ROMANIAN JOURNAL OF INFORMATION SCIENCE AND TECHNOLOGY
Volume 28, Number 1, 2025, 103–116

Mobile Robot Environment Representation Through Fuzzy Signatures-Integrated Quadtrees

Ahmet Mehmet KARADENIZ[1,*], Csaba HAJDU[2], and László T. KÓCZY[3]

[1]Doctoral School of Multidisciplinary Engineering Sciences, Széchenyi István University, Györ, Hungary
[2,3]Department of Informatics, Széchenyi István University, Györ, Hungary
Email: karadeniz.ahmet.mehmet@sze.hu*, hajdu.csaba@ga.sze.hu, koczy@sze.hu
* Corresponding author

Abstract. This paper presents an innovative environment representation technique for mobile robots, incorporating obstacle detection within their operational space. Leveraging the fuzzy signature method, this approach uses quadtrees for efficient data organization. A set of fuzzy rules evaluates feature points to ascertain the relevance of identified obstacles. These points and their fuzzy associations are systematically arranged using a quadtree structure. The environmental model is reconstructed by traversing this tree and applying the established fuzzy rules. This paper has achieved a high-resolution grid representation of 0.1m within a 20m×20m area. Notably, the inference operation completes in just 0.5 ms, underscoring the method's efficiency. Additionally, the technique is optimized for low memory consumption, demonstrating effective resource management even on older PCs, such as an Intel Core Duo 2 with 16 GB RAM. This representation is designed to support advanced robotic functions, such as obstacle navigation in a distributed computing environment.

Key-words: A* search algorithm; environment representation; fuzzy signatures; fuzzy systems; quadtrees; robotics.

1. Introduction

This paper presents a novel technique for mobile robot environment representation, leveraging fuzzy signatures and quadtrees for efficient obstacle detection within the operational space of the robot. The proposed method achieves rapid, high-resolution grid mapping with minimal memory usage, suitable for older computer systems. The potential to extend this representation into three-dimensional space is introduced, enhancing motion planning and navigation for autonomous robotic systems. These advancements represent a significant contribution to the field, offering a scalable and adaptable solution for complex robotic applications. The rest of the Introduction part is provided as a separate file in [1], and its references are shown in [2–4].

In the forthcoming sections, this paper will systematically explore the various aspects of the novel environment representation technique. Section 2 surveys the existing literature. Section 3 is a comprehensive exposition of the theoretical backgrounds of the proposed technique, detailing the intricacies of fuzzy signatures, fuzzy signature sets, fuzzy signature class, quadtrees and the A* graph-based pathfinding algorithm. Section 4 provides a synthesized view of the used algorithm, illustrating its application into mobile robots. Section 5 evaluates the performance and efficacy of the proposed technique through the results obtained. Section 6 concludes the paper, summarizing the key findings and discussing potential directions for future work aimed at further enhancing the capabilities of mobile robot systems.

2. Related Work

In the domains of robotics and vehicle engineering, a variety of methods exist for representing environments in both two and three dimensions. One such method is occupancy grid mapping, which offers a 2D depiction of the environment surrounding a given robot or vehicle. This technique is highly favored for its straightforward integration with mobile robots. Nonetheless, scalability poses a challenge, as the computational demands escalate with finer resolutions or larger map sizes. Consequently, occupancy grid mapping is predominantly utilized for representing static environments [5].

Moving to a different approach, graph-based simultaneous localization and mapping (SLAM), as detailed in [3], offers an alternative way to represent the environment. This method requires additional processes, including data filtering, to construct and optimize a map by correlating various sensor observations over time.

While some approaches prefer to maintain perceived data within a geometric framework, such as motion primitives [7] or probabilistic roadmaps [8], these are noted for their efficacy in managing large-scale environments. To improve feature indexing within these methods, spatial data structures like quadtrees and octrees are often utilized, offering a more sophisticated alternative to simple queues and arrays.

In the realm of navigation and localization for mobile robots, the adoption of advanced nonlinear modeling techniques such as dynamic Bayesian networks, recurrent neural networks, tensor product-based transformations and fuzzy systems based methodologies offer considerable potential [9–11]. These sophisticated modeling approaches enable the system to accurately represent complex cognitive processes, intricate spatial relationships and uncertain environmental conditions, which are crucial for navigating and understanding indoor spaces [12]. In order to integrate these diverse strategies for navigation and localization purposes, there is a need to develop a comprehensive and robust environment representation system that can effectively support tackling the complexities and challenges of real-time navigation and task execution in dynamic environments.

Concurrently, other research endeavors in path planning [13], indoor navigation [14] and positioning [15] have introduced flexible environment modeling and localization techniques. These studies have achieved considerable results, yet they highlight a development opportunity: the need for a more accurate representation of the environment to support task completion in autonomous systems.

On the other hand, fuzzy signatures have demonstrated remarkable success in diverse control scenarios within autonomous systems. Applications such as motion control in mobile robots [16] and autonomous vehicle steering angle control [17] have all benefited from the implementation of fuzzy signatures. Given their proven track record, fuzzy signatures could be the key to bridging the gap identified in the aforementioned studies. By incorporating fuzzy signatures into environment representation for mobile robots, there is potential to significantly enhance the capabilities of autonomous systems, ensuring more robust and reliable navigation and task execution.

3. Methodology

In the subsequent section, a comprehensive review of the research is conducted pertaining to the method introduced. An in-depth discussion of the foundational principles of fuzzy signatures is presented. Additionally, an overview of quadtrees is provided to establish the foundational concept behind the proposed approach.

3.1. Fuzzy signatures

Fuzzy signatures are considered a framework for representing symbolic data. By definition, fuzzy signatures are a distinct kind of multidimensional data structure where specific dimensions are interlinked, creating a group of variables that collectively define a complex feature [18]. The concepts of fuzzy signatures and fuzzy signature trees are founded on the following recursive definition of the set $S(n)$. Let R represent the set of real numbers. The set $S(n)$ is recursively established as follows as the Cartesian product of its elements S_i:

$$S(n) = \prod_{i=1}^{n} S_i \tag{1}$$

where each S_i is either the set of real numbers R for i ranging from 1 to n (denoted as $i \in 1, 2, \ldots, n$), or S_i is another set $S(m)$ with m being greater than or equal to 1. Here, $\prod$ denotes the Cartesian product operation.

Consider X as a nonempty set. The set of fuzzy signatures is then characterized by the function $A : X \to S(n)$. For any element $x \in X$, its fuzzy signature is $A(x)$, which is presented in the following nested vector form:

$$A(x) = \begin{bmatrix} \vdots \\ a_i \begin{bmatrix} a_{i+11} \\ a_{i+12} \end{bmatrix} \\ \begin{bmatrix} a_{i+21} \begin{bmatrix} a_{i+221} \\ a_{i+222} \end{bmatrix} \end{bmatrix} \\ \vdots \end{bmatrix} \tag{2}$$

The elements such as $a_1, a_2, \ldots, a_n, a_{i1}, a_{i2}, \ldots, a_{im}, \ldots, a_{jkl}, \ldots$ found within the fuzzy signatures defined in (2) are referred to as the values of the fuzzy signatures [19]. An example of a fuzzy signature in nested vectors form is:

$$A(x) = \begin{bmatrix} \begin{bmatrix} a_{11} \\ a_{12} \end{bmatrix} \\ \begin{bmatrix} a_{21} \\ \begin{bmatrix} a_{221} \\ a_{222} \end{bmatrix} \\ a_{23} \end{bmatrix} \\ a_3 \end{bmatrix} \quad (3)$$

and the corresponding structure of the fuzzy signature tree is depicted in Fig. 1.

In this structure, the subgroup $[a_{11}\ a_{12}]$ forms a composite variable at a higher level, representing a_1. Similarly, the elements $[a_{21}\ [a_{221}\ a_{222}]\ a_{23}]$ combine to constitute $[a_{21}\ a_{22}\ a_{23}]$, then a_2. Ultimately, this leads to the exemplary fuzzy signature being represented as $A(x) = [a_1\ a_2\ a_3]^T$.

The relationships between the various levels of the fuzzy signature are established through a series of fuzzy aggregation operations assigned to the intermediate nodes of the tree graph, including the root. The outcome of the parent fuzzy signature at each level is derived by appropriately aggregating the child fuzzy signatures from its branches. For instance, if $@_1$ is the aggregation operation that combines a_{11} and a_{12} to calculate a_1, then a_1 is obtained by $(a_{11})@_1(a_{12})$. As indicated in Fig. 1, the aggregations for the entire fuzzy signature structure in this example would involve $@_1$, $@_2$, $@_{22}$ and $@_0$. These aggregation operations, $@_1$, $@_2$, $@_22$ and $@_0$, could be different or the same. A simple choice for each could be the $minimum$ aggregation operator, a commonly used t-norm. Moreover, if the rule base of the fuzzy system is well-defined, it can guide the aggregation process. This process is crucial for merging several input values to yield a singular output value within the system. The demonstration of the operation will utilize the fuzzy signature values corresponding to the structure presented in the example, as shown in (3) and Fig. 1.

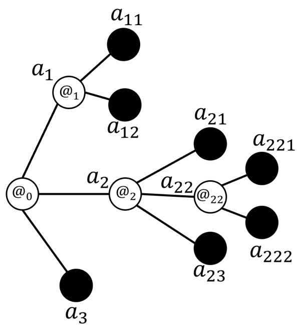

Fig. 1. Exemplary fuzzy signature tree.

Each fuzzy signature encapsulates information relevant to a particular data point, denoted as $A(x)$. As one ascends the structure of the fuzzy signature, the amount of detail encapsulated diminishes. In certain operations, it becomes necessary to summarize and amalgamate information derived from an alternative source, particularly when specific detail variables are absent or local details are omitted [20]. This need for condensation often occurs during the interpolation process within a fuzzy signature rule base, especially when the fuzzy signatures adjacent to a specific observation do not possess identical structures. Under such circumstances, it is imperative to identify the largest shared subtree and reduce all fuzzy signatures to that common level. This reduction enables the possibility of performing interpolation between the respective branches, or in some cases, directly between the roots, as outlined in [16].

A numerical illustration of the aforementioned exemplary fuzzy signature is presented in (4). For the sake of this example, let's assume that the $minimum$ (min) aggregation operator is applied at lower levels and $maximum$ (max) aggregation operator at the top level:

$$A(x) = \begin{bmatrix} \begin{bmatrix} 0.4 \\ 0.5 \end{bmatrix} \\ \begin{bmatrix} 0.3 \\ \begin{bmatrix} 0.5 \\ 0.7 \end{bmatrix} \\ 0.6 \end{bmatrix} \\ 0.9 \end{bmatrix} \tag{4}$$

The topmost level of the fuzzy signature structure can be determined as shown in (5) by executing the specific aggregation operations on the subordinate levels. For the lower levels, min aggregation operator is used which correspond to $@_1$, $@_2$ and $@_{22}$ while the max aggregation operator is applied at the top level, corresponding to $@_0$:

$$A(x) => \begin{bmatrix} 0.4 \\ \begin{bmatrix} 0.3 \\ 0.5 \\ 0.6 \end{bmatrix} \\ 0.9 \end{bmatrix} => \begin{bmatrix} 0.4 \\ 0.3 \\ 0.9 \end{bmatrix} => \begin{bmatrix} 0.9 \end{bmatrix} \tag{5}$$

3.2. Fuzzy signature sets

At their core, the structure of fuzzy signature sets bears a resemblance to that of fuzzy signatures. The primary distinction lies in the fact that each branch of the structure is composed of membership functions rather than fuzzy values. For example, (6) delineates the fuzzy signature sets $S(x)$ that was previously outlined in (3). Fig. 2 illustrates the tree structure of the fuzzy signature sets [21]. The only constraint imposed on the membership functions is that their domain must be confined to the interval $[0, 1]$:

$$S(x) = \begin{bmatrix} \begin{bmatrix} \mu_{11}(x) \\ \mu_{12}(x) \end{bmatrix} \\ \begin{bmatrix} \mu_{21}(x) \\ \begin{bmatrix} \mu_{221}(x) \\ \mu_{222}(x) \end{bmatrix} \\ \mu_{23}(x) \end{bmatrix} \\ \mu_3(x) \end{bmatrix} \tag{6}$$

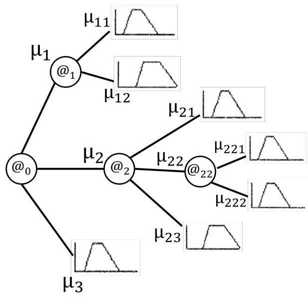

Fig. 2. Tree based representation of fuzzy signature sets.

3.3. Fuzzy signature class

Depending on the proximity of an obstacle, a robot will generate fuzzy signature values for each obstacle relative to their location within the robot's coordinate system. The fuzzy signature class P describes the positions of the each obstacle with fuzzy values and it can be represented in terms of:

$$P = \begin{bmatrix} x \\ y \\ F_x(x) \\ F_y(y) \end{bmatrix} \tag{7}$$

The elements of the position fuzzy signature class (P) are the fuzzy values that represent the distances in relation to the x, y directions as well as their corresponding fuzzy sets $F_x(x)$, $F_y(y)$, respectively. These fuzzy values provide a way for the robot to quantify and reason about the positions of obstacles in its environment in a manner that accounts for uncertainty and imprecision, which are inherent in real-world scenarios.

3.4. Quadtrees

Quadtrees [22], which are tree-structured data representations, efficiently store 2-dimensional spatial information. These structures have demonstrated their value in fields such as image processing and shape reconstruction, offering typical tree-based asymptotic benefits and enabling certain operations, like interval and point searches, to be conducted with high efficiency. It is important to recognize that in the tree-like perspective of a quadtree, each node invariably has four child nodes. Additionally, quadtrees are typically confined within a finite spatial boundary that undergoes recursive subdivision. The concept of quadtrees can be expanded to encompass 3-dimensional spaces, where the number of child nodes per parent increases to eight, akin to octrees [23], [24].

The core principle of the algorithm involves partitioning the space into distinct sectors. Each point is allocated to a particular sector based on its relative position to the sector's origin and the new point being included. Successive points are consistently inserted into the appropriate subsector after it is identified. This insertion process continues until a predetermined depth is achieved within a subsector. Unlike the original implementation of quadtrees, which splits the space into equal-sized sections, Fig. 3 illustrates a quadtree formed from a collection of points, visibly demonstrating the spatial subdivision according to the placement of geometric points.

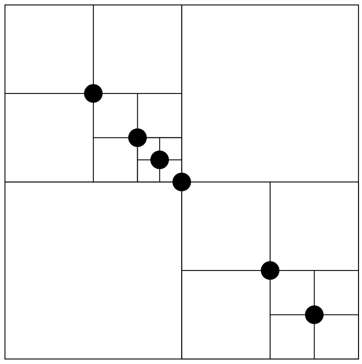

Fig. 3. Quadtree example with points inserted.

Consider a quadtree where each node represents a partition of the space. Initially, the quadtree is empty with a single node representing the entire area. As points are inserted into the quadtree, the space is subdivided into smaller regions.

In visual representation shown in the Fig. 3, a series of nested squares or rectangles can be seen, each subdivided into four smaller regions, with points (represented by dots) situated within the smallest applicable subdivision.

3.5. A* graph-based path finding algorithm

The A* algorithm, originally devised by Hart and his team [25], serves as a grid-based method to determine the shortest path for a given problem. It's widely utilized to pinpoint the least costly route from an origin to a destination. Renowned for its superior performance and reduced computational burden, A* has proven to be a potent path finding tool across various sectors, including satellite navigation for road vehicles as well as in the routing strategies for robots and vehicles [26]. The algorithm's core strategy involves calculating the cost $f(n)$ to access all neighboring nodes, with n denoting an adjacent node. The aim of A* is to progress into a state (node) that has the smallest $f(n)$ value, which is computed in terms of:

$$f(n) = g(n) + h(n) \tag{8}$$

In this formula, $g(n)$ refers to the actual distance or cost from the starting node to the node in question, n, while $h(n)$ is the heuristic predicted value of the node. Therefore, the search algorithm consistently selects the node with the lowest $f(n)$ value for further expansion in each search iteration. Steps of the A* graph based path finding algorithm is described in Section 2 of the supplementary material provided in [1].

4. Overview of the Approach

The methodology synergizes spatial data structures, particularly quadtrees, with fuzzy systems-based approaches for depicting environmental features, culminating in a unified and expansive format. The development of this method drew inspiration from Fuzzy Situational Maps (FSM) [27], which similarly segment the robot's operational space into zones of interest with varying levels of detail. The objective is to systematically arrange obstacle points and link them with fuzzy values, which are derived from rule sets that appraise the relative importance of each point. Upon completion of the construction process, every node within the quadtree encapsulates a $(x, y, F_x(x), F_y(y))$ 4-tuple, which includes the geometric coordinates (x, y) and the fuzzy sets $(F_x(x), F_y(y))$ evaluated on the respective X-Y axes, thereby containing data analogous to the fuzzy signature outlined in Fig. 2. If desired, the result of defuzzification can be appended to each node to facilitate inference. Furthermore, nodes may also store details about membership functions and their parameters to enhance the subsequent inference process. In the context of this article, the quadtree's central point is invariably positioned at the robot's center.

To translate node coordinates into fuzzy sets along each axis, fuzzy rules F_x and F_y have been established (with membership functions illustrated in Fig. 4). Each set is imbued with the following interpretations:

- *Negative Far (NF):* This signifies that the identified attribute is at a considerable distance from the robot on the negative-axis side, thus not presenting any immediate danger.
- *Negative Near (NN):* This denotes that the recognized feature is relatively close to the robot on the negative-axis side, thereby posing a significant risk.
- *Danger (D):* This indicates that the detected feature is at a perilously close proximity to the robot, necessitating immediate action to avoid a potential collision.
- *Positive Near (PN):* This is analogous to NN, but it pertains to the positive direction of the axis.
- *Positive Far (PF):* This corresponds to NF, but it relates to the positive side of the axis.

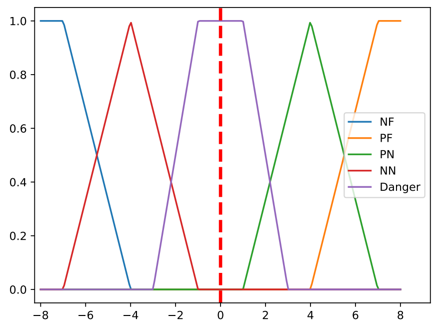

Fig. 4. Fuzzy membership functions associated with axes signatures.

The process of creating the environmental representation can be distilled into the following sequential steps, utilizing a collection of 2-dimensional obstacle points denoted as P:

1. Initialize the quadtree q to encompass the specified boundary range.
2. For each point p in set P, subdivide the space, making p the center of the new sector. Evaluate the fuzzy sets for p, assigning them alongside their coordinates, and customize

the obstacle's membership function based on its features. Initialize each node with four leaf nodes, each with defined subdivided boundaries but no values.

3. If a sector reaches its final depth limit, update the fuzzy signature value. Reparameterize the membership functions if necessary, such as when encountering a new obstacle with more significant features or based on the results of defuzzification.

The algorithm of the proposed method employing fuzzy signatures and quadtree organization can be seen in Algorithm 1:

Algorithm 1 Quadtree Construction with Fuzzy Signatures

Require: Set of points P, Boundary range for quadtree q
Ensure: Quadtree q with fuzzy signatures for environment representation
Initialize quadtree q with the specified boundary range
for each point $p \in P$ **do**
  Subdivide space, setting p as the center of the new sector
  Evaluate fuzzy sets for p and assign alongside coordinates
  Customize membership function for the obstacle based on features
  Initialize each node with four leaf nodes with defined boundaries
  if sector reaches final depth limit **then**
    Update fuzzy signature value
    Reparameterize membership functions if new obstacle is significant
  end if
end for

Fig. 5 illustrates a sample quadtree that has been constructed using the aforementioned method. The inferential procedure can leverage the tree characteristics of the formulated representation. For instance, during the generation of a grid or the employment of points from a Voronoi diagram, each point can undergo evaluation through the defuzzification process of the tree.

Taking into account the previously mentioned methodologies alongside the newly suggested concept, the integration of fuzzy signatures with a quadtree-based environmental representation emerges as a promising approach in terms of both computational efficiency and dimensional management. The initial outcomes of this proposed idea are showcased in Section 5.

5. Results

Utilizing the environment representation generated by the proposed technique, a grid was constructed to assess the preliminary efficacy of this representation. The grid was confined within the specified boundary limits. During the iterative process on each node, the inference was conducted based on a product-sum evaluation, which utilized inferred membership weights assigned during the insertion phase, in conjunction with the cylindrical extension of rules pertaining to the X-Y axes. The sub-grids obtained from this evaluation were then aggregated to form the final grid.

This approach allows for the generation of increasingly detailed grids. The algorithm continues to refine the grid until the necessary resolution is reached to identify each object within

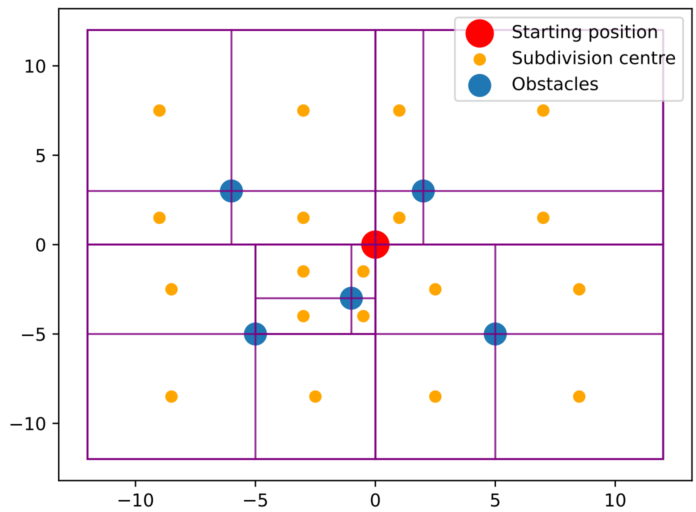

Fig. 5. An example of quadtree subdivision using the proposed method is illustrated, where the red circle indicates the starting position, blue circles represent obstacle points, orange circles denote the center points of leaf-node subdivisions, and purple rectangles outline the boundaries of each depth level.

the surroundings of the mobile robot. The process of grid creation is a complex one involving the organization of fuzzy signatures and quadtrees. This is done by dividing each section into four using quadtrees. The resulting information is then stored in fuzzy signatures. This operation is repeated iteratively until every object is detected, ensuring comprehensive coverage and representation of the environment.

For instance, the fuzzy signature graph that corresponds to the example illustrated in Fig. 5 is shown in Fig. 6. Here, S_1 denotes the top-right section, while S_2, S_3, and S_4 represent the subsequent sections in a counterclockwise direction starting from S_1. Additionally, S_{31}, S_{32}, S_{33} and S_{34} represent each subsection of the S_3 section. This indicates that there are closer obstacles in section S_3, necessitating a higher resolution to provide more detailed information about its subsections. This visual representation provides a clear picture of how the environment is divided and represented, as well as how the position information of the obstacles is stored, further demonstrating the effectiveness of the proposed approach.

Upon this grid, a well-known graph-based path finding algorithm (A*) was implemented. The algorithm commences from the central point and navigates towards a pre-determined target location. This demonstrates the practical application of the grid, showing how it can be used to guide a robot through its environment. To test the efficacy and efficiency of the approach, a rudimentary Python-based implementation was employed, and numerous input points were incorporated. This enabled the generation of a grid representation with a $0.1m$ resolution over a $20m \times 20m$ area. The inference operation was completed in approximately 0.5 ms, showcasing the speed of the proposed approach. Moreover, the process did not demand substantial memory

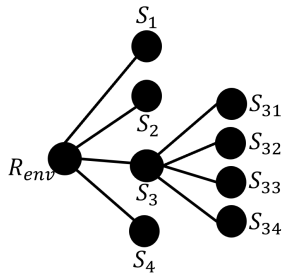

Fig. 6. Fuzzy signature graph of the shown example corresponding to the mobile robot environment representation.

resources, even on an older generation PC (Intel Core Duo 2, 16 GB RAM) at the specified grid resolution. This highlights the scalability and resource efficiency of this novel approach. The software developed for this purpose is accessible for download from the Github repository [1].

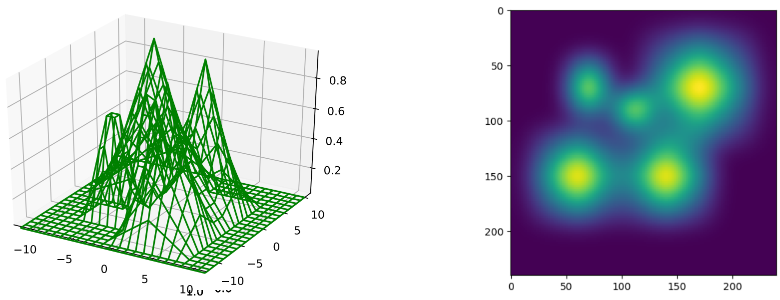

Fig. 7. Inference grid of the quadtree-based signature representation (grid and image-view).

6. Conclusions

This paper has explored a novel method for representing environments, specifically tailored for mobile robots. The approach amalgamates fuzzy signatures with quadtrees to efficiently encapsulate sparse data on obstacles and other geometric features that a mobile robot may encounter. The efficacy of the inference technique has been showcased with a simple grid-based example, which subsequently informs a general motion planning algorithm. Currently, the method is presenting a wealth of opportunities for enhancement by providing promising results for mobile robot environment representation task.

One potential application of the discussed quadtree structure is in the creation of a Voronoi diagram-based environmental model. Such a model would retain critical information, such as open spaces and obstacle positions, in a sparse and geometrically intuitive manner. This could

[1] https://github.com/robotlabor-nexus/fuzzy_signatures

directly benefit motion planning and navigation algorithms that rely on geometric data [28], [29]. Further advancements might also explore the integration of the representation's mapping output with machine learning systems, such as neural networks.

The current format has the potential for expansion to include three-dimensional data, along with a suitable inference interface. This extension could benefit from the principles of existing spatial data structures, such as octrees, which employ eight child nodes for each parent node. A specific application of this three-dimensional data expansion could be tasks performed by autonomous drone systems, such as monitoring [30], navigation [31] and path planning [32] activities.

From a technical standpoint, the implementation could be adapted for use within a distributed system, such as the Robot Operating System (ROS), which is widely used in robotics. Additionally, enhancing the solution to account for dynamic changes in the environment, such as moving or vanishing obstacles, could pave the way for the application of this method in real-world scenarios.

Acknowledgement. L.T. Kóczy is Professor Emeritus at Budapest University of Technology and Economics.

References

[1] A. M. KARADENIZ, C. HAJDU, and L. T. KÓCZY, *Supplementary material of the paper A.M. Karadeniz, C. Hajdu, and L.T. Kóczy, "Mobile Robot Environment Representation through Fuzzy Signatures-Integrated Quadtrees,"* Romanian Journal of Information Science and Technology, 2024. Accessed: October 24, 2024. [Online]. Available: https://github.com/robotlabor-nexus/ROMJIST/blob/main/Romjist-SupplementaryMaterial.pdf

[2] A. HENTOUT, A. MAOUDJ and A. KOUIDER, *Shortest path planning and efficient fuzzy logic control of mobile robots in indoor static and dynamic environments*, Romanian Journal of Information Science and Technology **27**(1), 2024, pp. 21–36.

[3] S. THRUN, W. BURGARD and D. FOX, *Probabilistic Robotics*, MIT Press, Cambridge, MA, 2005.

[4] A. M. KARADENIZ, C. HAJDU, L. T. KÓCZY and Á. BALLAGI, *Robot environment representation based on quadtree organization of fuzzy signatures*, Proceedings of 2021 IEEE 15th International Symposium on Applied Computational Intelligence and Informatics, Timisoara, Romania, 2021, pp. 509–514.

[5] J. POREBSKI, K. KOGUT, P. MARKIEWICZ, and P. SKRUCH, *Occupancy grid for static environment perception in series automotive applications*, IFAC-PapersOnLine **52**(8), 2019, pp. 148–153.

[6] P. K. PANIGRAHI and S. K. BISOY, *Localization strategies for autonomous mobile robots: A review*, Journal of King Saud University - Computer and Information Sciences **34**(8), Part B, 2022, pp. 6019–6039.

[7] J. BUTZKE, K. SAPKOTA, K. PRASAD, B. MACALLISTER and M. LIKHACHEV, *State lattice with controllers: Augmenting lattice-based path planning with controller-based motion primitives*, Proceedings of 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems, Chicago, IL, USA, 2014, pp. 258–265.

[8] L. E. KAVRAKI, P. SVESTKA, J.-C. LATOMBE and M. H. OVERMARS, *Probabilistic roadmaps for path planning in high-dimensional configuration spaces*, IEEE Transactions on Robotics and Automation **12**(4), 1996, pp. 566–580.

[9] E.-L. HEDREA, R.-E. PRECUP, R.-C. ROMAN and E. M. PETRIU, *Tensor product-based model transformation approach to tower crane systems modeling*, Asian Journal of Control **23**(3), 2021, pp. 1313–1323.

[10] C. POZNA and R.-E. PRECUP, *Aspects concerning the observation process modelling in the framework of cognition processes*, Acta Polytechnica Hungarica **9**(1), 2012, pp. 203–223.

[11] A. RAMATHILAGAM and P. PITCHIPOO, *Modeling and development of fuzzy logic-based intelligent decision support system*, Romanian Journal of Information Science and Technology **25**(1), 2022, pp. 58–79.

[12] N. YAPICI PEHLIVAN and I. B. TURKSEN, *A novel multiplicative fuzzy regression function with a multiplicative fuzzy clustering algorithm*, Romanian Journal of Information Science and Technology **24**(1), 2021, pp. 79–98.

[13] Y. HAN and T. SONG, *Path planning for shop-floor material transfer robots incorporating particle swarm and ant colony algorithms*, Studies in Informatics and Control **33**(3), 2024, pp. 49–59.

[14] D. P. KUN, E. BAKSÁNÉ VARGA, and Z. TÓTH, *Ontology based indoor navigation service for the ILONA system*, Infocommunications Journal **X**(3), 2018, pp. 21–29.

[15] D. TASKIN, C. TASKIN, S. YAZAR and A. COLAK, *Real-time low energy indoor positioning system to efficient use of operating theaters with medical asset and staff tracking*, Romanian Journal of Information Science and Technology **27**(2), 2024, pp. 151–165.

[16] Á. BALLAGI and L. T. KÓCZY, *Fuzzy signature based mobil robot motion control system*, Proceedings of 2008 6$^{\text{th}}$ International Symposium on Applied Machine Intelligence and Informatics, Herlány, Slovakia, 2008, pp. 29–33.

[17] A. M. KARADENIZ, Á. BALLAGI, and L. T. KÓCZY, *Transfer learning-based steering angle prediction and control with fuzzy signatures-enhanced fuzzy systems for autonomous vehicles*, Symmetry **16**(9), 2024, paper 1180.

[18] 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 **201**, 2012, pp. 86–104.

[19] L. T. KÓCZY, M. E. CORNEJO, and J. MEDINA, *Algebraic structure of fuzzy signatures*, Fuzzy Sets and Systems **418**, 2021, pp. 25–50.

[20] Á. BALLAGI, L. T. KÓCZY, and C. POZNA, *Robot systems with fuzzy signature rule bases*, Acta Technica Jaurinensis **4**(1), 2011, pp. 71-–88.

[21] K. TAMÁS and L. T. KÓCZY, *Mamdani-type inference in fuzzy signature based rule bases*, Proceedings of the 8$^{\text{th}}$ International Symposium of Hungarian Researchers (CINTI 2007), Budapest, Hungary, 2007, pp. 513–525.

[22] R. A. FINKEL and J. L. BENTLEY, *Quad trees a data structure for retrieval on composite keys*, Acta Informatica **4**(1), 1974, pp. 1–9.

[23] M. DONALD. *Geometric modeling using octree encoding*, Computer Graphics and Image Processing **19**(2), 1982, pp. 129–147.

[24] J. VÖRÖS, *Low-cost implementation of distance maps for path planning using matrix quadtrees and octrees*, Robotics and Computer-Integrated Manufacturing **17**(6), 2001, pp. 447–459.

[25] P. E. HART, N. J. NILSSON, and B. RAPHAEL, *A formal basis for the heuristic determination of minimum cost paths*, IEEE Transactions on Systems Science and Cybernetics **4**(2), 1968, pp. 100–107.

[26] E. M. MIYOMBO, Y.-K. LIU, C. M. MULENGA, A. SIAMULONGA, M. C. KABANDA, P. SHABA, C. XI, and A. AYODEJI, *Optimal path planning in a real-world radioactive environment: A comparative study of A-star and Dijkstra algorithms*, Nuclear Engineering and Design **420**, 2024, paper 113039.

[27] Á. BALLAGI, C. POZNA, P. FÖLDESI and L. T. KÓCZY, *Fuzzy Situational Maps: A new approach in mobile robot cooperation*, Proceedings of 2013 IEEE 17$^{\text{th}}$ International Conference on Intelligent Engineering Systems, San Jose, Costa Rica, 2013, pp. 287–292.

[28] G. C. CREȚ, A. STAN and C. GRAVA, *A short overview on navigation of mobile robots based on visual information*, Journal of Electrical and Electronics Engineering **13**(1), 2020, pp. 19–22.

[29] B. Y. SUPRAPTO, S. DWIJAYANTI and P. K. WIJAYA, *Mapping system autonomous electric vehicle based on Lidar-sensor using Hector SLAM algorithm*, Journal of Electrical and Electronics Engineering **17**(1), 2024, pp. 43–48.

[30] D.-M. DOBREA and M.-C. DOBREA, *An autonomous UAV system for video monitoring of the quarantine zones*, Romanian Journal of Information Science and Technology **23**(S), 2020, pp. 53–66.

[31] J. CHEN and S. SHEN, *Improving octree-based occupancy maps using environment sparsity with application to aerial robot navigation*, Proceedings of 2017 IEEE International Conference on Robotics and Automation, Singapore, 2017, pp. 3656–3663.

[32] M.-S. YUAN, T.-L. ZHOU, and M. CHEN, *Improved lazy theta* algorithm based on octree map for path planning of UAV*, Defence Technology **23**, 2023, pp. 8–18.