify which features are compared, how they are weighted and normalized, and what acceptance criterion (e.g., a threshold) determines whether two instances are considered sufficiently similar for reuse and result propagation—thereby directly shaping the trade off between computational savings and estimation accuracy.

The similarity measure based solely on the roadway width is shown in Figure 7. In this variant, two instances (e.g., road segments) are considered similar exclusively through the agreement of a single geometric parameter, which groups objects with comparable widths regardless of other characteristics such as segment length, road class, surrounding context, mounting geometry, or standard-driven requirements. As a result, this measure serves as a baseline for assessing how the choice of descriptors affects estimation quality: it makes it possible to examine to what extent a single geometric attribute is sufficient for transferring parameters and results and what risk of incorrect propagation arises when contextual and functional information is omitted. In contrast, the similarity measure based on both segment width and the lighting class is shown in Figure 8 When the lighting class is replaced by luminance level, the similarity measure exhibits a different behavior, the resulting trend is shown in Figure 9.

When comparing the three similarity measures, we observe that the width-only measure classifies substantially more segments as similar than the other approaches. As a consequence, fewer segments must be evaluated explicitly, because a larger portion of the network can reuse results within conformity defined groups. For example, at the similarity threshold of 0.6 the measure based on width only requires 15 segments to be computed, whereas the second

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29909-3_13

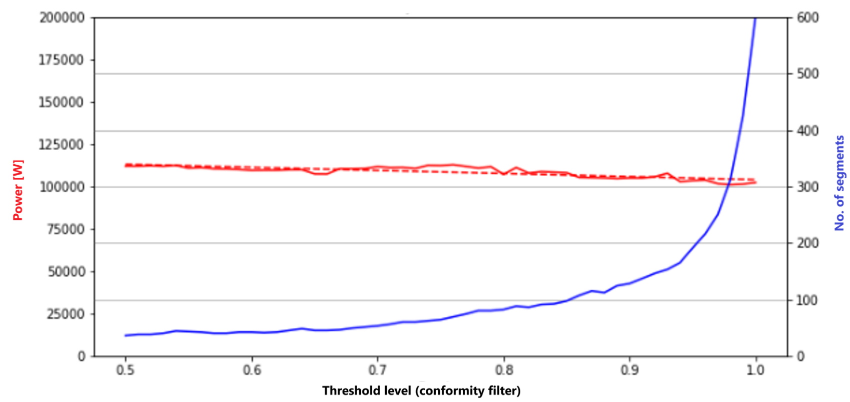


Fig. 9: Similarity measure based on road width and lighting level of luminance (f_{SIM3})

measure requires 60 segments and the third requires 42. At the same time, measures that incorporate additional descriptors (besides the roadway width) produce significantly more stable estimates. As the threshold increases, their results converge more consistently toward the reference optimum, indicating a better balance between reuse and compatibility. In practical terms, although the width-only measure provides the greatest computational reduction, it also increases the risk of transferring parameters across segments that are geometrically similar in width but differ in functional or normative requirements, which can degrade estimation fidelity.

Overall, the comparison highlights the expected trade-off: simpler similarity definitions maximize reuse and reduce computation, while richer, type-aware definitions produce more reliable and stable estimates, particularly at higher thresholds where conformity groups become more homogeneous. The quantitative results are summarized in Table 2.

5 Conclusion

This work presents a graph-grammar approach for standards aware estimation of the minimum achievable installed power in large scale outdoor lighting networks. The key idea is to separate the physical installation model (base graph) from similarity based retrieval structures (similarity and conformity graphs) that support controlled reuse and propagation of parameters and results. By formalizing the transformation pipeline as a graph grammar, the method yields a reproducible and implementation ready specification for constructing, filtering, partitioning, and reconciling these representations.

The future work will extend validation to multiple cities and diverse roadway typologies, include environmental descriptors (e.g., uplight and

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29909-3_13

Table 2: Comparison showing the number of segments requiring detailed calculation and the resulting estimated total power for different similarity measures f_{SIM} and thresholds

$TSH_{f_{\mathrm{CON}}}$	No of calculated segments			Power [W]		
	f_{SIM1}	f_{SIM2}	f_{SIM3}	f_{SIM1}	f_{SIM2}	f_{SIM3}
0.5	9	51	36	131,611	103,910	112,040
0.6	15	61	42	125,800	105,531	109,548
0.7	22	74	53	122,338	108,431	111,703
0.8	34	104	82	125,285	107,299	107,128
0.9	52	170	128	126,964	103,257	105,022
0.95	102	235	191	123,363	102,515	103,421
0.97	145	297	250	110,904	100,588	101,662
0.98	194	356	312	115,923	100,810	101,028
0.99	318	456	425	107,391	101,825	101,395
1.0	602	602	602	102,354	102,354	102,354

spill-light indicators), investigate automated or learning-based calibration of similarity weights and thresholds, and deepen the formal analysis of grammar properties (e.g., determinism, invariants, and correctness of merge/incorporation operations).

References

1. Basiura, A., Kotulski, L., Ziembinski, D.: Common Graph Representation of Different XBRL Taxonomies, Intelligent Information and Database Systems: Asian Conference - ACIIDS (2022) https://doi.org/10.1007/978-3-031-21743-2_40
2. Basiura, A., Sędziwy, A., Komnata, K.: Similarity and conformity graphs in Lighting Optimization and Assessment, 2021, International Conference on Computational Science - ICCS (2021). https://doi.org/10.1007/978-3-030-77970-2_12
3. European Committee For Standardization: CEN/TR 13201:1 Road lighting - Part 1: Guidelines on selection of lighting classes (2014)
4. European Committee For Standardization: CEN/TR 13201:2 Road lighting - Part 2: Performance requirements (2014)
5. European Committee For Standardization: CEN/TR 13201:3 Road lighting - Part 3: Calculation of performance (2014)
6. European Committee For Standardization: CEN/TR 13201:3 Road lighting - Part 4: Methods of measuring lighting performance (2014)
7. Ernst, S.: Graph-supported preparation of GIS machine learning datasets, Monografie Politechniki Łódzkiej (2023). https://doi.org/10.34658/9788366741928.15
8. European Transport Safety Council (ETSC): Ranking EU Progress on Road Safety: 19th Road Safety Performance Index (PIN) Report, 2025
9. Ernst, S., Łabuz, M., Środa, K., Kotulski L.: Graph-based spatial data processing and analysis for more efficient road lighting design, Sustainability (2018). https://doi.org/10.3390/su10113850

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29909-3_13

10. Estaji, A., Sauter, T.: Street Lighting Simulation for Energy Efficiency Improvement, International Conference on Emerging Technologies and Factory Automation - ETFA (2022). https://doi.org/10.1109/ETFA52439.2022.9921570
11. Illuminating Engineering Society (IES): ANSI/IES RP-8-25: Recommended Practice: Lighting Roadway and Parking Facilities (2025)
12. Jaskierny, L., Kotulski L.: A self-adapting IoT network configuration supported by distributed graph transformation, Applied Sciences (Basel) (2023). https://doi.org/10.3390/app132312718
13. Komnata, K., Basiura, A., Kotulski, L.: Graph-Based Street Similarity Comparing Method., Theory and Applications of Dependable Computer Systems DepCoS-RELCOMEX (2020). https://doi.org/10.1007/978-3-030-48256-5_36
14. Kotulski, L.: Distributed Graph Transformations, Theory and Applications, Wyd. Naukowe AGH (2013). ISBN 987-83-7464-630-7
15. Nelson, M.A., Anderson, B.P., Cai, H.: Selection methods and procedure for evaluation of LED roadway luminaires, Leukos (2017). https://doi.org/10.1080/15502724.2016.1256782
16. Rabaza, O., Peña-García, A., Pérez-Ocón, F., Gómez-Lorente, D.: A simple method for designing efficient public lighting, based on new parameter relationships, Expert Systems with Applications (2013). https://doi.org/10.1016/j.eswa.2013.07.037
17. Rabaza, O., Gómez-Lorente, D., Pérez-Ocón, F., Peña-García, A.: A simple and accurate model for the design of public lighting with energy efficiency functions based on regression analysis, Energy (2016). https://doi.org/10.1016/j.energy.2016.04.078
18. Sędziwy, A., Basiura, A., Wojnicki, I.: Roadway lighting retrofit: environmental and economic impact of greenhouse gases footprint reduction, Sustainability Special Issue: "Sustainable Lighting and Energy Saving" (2018). https://doi.org/10.3390/su10113925
19. Sędziwy, A., Kotulski, L., Basiura A.: Multi-agent support for street lighting modernization planning, Intelligent information and database systems : Asian conference ACIIDS (2019). https://doi.org/10.1007/978-3-030-14799-0_38
20. Sędziwy, A., Kotulski, L.: Graph-Based Optimization of Energy Efficiency of Street Lighting, Artificial Intelligence and Soft Computing - ICAISC (2015). https://doi.org/10.1007/978-3-319-19369-4_46
21. Peña-García, A., Sędziwy A.: Optimizing lighting of rural roads and protected areas with white light: A compromise among light pollution, energy savings, and visibility, Leukos (2020). https://doi.org/10.1080/15502724.2019.1574138
22. Wojnicki, I., Ernst, S., Kotulski, L.: Economic impact of intelligent dynamic control in urban outdoor lighting, Energies (2016), https://doi.org/10.3390/en9050314
23. Wojnicki, I., Kotulski, L.: Improving control efficiency of dynamic street lighting by utilizing the dual graph grammar concept, Energies (2018). https://doi.org/10.3390/en11020402

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29909-3_13