

Path Planning using Shi and Karl Level Sets

Randeep Singh

Reliable Care Solutions

Philips Research Asia – Bangalore (PRA-B)
Bangalore 560045, India.

Email: randeep.singh@philips.com

Tel: +91-80-4189 2252

Fax: +91-80-4189 2265

Nagaraju Bussa

Reliable Care Solutions

Philips Research Asia – Bangalore (PRA-B)
Bangalore 560045, India.

Email: nagaraju.bussa@philips.com

Tel: +91-80-4189 2237

Fax: +91-80-4189 2265

Abstract—Path planning for mobile robots is a well researched problem for over three decades. In this paper, we test and evaluate a new approach based on Shi and Karl Level Sets for mobile robot path planning. The evolution speed of these level sets is the fastest available implementation of level sets till date that prompted us to test the algorithm in the path planning domain. Level Sets have the advantage that they can support multiple robots and multiple goals in a single framework. This approach is similar to the Pixel Level Snakes but much faster and the resulting paths are different for some obstacle courses using these two approaches. To keep experiments simple we consider a holonomic point robot moving amidst stationary obstacles. The visual results presented in this paper are preliminary and tested in simulations only.

I. INTRODUCTION

Path planning for mobile robots is a well researched problem for over three decades. Solutions to path planning problems broadly involve either direct optimization based or heuristics based approaches. Most of the practical applications require a process that produces a reasonable result with the cost of the solution increasing with longer solve time. Very rarely, if ever is the globally optimal path ever required in practice.

Global path planning takes into account all the information in the environment whereas local path planning (reactive) algorithms are designed to avoid obstacles within a close vicinity of the robot. In this only information from on-board sensors and nearby obstacles is used.

The goal of path planning is to compute a collision free path between two configurations of a given robot. At a broad level, prior algorithms for path planning can be classified into roadmap methods, exact cell decomposition, approximate cell decomposition, potential field methods, randomized sampling-based methods etc. [1][2][3][4][5][6][7].

Recent work on robot path planning include [8] where the author has compared the three path planning approaches which include the older A* path planning [9] and the Distance Transform approach [10] and the newer Rapidly Exploring Random Trees (RRT) approach [11].

The paper is organized as follows. Section 2 discusses related research in global path planning methods that use level sets or marching propagating interface. Section 3 defines the Karl and Shi level sets that we will use for the path

planning problem. In section 4 we present our results of the path planning experiments on different obstacle courses. Few application scenarios are detailed in section 5. We give our conclusions and some directions for proposed future work in section 6.

II. RELATED RESEARCH

Jahanbin and Fallside [12] first introduced a wave propagation algorithm in the Configuration Space on discrete maps (Distance Transform). The distance transform is similar to a wave front expansion [13] where all the free cells take on the lowest value of their neighbour cells increased by one. Another novel approach to the problem of findpath using distance transforms was presented by [14]. This approach considers path planning to be finding all paths from the goal location back to the start location. This planning approach propagates a distance wave front through all free space from the goal. From any starting point in the environment the shortest path to the goal is traced by following path of steepest descent. If no downhill path exists then we can conclude the goal is unreachable. Despite the disadvantage of computing all paths from the goal location the distance transforms have a number of advantageous properties. They can support multiple robots and multiple goals in addition to supporting different types of findpath. For example paths which favour or avoid certain areas in the environment and paths of complete coverage of the environment. In [15], the author reported work that is also similar to distance transforms. A grid representation was used where each grid cell stores the lowest cost to the goal. From the grid a circulation map is derived which is represented with gradient fields to show all paths to the goal. Both the [14] and [15] approaches suffer the “too close” problem since they only consider the shortest path to the goal.

In [16], an extension to the distance transform called the “path transform” was presented. Using the path transform approach to findpath, instead of propagating a distance wave front from the goal, another wave front is propagated which is a combination of the distance from the goal together with a measure of the discomfort of moving near obstacles. This has the effect of producing a distance transform which has the properties of potential fields without local minima.

In [17], the authors present a dynamic wave expansion neural network (DWENN) model for path generation in a dynamic environment for both mobile robots and robotic manipulators, the model is parameter-free, and its complexity does not explicitly depend on the dimensionality of the configuration space.

In [18], path planning for robots was studied using level sets where there were objects to be avoided in the domain. The method of solution was to construct a weighted distance function over the entire domain and then, from a final position, back propagate the solution perpendicular to the level sets of the distance function, resulting in an optimally shortest path.

In [19], the authors use the level sets to determine both the shortest as well as the paths with the minimum costs. They extended this approach in [20], where they propose a robust path planning method comprising of two wave fronts traveling at different speeds. The first front propagates with a moderate speed to capture the map topology, while the second one propagates much faster at medial points such that the safest paths intersect the propagating fronts at those points of maximum positive curvature, which are identified by solving an ordinary differential equation (ODE). The motion of the front is governed by a nonlinear partial differential equation (PDE). There is a tuning parameter α by which planned paths can be made safer or shorter. No values for the speed of evolution of the wave fronts or the no. of iterations to reach the goal are given.

In [21] the path planning problem has been solved as the problem of visibility that involves the determination of regions in space visible to a given observer when obstacles to that sight are present. When the observer is replaced by a light source in the simplified geometric optics without reaction setting, the problem translates to that of finding illuminated regions.

In [22], a novel approach for solving the shortest path problem in 2D binary labyrinths based on cellular neural network (CNN) computation is proposed. Pixel level snakes (PLS) are a topographic and iterative active contour technique based on a pixel level discretization of the contours and on a massively parallel processing on every contour cell. According to this strategy, a labyrinth under study is explored by a traveling wave with constant speed initiated at the source point in such a way that the wavefront evolving along the shortest path will reach the target point without collisions. Any waves initiated at a junction located on the shortest path will reach the end of the branch or collide with other wave(s). In order to obtain the shortest path the paths are cut at the collision points destroying the closed loops. Finally, all branches are pruned in such a way that if there is a unique solution the result will be the shortest path. The PLS approach is the closest to our approach in terms of the shortest path found.

III. PROPOSED METHODOLOGY

The path planning problem is solved by first initializing the level set from the point which is the starting point of the robot. The level set is originated from this point and the wave front is propagated until it has reached the goal or the whole of the

obstacle course has been covered. We find the shortest path by finding the shortest path backwards between two iterations of the level set and finally to the starting point of the level set.

A. Shi and Karl Level Sets

The level set method, originally introduced by Osher and Sethian [23] is a general framework for the computation of evolving interfaces using implicit representations. We refer the interested reader to the monograph by Osher and Fedkiw [24] for a general introduction to the level set method and an overview of applications in several areas. The Shi and Karl level sets [25] implementation does not involve solving for the ODEs and PDEs that arise in the level sets domain. They demonstrate that their algorithm is approximately two orders of magnitude faster than even the optimized narrow band level set [26] algorithm for image segmentation tasks. The speed of these level sets is what prompted us to test the algorithm in the path planning domain.

In their implementation the object boundary is represented uniquely by two 1-pixel wide lists of points, L_{out} and L_{in} . L_{out} and L_{in} are formally defined as follows:

$$L_{out} = \{x \mid \phi(x) > 0 \text{ and } \exists y \in N(y) \text{ s.t. } \phi(y) < 0\} \quad (1)$$

$$L_{in} = \{x \mid \phi(x) < 0 \text{ and } \exists y \in N(y) \text{ s.t. } \phi(y) > 0\} \quad (2)$$

where $N(x)$ is a discrete 4-neighborhood of x .

The propagation of the wave is obtained by addition, deletion or switching of the point x between L_{out} and L_{in} . The interested reader is referred to the paper of Shi and Karl [25] for the implementation details.

IV. EXPERIMENTS

The path planning problem is solved by first initializing the level set from the point which is the starting point of the robot. In this, the starting point becomes the central pixel which is the only pixel in L_{in} and the 4-neighbors are put in L_{out} . The wave front is propagated until it has reached the goal or the whole of the obstacle course has been covered. In normal level sets, the shortest path to goal from any starting point in the environment is traced by following path of steepest descent. We find the shortest path by finding the shortest path backwards between two iterations of the level set and finally to the starting point of the robot.

We highlight the difference between the normal level sets and the Shi and Karl level sets by an example. Fig. 1 shows the expanding wave front of normal level sets and Fig. 2 shows the Shi and Karl level sets for the same problem. The normal level set also takes into consideration the curvature of the expanding wave front and hence is slower but accurate description of the flow than the Shi and Karl level sets.

Fig. 3 shows 7 robots converging on a pre-defined goal point. Please note here that we need only one level set starting from the goal point for all the robots to find their way back to the goal. Another way to extend the overall approach is to pre-define ‘‘way points’’ which the user wants the robot to visit. The robot can be made to visit the way points by initiating



Fig. 1: Expanding wave front of Level Sets

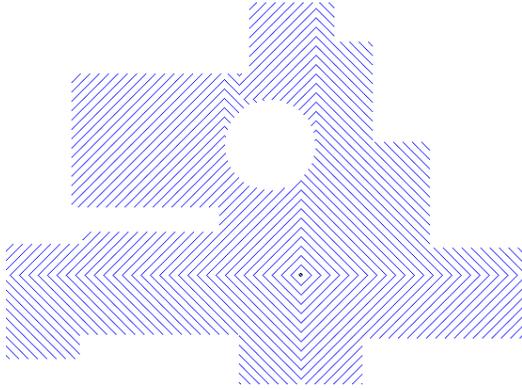


Fig. 2: Expanding wave fronts of Shi and Karl level sets (every 10^{th} iteration is shown) for Fig. 1

the level sets from the starting point until it reaches the next way point. Different level sets have to be initiated at each way point for this to accomplish. In general, we need $N - 1$ level sets for N way points. The result for this is shown in Fig. 4.

V. APPLICATION SCENARIOS

We detail few application scenarios where our work will be helpful. The multi robot rendezvous problem [27]; the context of the rendezvous is an unknown environment, with

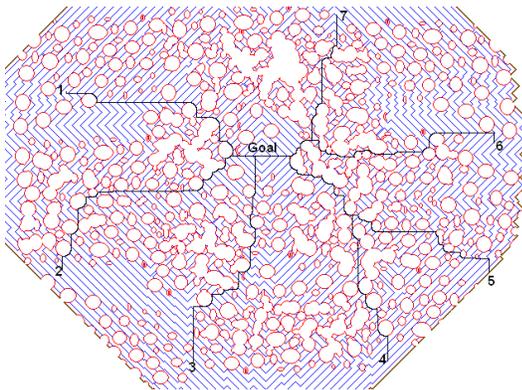


Fig. 3: Multi robots converging to a single goal

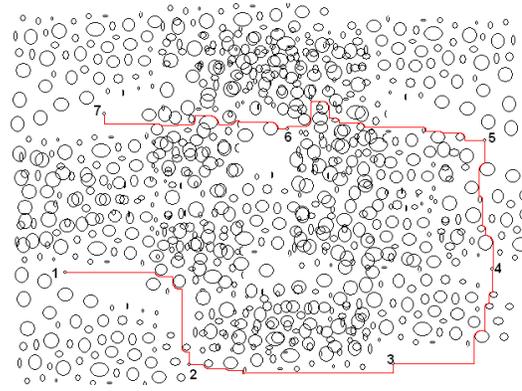


Fig. 4: Robot traveling through pre defined way points

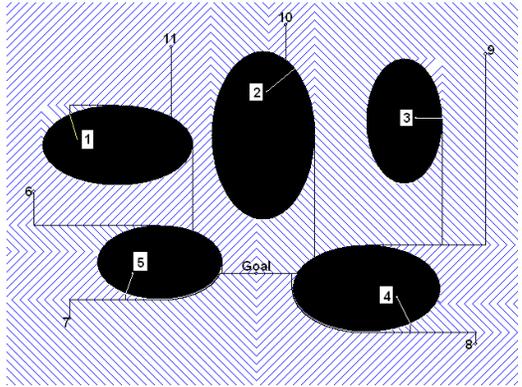


Fig. 5: Multi robots emerging out of obstacles and converging on the goal

no shared spatial information between agents, and there is no communication until rendezvous. This problem can be separated into two separate sub-problems. The first is how to select points in the environment for potential rendezvous. The second subproblem that can greatly help from our work is how to reach the potential rendezvous point once it has been identified and selected. A practical application of our approach can be in the future battle field scenario where a flying robot is helping multiple ground robots to converge and attack. We extended the level sets approach to also include the scenario where a robot is stuck *inside the obstacle*. In this case, the robot first moves out of the obstacle. The path it takes inside the obstacle is the minimum path to the nearest level set. Once the robot comes out of the obstacle, it then moves towards the goal point as shown in Fig. 5. In this figure, robots 1–5 are stuck in the obstacle and manage to meet at the goal point.

Another area where our approach can be useful is in the domain of circuit path planning for PCBs layout and VLSI designing. Lee [28] presents a good survey for techniques available for rectilinear path planning among rectilinear obstacles. Mitchell [29] proposes an approach called 45 degree wave front propagation that is very similar to our approach. Applying the Shi and Karl level sets to this domain will certainly make the process faster. Fig. 6 shows the rectilinear

