

HYPERGRAPH FORMALISM FOR FUZZY SIGNATURE-BASED ROBOT ENVIRONMENT REPRESENTATION

Ahmet Mehmet Karadeniz^{1,*}, Csaba Hajdu², Danuta Rutkowska³, László T. Kóczy⁴

¹*Doctoral School of Multidisciplinary Engineering Sciences, Szechenyi István University, Egyetem Tér 1., 9026 Győr, Hungary*

²*Department of Informatics, Szechenyi István University, Egyetem Tér 1., 9026 Győr, Hungary*

³*Information Technology Institute, University of Social Sciences, 90-113 Lodz, Poland*

⁴*Department of Informatics, Szechenyi István University, Egyetem Tér 1., 9026 Győr, Hungary*

**E-mail: karadeniz.ahmet.mehmet@sze.hu*

Submitted: 11th April 2025; Accepted: 30th July 2025

Abstract

This paper introduces a pioneering approach to robot environment representation by integrating a novel hypergraph-based method for modeling fuzzy signatures with a quadtree-like structure for obstacle detection. By structuring fuzzy signatures through hypergraphs, we establish a robust framework that not only streamlines information representation but also simplifies the aggregation-based decision-making process. This synergy is applied to the domain of mobile robotics, where accurate and efficient environment representation is essential. Utilizing a quadtree-like structure for data organization, our technique systematically evaluates feature points against a set of fuzzy operations, determining the significance of obstacles and reconstructing the environmental model through the traversal of the quadtree-like structure. Furthermore, the hypergraph-based formalism sets the stage for a future transition to a tensor-based representation of fuzzy signatures, as envisioned in future work.

Keywords: decision making; environment representation; fuzzy signatures; fuzzy modeling; hypergraphs; robotics

1 Introduction

Robotic environment representation constitutes a fundamental challenge in autonomous navigation systems [1], requiring the simultaneous handling of spatial precision and uncertainty modeling. Traditional approaches like occupancy grids

[2] and geometry-based representations face limitations in dynamic environments due to their computational complexity and inability to capture multi-dimensional feature relationships. While the integration of fuzzy signatures with Quadrees [3] introduced adaptive resolution and uncertainty-aware encoding, critical gaps remain in modeling complex

feature interactions across multiple spatial scales.

The evolution of environment representation paradigms has progressed from uniform grid maps [4] with quadratic collision-checking complexity to adaptive quadtree structures. Subsequent integration of fuzzy signatures enabled tree-structured aggregation of multiple sensor modalities. However, these approaches remain constrained by binary parent-child relationships in Quadtrees and fixed aggregation rules in traditional fuzzy inference systems.

This work presents a hypergraph formalism that fundamentally extends our previous quadtree-based fuzzy signature framework [3], based on the synergies between fuzzy signatures and hypergraphs [5]. By modeling environment features (obstacles) as hypernodes and sections of the overall environment as hyperedges, we overcome critical limitations of prior approaches: restricted relationship modeling through n-ary connections and static aggregation rules via context-sensitive hyperedge weights.

2 Methodology

In this section, Fuzzy Signatures, Quadtrees, Hypergraphs, and the synergies between Fuzzy Signatures and Hypergraphs through their specific features are introduced, respectively.

2.1 Fuzzy Signatures

Fuzzy Signatures offer a hierarchical structure for representing multidimensional data with inherent uncertainties, effectively capturing interrelated subsets. These subsets work together to define attributes of higher complexity or level [6]. Figure 1 showcases an example of the nested graph structure characteristic of a standard Fuzzy Signature.

Fuzzy Signatures can be depicted in two ways: as nested vectors or as graphical structures. The nested vector format, which corresponds to the hierarchical graph representation of Fuzzy Signatures shown in Figure 1, is detailed in Equation (1). In this representation, the first level is denoted by the transposed vector $[x_1 \ x_2 \ x_3]^T$.

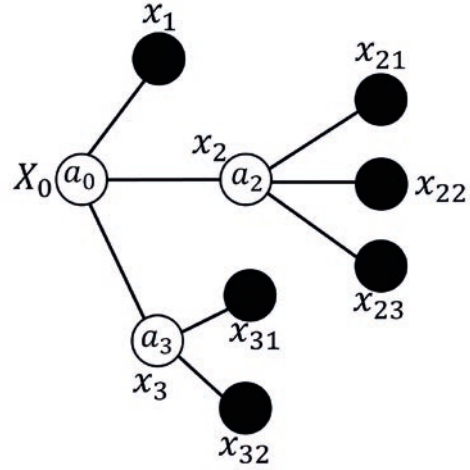


Figure 1. Fuzzy signature graph

$$X_0 = a_0 \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = a_0 \begin{bmatrix} x_1 \\ a_2 \begin{bmatrix} x_{21} \\ x_{22} \\ x_{23} \end{bmatrix} \\ a_3 \begin{bmatrix} x_{31} \\ x_{32} \end{bmatrix} \end{bmatrix} \quad (1)$$

The structure of the fuzzy signatures, presented in Equation (1), shows that the elements $[x_{21} \ x_{22} \ x_{23}]$ are combined by the aggregation a_2 and result in the component x_2 , and the elements $[x_{31} \ x_{32}]$ are combined by the aggregation a_3 to form x_3 . These elements are integrated using specific fuzzy aggregation operators, the choice of which is critical for the effective combination of data within the fuzzy signatures framework. This is especially true in situations where certain details of variables (represented as graph nodes or leaf nodes) might be missing in the subsets. The *min* and *max* functions are among the most common aggregation operators applied to fuzzy signatures, known for being the primary t-norm and s-norm operators, respectively, along with the weighted generalized mean [7].

To illustrate this concept, a numerical example of the fuzzy signature discussed is provided in Equation (2). In this example, it is assumed that the *max* aggregation operators are used at the deeper levels, while the *min* aggregation operator is applied at the first level.

$$X_0 = \min \left[\begin{array}{c} 0.9 \\ \max \left[\begin{array}{c} 0.6 \\ 0.4 \\ 0.8 \end{array} \right] \\ \max \left[\begin{array}{c} 0.5 \\ 0.7 \end{array} \right] \end{array} \right] \quad (2)$$

The first level of the fuzzy signature hierarchy can be determined, as shown in Equation (3), by applying the *max* aggregation operator to the branches:

$$X_0 = \min \left[\begin{array}{c} 0.9 \\ 0.8 \\ 0.7 \end{array} \right] \Rightarrow [0.7] \quad (3)$$

Considering the graph representation of the provided numerical example, it is visualized as shown in Figure 2.

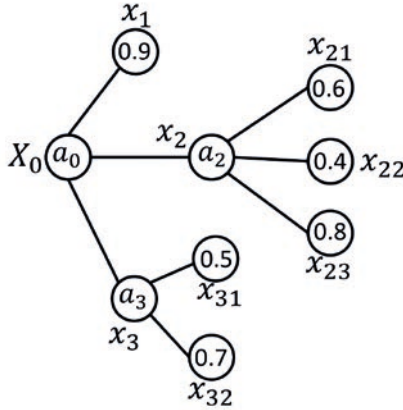


Figure 2. Graph representation of the given example.

In the fuzzy signatures graph shown in Figure 2, the fuzzy aggregation operators are denoted as a_0 , a_2 , and a_3 . The operator a_2 combines $x_{2,1}$, $x_{2,2}$, and $x_{2,3}$ to form x_2 , while a_3 merges $x_{3,1}$ and $x_{3,2}$ to produce x_3 . Both a_2 and a_3 utilize *max* aggregation operators.

Through the application of these fuzzy aggregation processes on the deeper-level components, we can extrapolate the values for the higher levels. This hierarchical computation culminates in the root-level representation of the fuzzy signature graph, as shown in Figure 3. Here, a_0 is the *min* aggregation operator, which determines the final root-level value of the resulting membership degree (X_0) for the fuzzy signature graph.

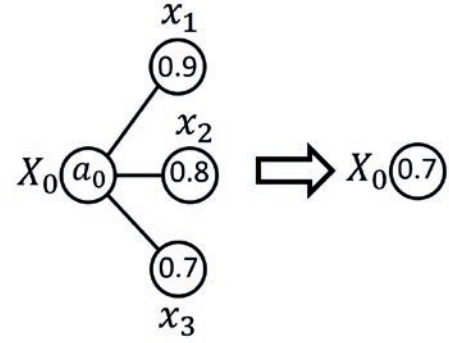


Figure 3. Obtained root level graph representation and the resulting membership degree.

2.2 Quadrees

Quadrees [8], which are hierarchical data structures, effectively capture two-dimensional spatial information. These structures have proven invaluable in disciplines such as image processing and shape reconstruction, offering the typical advantages of tree-based data structures and facilitating efficient execution of operations like interval and point searches. It is essential to note that in a quadtree's hierarchical layout, each node consistently branches into four child nodes. Quadrees are generally bounded within a limited spatial domain, which is recursively divided into four smaller sections. This quadtree concept can be extended to three-dimensional contexts, resulting in structures known as octrees [9, 10], where each node has eight child nodes.

The fundamental algorithmic approach involves dividing the space into distinct segments. Points are assigned to a specific segment based on their spatial relation to the segment's origin and the new point being added. Subsequent points are systematically placed into the correct subsegment as determined by their position. This insertion process is repeated until a specified depth level within a subsegment is reached. Unlike traditional Quadrees that divide space into equally sized quadrants, Figure 4 depicts a quadtree created using a set of points, clearly showing the spatial partitioning relative to the distribution of the points.

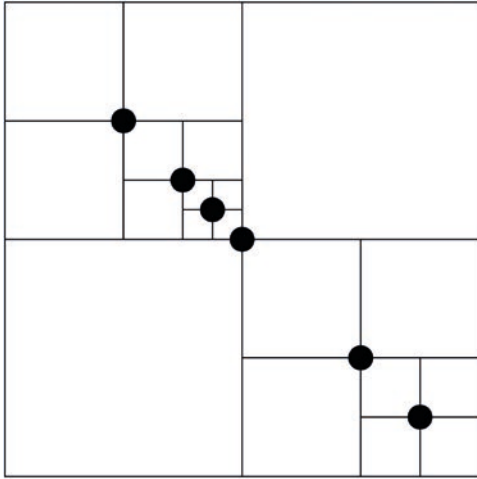


Figure 4. Quadtree example with points inserted

Imagine a quadtree in which each node corresponds to a distinct segment of space. Initially, the quadtree is a blank slate, with just one node encompassing the entire space. As new points are added, the quadtree progressively subdivides the space into increasingly smaller sections.

The visual depiction in Figure 4 illustrates a collection of nested squares or rectangles. Each of these is further divided into four smaller quadrants, and within the smallest possible quadrant, points (indicated by dots) are placed.

2.3 Hypergraphs

Hypergraphs $H(V;E)$ [11] extend the concept of traditional graphs, consisting of *hypernodes* V and *hyperedges* $E = (e_i)_{i \in I}$ indexed by a finite set I . These structures permit edges to encompass relationships among multiple nodes, thereby representing many-to-many connections. In essence, a hyperedge is a collection that includes vertices as its members. An illustrative hypergraph with 7 vertices and 4 hyperedges is depicted in Figure 5.

Directed hypergraphs, or *dirhypergraphs*, introduce directionality to the relationships between set elements in a hypergraph, denoted as $\vec{H} = (V; \vec{E} = \{\vec{e}_i : i \in I\})$, where I is a finite index set. Here, $\vec{E}$ represents a collection of *hyperarcs*. Each hyperarc $\vec{e}_i$ within $\vec{E}$ corresponds to an ordered pair $\vec{e}_i = (\vec{e}_i^+ = (e_i^+, i); \vec{e}_i^- = (e_i^-, i))$. Directed hyperedges differentiate the connected nodes into two distinct groups, or 'limbs':

- $e_i^+ \subseteq V$ of vertices of $\vec{e}_i^+$, representing the *tail* of the hyperarc.
- $e_i^- \subseteq V$ of vertices of $\vec{e}_i^-$, representing the *head* of the hyperarc.
- Ensure that all $\vec{e} = (e^+, e^-) \in \vec{E}$ and $e^+ \cap e^- = \emptyset, e^+ \neq \emptyset, e^- \neq \emptyset$

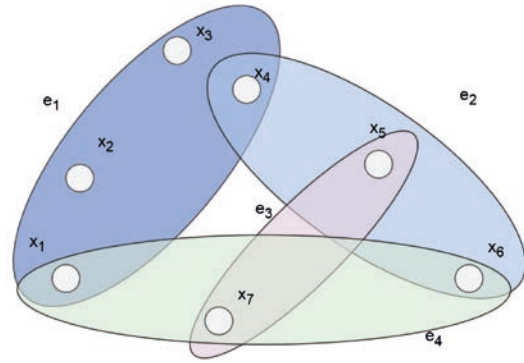


Figure 5. Example of a hypergraph

An example of a dirhypergraph is depicted in Figure 6, with 9 hypernodes, 3 dirhyperedges.

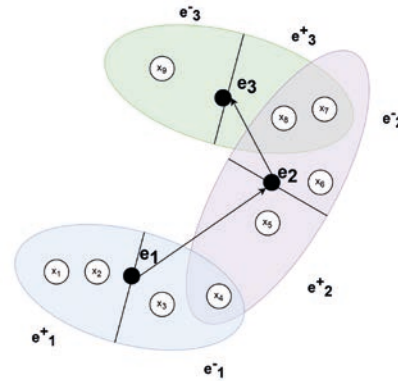


Figure 6. An example of directed hypergraph

2.4 Fuzzy decision making

Decision-making using fuzzy inference systems entails the mathematical manipulation of fuzzy linguistic values L , which result from the fuzzification of typically observed crisp values, in line with the principles of conventional logic:

- *Fuzzy rule:* This defines a relation (**if** ... **→ then** ...) by utilizing a membership function that maps a set of crisp values to a *fuzzy*

linguistic value. Membership functions typically operate within normalized domains, indicating the degree to which a value is associated with a fuzzy linguistic term. Common shapes for membership functions are triangular, bell-shaped, and trapezoidal. Instead of using standard logical conjunctions or disjunctions, a rule may employ an S-norm or T-norm.

- *S-norm operations*: These combine information in a non-exclusive manner (disjunction, $\vee$), analogous to the union in set theory or the logical **OR** operation in Boolean logic.
- *T-norm operations*: These handle information in an exclusive fashion (conjunction, $\wedge$), akin to the intersection in set theory or the logical **AND** operation in Boolean logic.

In summary, standard fuzzy decision-making processes formulate rules in the following format (where μ_i represents a membership function from the collection $\bigcup_{i=1}^n \mu_i$) and crisp (or fuzzy) input values $X \in \mathbb{R}^n$: $\text{if } \mu_1(X) \vee \mu_2(X) \dots \mu_n(X) \rightarrow L$. This structure essentially states that if the input X satisfies the conditions defined by the membership functions $\mu_1, \mu_2, \dots, \mu_n$ to a certain degree, then the output is the fuzzy linguistic value L .

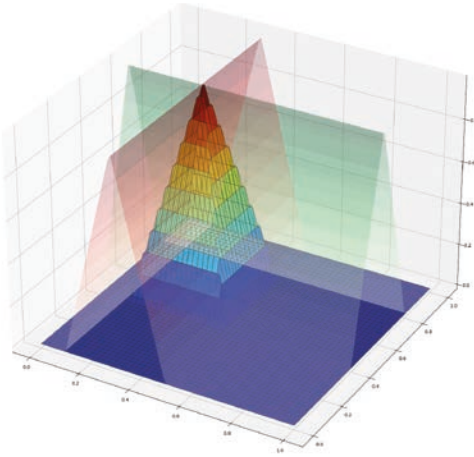


Figure 7. Cylindrical extension of projected triangular terms of one rule.

When dealing with crisp inputs that are typically presented in a vector format, and logical operations that necessitate the Cartesian product of a set of rules, the decision-making process often involves the projection to yield the feasible set between $\mathcal{X}$ and $\mathcal{Y}$. This is represented as $[R \downarrow \mathcal{Y}]$,

with the relation R on $\mathcal{Y}\{X_j | j \in J \subset \mathbb{N}_n\}$ and the combined rule definition of the smallest set of interpretable outcomes. The *cylindrical extension* $[R \uparrow \mathcal{X} - \mathcal{Y}](x) = R(y)$ creates the projection that is minimal for the given set of rules. The concept of the cylindrical extension of one rule (and correspondingly the projection of the one rule) is illustrated in Figure 7.

3 Robot Environment Representation Approach

The robot environment representation method integrates spatial data structures, particularly Quadrees, with fuzzy systems-based techniques to represent environmental characteristics, resulting in a comprehensive and extensive framework. The approach is influenced by Fuzzy Situational Maps (FSM) [12], which partition the robot's working space into zones of varying interest and granularity. The aim is to systematically identify obstacle points and associate them with fuzzy values, obtained from rule sets that evaluate the significance of each point.

Upon quadtree construction, each node contains a 4-tuple $(x, y, F_x(x), F_y(y))$, which includes the spatial coordinates (x, y) and the fuzzy sets $(F_x(x), F_y(y))$ evaluated along the X and Y axes, respectively. Optionally, the outcome of the defuzzification process can be included in each node to support decision-making. Nodes may also hold information about the membership functions and their parameters to improve inference capabilities.

For the purposes of this discussion, the origin of the quadtree-like structure is consistently placed at the robot's central position.

To convert uncertain node coordinates into fuzzy sets along each axis, fuzzy operations (aggregation operators) for F_x and F_y are defined, with their corresponding membership functions depicted in Figure 8. In this study, the *product* aggregation operator is used for both axis. Each fuzzy set conveys a specific meaning:

- *Negative Far (NF)*: Represents a scenario where the attribute in question is located far from the robot on the negative side of the axis, indicating no immediate threat.

- *Negative Near (NN)*: Indicates that the attribute is relatively close to the robot on the negative side of the axis, presenting a considerable risk.
- *Danger (D)*: Suggests that the attribute is dangerously close to the robot, requiring swift action to prevent a collision.
- *Positive Near (PN)*: Similar to NN, but on the positive side of the axis, indicating a significant risk.
- *Positive Far (PF)*: Similar to NF, but on the positive side of the axis, suggesting no immediate danger.

These interpretations assist in evaluating the proximity and potential threat posed by various attributes in the robot's environment.

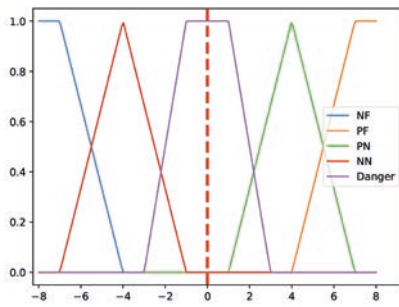


Figure 8. Fuzzy membership functions associated with axes signatures

The procedure for constructing an environmental representation using a set of two-dimensional obstacle points, referred to as P , is outlined in the following steps:

1. Begin by setting up the quadtree q to cover the designated boundary area.
2. Iterate over each point p within the set P and subdivide the space accordingly, positioning p at the center of the newly created sector. Determine the fuzzy sets for p and store them along with the point's coordinates. Adapt the obstacle's membership function to reflect its characteristics. Initialize each node with four child nodes, which have predetermined subdivided boundaries but no assigned values yet.
3. Once a sector reaches its maximum depth limit, update the fuzzy signature value for that

node. Adjust the membership functions if needed—this may occur when encountering a new obstacle with more pronounced attributes or based on the outcomes from the defuzzification process.

Through these steps, the quadtree is populated with nodes that represent the spatial distribution of obstacles, each node containing fuzzy values that provide a nuanced description of environmental features relative to the robot's position.

Figure 9 typically illustrates an example of a quadtree-like Fuzzy Situational Map developed using the method described above. The inferential process can leverage the tree structure of the created representation. For example, when generating a grid or utilizing points from a Voronoi diagram, each point can be evaluated through the defuzzification process applied to the tree. This enables the extraction of specific, actionable information from the fuzzy environmental model, facilitating decision-making processes such as navigation [13] or obstacle avoidance [14] for a robot or autonomous system.

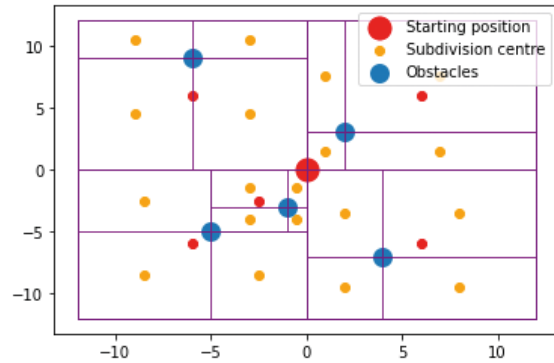


Figure 9. Visualization of quadtree-like fuzzy situational map subdivisions for environmental representation. The red circle marks the robot's starting position, blue circles indicate obstacle points, orange circles represent the centers of leaf-node subdivisions, and purple rectangles show the boundaries of each depth level within the quadtree structure.

Considering the methodologies outlined earlier and the introduction of the novel concept, the fusion of fuzzy signatures with a quadtree-based representation of the environment presents a compelling strategy that balances computational efficiency with effective dimensionality management.

4 Hypergraph Modeling of the Approach

Utilizing the HyMeKo description framework—based on LALR and developed by one of the authors to delineate hypergraphs¹—a prototype of a fuzzy signature example can be articulated. Key aspects of describing a fuzzy signature with this language include:

- Defining fuzzy concepts such as T-norms and S-norms within a metaelement description.
- Organizing information value distinctly, with the capability to include multidimensional values in each node.
- Structuring edges to align with the architecture of the fuzzy signature.
- Specifying fuzzy aggregation operations as stereotypes, with edges adapting these stereotypes to represent aggregations via S-norms (for disjunctions) or T-norms (for conjunctions).
- Linking hyperedges to informational nodes to illustrate the sequence of the fuzzy aggregation process.

Figure 10 illustrates a representation of a fuzzy signature using a hypergraph. In this representation, e and e_3 denote hyperedges where aggregation is performed by specific aggregation operators. R_{env} represents the root-level hypernode, while S_1 , S_2 , S_3 , and S_4 represent first-level (child) hypernodes. Each hypernode contains relevant information about its associated section of the robot environment. Since the third section contains more obstacles and therefore requires finer resolution, S_3 is further divided into four sub-hypernodes: S_{31} , S_{32} , S_{33} , and S_{34} . These sub-hypernodes hold more detailed information about the corresponding section of the robot environment. In this example, the *max* aggregation operator is chosen for the e_3 hyperedge, while the *min* aggregation operator is chosen for the e hyperedge. In this hypergraph representation, the decision-making process is conducted as follows:

1. Begin by gathering information from all leaf nodes, starting from the root hyperedge and working upwards in a bottom-up fashion.

2. Utilize the leaf hyperedges to carry out the corresponding fuzzy aggregation process. This is achieved by applying the cylindrical extension to the rules associated with the leaf or intermediate variables.
3. Record the derived fuzzy value by either updating the weight of the hyperedge or adding a new attribute to the graph.
4. Proceed to the next level, using the newly obtained intermediate fuzzy values as inputs for subsequent aggregation operations, which may involve T-norms or S-norms, until reaching the root hyperedge.
5. When necessary, convert the ultimate fuzzy value into a crisp value through the defuzzification process.

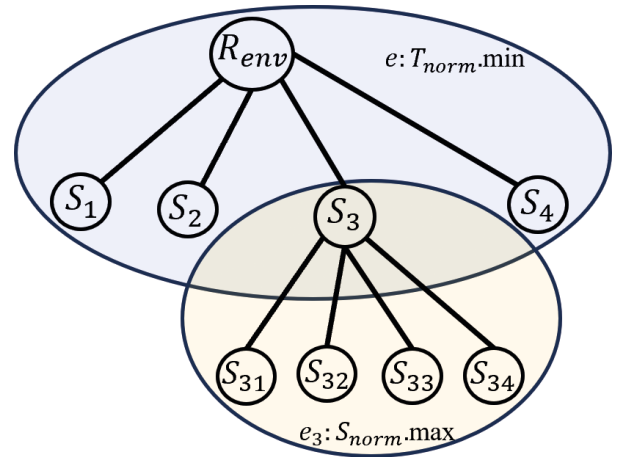


Figure 10. Representation of a Robot's Environment Using Fuzzy Signatures Modeled as Hypergraphs

The synergy between hypergraphs and fuzzy signatures opens up the potential for representing fuzzy signatures within a tensor-based framework. Recent studies have explored the concept of modeling hypergraphs as tensors [15], with practical applications also being investigated [16]. This approach could facilitate the integration of data derived from fuzzy signatures into neural networks, which typically rely on vector-based inputs and outputs.

The method for describing the robot's environment, represented as hypergraphs, is illustrated in

¹https://github.com/kyberszittya/himeko_lang

Figure 10 and detailed in Listing 4. Here, the node x may optionally contain data beyond mere numerical values. It is noteworthy that the attribute values of leaf nodes can encompass multi-dimensional data, such as S_1, S_2, S_{31}, S_{32} . Observe the repeated use of meta elements in the hyperedges and how the information is clearly distinguished from the fuzzy signature itself.

In Listing 4, it can be observed that the *min* operation is applied at hyperedge e , while the *max* operation is applied at hyperedges e_1, e_2, e_3 , and e_4 .

Listing 1. Hypergraph description of the fuzzy signature for the robot environment representation

```
[fuzzy_signature_example import
  ↪ "fuzzy_signature_meta.himeko"]
fuzzy_1 use fuzzy_signature.tnorm
  ↪ use fuzzy_signature.snorm
{
  R_env {
    name "x"
    description "Input variable"
    ↪ "x"
    S1 { value [S1] }
    S2 { value [S2] }
    S3 {
      value [S31, S32, S33,
        ↪ S34]
    }
    S4 { value [S4] }
  }
  @e: min {
    — x.S1, — x.S2, — x.S3,
    ↪ — x.S4,
    @e1: max {— x.S1},
    @e2: max {— x.S2},
    @e3: max {— x.S3},
    @e4: max {— x.S4},
  }
}
```

Conclusions

This paper proposes a hypergraph-based model for a novel approach to robot environment representation [17], which provides a general structure for organizing information and incorporates fuzzy meta-information to support contextual decision-making processes. The approach introduces a new method for representing environments specifically designed for mobile robots, combining fuzzy signa-

tures with Quadrees to create quadtree-like Fuzzy Situational Maps. This method effectively encapsulates sparse data related to obstacles and geometric features.

The goal is to integrate general ontology processes and traditional information representation with fuzzy signatures, creating a unified approach. The effectiveness of this combined method is demonstrated through a grid-based example, which informs a general path planning algorithm. The results include a mathematical definition and a description of the robot environment representation approach using a hypergraph-based language. The paper also highlights the relationship between fuzzy signatures and their potential tensor-based representation for further algebraic analysis.

Overall, the proposed method demonstrates promising results for mobile robot environment representation and opens up numerous opportunities for enhancement, such as tensor-based representations that could later be integrated with machine learning and metaheuristic algorithms [18]. This innovative approach merges two powerful techniques, offering a robust and efficient solution for complex environmental representation tasks. Consequently, it could significantly benefit motion planning and navigation algorithms [19], as well as transportation-related problems [20, 21].

Future work may include exploring additional use cases for the introduced hypergraph representation, such as robot applications [22] involving knowledge sharing between agents and sensor fusion (e.g., for IMU data) or environment mapping tasks [23] that incorporate multiple sensory inputs. Further research is also planned to generalize hypergraphs for handling more complex structures and to explore broader application areas. Additionally, the possibility of representing fuzzy signatures as sparse tensors could be investigated, given the potential of tensor-based representations of hypergraphs [16, 24]. Such a tensor-based representation could help reduce dimensionality, thereby decreasing computational load and increasing processing speed.

References

- [1] N. Promkaew, S. Thammawiset, P. Srisan, P. Sanitchon, T. Tummai, and S. Sukpancharoen, "De-

- velopment of metaheuristic algorithms for efficient path planning of autonomous mobile robots in indoor environments,” *Results in Engineering*, vol. 22, p. 102280, 2024.
- [2] A. Rogers, K. Eshaghi, G. Nejat, and B. Benhabib, “Occupancy grid mapping via resource-constrained robotic swarms: A collaborative exploration strategy,” *Robotics*, vol. 12, no. 3, 2023.
 - [3] A. M. Karadeniz, C. Hajdu, L. T. Kóczy, and Á. Ballagi, “Robot environment representation based on Quadtree organization of Fuzzy Signatures,” in 2021 IEEE 15th International Symposium on Applied Computational Intelligence and Informatics (SACI), pp. 509–514, 2021.
 - [4] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics (Intelligent Robotics and Autonomous Agents)*. The MIT Press, 2005.
 - [5] C. Hajdu, A. M. Karadeniz, and L. T. Kóczy, “Synergies between fuzzy signatures and hypergraphs,” in 2025 IEEE 23rd World Symposium on Applied Machine Intelligence and Informatics (SAMI), pp. 495–500, 2025.
 - [6] C. Pozna, N. Minculete, R.-E. Precup, L. T. Kóczy, and Á. Ballagi, “Signatures: Definitions, operators and applications to fuzzy modelling,” *Fuzzy Sets and Systems*, vol. 201, pp. 86–104, 2012.
 - [7] L. T. Kóczy, M. E. Cornejo, and J. Medina, “Algebraic structure of fuzzy signatures,” *Fuzzy Sets and Systems*, vol. 418, pp. 25–50, 2021.
 - [8] R. A. Finkel and J. L. Bentley, “Quad trees a data structure for retrieval on composite keys,” *Acta Informatica*, vol. 4, no. 1, pp. 1–9, 1974.
 - [9] M. Donald, “Geometric modeling using octree encoding,” *Computer Graphics and Image Processing*, vol. 19, no. 2, pp. 129–147, 1982.
 - [10] J. Vörös, “Low-cost implementation of distance maps for path planning using matrix quadrees and octrees,” *Robotics and Computer-Integrated Manufacturing*, vol. 17, no. 6, pp. 447–459, 2001.
 - [11] A. Bretto, *Hypergraph Theory: An Introduction*. Springer Publishing Company, Incorporated, 2013.
 - [12] Á. Ballagi, C. Pozna, P. Földesi, and L. T. Kóczy, “Fuzzy situational maps: A new approach in mobile robot cooperation,” in *Proceedings of 2013 IEEE 17th International Conference on Intelligent Engineering Systems (INES)*, (San Jose, Costa Rica), pp. 287–292, 2013.
 - [13] E. Szmídt, J. Kacprzyk, P. Bujnowski, J. T. Starczewski, and A. Siwocha, “Ranking of alternatives described by atanassov’s intuitionistic fuzzy sets – reconciling some misunderstandings,” *Journal of Artificial Intelligence and Soft Computing Research*, vol. 14, no. 3, pp. 237–250, 2024.
 - [14] L. Wang and H. Wei, “Bending path understanding based on angle projections in field environments,” *Journal of Artificial Intelligence and Soft Computing Research*, vol. 14, no. 1, pp. 25–43, 2024.
 - [15] X. Ouvrard, J.-M. L. Goff, and S. Marchand-Maillet, “Adjacency and Tensor Representation in General Hypergraphs Part 1: e-adjacency Tensor Uniformisation Using Homogeneous Polynomials,” *CoRR*, vol. abs/1712.08189, 2017. arXiv: 1712.08189.
 - [16] C. Hajdu and Á. Csapó, “Tensor-based Format for Exchanging Hypergraphs between Cognitive Entities,” in 2022 IEEE 1st International Conference on Internet of Digital Reality (IoD), pp. 33–38, June 2022.
 - [17] A. M. Karadeniz, C. Hajdu, and L. T. Kóczy, “Mobile robot environment representation through fuzzy signatures-integrated quadrees,” *Romanian Journal of Information Science and Technology*, vol. 28, no. 1, pp. 103–116, 2025.
 - [18] Y. Dou, C. Jin, Y. Dou, H. Fan, and X. Zhang, “Hybrid coyote optimization with differential evolution and its application to the estimation of solar photovoltaic cell parameters,” *Journal of Artificial Intelligence and Soft Computing Research*, vol. 15, no. 4, pp. 385–412, 2025.
 - [19] Y. Liu, G. Zeng, X. Du, K. Fang, and J. Yu, “Navigated locomotion and controllable splitting of a microswarm in a complex environment,” in 2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 1337–1342, 2024.
 - [20] M. Zajecka, M. Mastalerczyk, S. Y. Chong, X. Yao, J. Kwiecien, W. Chmiel, J. Dajda, M. Kisiel-Dorohinicki, and A. Byrski, “Portfolio optimization with translation of representation for transport problems,” *Journal of Artificial Intelligence and Soft Computing Research*, vol. 15, no. 1, pp. 57–75, 2025.
 - [21] L.-J. Liu, G.-M. Bai, and H. R. Karimi, “Advanced traffic signal control system using deep double q-learning with pedestrian factors,” *Journal of Artificial Intelligence and Soft Computing Research*, vol. 15, no. 3, pp. 239–255, 2025.
 - [22] N. F. Koçak and A. Saygın, “Real-time laser weld point seam tracking system for robotic welding,” *Journal of Computer & Electrical and Electronics Engineering Sciences*, vol. 3, no. 1, pp. 29–34, 2025.

- [23] J. Niedzwiedzki, P. Lipinski, and L. Podsedkowski, "Idtmm: Incremental direct triangle mesh mapping," *IEEE Robotics and Automation Letters*, vol. 8, no. 9, pp. 5416–5423, 2023.

- [24] F. Wang, K. Pena-Pena, W. Qian, and G. R. Arce, "T-hypergnns: Hypergraph neural networks via tensor representations," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 36, no. 3, pp. 5044–5058, 2025.



Ahmet Mehmet Karadeniz completed his B.Sc. in Mechatronics Engineering at Kocaeli University in 2017 and his M.Sc. in the same field at the University of Debrecen in 2019. He then began his Ph.D. studies at Széchenyi István University, where he is currently conducting research on robot environment representation. His research interests include Machine Learning, Deep

Learning, Computational Intelligence, Fuzzy Systems, and Systems Engineering.

<https://orcid.org/0000-0002-6851-6033>



Csaba Hajdu completed his B.Sc. and M.Sc. in Computer Science at Budapest University of Technology and Economics. He then began his Ph.D. studies at Széchenyi István University, where he is serving as a Teaching/Research Associate and currently conducting research on Artificial Intelligence, Cognitive Robotics and Hypergraphs

Framework. His research interests include Deep Learning, Cognitive Agents, and Corresponding Robots.

<https://orcid.org/0000-0002-1058-7525>



Danuta Rutkowska is a professor of computer science. She graduated from Wrocław University of Science and Technology, Wrocław, Poland, from where she also received her Ph.D. in automation and then D.Sc. in computer science. In 2002 she was given the title of professor conferred by the President of Poland. She is an author or co-author

of numerous publications mostly in computational intelligence, including several books, book chapters, and many scientific papers. Her research interests are in the area of artificial intelligence, especially computational intelligence, focusing on artificial neural networks, fuzzy systems, genetic/evolutionary algorithms, as well as hybrid intelligent systems, i.e. neuro-fuzzy or genetic-neuro-fuzzy systems, and their applications. After working for many years at the Częstochowa University of Technology, Częstochowa, Poland, she is currently a professor in the Institute of Information Technology at the University of Social Sciences in Łódź, Poland. Since 2022 she has been serving as vice-rector for science at this university.

<https://orcid.org/0000-0003-0217-2589>



László T. Kóczy received the M.Sc., M.Phil. and Ph.D. degrees from the Technical University of Budapest (BME) in 1975, 1976, and 1977, respectively; and the D.Sc. degree from the Hungarian Academy of Science in 1998. He spent his career at BME until 2001 and from 2002 at Széchenyi István University (Győr, SZE). He has

been a visiting professor in Australia, Japan, Korea, Austria, Italy, etc. His research interests are fuzzy systems, evolutionary and memetic algorithms, and neural networks, as well as applications in infocommunications, logistics, management, and others. In the last years, he has focused on NP-complete problems, especially route selection and optimization and the application of metaheuristics for approximate solution of such complex tasks. He has published over 800 articles, most of those being refereed papers, and several text books on the subject. His Hirsch-index is 44 by Google Scholar (based on 9600 citations there).

<https://orcid.org/0000-0003-1316-4832>