www.editada.org

International Journal of Combinatorial Optimization Problems and Informatics, 16(1), Jan-April 2025, 90-99. ISSN: 2007-1558.
https://doi.org/10.61467/2007.1558.2025.v16i1.588

20 Years of Integrated Exploration

Abraham Sánchez López[1], Alfredo Toriz Palacios[2], Rogelio González Velázquez[1]

[1] Benemérita Universidad Autónoma de Puebla. Facultad de Ciencias de la Computación, Av. 14 Sur y San Claudio, Col. San Manuel, CP 72570, Puebla, Pue. México
[2] Universidad Popular Autónoma del Estado de Puebla, Calle 21 Sur, Barrio de Santiago, CP 72410, Puebla, Pue. México
abraham.sanchez@correo.buap.mx, rogelio.gonzalez@correo.buap.mx
alfredo.toriz@upaep.mx.

Abstract. The autonomous construction of environment maps with the help of mobile robots is an important problem in modern robotics; because practically all tasks performed by robots require a representation of the working environment. Many solutions have been proposed to solve this problem known as SLAM (Simultaneous Localization and Mapping). The inclusion of a motion planner to the classical SLAM problem gives way to a new approach known as “Integrated Exploration”, in which a robot gradually builds a map while simultaneously localizing itself and making local decisions on where to go to maximize the acquisition of map information. In this paper we will analyze the proposals that have been developed in the last 20 years in this area, having as a primary interest to show the advances that have been made in the area of motion planning and the challenges presented by its integration and coordination with the SLAM problem.

Keywords: Exploration, localization, environment maps, mobile robots, integrated exploration.

Article Info
Received December 2, 2024
Accepted Feb 2, 2025

1 Introduction

A well-known topic in the field of robotics is motion planning, the main goal is to determine the best path for a robot to autonomously navigate a working environment. Research in this field has helped many areas of robotics, but one of the most recent is its application to the problem of autonomous construction of environment maps, also known as integrated exploration or active SLAM. The basic working principle of this problem is a mobile robot that must move through an unknown environment while creating an environment map of the environment. Many papers have been presented over the years to try to optimize motion planning in mobile robots and adapt them to the exploration of unknown environments. These proposals will be reviewed in the following sections, from the origins of the autonomous motion planning approach to its inclusion in the field of integrated exploration.

1.1 Motion Planning, the First Steps

Sampling-based motion planning methods (Figure 1), such as Probabilistic Motion Planning (PRM) methods or Rapidly Random exploring Trees (RRT) approaches, have proven to be very effective in robot motion

© Editorial Académica Dragón Azteca S. de R.L. de C.V. (EDITADA.ORG), All rights reserved.

planning with a high degree of freedom. In recent years, the community has proposed interesting algorithms that contribute to the state of the art. For example, strategies have been proposed to redirect sampling to the most promising regions, improving efficiency and solving difficult motion planning problems. A summary of the most important ideas and proposals in the field up to 2008 is given in [1], specifically highlighting their theoretical and practical implications, as well as their application to other topics besides robotics.

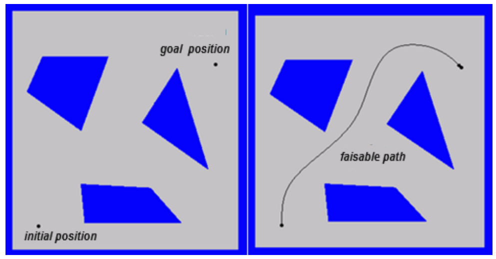

Figure 1. The basic motion planning problem.

Probabilistic sampling-based planners are capable of successfully addressing a wide range of problems. The main function of these planners is to identify where the robot collides with obstacles in the environment. Effective collision detectors can perform this procedure. Samples and edges are generated to connect them, which are stored in a suitable data structure, to obtain information about the configuration space. Numerous algorithmic techniques based on this paradigm have been proposed, some of which are now widely accepted in the standard literature of the field. One promising direction for speeding up PRM-based planners is to create better sampling strategies (and possibly also connection strategies) by using partial knowledge gained during motion plan creation and applying this knowledge to adjust the online sampling measure to be more effective [2]. When important configurations are in and around narrow areas of the configuration space, sampling-based planners often have a narrow-passages problem. Concentrating samples in difficult areas and/or generating samples in large open areas can solve the narrow passages problem. The uniform sampling strategy is not a good option in environments with narrow passages. A more competent local planner is an additional strategy [3]. Although rarely used in real time, these algorithms are an important source of inspiration for learning mobile robotics because the working environment is controlled and not real. Motion planning applications will increase. New motion planning problems will be investigated simultaneously. It is very likely that the fundamental motion planning problem, which has been the focus of motion planning research for more than two decades, will soon disappear. There is no other problem; however, it seems fundamental enough to play the same role in the future.

1.2 Motion Planning in Exploration Tasks

One of the main functions of mobile robotics is the autonomous creation of maps. Many successful robotic systems use maps of the environment to perform their functions. Therefore, research on how to optimally traverse an unknown environment, also known as the environment exploration task, while simultaneously constructing a map of it, is ongoing [4]. In recent years, the RRT method, presented by S. LaValle [3], has become the most popular single-query motion planner. RRT-based algorithms were first developed for non-holonomic and kinodynamic planning problems, where the space to be explored is the state space (i.e., a generalization of the configuration space that includes time). Another widely spread solution for the unknown environment exploration problem is the Sensor-based Random Tree (SRT) presented by Oriolo, Freda and Franchi in [5].

This method is based on the random generation of robot configurations within a local safety area detected by the robot sensors. From these configurations, a compact tree-like data structure representing the path of the explored area is created. The SRT method randomly chooses free boundaries at the robot's current position so that it can continue the exploration task. If it cannot find any, the robot will automatically return to its parent node to search for new areas with exploration possibility. When the backtracking behavior brings the robot to the root of the tree, the process ends. Each SRT node has a collision-free robot configuration, and its associated Local Safe Region (LSR), which is reconstructed in the perception system. The LSR is an estimate of the free space surrounding the robot in each configuration; its shape generally depends on the sensor characteristics but may also reflect different attitudes towards perception. SRT-Ball and SRT-Star are two techniques developed, where the shape of the RSL depends more on the characteristics of the robot used in the sensing than on the characteristics of the perception system. The use of SRT-Radial (Figure 2) is recommended because it allows considering robots with non-holonomic constraints in a study suggested in [6].

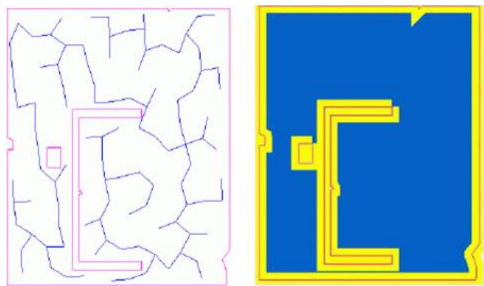

Figure 2. Example with SRT-Radial: on the left the SRT and the sensed regions and on the right the safe region with its respective safety band.

Despite its popularity, the SRT scheme presents some problems that must be considered. The first is that the state of the structure being built is not known, so full coverage of the environment cannot be guaranteed because it is not known whether the nodes of the structure left behind contain more areas available for exploration. The second problem is related to the first, as the robot must return to the parent nodes to determine if further exploration is possible, which requires the structure to be traversed twice and thus increases exploration time.

Given the above, Franchi et al. [7] create a new approach for the multi-robot case called Sensor-based Random Graph (SRG) which is based on the exploration philosophy of SRT. When the robot finds a safe path to travel between two nodes, this method transforms the tree structure created by the SRT method into an exploration graph. This method uses a probability proportional to the arc length of the free edges at the node where the robot is located to determine which position to explore next. In addition, the way to verify the structure to establish already explored zones where the exploration can be continued is done by generating a minimum spanning tree with the adjacent nodes of the graph, choosing the one of the adjacent nodes with the highest weight with respect to the length of the free boundaries of the frontiers.

The SRG method has similar problems as the SRT method in that, although the data structure is transformed into an exploration graph, the structure is not fully exploited to make the exploration more efficient because the method of revisiting nodes to verify unexplored areas creates a tree structure, which creates a discontinuous path that forces the robot to go through the parent nodes, ignoring the versatility of the graph. Moreover, as in the SRT method, the robot decides the next position to explore without considering that the random selection causes too many orientation changes, which directly affects the odometric system.

2 Integrated Exploration for Unknown Environments

Currently, the unknown environment exploration problem and the construction of environment maps are current due to their importance for mobile robotics. Classical exploration algorithms and SLAM algorithms are combined to achieve robust and efficient exploration algorithms [4]; however, although SLAM algorithms rely heavily on the trajectories performed by the robots, classical exploration algorithms do not take into account the uncertainty that the robot's movements generate about its location when it moves through unknown environments, which may result in partially constructed or low-quality maps. From the above, it is necessary to consider an integrated approach where the characteristics of the exploration algorithms are considered to explore the working environment efficiently, which gives way to the concept of "integrated exploration" or SPLAM (Simultaneous Planning Localization and Mapping), where the robot explores the environment efficiently considering the requirements of the SLAM algorithm [8], [9], [10].

Considering the last paragraph, an integrated exploring method was presented in [8] to achieve the trade-off between exploring speed and map accuracy using a single robot, here, an occupancy grid and a sparse mark SLAM are used. A multi-objective function is proposed to account for the expected information gain in the occupancy grid and SLAM system in a single action. A linear combination is used to evaluate and combine both information gains with predetermined values to be adapted to each situation.

The proposed work in [10] shows a novel SLAM algorithm based on laser data using B-Splines as a way of representing features present in the map. The Extended Kalman filter (EKF) was used to ensure the robot localization in the proposed BS-SLAM algorithm and the state vector contains the current pose of the robot along with the spline control points. The observation model used for the EKF update is the intersections of the laser beams with the splines contained in the map. In this proposal, the authors use an integrated approach based on SRT exploration [5], called SRT-BSplines.

With this integrated exploration approach, interesting results were obtained at simulation and real time level, it can be mentioned that the approach is not limited to environments with linear features. Also, the localization method is perfectly adapted to the new curves that can be seen more and more in everyday life. The theory and implementation of B-splines was a powerful tool in the approach and can be adapted to environments where previous methods considered only simple descriptions.

Toriz et al. in [11], presented a new method called Random Exploration Graph (REG), which maximizes the map coverage during the exploration process. This method adapts the working principle of the SRT method to create an exploration graph structure. Although this method has a probabilistic nature that may result in excessive robot movements to complete the task and prolong the exploration time, one of its main advantages is the accumulation of knowledge through the concept of frontier control, which stores information about the areas that the robot left behind in the exploration process and needs to revisit to complete the exploration.

The relevance of frontier control lies in identifying only the information of the nodes not fully explored during the development of the exploration, saving the unanalyzed free frontiers contained in the nodes. This concept allows the systematic exploration of the acquired knowledge, allowing the REG method to plan paths towards well-identified zones that need to be explored.

The REG method works without the need to go back and physically verify each of the nodes within the structure, as required by the SRT method. Using REG, there is also no need to perform a complete analysis of the exploration structure generated during the execution of the task every time there is a need to find a new area with possibilities to be explored, as required by the SRG method. These differences with SRT and SRG represent a substantial computational improvement of REG in terms of execution time.

Motion planning is executed with a bidirectional A* method, using the graph structure created by planning the path from the current node to the nodes contained in the frontier control and from the nodes in the frontier control to the current node, ending when the path is between the current node and any node in the list. The use of the bidirectional A* algorithm extends the path from the initial position or the desired position to the final position reached by the opposite side, ending when both paths are at the same node.

This strategy is used simultaneously in our method, with all nodes contained in the list and terminates when a path is found. The reason for looking for individual paths from the current position to each node with the possibility of exploration, rather than simply a Euclidean distance to the nearest node, is the existence of inaccessible spaces, one might think that the distance to the node is short, but perhaps the distance to reach it could be very large.

Once a path is obtained, the control method will allow the movement of the robot from the current node to the node that will allow the robot to continue with the exploration Finally, the index of the selected node is removed from the list of nodes that can be explored. With the new node to be explored, the method will continue with the same process described above until there is no unexplored frontier.

Figure 3 shows an example obtained with REG, in an office-garden environment (widely used in testing SLAM algorithms).

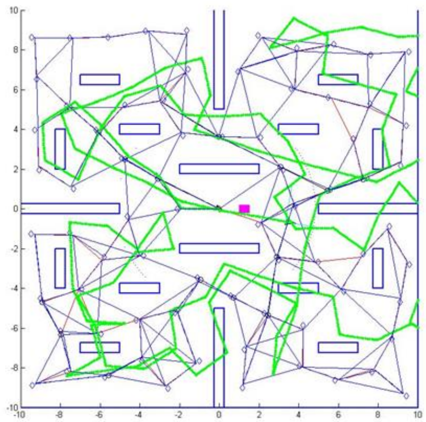

Figure 3. Graph structure and path followed by the REG approach.

This figure shows the algorithm monitor and the execution is in real time, a Pioneer P3DX robot equipped with a laser sensor was used. We can also comment that we do not include more real tests due to the space we have, and it would be interesting for the reader to review the suggested bibliography. It would be important to include comparative tables, but in themselves, these proposals are not feasible for such comparisons, since they have been developed with different approaches and although we can use the same robot, the working environments change significantly. Given that one of the fundamental objectives of the different proposed works is the consideration of complex environments, environments that are out of consideration in many works of the state of the art that only consider representations based on points and lines.

The extended REG method was designed to be integrated as part of the SLAM method, facilitating the construction of maps in complex environments. As can be seen, the methods shown use randomness to determine the next position to explore. The problems with these algorithms are the number of movements required to traverse the work environment, the time required to complete the task, and in some cases as in the SRT and SRG methods, the uncertainty about the total coverage of the exploration area.

Given these limitations, in [12] a new Deterministic Exploration Graph strategy known as the DEG method is presented, which results from a modification to the REG method, where the main difference lies in how the robot will plan the exploration path by performing a deterministic analysis of the next position to be explored. In the algorithm, the start and end node will be the start and end node. As in REG method, the exploration structure will include a position of the robot reached, as well as a representation of the local safety region (LSR), where the robot can navigate without risk of colliding with any obstacle. The cycle controls the exploration process with this created node. Then, at each iteration of the algorithm, the frontiers of the adjacent nodes to the current node are evaluated to determine which free frontier segments with the possibility to explore belong to the current LSR. To avoid considering these intersections in a possible return of the robot for further exploration, the free frontier segments of the neighboring node and the current node will be removed. In addition, intersection checking between nodes is used to modify the structure of the exploration graph by adding edges between non-adjacent nodes if there are safe paths to travel between them.

After analyzing the frontiers of neighboring nodes covered by the new LSR and modifying the exploration structure with new edges, the next step is to identify the remaining free frontiers of the current position. An approximation point will be determined for each of the frontiers found, if any, which will serve to prioritize the free frontiers, ranking them according to the effort required to reach them and selecting a new frontier that has the highest hierarchy. The approach point is defined as the midpoint of the arc segment formed by the frontiers, if these can be covered in their entirety by the threshold defined by the LSR area. If the criterion for choosing the approach point is not met, it will be redefined by taking the midpoint of the arc length proportional to the area that can be covered by the LSR, from the initial end of the frontier. With this new point selected to continue the exploration, the frontier or a segment of the frontier is removed from the group of free frontiers found.

When the robot does not find a new position to explore at the current node, i.e. there are no more free frontiers, the robot will choose one of the nodes contained in the frontier control to continue the exploration. The choice of the position will be determined by the bidirectional A* search algorithm, which will extend a path from the current node to the nodes of the frontier control and from the nodes of the frontier control to the current position. At this point, the frontier control will remove the index of the node where the path was found. In this way, the method will continue executing the described process, until there are no more free frontiers at the current node, and the frontier control list is empty; at this point, the robot will search for a path to return to the initial node from where the exploration process starts. Figure 4 shows the result of the application of the DEG algorithm in the exploration of an unknown environment.

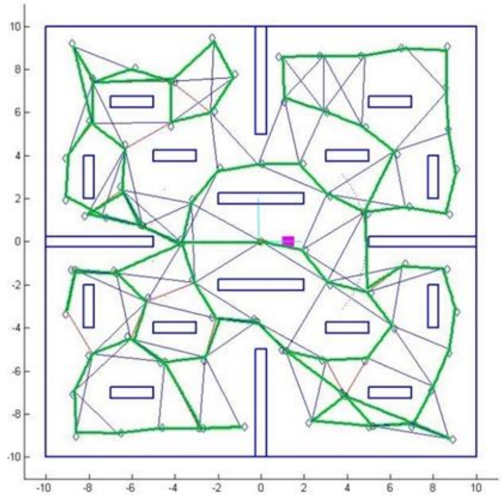

Figure 4. Graph structure and path followed by the DEG approach.

3 Tests and Results

Numerous experiments were performed with the intention of evaluating the accuracy and consistency of the integrated exploration proposals analyzed in this article; in addition, quantitative variables used in the field of exploration methods, such as exploration time, odometric error, and total environment coverage, were analyzed and compared with data obtained by the SRT (Sensor-based Random Tree) [5], SRG (Sensor-based Random Graph) [7], REG (Random Exploration Graph) [11], and DEG (Deterministic Exploration Graph) [12] methods, which allows explaining the characteristics of the methods.

With respect to the integrated exploration paradigm, the exploration methods were adapted to work with the idea of any SLAM method; however, the tests performed opted to use the method presented in [13] due to the comprehensive way of data exploitation of the working environment.

The P3DX pioneer differential robot, equipped with a Hokuyo URG-04LX range sensor with a sensing range of 4 meters, an angular resolution of 0.360° and a sweep angle of 240°, was used to perform the tests using simulated data. The experiments used a modified environment of the corridors of the Laboratory of Informatics, Robotics and Microelectronics (LIRMM) in Montpellier - France (see Figure 5).

Figure 5. Example of an environment where we performed experimental tests with the developed algorithms.

Table 1 shows the comparative results of the variable time required by the SRT, SRG, REG and DEG exploration methods to complete the exploration of the environment map, the results were obtained from 30 trials. In this table, it is easy to observe that the DEG method requires approximately 25% less time than the best mean time of the other three methods. In addition, it is possible to observe that the standard deviation for the DEG method is very low compared to the other methods due to the deterministic way of choosing the next position to explore. By brevity of the table, we only show the first 10 results, without forgetting that 30 executions of each proposed method were carried out.

Table 1. Time required for the DEG, REG, SRG an SRT exploration methods to explore the LIRM environment based on 30 tests.

Test number	*Total time required to complete the exploration with DEG.*	*Total time required to complete the exploration with REG*	*Total time required to complete the exploration with SRG*	*Total time required to complete the exploration with SRT*
1	214.72	261.83	590.81	691.36
2	206.95	299.34	426.42	417.04
3	218.46	312.00	492.87	548.34
4	219.97	275.01	611.48	559.18
5	227.20	302.86	572.78	590.68
6	207.81	264.34	485.68	788.16
7	212.47	293.49	452.87	587.40
8	202.14	315.21	601.09	560.75
9	209.30	320.43	450.02	751.03
10	203.5	313.92	544.92	592.65
Average	215.16	287.95	521.70	596.32
Standard deviation	7.94	23.77	58.45	108.72

In addition, tables 2, 3 and 4 show the odometric errors in the X-axis, Y-axis and in the orientation θ obtained after performing the exploration of the environment by each of the methods. In these tables it can be observed that the odometric error is lower in the case of the DEG method because it does not require too many orientation changes to perform the exploration of the environment, a characteristic that is not shared by the REG, SRG and SRT methods due to their random nature. Once again, not to use much space, only the first 10 tests are reported.

Table 2. Odometric error in the X-axis reported by DEG, REG, SRG an SRT exploration methods to explore the LIRM environment based on 30 tests.

Test number	*Maximum accumulated odometric error in the y-axis obtained by DEG.*	*Maximum accumulated odometric error in the y-axis obtained by REG*	*Maximum accumulated odometric error in the y-axis obtained by SRG*	*Maximum accumulated odometric error in the y-axis obtained by SRT*
1	1.2201	5.2554	7.1789	8.0055
2	1.0918	6.2265	5.9925	9.9349
3	1.9713	6.3740	7.3390	4.7194
4	1.7877	5.7652	5.4158	4.3211
5	1.3591	6.0856	6.4277	6.1871
6	1.8465	5.7474	5.5948	6.4377
7	1.8850	6.7594	5.0442	7.2604
8	1.4895	6.3329	4.1114	6.7849
9	1.2506	5.1962	5.1091	8.7728
10	1.1168	4.1123	5.2357	4.9936
Average	1.5172	5.7558	6.2200	8.1979
Standard deviation	0.3169	0.7847	1.4181	2.3366

Table 3. Odometric error in the Y-axis reported by DEG, REG, SRG an SRT exploration methods to explore the LIRM environment based on 30 tests.

Test number	*Maximum accumulated odometric error in the y-axis obtained by DEG.*	*Maximum accumulated odometric error in the y-axis obtained by REG*	*Maximum accumulated odometric error in the y-axis obtained by SRG*	*Maximum accumulated odometric error in the y-axis obtained by SRT*
1	1.0510	5.7380	7.4345	8.3685
2	1.5145	6.6229	8.4326	10.3981
3	1.1518	6.6694	8.6794	6.8330
4	1.2032	4.8037	9.9912	8.4258
5	1.8662	7.1203	8.7447	6.9073
6	1.5301	6.1926	8.0771	9.2429
7	1.8517	5.4546	9.5684	13.7404
8	1.4147	5.1956	7.9154	9.0708
9	1.8892	4.9849	9.9837	11.0256
10	1.9341	6.1731	7.2319	
Average	1.5085	6.1873	8.4668	9.6757
Standard deviation	0.3416	0.9353	1.2185	2.3434

Table 4. Odometric error in orientation θ reported by DEG, REG, SRG an SRT exploration methods to explore the LIRM environment based on 30 tests.

Test number	*Maximum accumulated odometric error obtained by DEG (radians)*	*Maximum accumulated odometric error obtained by REG (radians)*	*Maximum accumulated odometric error obtained by SRG (radians)*	*Maximum accumulated odometric error obtained by SRT (radians)*
1	0.1407	0.5770	0.8508	1.0046
2	0.1271	0.4539	0.8036	0.9469
3	0.1305	0.6855	0.8122	1.0275
4	0.1143	0.5256	0.8206	0.9315
5	0.1373	0.6538	0.8080	0.8406
6	0.1485	0.5075	0.8225	0.9286
7	0.1563	0.6985	0.7495	1.0209
8	0.1596	0.4655	0.6619	0.9663
9	0.1156	0.6029	0.6313	0.9568
10	0.1349	0.6725	0.6725	0.9402
Average	0.1329	0.5600	0.7638	0.9320
Standard deviation	0.0153	0.1013	0.0884	0.0688

Another important factor to be considered by the methods is the coverage of the environment during exploration. Thus, to determine the percentage of exploration of the environment by the methods, the environment was divided into grids to determine which had been explored.

Table 5 shows that both the DEG method and the REG method achieved 100% coverage of the environment in all tests, thanks to the integrated boundary control in both methods, which allows constant knowledge of the unexplored areas of the environment. On the other hand, both the SRT and SRG methods cannot guarantee full coverage, as their randomness may omit unexplored areas during the exploration.

Table 5. Explored surface of the LIRM environment for DEG, REG and SRG exploration methods based on 30 tests.

Test number	*Map coverage obtained by DEG*	*Map coverage obtained by REG*	*Map coverage obtained by SRG*
1	100 %	100 %	96 %
2	100 %	100 %	91 %
3	100 %	100 %	96 %
4	100 %	100 %	96 %
5	100 %	100 %	93 %
6	100 %	100 %	91 %
7	100 %	100 %	90 %
8	100 %	100 %	90 %
9	100 %	100 %	97 %
10	100 %	100 %	92 %
Average	100 %	100 %	94 %
Standard deviation	0%	0 %	3 %

4 Conclusions and Future Work

In the works on the subject of exploration and map creation in robotics, many authors emphasize that the odometry is one of the most important aspects in a navigation system and that it is important that the accuracy of the odometry is a point to be improved [15], which would help significantly in selecting the marks that will be used in the localization of the robot [14]. It is important to emphasize an important detail of most of the works published in recent years, the reduction of the odometric error is performed with different algorithms and often omits the consequences of the robot motion control [12].

As an important conclusion, we can state that the way of exploring environments has a strong impact on the correctness of the odometry of the mobile robot and this has important effects on the localization and the construction of the map. And perhaps the most catastrophic effect of this, is the association of the data, which would lead to a proposal of an incorrect SLAM algorithm and not applicable to real problems.

We can affirm that this new way of exploring unknown environments (the REG and DEG methods) minimizes the time required to perform this task, but above all it significantly minimizes the accumulated odometric error on the robot system.

For many years, one of the important goals for the mobile robotics community is to have truly autonomous robotic systems in acquiring maps of unknown environments to perform their tasks, i.e. to have robust algorithms to solve the SLAM problem. And by way of conclusion, the reader will find in many references interesting proposals to solve the SLAM problem, however, there is little literature and therefore proposals for integrated exploration, i.e. SLAM plus motion control. What we have developed for 20 years has been evolutionary and we continue to improve aspects that are emphasized in different academic forums when we present papers on the subject.

It is clear that further detailing the work that we as a team have done since the beginning of this century, would be long and tedious. The interesting thing in this work is to show how step by step we have reached a point where integrated exploration can be an interesting solution that can be adopted to other classic robotics problems. As future work we could mention: the famous robot abduction problem (while the robot is performing exploration, it is abducted and moved elsewhere, the robot must detect this abduction situation and perform a global location in order to return to its local exploration situation) and the multi-robot version of DEG.

It should be noted that recent work on integrated exploration continues to use the methods proposed at the beginning of the 2000s and that it is interesting to emphasize that they are the ones that we use as a basis at the beginning of our research, it is clear that many recent methods highlight the good results obtained with topological methods [16], [17] y [18] and this is where the most recent contributions we have made to the state of the art are located.

References

1. Sánchez López A., Zapata René., Osorio Lama María A. (2008). Sampling-based motion planning: A survey. Computación y Sistemas, 12-1 (5 – 24).
2. Sánchez López A., Zapata René., Lanzoni Claudio. (2003). On the use of low-discrepancy sequences in nonholonomic motion planning. Proc. of the IEEE International Conference on Robotics and Automation, (3764-3769).
3. Lavalle S. M. (2006). Planning algorithms. Cambridge University Press.
4. Siegwart R., Nourbakhsh I. R., Scaramuzza D., Arkin R. C. (2011). Introduction to autonomous mobile robots. The MIT Press.
5. Oriolo G., Vendittelli M., Freda L., Troso G. (2004). The SRT method: Randomized strategies for exploration. IEEE/RSJ International Conference on Intelligent Robots and Systems. (4688-4694).
6. Espinoza L. Judith., Sánchez L. Abraham., Osorio L. Maria A. (2007). Exploring unknown environments with mobile robots using SRT-Radial. IEEE/RSJ International Conference on Intelligent Robots and Systems. (289-294).
7. Franchi A., Freda L., Oriolo G., Venditelli M. (2007). A randomized strategy for cooperative robot exploration. IEEE/RSJ International Conference on Intelligent Robots and Systems. (534-539).
8. Makarenko A. A., Williams S. B., Bourgante F., Durrant-Whyte H. F. (2002). IEEE/RSJ International Conference on Intelligent Robots and Systems. (534-539).
9. Pedraza L., Dissanayake G., Valls-Miro J., Rodriguez-Losada D., Matia F. (2009). Extending the limits of feature-based SLAM with B-Splines. IEEE Transactions on Robotics, 25, (353-366).
10. Sánchez L. Abraham, Toriz P. Alfredo, Zapata René, Osorio L. Maria A. (2011). Mobile robots SPLAM for robust navigation. Reserach in Computing Science, 54 (295-304).
11. Toriz P. Alfredo, Sánchez L. Abraham, Bedolla C. J. Maria. (2017). The random exploration graph for optimal exploration of unknown enviroments. International Journal of Advanced Robotic Systems, (1-11).
12. Toriz P. Alfredo, Sánchez L. Abraham. (2019). Extending the limits of the random exploration graph for efficient autonomous exploration in unknown enviroments. IntechOpen, (1-14).
13. Pedraza L., Dissanayake G., Miro J. V., Rodriguez-Losada D., Matia F. (2007). BS-SLAM: Shaping the world. Robotics: Science and Systems, (1-8).
14. Potthast C., Gaurav Sukhatme S. (2014). A probabilistic framework for next best view estimation in a cluttered environment. 25-1, (149-164).
15. Toriz P. Alfredo, Sánchez L. Abraham (2020) Sobre la mejora esperada de la estimación de la odometría en exploración integrada. Revista Iberoamericana de automática e Informática Industrial, 17-2, (27-40).
16. Brugali Davide, Luizaga Y. J. Raul (2023). Simultaneous frontier-based exploration and topological mapping, Seventh IEEE International Conference on Robotic Computing, (219-222).
17. Li Bo, Pan Feng, Zhang Shi, Song Xiaofei, Cui Rongxin. (2023). Frontier-based exploration planning for amphibious robot with Gaussian process occupancy map. Proc. of the 42nd Chinese Control Conference, (3889-3894).
18. Wu Cheng-Yan, Lin Huei-Yung. (2019). Autonomous mobile robot exploration in unknown indoor environments based on rapidly-exploring random tree. IEEE International Conference on Industrial Technology, (1345-1350).