This is a repository copy of *A Genetic- Based Approach to Robot Motion Planning Considering Path Safety*.

White Rose Research Online URL for this paper:
http://eprints.whiterose.ac.uk/79767/

Monograph:

Mungwu, Chen and Zalzala, A.M.S. (1994) A Genetic- Based Approach to Robot Motion Planning Considering Path Safety. UNSPECIFIED. ACSE Research Report 537 . Department of Automatic Control and Systems Engineering

Reuse

Unless indicated otherwise, fulltext items are protected by copyright with all rights reserved. The copyright exception in section 29 of the Copyright, Designs and Patents Act 1988 allows the making of a single copy solely for the purpose of non-commercial research or private study within the limits of fair dealing. The publisher or other rights-holder may allow further reproduction and re-use of this version - refer to the White Rose Research Online record for this item. Where records identify the publisher as the copyright holder, users can verify any specific terms of use on the publisher's website.

Takedown

If you consider content in White Rose Research Online to be in breach of UK law, please notify us by emailing eprints@whiterose.ac.uk including the URL of the record and the reason for the withdrawal request.

eprints@whiterose.ac.uk
https://eprints.whiterose.ac.uk/

629.8 (S)

A Genetic-Based Approach to Robot Motion Planning Considering Path Safety

Mingwu Chen[+] and A.M.S. Zalzala[*]

Abstract

This paper proposes a genetic algorithm for robot path planning by considering both the travelling distance and a safety criterion. Incorporation of the safety issue into path planning is important not only because of the uncertainties in the robot dynamics during path execution, but also because of the inaccuracies in the geometric modelling of obstacles. The approach uses a wave front expansion algorithm to build the numeric potential fields for both the goal point and the obstacles by representing the workspace as a grid. The safety value of a node in the grid is defined as the numerical potential from obstacles. A genetic algorithm is developed to search for near optimal paths. Computer simulation results are presented to demonstrate the effectiveness of the algorithm.

Key Words: Robotics, Motion Planning, System Safety, Multi-Criteria Optimisation, Genetic Algorithms.

Research Report No. 537
26 September 1994

+ : Mr. Mingwu Chen is a PhD student with the Robotics Research Group in the Department of Automatic Control and Systems Engineering, the University of Sheffield, Sheffield S1 4DU, UK. His email address is Mingwu.Chen@sheffield.ac.uk. Tel: 0044 (0)742 825237. Fax: 0044 (0)742 731729

* : Dr. A.M.S. Zalzala is a lecturer with the Robotics Research Group in the Department of Automatic Control and Systems Engineering, the University of Sheffield, Sheffield, S1 4DU, UK. His email address is A.Zalzala@sheffield.ac.uk. Tel. 0044 (0)742 825136 Fax: 0044 (0)742 731729

1. Introduction

The motion planning of mobile robots is concerned with providing a feasible and efficient path to accomplish a given task. Depending on the system objective, the robot path planning has to be solved by optimising suitable criteria subject to some constraints. This paper presents a near optimal solution to the robot path planning problem with a weighted distance-safety criterion.

The travelling distance has been the primary object to minimise in most conventional robot path planning approaches because the shortest distance path may reduce the robot's travelling time and consequently reduce the computational complexity of path planning. However, another factor which should not be ignored during planning is robot safety during path execution . Robot safety becomes important, especially when there are non negligible uncertainties in both the robot dynamics during path execution and the environmental information such as obstacles. Thus the simultaneous consideration of distance and safety needs to be called for during robot path planning.

The safety of a robot path can be quantified by the clearance between the path and obstacles. If robot safety is the only concern, one would choose a path providing the maximum clearance from obstacles. However, such a path could be considerably longer than the shortest one, and it is not desirable to consider safety only when the robot's travelling distance is of significant importance.

The safety of a path has not been considered explicitly in most known path planning approaches. In most papers([1],[2]), the path safety was obtained by enlarging each obstacle by a specified amount. Though the method of growing obstacles is simple and attractive in many cases, a potential problem with this method is that some good paths would have been eliminated as a result of growing obstacles. Moreover, it may be very difficult to determine the degree of enlargement of obstacles during path planning because of its independence on the utilisation of the workspace as well as the uncertainties in the robot dynamics during path execution. Paper [3] presented a variational dynamic programming approach to robot path planning with a distance-safety criterion. The method represented free workspace as channels and the safety cost of a path is defined as the deviation of the path from the centre-line path.

In this paper, we represent a cluttered environment as a grid through cell decomposition. Two numerical potential fields are built for the obstacles and the goal point by using "wave front expansion" algorithm. Each point in the grid has a safety value from the obstacles and a distance value from the goal point. The safety value of a node is defined as the numerical potential from the obstacles. A feasible path is a set of adjacent nodes in the free space connecting the start point and the goal point. The cost of a path is defined as the sum of the travelling distance and the average safety of all the points in the path.

There often exist a large, (even infinite), number of paths between the initial position and final position and path planning is not necessarily to determine the best solution but to obtain a good one according to certain requirements. Different optimisation methods have been developed (e.g. calculus based methods, enumerative schemes, random search algorithms, etc.) for solving the path planning problem. These conventional optimisation methods are shown not to be very effective in certain applications. Genetic algorithms are very robust search and optimisation methods.

SHEFFIELD UNIV.
LIBRARY
ST. GEORGE'S

choosing them can help searching for safe paths. Fig(1b) is a numerical potential field for obstacles.

29	27	25	23	21	19	17	15	13	11	9	7	5	3	5	7
27	25	23	21	19	17	15	13	11	9	7	5	3	1	3	5
25	23	21	19	17	15	13	11	9	7	5	3	1	0	1	3
27	25	23	*	*	17	15	13	11	9	7	5	3	1	3	5
29	27	*	*	*	*	17	15	*	*	*	*	*	3	5	7
31	29	*	*	*	*	19	17	19	*	*	*	7	5	7	9
33	31	33	*	*	23	21	19	21	19	*	11	9	7	9	11
35	33	31	29	27	25	*	*	19	17	15	13	11	9	11	13
37	35	33	31	29	27	*	*	21	19	17	15	13	11	13	15
39	37	*	*	*	29	*	*	23	21	*	*	*	13	15	17
41	39	*	*	*	31	29	27	25	*	*	*	17	15	17	19
43	41	43	*	*	33	31	29	*	*	*	21	19	17	19	21
45	43	45	*	*	35	33	31	29	27	25	23	21	19	21	23
47	45	43	41	39	37	35	33	31	29	27	25	23	21	23	25
49	47	45	43	41	39	37	35	33	31	29	27	25	23	25	27
51	49	47	45	43	41	39	37	35	33	31	29	27	25	27	29

10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10
10	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	10
10	9	7	7	7	7	7	7	7	7	7	7	7	7	7	7	9	10
10	9	7	7	9	9	7	5	5	7	7	7	7	7	5	7	9	10
10	9	7	9	10	10	9	7	7	9	9	9	9	9	7	7	9	10
10	9	9	10	*	*	10	9	9	10	10	10	10	10	9	7	9	10
10	9	9	10	*	*	10	9	7	9	10	*	10	9	7	7	9	10
10	9	7	9	10	10	9	9	9	7	9	10	9	7	5	7	9	10
10	9	7	7	9	9	9	10	10	9	7	9	7	7	5	7	9	10
10	9	7	9	9	9	9	10	10	9	7	9	9	9	7	7	9	10
10	9	9	10	10	10	9	10	10	9	9	10	10	10	9	7	9	10
10	9	9	10	*	10	9	9	9	9	10	*	10	9	7	7	9	10
10	9	7	9	10	10	9	7	9	10	10	10	9	7	5	7	9	10
10	9	7	9	10	10	9	7	7	9	9	9	7	5	5	7	9	10
10	9	7	7	9	9	7	5	5	7	7	7	5	5	5	7	9	10
10	9	7	7	7	7	7	7	7	7	7	7	7	7	7	7	9	10
10	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	10
10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10

(a) $d_1 = 1, d_2 = 3, d_3 = 5, \cdots$ (b) $o_0 = 10, o_1 = 9, o_2 = 7, \cdots$

Fig(1) Two numerical potential fields

Every node in the free space is assigned two values. One is the distance value from the goal, and the other is the distance value from obstacles which represents the safety of a node. The farther away a node is from the obstacles, the lower the safety value of a node is and the safer it is for a robot to move through it.

3. The Cost Function

Let a path P be described by a set of neighbouring nodes connecting the starting node and goal node, denoted by $\{n_i, i=1,2,...,N\}$.

Considering both distance and safety, the cost function is represented by

$$C(P) = \varepsilon D(P) + \lambda S(P) \qquad \varepsilon, \lambda \geq 0$$

where D(P) and S(P) represent costs associated with length and safety of P, respectively, and ε, λ are the relative weightings between the two, thus

$$D(P) = \sum_{i=2}^{N} \| n_i - n_{i-1} \|$$

and

$$S(P) = \frac{1}{N} \sum_{i=1}^{N} S_i \qquad S_i \leq S_u$$

where S_i is the safety value of the node n_i and S_u is the given upper limit of the safety value, which is introduced to prevent a path from getting too close to obstacles.

In a given workspace there usually exist a large (even infinite) number of feasible paths connecting the starting point and the goal point. By adjusting the relative weightings ε and λ different paths can be searched for. In this paper, we use genetic algorithms to optimise the cost function and search for optimal paths.

4. The Genetic Algorithm Approach

To solve the mobile robot path planning problem by a genetic algorithm, we need a coding scheme to encode the parameters of the problem into genetic strings. In our problem, the robot path is coded as a string of N number of points represented by their Cartesian co-ordinates as

$$\{(x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N)\}$$

with all the values stored in a decimal form. This coding method yields variable-length paths and a proper genetic structure is required to deal with it, in particular when performing crossover.

First, a set of valid random paths are generated as the initial generation. In order to prevent the robot wandering endlessly inside the work space, a weighted vector of motion direction is employed according to the goal distance values of the 8 neighbouring nodes. A neighbouring node which has a lower goal distance value, i.e. a node closer to the goal, has more chance to be selected as the next node in the path. A fitness value is assigned to each string according to its distance and safety. In our algorithm, the fitness function is defined as

$$f = C_m - C(P)$$

where C_m is a selected positive real number not less than the maximum cost of C(P).

Second, a reproduction approach is applied to select strings for the next generation. Genetic algorithms use the fitness value of each string of the current generation to decide if and how many copies of the string should be passed to the next generation. The larger the fitness value of one string, i.e. the lower the cost of the path, the higher probability of the string being chosen for the next generation.

When in early generations there is a tendency for a few superstrings to dominate the selection process. Later on when the population has largely converged, competition among population members is less strong and the simulation tends to wander. In these two cases, fitness values must be scaled to prevent the takeover of the population by a few superstrings in the early generations and to accentuate differences between population members to continue to reward the best performers. In this paper we use linear scaling to calculate the scaled fitness f' from the raw fitness f using a linear equation of the form

$$f' = a f + b$$

In this equation, the coefficients a and b are chosen to do two things: enforce equality of the raw and scaled average fitness values and cause the maximum scaled fitness to be two times of the average fitness. These two conditions ensure that the average strings receive one offspring copy on average and the best receive two on average. When the scaled fitness value of a string becomes negative, we simply set it to zero. To reduce the stochastic error associated with the selection, we implement the stochastic remainder sampling without replacement.

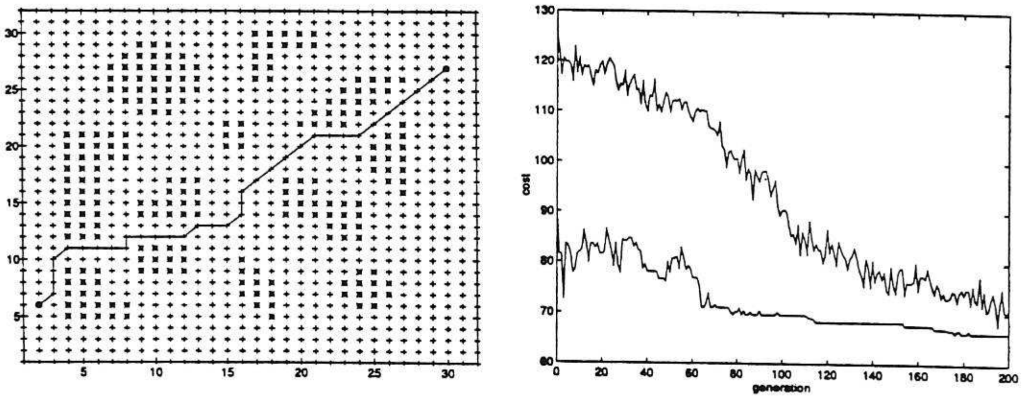
Performing crossover is not straight forward because of the variable-length coding and, more important, since a random crossover would produce a discontinuous path. Thus the selected path pair is checked for nodes with a certain proximity (coincident, one or two nodes apart). If one is found and is not coincident for both paths, a random segment is generated to connect both nodes, and exchange the remainders of the both paths. If the path pair have same points, then select one randomly as the crossover site. To perform mutation, select a node randomly along a path, and the remainder of the unmutated path between that node and goal is destroyed and replaced by a randomly generated segment.

5. Computer Simulation Results

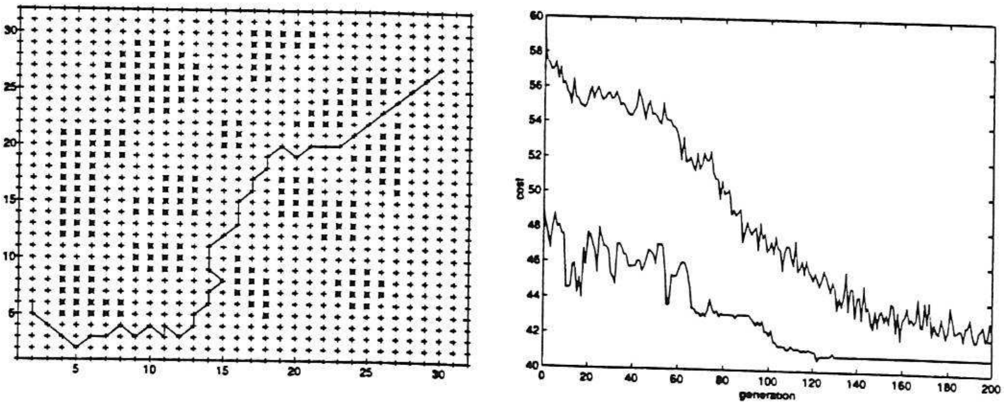
In order to demonstrate the efficiency of the algorithm, we built a workspace with cluttered obstacles as shown in Fig(2). The workspace is represented by a grid with 32×32 nodes. The staring point is (2,6), and the goal is (30,27).

Three genetic parameters have significant effects on the performance of the genetic algorithms. These parameters are n (population size), p_c (crossover probability), and p_m (mutation probability). As a reasonable compromise between the speed to reach the acceptable result (on-line performance) and convergence (off-line performance), we choose n as 100, p_c as 0.8 and p_m as 0.03 in the simulation.

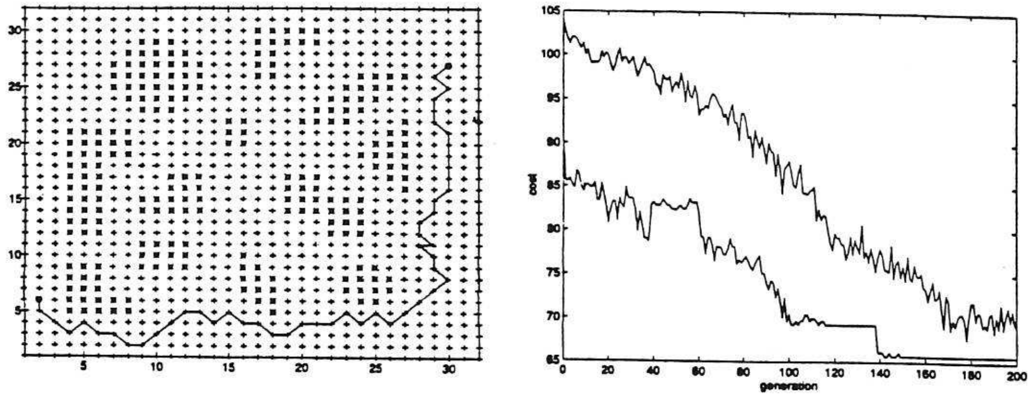
Fig(2) shows the paths and the cost functions obtained by the algorithm. When $\varepsilon=1$, $\lambda=0$, i.e. not considering safety, a nearly shortest path is obtained. When $\varepsilon=0.1$, $\lambda=10$, there is a compromise between distance and safety, a relative safe and short path is obtained. When λ is further increased to 20, i.e. safety is supposed to be a major concern, the algorithm generates a very safe path but it is much longer than the shortest one.



(a) $\varepsilon=1$ $\lambda=0$



(b) $\varepsilon=0.1$ $\lambda=10$



(c) ε=0.1   λ=20

Fig(2) Robot paths and their costs

In the figures showing the costs, the upper lines show the average costs of each generation and the lower lines show the lowest costs within each generation. All the costs decrease steadily and converge as the generations increase.

6. Conclusion

In this paper, we have developed a genetic algorithm approach to search for optimal paths for a mobile robot in a cluttered environment with a distance-safety criterion. The incorporation of robot safety into path planning is practically important not only because of the uncertainties in robot dynamics during path execution, but also because of the inaccuracy in the geometric modelling of obstacles. We have demonstrated the effectiveness and efficiency of the algorithm to reach near optimal solutions to different weighting path planning problems through various simulations. Although it is developed mainly for two dimensional problems the algorithm can be easily extended to the class of three dimensional problems.

Acknowledgement

The first author greatly acknowledges the support of the Sino-British Friendship Scholarship Scheme.

References

[1] T. Lozano-Perez and M.A. Wesley, "An algorithm for planning collision-free paths among polyhedral obstacles", Commun. ACM, Vol.22, No.10, pp.560-570, 1979.

[2] K.G. Shin and N.D. Mckay, "Selection of near-minimum time geometric paths for robotic manipulators", IEEE Transaction on Automation and Control, Vol.AC-31, No.6, pp.501-511, 1986.

[3] Suk-Hwan Suh and K.G. Shin, " A variational dynamic programming approach to robot-path planning with a distance-safety criterion", IEEE Journal of Robotics and Automation, Vol.4, No.3, pp.334-349, 1988.

[4] J. Barraquand and J.C. Latombe, " Robot motion planning: A distributed representation approach", The International Journal of Robotics Research, Vol.10, No.6, pp.628-649, 1991.

[5] T. F. Cleghorn, P.T. Baffes and L. Wang, "Robot path planning using a genetic algorithm", Proc. 2nd Annual Workshop on Space Operations, Automation and Robotics, pp.383-390, 1988.

[6] C.H. Leung and A.M.S. Zalzala, "A genetic solution for the control of wheeled robotic systems in dynamic environments", Proc. IEE International Conference on Control'94, 1994.

[7] K.K. Chan amd A.M.S. Zalzala, " Genetic-based minimum-time trajectory planning of articulated manipulators with torque constraints", IEE Colloquium on Genetic Algorithms for Control and Systems Engineering, May 1993.

[8] D.E. Goldberg, Genetic Algorithms in Search, Optimisation and Machine Learning, Addison Welsey, 1989.

SHEFFIELD UNIV.
LIBRARY
ST. GEORGES