IBIMA Publishing
Communications of International Proceedings
https://ibimapublishing.com/p-articles/45AI/2025/4522125/
Vol. 2025 (20), Article ID 4522125

Multi Sampling-Strategy RRT Path Planning Optimization Using Genetic Algorithm*

Bartłomiej Bonar, Tomasz Buratowski and Mariusz Giergiel

AGH University of Kraków, Aleja Adama Mickiewicza 30, Kraków, Poland

Correspondence should be addressed to: Bartłomiej Bonar, bbonar@agh.edu.pl

* Presented at the 45th IBIMA International Conference, 25-26 June 2025, Cordoba, Spain

Abstract

The paper presents optimization experiment performed on custom implementation of multi-sampling strategy approach for Rapidly exploring Random Tree path planning algorithm. Implementation of algorithm, of which detailed explanation and pseudocode are provided, was developed to perform with increased robustness and intended for dynamic environment navigation task. Genetic Algorithm is used to search for optimal weights for choosing sampling strategy based on mean path length from multiple executions of the algorithm. Optimization experiment is performed on multiple testing boards, results are analyzed and presented, showing strong tendency towards an unstable solution creating statistically shortest path.

Keywords: path planning, mobile robot, RRT

Introduction

Rapidly exploring random tree (RRT) is currently one of the most frequently used algorithms for mobile robot path planning. As described by its author, in contrast to other popular stochastic approaches to path planning, RRT naturally extend to general nonholonomic problems [1]. This property allows a trajectory that is generated on the basis of planned path to more closely resemble the originally planned path, which allows for finer *apriori* control of robot actions, which could be easily seen in known static environment, where path should not be subjected to changes during robot work.

One of the most important modifications for RRT is RRT*, which changes the strategy of creating connections (branches) between nodes [2]. Instead of choosing the node closest to the sampled point, RRT* searches within radius larger than the native distance used to sample new points and looks for the node closest to the root of the tree. This way of creating paths allows for visibly shorter end-result without any post-processing.

When considering the implementation of the RRT* algorithm, sampling strategy is one of its vital properties to decide on. Its original version uses global uniform random sampling [3], which results in strong exploratory search tendencies, but at the same time has a slow convergence rate towards the target. To solve this problem, extensive research on sampling strategies for RRT* has been conducted over the years.

Informed RRT modification uses elliptical sampling, which allows a much faster convergence rate [4]. Instead of sampling towards random directions, in most iterations the tree will be growing towards its target. At the same time some exploratory behavior will be retained, especially in the area directly between initial state and target.

The authors of [5] proposed introduction of a density gradient area to guide samples. This strategy of guiding samples allows for much faster convergence towards the target than in case of always using uniform distribution. Samples will be confined to a progressively smaller area around the target, thus forcing a faster search. To retain

Cite this Article as: Bartłomiej Bonar, Tomasz Buratowski and Mariusz Giergiel, Vol. 2025 (20) "Multi Sampling-Strategy RRT Path Planning Optimization Using Genetic Algorithm " Communications of International Proceedings, Vol. 2025 (20), Article ID 4522125, https://doi.org/10.5171/2025.4522125

exploratory behavior and effectiveness of algorithm in more complex environments, uniform random sampling will still be used with some probability.

The research described in [6] is focused on analysis of the usage of multiple sampling strategies to obtain results much faster than when using a single sampling strategy. The data presented show that the result of using hybrid sampling is a statistically much shorter path prepared using significantly fewer nodes during path planning.

In addition to using multiple sampling strategies, another popular modification introduced to the RRT* algorithm is to grow more than one tree during the path planning phase. This approach allows for bi-directional search, which finishes when trees connect instead of tree starting from robot finding path to target. Statistically, it results in multiple smaller search areas, significantly reducing search time. The authors of [4] show an example of using a tree grown from the initial state with elliptical sampling (Informed RRT*) and directed sampling. As shown in the results, this approach allows for much faster execution of the path planning phase, which ends before the algorithms start searching in areas distant from both the initial state and the target.

In [7] an approach with auto-generation of new trees is presented based on environment analysis. Stochastic path planning algorithms have a tendency to struggle with traversing cluttered, complex environments, as most sampling strategies do not take into account local geometry of terrain. As per description of benefits of using multiple trees at once described above, creating new trees in such areas allows for much easier creating connections in locally complex areas, as trees with trunks located inside them will naturally create paths connecting opposite sides of such areas.

The solution presented in this research is the result of using uniform random sampling, directional search and elliptical sampling in BI-RRT* to achieve balance between exploratory behavior and fast-converging planning. Initial tests are performed with manually tuned weights of differ- ent sampling strategies based on expert knowledge, and then optimization techniques are used for finding the best settings in multiple test environments.

MS-RRT*

Before describing the proposed modified version of RRT* planer - Multi Strategy RRT* (MS- RRT*), the original algorithm will be explained. Apart from the sampling stage, which is the main focus of this work, the algorithm creates new nodes based on sampled points, tries to connect them with the tree, and verifies if the path to the target has been created. Based on the explanation provided in [8], RRT* can be described as:

Algorithm 1 RRT*

while *Pathfinding* **do**
    Sample point x_{sample}
    Find closest node x_{c-node}
    Try new node x_{n-node} in distance r from x_{c-node}
    if $x_{c-node} \rightarrow x_{n-node}$ CollisionFree **then**
        Create new node at x_{n-node}
    else
        Back to sampling point
    end if
    In range R look for visible node x_{s-node} with shortest path to trunk
    Connect x_{n-node} to x_{s-node} and calculate new path length
    if Target in range r from x_{n-node} **then**
        Finish pathfinding
    end if
end while

It can be seen from the pseudocode description that most of the algorithm is related to the validation of new nodes and connecting them with the rest of the tree. There is known research on moving the already planned sample based on local obstacle analysis if no nodes or connections can be created [9]. Such an approach might prove useful when considering a dynamic environment.

Sampling Strategies

As described in the introduction, the algorithm presented in this work uses three different sampling strategies:

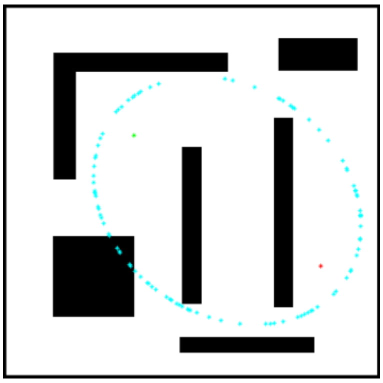
- uniform random sampling

- directed search sampling
- elliptical sampling

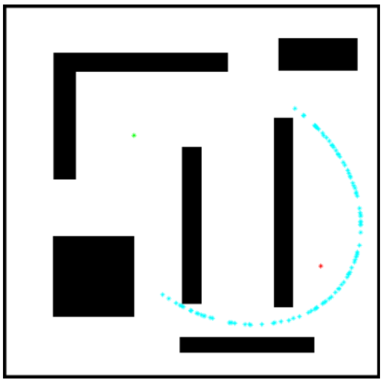
Uniform random sampling was used in original work [1] and is the most simple sampling strategy. As the name describes, a random point is chosen on the board and is used if it can be connected to the tree. This sampling strategy is introduced to maintain exploratory behavior. This strategy will contribute to increased search time, but has been used for increasing robustness of algorithm in complex environments, where other strategies may prove insufficient.

Directed search sampling is a strategy that can be compared with the heuristic used in the A* algorithm [10]. The samples are chosen on the basis of the cost of the path to the initial node and the distance from the beeline to the target. This sampling strategy samples a point in proximity to the target and then chooses one of the closest nodes in the tree to branch from. This strategy is introduced to increase the convergence rate of the algorithm towards the target, reducing the required amount of calculations for path planning. Its major drawback is that sampling will not be effective if there are obstacles directly on the line between the target and the nodes closest to it.

Elliptical sampling strategy, typically used in Informed RRT* [4] provides common ground between fast convergence and exploratory behavior. The samples are located on an ellipse that



(a) Full-ellipse sampling



(b) Half-ellipse sampling

Figure 1: Comparison of a) full ellipse sampling, b) half-ellipse sampling

encompasses the initial state and target. It is noteworthy that this type on sampling can be done on full ellipse as well as part of it. In Fig. 1 is presented sampling done on the whole ellipse as well as half of it on the side of the target. It can be easily seen that sampling on the part of ellipse close to initial state does not contribute much to path planning process, as tree is inclined to grow away from its target, which is reason for using only half of ellipse in this work.

*Bi-tree RRT**

The idea of bi-directional RRT, as introduced above, is based on using multiple trees at once to make it more natural for the algorithm to explore workspace around areas selected by locating the trunk there. This approach in a seamless way decomposes search space into a number of subspaces corresponding to the number of trees, thus creating better conditions for working with complex spaces, which alternatively would have to be solved by one heavily modified tree to solve such problems.

There are known examples of using more than two trees to address difficult areas in robot workspace, one of which is provided by authors of [7]. This work will focus on adjusting work parameters for the two-tree variant, but implementation of the algorithm had been prepared in a way that allowed any number of trees to work together, which will be the subject of future research. In Fig. 2, three stages of tree growth can be seen. The parameters for the trees in this case were arbitrarily chosen and asymmetrical as presented in the tab. 1.

Table 1: Parameters for bi-tree example

Tree trunk location	**Uniform**	**Directed**	**Elliptical**
Robot	6	4	2

Target	2	3	6

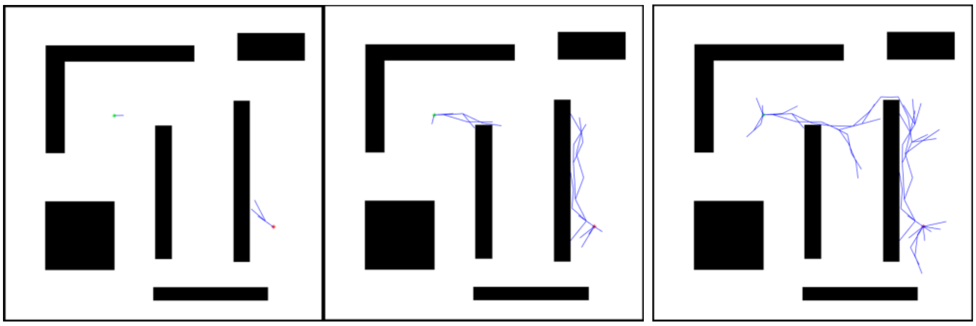


Figure 2: Successive stages of bi-tree finding path

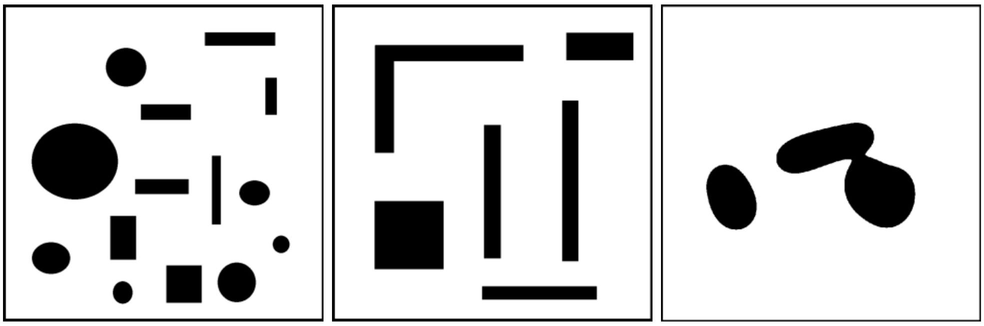


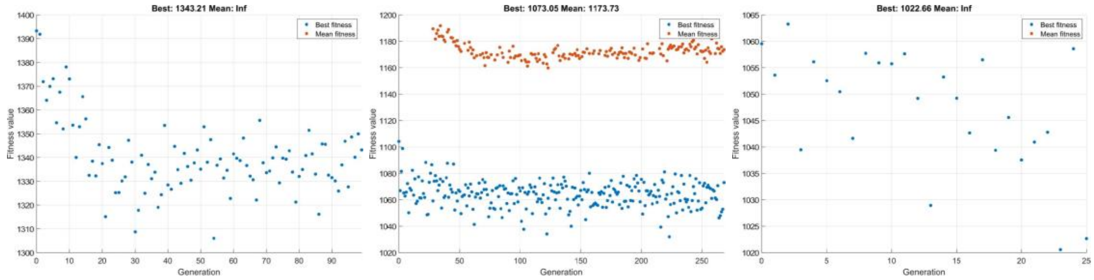
Figure 3: Set of testing boards

Results

To find the best set of parameters for RRT implementation, a Genetic Algorithm (GA) was used. Proposed for optimization purposes in [11], GA is currently one of the most important algorithms used to find optimal solutions to mathematical problems. In addition to typical optimization problems, it is also widely used in global path planning [12], but it should be known that such an approach usually requires relatively long computation times.

To optimize the algorithm, three boards with different types of obstacles present in the workspace were prepared (Fig. 3). The total length of the path was used as the value to be minimized and the quality function that GA uses was a wrapper for the RRT implementation to make it run 10 times and returns the medium value of the path length to ensure that results are statistically meaningful despite the algorithm being of stochastic nature. In each case, all sampling types weights were limited to range from 0.33 to 3 and timeout was set for 4 hours. The calculations were performed on a laptop with an Intel i5-12500H. For all simulations, a half-ellipse was used.

The optimization results can be seen in Fig. 4 and in the tab. 2. Despite the weight of uniform distribution being visibly higher for tree 2 in set 2 and tree 1 in set 3, there is a visible trend of GA putting much greater value in directed search than for other methods, which is understandable from medium path length being a quality function. For analysis purposes, it is important to note that the penalty function was also introduced, which can be seen in Fig. 4, where only the second set has a mean value that is not infinite. This is because RRT had a safety mechanism that will interrupt path planning if path could not be found within a set amount of operations, which prevented e.g.



(a) Results for first board (b) Results for second board (c) Results for third board

Figure 4: Results of RRT optimization using GA

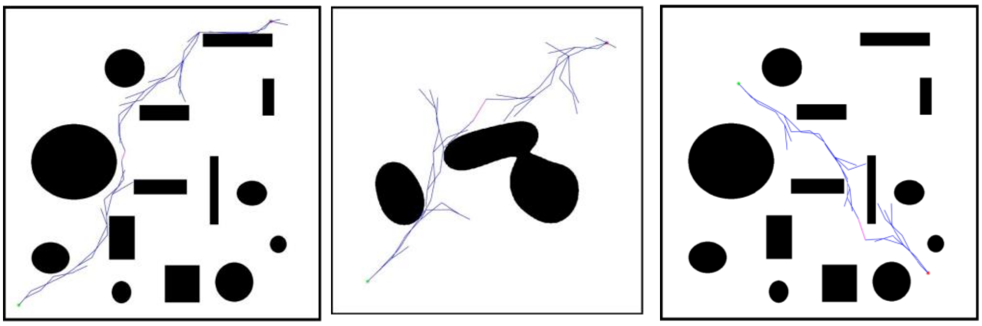


Figure 5: Results of path planning using optimized RRT

calculating indefinitely when only directed search was performed and there was an obstacle with local minimum on the road between robot and target.

Such situation seems to have happened in case of set 3. It can be seen that GA calculated significantly fewer generations than in other cases. Also, for the first and second sets, the optimiza- tion was interrupted due to a minimal change in the trend of the fitness value after around 1,5h. In case of third set, optimization timeouted after 4 hours without achieving satisfactory results. Analysis showed that obstacle creating local minimum from both robot and target sides caused a lot of samples to be located in those areas through directed search, which as mentioned before, was highly favored by GA due to it creating shortest path possible. Examples of planned path using optimized parameters can be seen in Fig. 5

Table 2: Parameters found by GA optimization

Set number	T1-Elli	T1-Uni	T1-Dir	T2-Elli	T2-Uni	T2-Dir
1	0.372	0.512	2.456	0.689	0.745	2.801
2	0.749	0.407	2.936	0.414	2.087	2.939
3	1.013	2.834	1.757	0.608	0.635	2.043

Cite this Article as: Bartłomiej Bonar, Tomasz Buratowski and Mariusz Giergiel, Vol. 2025 (20) "Multi Sampling-Strategy RRT Path Planning Optimization Using Genetic Algorithm " Communications of International Proceedings, Vol. 2025 (20), Article ID 4522125, https://doi.org/10.5171/2025.4522125

Conclusions

This paper presents the development of an RRT implementation using multiple sampling strate- gies, which are randomly chosen based on weight parameters. Those parameters are optimized using GA on a number of prepared testing boards, and some examples of paths planned using optimized weights are presented and analyzed at the end.

To analyze results, it is important to understand that medium length of set of paths was used as quality function. Because RRT is stochastic in its nature, for results to be statistically mean- ingful, an experiment for each parameter set was performed 10 times. RRT also had built-in safety mechanism working in similar way to watchdog for CPU - if for long enough time results could not be achieved, algorithm work was deemed unsuccessful, interrupted and infinite length of path was returned instead. This caused a result visible in Fig. 4, where only the second optimization run achieved results stable enough for subsequent results to have visible mean value of quality function. For every optimization experiment directed search was highly favored, as creating bee-line path will naturally lead to shorter total length. It turned out to be problematic on the third board. In that case, on both sides of the obstacle were concave local minimum areas, which caused tree expansion to get stuck there in case of using directed search. It resulted in algorithm reaching internal watchdog to intervene after too much time passed, thus resulting in not calculating as many generations as other runs despite spending much more time of calculations before achieving timeout.

The optimization experiment as a whole has shown that when dealing with stochastic algorithms, it is important to perform simulation multiple times and analyze only mean of results. Even tough for this research path planning was performed multiple times for each set of parameters, it still can be seen in figures presenting history of optimization that results have significant spread. Increasing the number of repetitions for each set of parameters will decrease the spread, resulting in more orderly outcomes.

References

- S. M. LaValle, Rapidly-exploring random trees: A new tool for path planning, Technical Report 11 (98) (1998).
- F. E. Karaman S, Sampling-based algorithms for optimal motion planning, The International Journal of Robotics Research 30 (7) (2011) 8460894. doi:doi:10.1177/0278364911406761.
- S. M. LaValle, J. J. J. Kuffner, Randomized kinodynamic planning, The International Journal of Robotics Research 20 (5) (2001) 378–400. doi:https://doi.org/10.1177/027836401220674.
- X. H. H. Z. H. S. Honghui Fan, Jiahe Huang, Bi-rrt*: An improved path plan- ning algorithm for secure and trustworthy mobile robots systems, Heliyon 10 (2024). doi:https://doi.org/10.1016/j.heliyon.2024.e26403.
- W. S. Tai Huang, Kuangang Fan, Density gradient-rrt: An improved rapidly exploring ran- dom tree algorithm for uav path planning, Expert Systems With Applications 252 (2024). doi:https://doi.org/10.1016/j.eswa.2024.124121.
- R. E. M. Sivasankar Ganesan, Balakrishnan Ramalingam, A hybrid sampling-based rrt* path planning algorithm for autonomous mobile robot navigation, Expert Systems With Applica- tions 258 (2024). doi:https://doi.org/10.1016/j.eswa.2024.125206.
- Q. L. M. S. X. Z. Haiyan Tu, Yizhao Deng, Improved rrt global path plan- ning algorithm based on bridge test, Robotics and Autonomous Systems 171 (2024). doi:https://doi.org/10.1016/j.robot.2023.104570.
- X. H. K. S. S. L. L. W. Jun Ding, Yinxuan Zhou, An improved rrt* algorithm for robot path planning based on path expansion heuristic sampling, Journal of Computational Science 67 (2023). doi:https://doi.org/10.1016/j.jocs.2022.101937.
- Z. W. Xiangkui Jiang, C. Dong, A path planning algorithm based on improved rrt sampling region, Computers, Materials Continua 80 (3) (2024). doi:10.32604/cmc.2024.054640.
- N. R. B. Hart, P. E.; Nilsson, A formal basis for the heuristic determination of mini- mum cost paths, IEEE Transactions on Systems Science and Cybernetics 4 (2) (1968). doi:10.1109/TSSC.1968.300136.
- D. E. Goldberg, Genetic Algorithms in Search, Optimization and Machine Learning, 1st Edi- tion, Addison-Wesley Longman Publishing Co., Inc., USA, 1989.
- R. Sarkar, D. Barman, N. Chowdhury, Domain knowledge based genetic algorithms for mobile robot path planning having single and multiple targets, Journal of King Saud University - Computer and Information

Sciences 34 (7) (2022) 4269–4283. doi:https://doi.org/10.1016/j.jksuci.2020.10.010.