Chapter 1
Robots That Do Not Avoid Obstacles

Kyriakos Papadopoulos and Apostolos Syropoulos

Abstract The motion planning problem is a fundamental problem in robotics, so that every autonomous robot should be able to deal with it. A number of solutions have been proposed and a probabilistic one seems to be quite reasonable. However, here we propose a more adoptive solution that uses fuzzy set theory and we expose this solution next to a sort survey on the recent theory of soft robots, for a future qualitative comparison between the two.

1.1 Introduction.

According to Latombe [9], “the ultimate goal of robotics is to create autonomous robots”. Furber [6] adds that

\[
\ldots \text{such robots should be able to accept high-level description of tasks and execute them without further human intervention. The input description specifies what should be done and the robot decides how to do it and performs the task. One expects robots to have sensors and actuators.}
\]

Typically, robots should be programmed so to be able to plan collision-free motions for complex bodies from some point \(A\) to another point \(B\) while having a collection of static obstacles in between. This task is called motion planning. Naturally, motion planning is very interesting but there are many cases where this is not even desirable. For example, a rover moving on the surface of a planet should be able to go above obstacles or to even pass through obstacles.

K. Papadopoulos
Department of Mathematics, Kuwait University, PO Box 5969, Safat 13060, Kuwait
e-mail: kyriakos.papadopoulos1981@gmail.com

A. Syropoulos
Greek Molecular Computing Group, Xanthi, Greece e-mail: asyropoulos@yahoo.com
Dynamical systems are characterized by equations that describe their evolution. A dynamical system is called linear when its evolution is a linear process. A process is linear when a change in any variable at some initial time produces a change in some variable at some later time, however, if the initial variable changes \( n \) times, then the new variable will change \( n \) times at the later time. In other words, any change propagates without any alterations. Any system that is not linear is called a nonlinear dynamical system [13]. A basic characteristic of these systems is that any change in a variable at some initial moment leads to a change to some variable at a later time, which is not proportional to the initial change. For example, the logistic map [12]

\[
x_{n+1} = rx_n(1 - x_n),
\]

where \( x_n \in [0, 1] \) is the magnitude of population in generation \( n \) and \( x_{n+1} \) the magnitude of population at generation \( n + 1 \), is a typical example of an equation that describes a nonlinear system. In this case, the system is the population of some species and the dynamics the changes from one generation to another.

Although a robotic system can be either linear or nonlinear, it seems that nonlinear systems are more interesting in terms of applications. A robotic system is called nonlinear when its control is not nonlinear. In particular, a control system is called nonlinear when it contains at least one nonlinear component [14]. For example, a soft robot [2], that is, a robotic system that consists of several deformable spherical components, is a nonlinear robotic system [15]. Unlike (some) rigid robots, a soft robot can in general go through or above an obstacle. Consider a robot, rigid or soft, that moves on a specific path. Assume that we assign to each obstacle which is on this path a penetrability degree. Then, the degree to which the robot will not deviate from its path to avoid the obstacle will depend on this degree. If the robot can go through the obstacle or above it, then we have a nonlinear system moving on a “vague” environment. Thus one can say that the motion of a soft robot can be described also by using fuzzy “mathematics” (i.e., a very popular mathematical formulation of vagueness).

The central problem of robotics is how to go from point \( A \) to point \( B \). As explained above, avoiding obstacles by deviating from a “predetermined” path is the “classical” way to solve this problem. However, this is not an interesting problem for us. We are interested in systems that can use an extended form of the motion planning algorithm able to describe robots that go through or above obstacles. But first, let us examine what is the “classical” motion planning algorithm.
1.2 Obstacle Avoiding: an up-to-date mathematical formulation.

Given a vehicle $V$, a starting point $A$ (usually called an initial configuration) and an ending point $B$ (called a final configuration), one can form the set $P$ of all paths that $V$ can follow, starting from $A$ and ending in $B$. Clearly, one can define a number of fuzzy subsets of $P$, for example, the fuzzy subset of easy paths, the fuzzy subset of smooth paths, etc. Obviously, the problem is how to choose a path in order to go from $A$ to $B$. This problem is called the motion planning problem [9].

A motion-planning algorithm [9] is a solution to the motion planning problem. Before giving a formal definition to this problem and to its solution, we describe these notions intuitively. The main task is to find a path starting at a point $A$ and ending at point $B$. The path has to avoid collisions with a known set of stationary obstacles. At any given moment, a robot moving on this path is on a specific robot configuration (i.e., a point of this path). In order to solve this problem one needs a geometric description of both the vehicle and the space where the vehicle moves. The configuration $q$ of a vehicle is a specification of the positions of all vehicle points relative to a fixed coordinate system. The configuration space is the space of all possible configurations.

Assume that $W \subset \mathbb{R}^3$ is the configuration space on which the vehicle moves, where $\mathbb{R}^3$ is the Euclidean space of dimension 3, and denote by $O \in W$ the set of all possible obstacles that the vehicle can meet. Such obstacles will be presented in terms of neighborhoods in $\mathbb{R}^3$. The expression $A(q)$ is used to denote that the vehicle is in configuration $q \in C \subseteq W$. Then,

$$ C_{\text{free}} = \{ q \in C \mid A(q) \cap O = \emptyset \} $$

$$ C_{\text{obs}} = C / C_{\text{free}}. $$

Let $q_S$ be the initial configuration and $q_G$ the final configuration. Then, the motion planning problem is the process of finding a continuous path $p : [0, 1] \to C_{\text{free}}$, where $p(0) = q_S$ and $p(1) = q_G$.

One approaches the motion planning problem using different tools and methodologies and, thus, there are different solutions to it. For example, Lozano-Pérez [10] presented a simple solution, Ashiru and Czarnecki [1] discussed motion planning using genetic algorithms and Farber [6] presented a probabilistic solution. Most of all these approaches assume that the vehicle should always avoid obstacles, but there has not been a study of cases where the vehicle can pass through (penetrate) an obstacle.
1.2.1 A Mathematical Formulation.

We will use Farber’s [6] notation and mathematical description of robot motion planning algorithm. For topological notions like path-connected spaces, compact-open topology, etc., see [5].

Let $X$ be a path-connected topological space and denote by $PX$ the space of all continuous paths. $PX$ is supplied with the compact open topology. Consider the map $\pi : PX \to X \times X$, which assigns to a path the pair $(\gamma(0), \gamma(1))$ of the so-called initial-final configurations. $\pi$ is a fibration in the sense of Serre.

**Definition 1.** A \textit{motion planning algorithm} is a section $s : X \times X \to PX$ of fibration, that is, $\pi \circ s = 1_{X \times X}$.

One of Farber’s research goals was to predict the character of instabilities of the behavior of the robot, knowing several topological properties of the configuration space, such as its cohomology algebra. Here we will not concern ourselves with this approach. We will stick in Farber’s declaration that there may exist a better mathematical notion of a configuration space, describing a partially known topological space, whose (geometric and topological) properties are being gradually revealed. We believe that fuzzy set theory is the key tool for this.

Farber introduced four numerical invariants $TC_i(X)$, $i = 1, 2, 3, 4$, measuring the complexity of the problem of navigation of a robot configuration space. These invariants coincide for “good” spaces, such us for simplicial polyhedra. We will now present $TC_4(X)$, for our purposes, since it is linked with random motion planning algorithms.

**Definition 2.** A random $n$-valued path $\sigma$, on a path-connected topological space $X$, starting at $A \in X$ and ending at $B \in X$ is given by an ordered sequence of paths $\gamma_1, \cdots, \gamma_n \in PX$ and an ordered sequence of real numbers $p_1, \cdots, p_n \in [0, 1]$, such that each $\gamma_j : [0, 1] \to X$ is a continuous path in $X$ starting at $A = \gamma_j(0)$ and ending at $B = \gamma_j(1)$, such that $p_j \geq 0$ and $\Sigma_{i=1}^n \gamma_i = 1$.

The notation $P_nX$, of Farber, refers to the set of all $n$-valued random paths in $X$. This set is a factor-space of a subspace of the Cartesian product of $n$ copies of $PX \times [0, 1]$.

**Definition 3.** $TC_4(X)$ is defined as the minimal integer $n$, such that there exists an $n$-valued random motion planning algorithm $s : X \times X \to P_nX$.

**Remark 1.** It has been proved that $TC_{n+1}(X) = \text{cat}(X^n)$, for $n \geq 1$, where $\text{cat}(X^n)$ is the Lusternik-Schnirelmann category [8]. These categories have been used to solve problems in nonlinear analysis (e.g., see [7]).
1.2.2 Remarks on this Formulation

No one can doubt the usefulness of Farber’s approach, both in the field of Topology and in Robotics. The instabilities in the robot motion planning algorithm are linked to topological invariants and the universe where the robot moves is seen through the eyes of a topologist who sees configuration spaces. When it comes to engineering though, an interpretation of the invariant $TC_4(X)$ is tough. What does it mean for a vehicle to take a random path? Is it better to talk about a plausible path? Moreover, instead of bypassing obstacles, can we assume that a robot can go through obstacles?

In what follows, we describe a fuzzy motion planning problem and explain how it can be solved. These ideas are explained practically and we conclude with some questions and problems related to this approach.

1.3 Questioning an Even More Theoretical Approach to Motion Planning Problem.

Here we ask for the possibility of investigating purely topological properties of robot motion planning algorithms via function spaces, based on the study in [4] and on the results by Farber. Considering a function space $F(X,Y)$, there are several topological problems one can study. Knowing topological properties of $X$ (or $Y$), what are the topological properties of $F(X,Y)$ and vice versa.

Let $X$ be an arbitrary topological space. Let $PX = C([0,1],X)$ be the function space of all continuous paths $\gamma : [0,1] \to X$, supplied with the compact-open topology. Let $\pi : PX \to X \times X$ be the map which assigns to a path $\gamma$ the pair $(\gamma(0), \gamma(1)) \in X \times X$ of the so-called “initial-final configurations”. Consider the function space $F(PX, X \times X)$. A motion planning algorithm is a map $s : X \times X \to PX$, such that $\pi \circ s = 1_{X\times X}$. Consider the function space $FM'(X \times X, PX)$, consisting of motion planning algorithms. Notice that this is a subspace of the function space $F(X \times X, PX)$.

Question 1

Farber questions under what conditions there exist motion planning algorithms which are continuous, and gives an answer through contractibility. More generally, add (the minimum number of) topological conditions on the function space $C(X \times X, PX)$, so that its functions to be motion planning algorithms, and thus study topological properties of the function space $CM(X \times X, PX)$ of continuous motion planning algorithms. Here we should remark that we did not recommend $X$ to be path-connected (which practically means that
one can fully control the system by bringing it to an arbitrary state from a given state) as an initial condition.

**Question 2**

Start with a topological space $X$, as the configuration space of a mechanical system, with no explicit information about its local or global topological properties. Apply Step 0 to Step $n$ of the construction given in [4], to the motion planning algorithms space $\mathcal{F}(X \times X, P_X)$. Study the possibility for the existence of a minimal integer $n$ “revealing as much as possible topological information about $X$”. This will give a partial answer to Farber’s question on robot motion planning algorithms, on whether there exists a way to study very complex configuration spaces which are gradually revealing their topological properties.

**Question 3**

Given answers to our Question 1, a further theory can be developed, studying the topological complexity of tame motion planning algorithms, in the language of function spaces (see [6]).

**Question 4**

If a space $X$ is path-connected, one can “fully control it”, in a sense that for any two fixed points there is a path joining them. One could define a topological space, so that for any two points $A$ and $B$ there exists a *linear ordered topological space* (lots) starting from $A$ and ending at $B$, and this would generalize path-connected spaces and furthermore motion planning algorithms.

Can one achieve this in a different way rather than refining the definition of a continuous path $\gamma$, by adding the extra property that the path $\gamma$ should be also order preserving (taking in $[0, 1]$ the natural order $<$)?

One can consider the space of all such lots on $X$, say $P_X$, mapped to $X \times X$ as a fibration $\pi$, and define a section $s : X \times X \to P_X$, such that $\pi \circ s = id_{X \times X}$. One could then study its Schwartz genus, as a notion of a topological complexity of $X$, and link notions of order theory and general topology to algebraic topological ideas.

There will be a problem if one considered an arbitrary lots. Consider for example the lots consisting of just two points can be mapped into any space $X$ with two points $A$ and $B$ and that mpa will be a homeomorphic embedding, if and only if $X$ is $T_1$. One does not want this sort of “teleporting” behavior to be possible, that perhaps one wants there to be many points linking $A$ to $B$ along what “resembles a path”. A general way to achieve this is to require that the lots to be a dense order. If one follows this route, it would be most natural to require paths to be closed subsets and the map to
be a homeomorphic embedding. Alternatively, one could fix a lots $L$ that is
to work for all pairs of points in the space:

1. when $L = \{0, 1\}$ then we have a $T_1$ space and
2. when $L = [0, 1]$, then we have a path-connected space.

What if $Y = \mathbb{Q} \cap [0, 1]$? What if $Y$ is the Cantor set $C$? What if $Y = \omega + 1$.
In either cases, the “interesting” spaces are going to be totally disconnected

1.4 Further Topological Remarks

For a more detailed discussion, see [6]. Here we add a few more questions of
topological nature.

Consider a path-connected topological space $X$. A random $n$-valued path
$\sigma$, in $X$, which starts at point $A$ and ends at point $B$, is given from a sequence
of paths $\gamma_1, \gamma_2, \ldots, \gamma_n$ which belong to $PX$ (the space of all continuous paths
on $X$) and a sequence of real numbers $p_1, p_2, \ldots, p_n$ in $[0, 1]$, such that every
path $\gamma_j : [0, 1] \to X$ is continuous, where $\gamma_j(0) = A$ and $\gamma_j(1) = B$ and also
$p_j \geq 0$ and $p_1 + p_2 + \cdots + p_n = 1$. From the third Axiom of probability theory,
one induces that $\sigma = p_1 \gamma_1 + p_2 \gamma_2 + \cdots + p_n \gamma_n$.

Consider now the map $\pi : P_n X \to X \times X$, where $P_n X$ denotes the set of
all random $n$-valued paths on $X$. An $n$-valued random algorithm is a map
$s : X \times X \to PX$, such that $\pi \circ s = 1_{X \times X}$.

In other words, if one considers the pair $(A, B)$ in $X \times X$ (input), the output
is an ordered probability distribution $s(A, B) = p_1 \gamma_1 + p_2 \gamma_2 + \cdots + p_n \gamma_n$, that
is the algorithm $s$ induces the path $\gamma_j$ with probability $p_j$.

A first question, is which probability distributions are outputs of such mo-
tion planning algorithms. It would be of a theoretical interest to characterize
probability distributions via motion planning algorithms. What about if the
number of paths is not countable? If one can define such motion planning
algorithm, then what kind of probability distribution can one expect as an
output? This is a good point to pass into the next section, which is the
approach to the motion planning problem through fuzzy logic.

1.5 Obstacle Avoiding: a Fuzzy Logic approach.

A fuzzy motion planning problem is a problem that asks how a vehicle
can move from a point $A$ to a point $B$ by possibly going through/climb
over/penetrate and so on, a number of obstacles, instead of avoiding them.
All obstacles, which are represented mathematically by neighborhoods, are
associated with a traversal difficulty degree that specifies how difficult it is
to go over a specific obstacle. This degree is a number drawn from $[0, 1]$ and
when it is equal to 1 for a given obstacle \( O \), this implies that \( O \) is actually not an obstacle. On the other hand, a traversal difficulty degree equal to 0 means that it is impossible to go over \( O \), so the robot will have to find ways to avoid it.

**Definition 4.** A fuzzy continuous path is a map \( p^{\lambda, \ell} : [0, 1] \to C \) that goes over obstacles \( O_1, \ldots, O_n \in C_{\text{obs}} \), where the traversal difficulty degree of each obstacle \( O_i \) is \( \lambda_i \), has a plausibility degree that equals \( \lambda = \min_{i=0} \lambda_i \) and its length is \( \ell \).

Clearly, the smaller the value of \( \lambda \) is, the less plausible a specific path is.

Figure 1.1 depicts a terrain with some obstacles. The vehicle’s task is to go from \( A \) to \( B \). Obviously, the dotted path is one that avoids all obstacles but it is quite long. On the other hand, the straight line is a path that goes over three obstacles but it is the shortest possible path. Thus, the ideal path is the one that it will be as short as possible and as easy to traverse as possible.

![Fig. 1.1 The problem of moving a vehicle from A to B and two possible solutions.](image)

**Definition 5.** A fuzzy \( n \)-valued path \( \sigma \), on \( X \), starting at \( A \in X \) and ending at \( B \in X \) is an ordered sequence of paths \( p_1^{\lambda_1, \ell_1}, p_2^{\lambda_2, \ell_2}, \ldots, p_n^{\lambda_n, \ell_n} \in PX \), where

\[
\sigma = \min_{\ell_i} \max_{\lambda_i} p_i^{\lambda_i, \ell_i}, \forall i = 1, 2, \ldots, n.
\]

Assume that \( P_n X \) is the set of all fuzzy \( n \)-valued paths. Then, the function:

\[
\pi : P_n X \to X \times X
\]

maps to a fuzzy path its starting and end points.
Definition 6. An $n$-valued fuzzy motion planning algorithm is defined as the map:

$$s : X \times X \rightarrow P_n X.$$ 

Thus, the algorithm is a two-fold process: first it identifies $n$ distinct paths and it then chooses the most plausible one, not just someone “in random”.

Remark 2. The function $s$ is a continuous section of the fibration $\pi$.

Having given the above definition of an $n$-valued fuzzy motion planning algorithm, we now have a clearer picture of how one can define an invariant, similar to $TC^4(X)$ but more realistic, describing its navigational complexity. Let us call such an invariant $TC^\ast_4(X)$. This invariant will depend on both parameters $\lambda$ and $\ell$ of Definition 5. So, it will be sufficient to declare it as the “smallest integer $n$, such that an $n$-valued fuzzy motion planning algorithm exists”. $TC^\ast_4(X)$ certainly describes a wider range or properties of the configuration space. Sometimes, in real situations, it will be better to go through an obstacle, e.g. a vehicle towards water, provided that in such a way $\ell$ is small, even if $\lambda$ is small too. A mission running out of time, for example, will put a vehicle into such a risk. In other cases, it might be better for $\ell$ to be big in order $\lambda$ to be big, too; for instance, a short distance and a harsh obstacle might put the vehicle into a great risk or might force it to spend a sufficiently big amount of fuel, etc.

An Example

Imagine that a vehicle, like NASAs Curiosity, is on the surface of planet Mars. Assume that this vehicle can recognize obstacles and it can assess whether it is possible to go over an obstacle or not. For example, the rover might have access to an on-board databank with pictures of obstacles, which have been rated somehow (e.g., by a human expert), and using some sort of object recognition algorithm, then it can assign traversal difficulty degrees to various objects and so it can “deduce” whether a specific path is traversable or not. More generally, the vehicle can perform this action several times to find different traversable paths and to choose the best path. Of course, the system should be able to retract and make another choice since it is quite possible that some initial estimation was more vague than expected.

1.6 Soft Robots or Fuzzy Motion Planning Algorithms?

On the one hand each obstacle in the path of a robot can be associated with a number that will show to what extend it is possible to go through or above the obstacle but on the other hand we have soft robots that are able to go through obstacles. What is really missing here is that even for soft robots
it would not be absolutely sure that one can go through a specific obstacle. Thus even for soft robots, each obstacle should be associated with a number whose value would indicate to what degree it is possible to go through it. In different words, the behavior of soft robots can be better described with the use of fuzzy set theory. Let us roughly describe how this can be realized.

First we chose the path our robot with follow. Then we assign to each obstacle an “absolute” traversal degree, as if our robot is a rigid one. Depending on the shape of the robot and how flexible it is, we modify the absolute traversal degree so to take into account the capabilities of the soft robot. The modified traversal degrees can be used to define a fuzzy motion planning algorithm. The interest thing here is that the dynamics of the robot are nonlinear and we can use fuzzy sets to-described a motion planning algorithm.

1.7 Conclusions and Open Questions.

fter describing the motion planning problem problem, we briefly discussed a more “realistic” solution and commented on its unsuitability. Next, we presented a formulation of the problem that uses “vagueness” and proposed a solution that makes use of fuzzy set theory. The result is more natural as it coincides with the procedure that humans follow in order to choose the most suitable path. We then gave a first comparison of the fuzzy formulation with that one that uses soft robots. Here we list a list of open problems which, in our own opinion, are interesting both from a theoretical perspective as well as in applications.

1. Implement the methodology given in the section “Obstacle Avoiding: a Fuzzy Logic approach” with simulation(s) and (an) experiment(s), and see how it works in practice, comparing it with a similar methodology referring to soft-robots.

2. Nonlinear analysis has been used to analyze fuzzy systems (e.g., see [3]). Also, tools used to analyze fuzzy systems have been used to analyze nonlinear systems (e.g., see [11]). The question is: Can use use both methodologies to assist us to build and test a flexible robot?

References

1. Ashiru, I., Czarnecki, C.: Optimal Motion Planning for Mobile Robots Using Genetic Algorithms. In: Industrial Automation and Control, 1995 (I A amp; C’95), IEEE/IAS International Conference on (Cat. No.95TH8005), pp. 297–300 (1995)

2. Cecilia Laschi and Barbara Mazzolai and Matteo Cianchetti: Soft robotics: Technologies and systems pushing the boundaries of robot abilities. Science Robotics 1(1) (2016). DOI: 10.1126/scirobots.aah3690
3. David F. Jenkins and Kevin M. Passino: An introduction to nonlinear analysis of fuzzy control systems. Journal of Intelligent and Fuzzy Systems 7, 75–103 (1999)
4. Dimitris Georgiou and Athanasios Megaritis and Kyriakos Papadopoulos and Vasilios Petropoulos: A study concerning splitting and jointly continuous topologies on $C(Y, Z)$. Science Robotics 39(3), 363–379 (2016)
5. Engelking, R.: General Topology. Heldermann Verlag, Berlin (1989)
6. Farber, M.: Topology of Robot Motion Planning. In: P. Biran, O. Cornea, F. Lalonde (eds.) Morse Theoretic Methods in Nonlinear Analysis and in Symplectic Topology, NATO Science Series II: Mathematics, Physics and Chemistry, vol. 217, pp. 185–230. Springer Netherlands (2006)
7. Felix E. Browder: Lusternik-Schnirelman Category and Nonlinear Elliptic Eigenvalue Problems. Bulletin of the American Mathematical Society 71(4), 644–648 (1965)
8. Gregory Lupton and Jerome Scherer: Topological Complexity Of H-Spaces. Proceedings of The American Mathematical Society 141(5), 1827–1838 (2013)
9. Jean-Claude Latombe: Robot Motion Planning. Springer New York (1991)
10. Lozano-Pérez, T.: A Simple Motion-Planning Algorithm for General Robot Manipulators. IEEE Journal of Robotics and Automation 3(3), 224–238 (1987)
11. Mei, F., Man, Z., Nguyen, T.: Fuzzy modelling and tracking control of nonlinear systems. Mathematical and Computer Modelling 33(6), 759–770 (2001)
12. Robert M. May: Simple mathematical models with very complicated dynamics. Nature 261, 459–467 (1976)
13. Sastry, S.: Nonlinear Systems: Analysis, Stability, and Control. Springer-Verlag New York (1999)
14. Slotine, J.J.E., Li, W.: Applied Nonlinear Control. Prentice Hall, Englewood Cliffs, New Jersey 07632 (1991)
15. Yanqiong Fei and Xingyao Shen: Nonlinear analysis on moving process of soft robots. Nonlinear Dynamics 73(1), 671–677 (2013)