Fuzzy Logic for Mobile Robot Navigation Applications

Ben Church – 10006197

**Abstract:** Mobile robot navigation is a mature field with many applications in industry and academia, and with even more approaches to implementation than there are applications. Fuzzy logic has attracted research attention within the field for allowing natural navigation processes to be expresses and implemented with familiar human language. Fuzzy logic’s applicability to a wide range of navigation topics has caused a similar proliferation of work within the sub-field of fuzzy logic for mobile robot navigation. While the natural familiarity of fuzzy rule based navigation processes make them an attractive option to a designer wishing to implement navigation functionality, the breadth of topics and approaches can make their application daunting. This survey presents an analysis of mobile robot navigation functionality, breaking it into processes and sub-processes which can be implanted in a modular fashion. Approaches to these processes and sub-processes from literature are explored to provide the designer with tools which they may pick and choose to design general navigation schemes.

Contents

[1. Introduction 1](#_Toc479963878)

[Motivation 1](#_Toc479963879)

[Scope 1](#_Toc479963880)

[Background 2](#_Toc479963881)

[Proposed Framework 3](#_Toc479963882)

[2. Perception 4](#_Toc479963883)

[Measurement 4](#_Toc479963884)

[Heading 4](#_Toc479963885)

[Range 5](#_Toc479963886)

[3. Modelling 5](#_Toc479963887)

[Measurement to Features 5](#_Toc479963888)

[Obstacles from range disjunction 6](#_Toc479963889)

[Desired area from GPS heading 6](#_Toc479963890)

[Terrain roughness from vision 6](#_Toc479963891)

[Features to Model 7](#_Toc479963892)

[Obstacles and goal-direction to desired direction 7](#_Toc479963893)

[Roughness, tilt, and discontinuity to traversability 7](#_Toc479963894)

[4. Behavior 8](#_Toc479963895)

[Behaviors for functionality 8](#_Toc479963896)

[Goal seeking and obstacle avoidance for end-goal navigation 8](#_Toc479963897)

[End-goal navigation with wall following and emergency situation behaviors 8](#_Toc479963898)

[Output behavior control 8](#_Toc479963899)

[Fuzzy context behavior supervision 9](#_Toc479963900)

[5. Summary 10](#_Toc479963901)

[6. References 11](#_Toc479963902)

# Introduction

## Motivation

Mobile robot navigation is a broad field with around a century of invested research dating back to early autopilot systems, depending on what exactly one considers as the first mobile robot with navigation capabilities. We now profit from many applications using this technology such as scientific exploration with rovers, autonomous warehouses, and modern autopilot systems. The self-driving car is an example of how mobile robot navigation technology could impact many private lives, if present trends in the industry continue. While some may find the scope of its applications inspiring, the number of problems in mobile robot navigation may be daunting to potential innovators. This paper is written to the end of bridging the gap between user and designer, exploring mobile robot navigation functionality based on common-sense human experience, rather than various abstract mathematical treatments of the navigation problem removed from human experience.

Fuzzy logic applications are chosen as the focus of this paper because the author believes that mobile robot technologies stand to benefit from integration of a fuzzy rule-based programming interface, suitable for programming by average consumers. Simplified mobile robot customization through a fuzzy programming language consisting of perhaps several verbs, measurements, logical operators, and qualitative magnitudes, could offer typical consumers more sophisticated or better tuned robot behavior than technically knowledgeable professionals can pre-program. The author believes further that a guide matching mobile robot navigation problems with fuzzy logic solutions could be informative in designing such a fuzzy programming interface.

## Scope

While greater user freedom through simple programming is desirable, the designer may not wish to make all functionality open for customization. For example, drone hover stability controls are unlikely to improve with user tuning, whereas a user may wish to change the relationship between their robot’s velocity and the distance it follows the user from. The particular robot functionality made customizable through a fuzzy programming interface will always depend on the total functionality available for customization, determined by the application, and individual designer choices. This work is not meant to indicate when functionality ought to be implemented with fuzzy logic or be made user-programmable, much less to explore fuzzy navigation solutions in an exhaustive manner. It is instead intended as a reference, a toolbox, for designers who wish to implement navigation functionality with fuzzy logic, with consideration given to how such functionality could be made user-programmable.

To this end, the Background section provides an overview of mobile robot navigation, while it is assumed that the reader is familiar with fuzzy logic. The overview introduces navigation models, behaviors, and problems from literature. It serves to define concepts explored in the context of fuzzy logic by works considered throughout the rest of the paper, and to inspire the organization of our fuzzy navigation framework. The navigation framework is then presented in the Proposed Framework section, where its organization is described in reference to the ideas from mobile robot navigation explored in the Background section. Subsequent sections explore works which deal with various aspects of navigation using fuzzy logic, and their approaches are conveyed in our proposed framework. The approaches are summarized under their corresponding topics in the Summary section for convenient reference.

## Background

We can consider mobile robots as engineering systems like any other. A set of measurements taken from the environment are the inputs which determine an output, a chosen action. To achieve useful, sustained navigation behavior, this systemic conception of robot behavior may be incorporated into a closed-loop control system process. The closed-loop aspect of the system means that the robot determines subsequent action based on how the relationship between itself and the environment has changed as a result of previous behavior. Arkin represents behavior using a cognitive-inspired schema flowchart [1], which amounts to a closed-loop control system using processes and information familiar to human experience. Figure 1 shows a general action-perception cycle, extended to include perception as a process.

Navigation may be seen as a special case of an action-perception cycle where the changes caused by behavior are limited to changes in the robot’s spatial relationship with the environment. Many fuzzy logic solutions compartmentalize the navigation problem into such processes, more or less explicitly, therefore the proposed programming framework should do so as well.

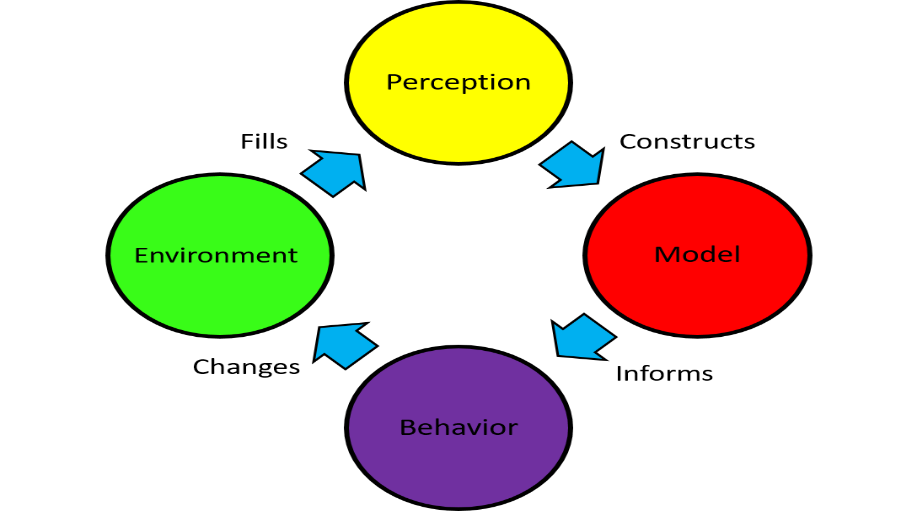


Figure : Action-perception cycle for environmental interaction

The environment itself exists independently of the robot, and therefore cannot be incorporated into a navigation scheme, fuzzy or otherwise. This leaves three processes in the cycle for exploration. However, the distinction between perception and the internal model of the environment is not immediately clear; at which point is information taken from the environment considered a model thereof? For the purpose of organization, we consider perception as sensor readings which have not to be interpreted in the context of other readings. The environmental model begins when perception data is integrated and taken in context. A robot with a single sensor constitutes a special case; where no measurement context is possible, the robot’s perception is equivalent to its model of the environment. Solutions are more likely to fall under modelling than perception because of how they are classified here, however applications for both may be considered. This leaves the behavior component, which has received much attention in research due to its breadth, difficulty, and the natural suitability of fuzzy logic to its problems.

The modelling component of navigation may be further divided into producing features from environmental measurements the robot perceives, and combining features into the robot’s complete environmental model. The distinction here between features and models is similar to the distinction between the perception and modelling processes; a feature is homogeneous in its dimensions, which may be quite abstract, while the model is the result of the synthesis of all features under consideration. For example, multiple adjacent range sensor readings can be perceived and grouped into an obstacle feature. The robot may construct multiple, separate obstacle features from additional sensor readings. A model could be the combination of these obstacle features. It is from this model that behavior is determined.

Robot control schemes are designed based on desired functionality, and functionality can be described in relatable terms such as “exploration” or “end goal seeking”. Multiple behaviors may be made available to the robot to implement a given functionality. Behaviors can likewise be described in recognizable language such as “wall-following” or “obstacle avoidance”. This use of behaviors breaks the higher level functionality down into components which can be implemented more easily since they are smaller, less abstract tasks. Working in the other direction, behaviors can be combined into new ways to produce custom functionality, providing a level of abstraction above hardware action for programming. This treatment of navigation as behaviors was used by authors such as Brooks [2] who used behavior subsumption, selecting the single most important behavior to act upon, to address problems such as behavior selection or arbitration. The programming framework should be able to treat each of these levels of behavior programming: Which behaviors, when made available to the robot are sufficient, in some combination, to implement the desired functionality, and determining how to obtain a single, actionable output from these multiple behaviors.

## Proposed Framework

To the end of producing a useful guide to implementing fuzzy mobile robot navigation, the solutions and approaches surveyed in this paper are categorized under perception, modelling, or behavior processes. Solutions are further categorized into components of these processes because of the natural modularity of the processes’ building blocks. This provides a convenient organization scheme which presents solutions as they relate to the particular aspect of navigation which the designer wishes to make available to the user for fuzzy programming. This organization scheme is shown in Figure 2.

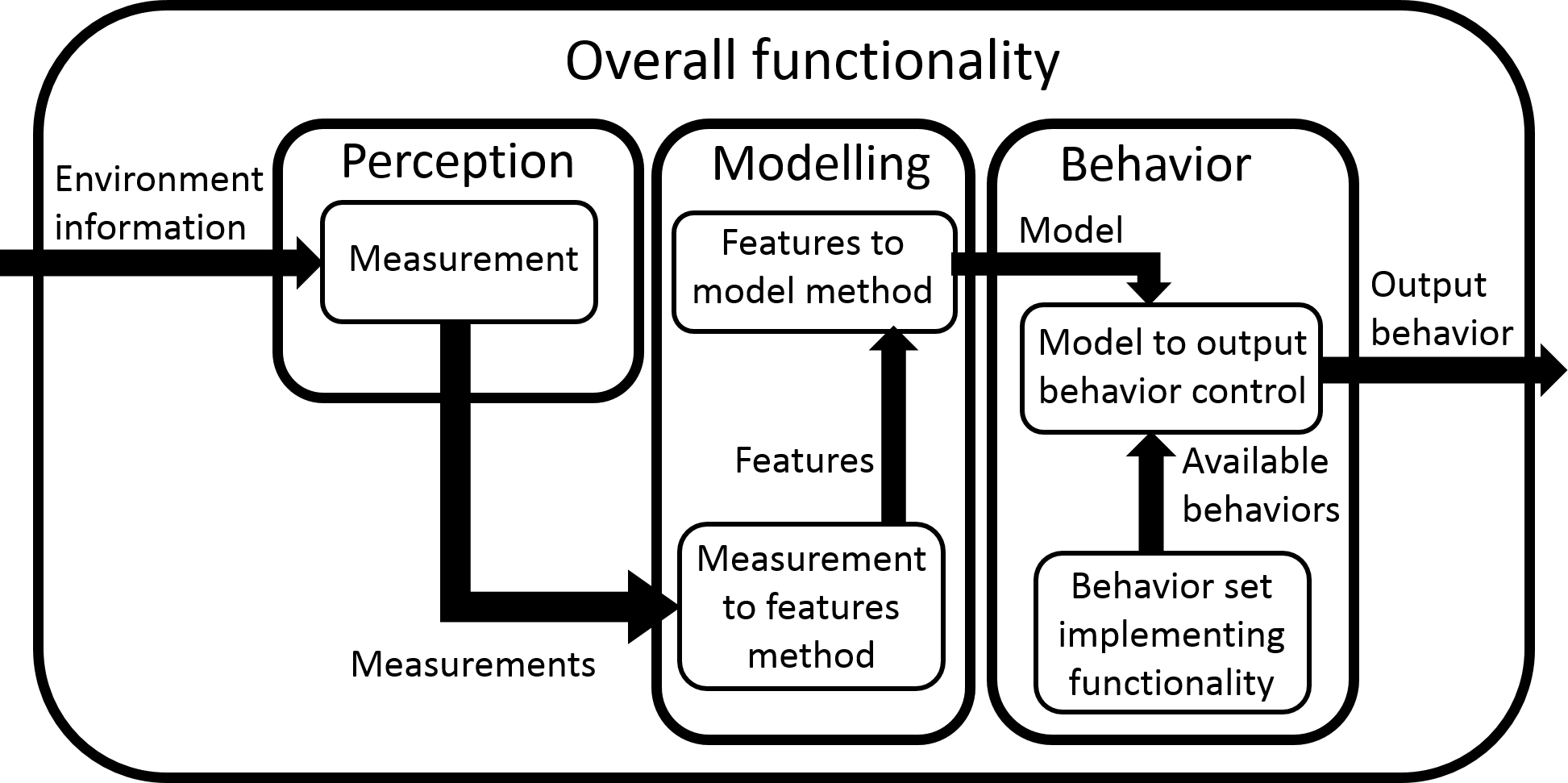


Figure : Topic categorization scheme, organized as a robot navigation process model

This is essentially a more detailed navigation model than shown in Figure 1. The process in Figure 2 is still closed loop, when the omitted environment is considered. The environment relates output behavior to environmental change relative to the robot, which is then perceived, closing the loop. The rounded rectangles in Figure 2 represent navigation process components which may be implemented in a fuzzy manner, and are therefore surveyed here. The arrows represent data flow from between modules. Individual works often present solutions or approaches to multiple components of this framework. This is because a functional navigation implementation requires multiple components, conceptualized in this framework or another, and authors are typically providing insufficient information if they address only one such topic.

# Perception

Works which employ a fuzzy treatment of perception separately from modelling are somewhat rare because of how the two are distinguished in this organization scheme. Because of our definition of perception pertains to measurements of environmental information, the process consists of only one component, the type of measurement performed. In fact fuzzy perception is little more than the determination of membership set values using a physical measurement as the crisp input. For the similarity of this topic to fundamental fuzzy logic theory, it is treated briefly. Integrating the resulting fuzzy set memberships, whatever measurement they fuzzyify, is treated in the Modelling Section. The final interpretation of the fuzzy set memberships will often fall under Behavior.

## Measurement

Measurement is characterized by the physical quantity which is being measured. For mobile robot navigation, this is almost invariably a linear range, or angular heading. Exceptions exist, usually in a non-fuzzy context, such as Xu *et al.* [3] and Memon *et al.* [4] who used infrared thermal sensors to detect fires with mobile robots. Since navigation is often implemented with multiple range sensors and a single GPS for perception, fuzzy treatment of angular headings exclusively at the perception level are more common than treatment of range at this level. The following treatment of fuzzy heading and range measurements should generalize to other quantity measurements.

### Heading

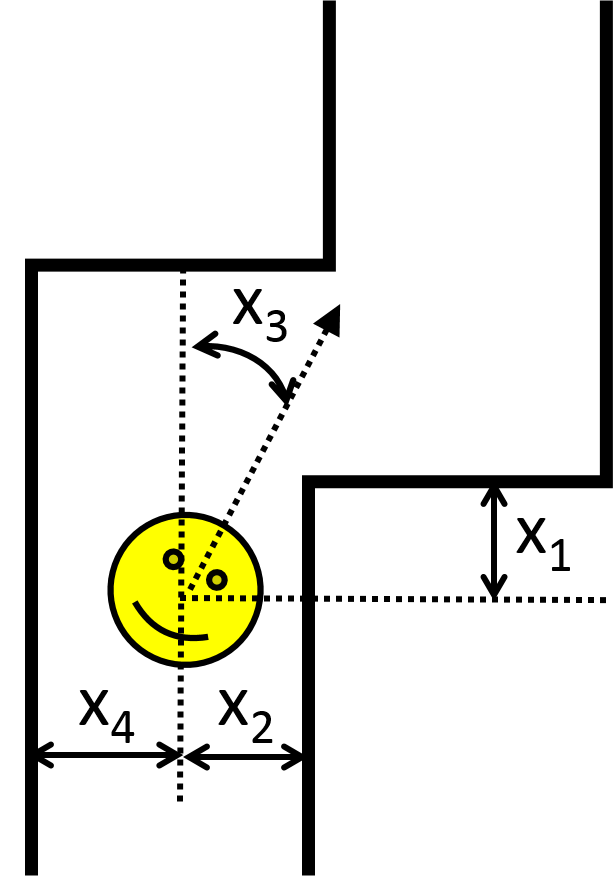


Figure : Model of the arena used by [5]

Perhaps because it is one of the earliest papers on fuzzy mobile robot navigation, Sugeno and Nishida [5] provide some treatment of fuzzy navigation at the perception level. They designed a control scheme to navigate a robotic model car through a crank-shaped course. Measurements were captured from ultrasonic range sensor readings. Measured distances were captured directly while the heading angle was deduced from a rotating range sensor. Their treatment of measured distances is similar to other sources, and their treatment of angular heading, so they are mentioned only briefly to explain their approach.

The measurement quantities, shown in Figure 3, all had corresponding fuzzy sets. The measurement values determined the crisp input values to these sets. For example, the heading variable, x3, determined the membership values of Out, Forward, and In sets, shown in Figure 4. The particular nature of the course allowed these quantities’ rules to produce the output steering angle’s fuzzy set’s membership immediately. The simplicity of the application necessitated only two rules, using only x3 and x4 for wall avoidance, but 18 rules using all measurement except x4 for smooth steering. The nature of the steering sets, and the operations used to compute them from the rules are not provided. The designer may experiment with these mathematical details to tune performance.

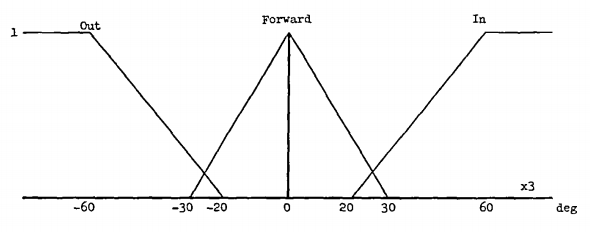


Figure : Angular heading fuzzy sets from [5]

### Range

In a more recent work, Li and Wang [6] used fuzzy perception to navigate a robot in and around 90° corners, both concave and convex. The specific nature of the navigation problem allowed them to use fuzzy navigation at the perception level. Range measurements were taken from 3 laser range sensors, arranged on a nonholonomic (non-symmetric, subject to rolling steering dynamics) mobile robot as shown in Figure 5. The Front and Rear range sensors had corresponding Near, Mid, and Far fuzzy range sets, such as those shown in Figure 6. Their fuzzy set values are then used with the front range sensor reading to determine whether the robot is in a concave corner, or outside a convex one.

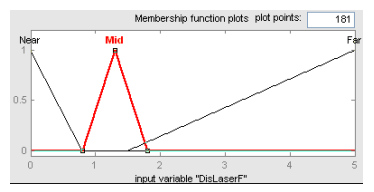


Figure 6: Sample of fuzzy range sets, taken from [6]. “DisLaserF” corresponds to the “Front” sensor in Figure 5.

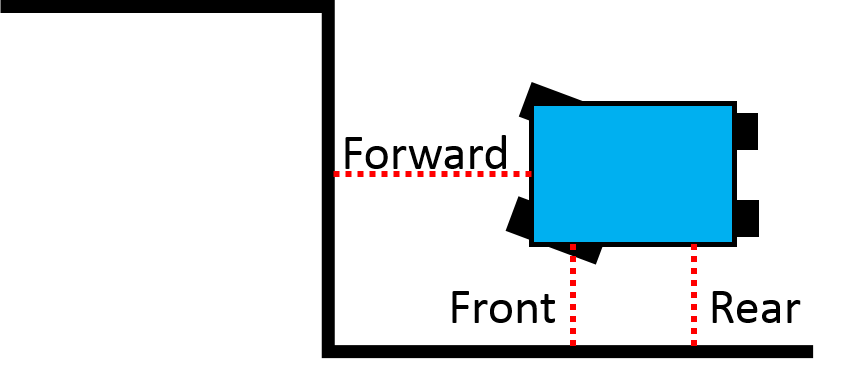


Figure 5: Robot and range sensor layout used by [6]

# Modelling

Once physical aspects of the environment are perceived, in a fuzzy manner or not, they may be combined or interpreted together to construct environmental features. A common environmental feature dealt with in literature is the obstacle, a physical barrier to motion. Other, more abstract features include terrain rockiness.

## Measurement to Features

This subsection deals with the construction of features from measurements. Robots must recognize features such as obstacles either as a component of, or the entirety of, the environmental model from which it will later determine behavior.

### Obstacles from range disjunction

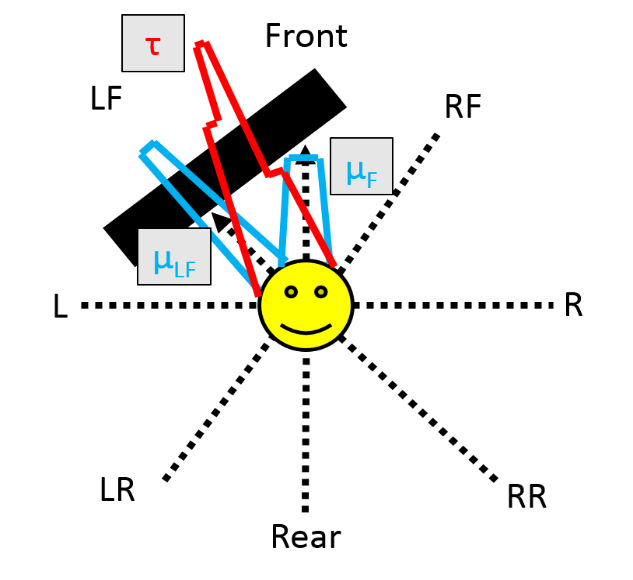


Figure : Robot sensor layout used by [7]. Sensor fuzzy sets are shown in blue, the resulting obstacle set, τ, in red.

Naturally, navigation often entails recognizing physical obstacles around the robot and avoiding them. Yang *et al.* [7] describe a method for representing environmental obstacles using fuzzy sets. Each fuzzy set corresponds to a direction about a radially symmetric robot with a range sensor in each such direction, as shown in Figure 7. Each sensor has a fuzzy set whose membership value depends on the activation of that sensor, indicating the proximity of the obstacle. The sensor sets are combined into obstacle feature sets with a fuzzy disjunction operation. The response of these fuzzy sets is exemplified is Figures 7 and 8. These obstacle sets are used later in [modelling](#_Obstacles_and_goal-direction).

### Desired area from GPS heading

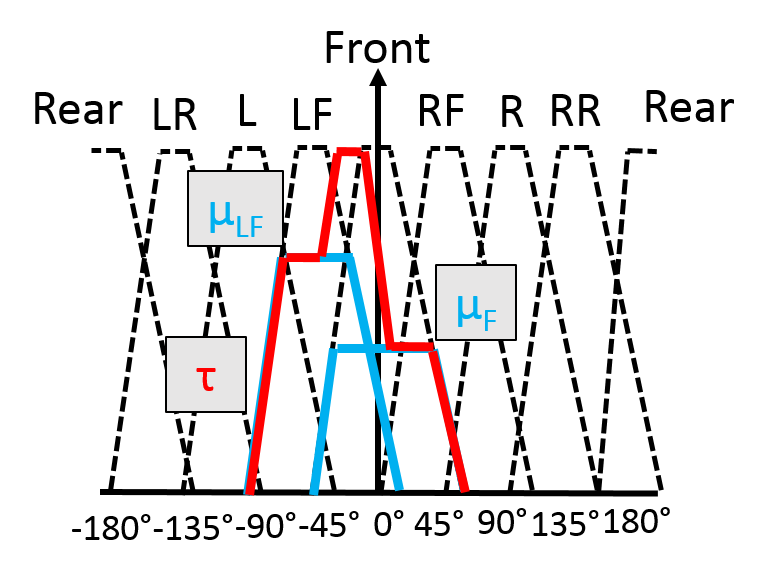


Figure : Sensor fuzzy set response corresponding to situation in Figure 7. Obstacle feature set τ derived from sensor set disjunction also shown.

Yang *et al.* [7] sought to have their robot navigate to an end-goal point. A GPS was used to provide the robot with its current angular heading relative to the direction of the end-goal. This angle determined the memberships of Left, Front, Right, and Read direction sets. See [Heading](#_Heading) for a fuzzy treatment of angular perception. The fuzzy direction sets representing the direction of the end-goal were aggregated into a desired area set, Ω, as in equation 1:

(1)

where u1 and μ2 and the fuzzy membership functions for the two direction sets which the heading lies between. This desired area was combined with a traversable area set for an environmental model representing the area which is both desired and traversable.

### Terrain roughness from vision

Seraji and Howard [8] proposed a navigation scheme allowing mobile robot navigation in more natural environments. For example, a Mars rover must navigate terrain with varying degrees of rockiness and slope, unlike a robot on a warehouse floor. They used a stereoscopic computer vision system for perception, rather than the usual range sensors. The vision system computed an estimate of typical rock size and rock concentration in view. The rock size measure was the input to Small and Large fuzzy membership functions. The rock concentration determined membership values for Few and Many sets. The set of four possible fuzzy inference rule combinations was used to determine membership values for Smooth, Rough, and Rocky sets. These sets could be defuzzified to obtain a crisp measure of terrain ‘rockiness’, but they used it with other fuzzy sets to model terrain traversability, [described](#_Roughness,_tilt,_and) in the Features to Model subsection.

## Features to Model

The Features to Model subsection addresses ways in which disparate features are combined into a final model of the environment.

### Obstacles and goal-direction to desired direction

Yang *et al.* [7] combined their obstacle features with a fuzzified representation of the direction of the global goal relative to their heading. The obstacle feature’s sets were aggregated into the overall untraversable area set, Γ, with the function in equation 2:

(2)

where τi is obstacle feature set i out of n. The traversable area set Γ was combined with a fuzzy set Ω (see equation 1), representing the desired direction of travel based on a GPS heading relative to the end-goal direction. They were combined with a *min* t-norm fuzzy conjunction operator, as in equation 3:

(3)

where γ ̂ represents the area which is both traversable and desired. This was then defuzzified to obtain the angle of steering for a mid-range sub-goal point.

### Roughness, tilt, and discontinuity to traversability

Seraji and Howard [8], in addition to considering [terrain roughness](#_Terrain_roughness_from) for natural environment navigation, used stereoscopic computer vision to measure regional ground tilt relative to the robot. The tilt measure was the crisp input value for Flat, Sloped, and Steep fuzzy sets. Roughness and tilt provide useful information about whether the robot will be able to negotiate the terrain, however they are not always sufficient. Special cases occur when two patches of similar roughness and slope are separated by a ditch or cliff. The computer vision may recognize the regions as homogeneous; the discontinuity may be insufficient to deter the robot if it is only treated as a small region of high roughness. Therefore, the authors introduced a discontinuity measure, a crisp value estimating the distance between adjacent areas. The discontinuity measure was the input to Small and Large fuzzy sets. The roughness, tilt, and discontinuity sets were used in a series of fuzzy rules to determine memberships for High, Medium, and Low traversability sets. Traversability was determined in this manner for each of seven sectors in front of the robot, as shown in Figure 9. The methods used to determine the crisp tilt and discontinuity measures use computer vision, not fuzzy logic, and so are beyond the scope of this work. Seraji and Howard [8] refers the reader to the appropriate references for details on these methods.

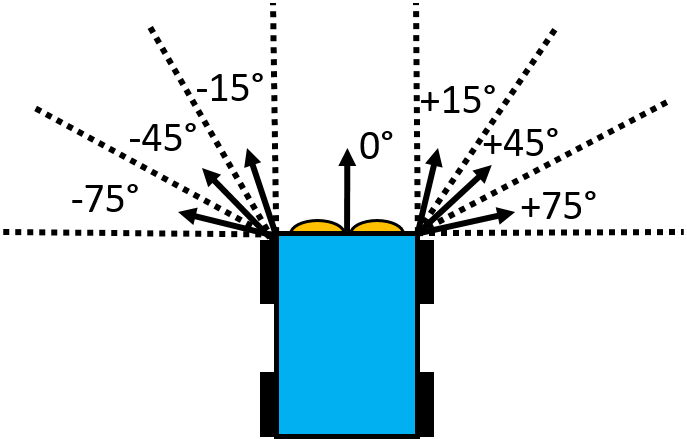


Figure : Robot layout used by Seraji and Howard [8]. Sectors each corresponds to a fuzzy traversability set

# Behavior

## Behaviors for functionality

Within general mobile robot navigation, there are numerous, more specific, functionalities which authors have sought to implement. Navigating an unknown environment to an end-goal point is a common one. Exploring and mapping the robot’s environment is another. Although navigation behaviors used to produce to desired functionality are themselves not intrinsically fuzzy, behavior arbitration schemes may be. Different combinations of behaviors, used to implement different functionalities are treated briefly here for their modularity and extensibility, especially when fuzzy logic is used to produce a meaningful output action from multiple behaviors.

### Goal seeking and obstacle avoidance for end-goal navigation

Ravangard [9] used perhaps the simplest behavior set possible to achieve end-goal navigation functionality. Goal seeking behavior was used to compel the robot to fulfill its desired purpose, by moving it towards the goal. Obstacle avoiding behavior was used to keep the robot from behaving in undesired ways, i.e. by colliding with obstacles. The goal seeking behavior was implemented in its usual manner where GPS based heading and distance measures as inputs to fuzzy membership sets which determine the direction desired by the goal seeking behavior. Obstacle avoidance used radially arranged range sensors, as did Yang *et al.* [7], to detect and avoid nearby obstacles.

### End-goal navigation with wall following and emergency situation behaviors

In addition to the usual goal seeking and obstacle avoidance behavior for end-goal navigation functionality, Dongshu *et al.* [10] incorporated wall following and emergency situation behaviors. Wall following is a natural addition to a robot with obstacle avoidance behavior. While simple obstacle avoidance may be sufficient to prevent collisions, wall following can allow robots to navigate past obstacles more quickly, such as when a robot finds itself too near to a large obstacle for steer-past-the-edge avoidance. Even with both obstacle avoidance and wall following, certain situations can still cause navigational failure. Consider the situation in Figure 10, where the robot has navigated into a narrow dead-end. The robot may not have room to follow a wall to escape, and obstacle avoidance behavior typically does not recognize the need for backwards motion. When a robot’s range sensors indicate this kind of scenario, special emergency situation behavior can solve the problem. The exact action the robot will use depends on the contingency situations the designer wishes to account for. A “trap” situation, like Figure 10, is usually addressed with backwards motion followed by a turn.

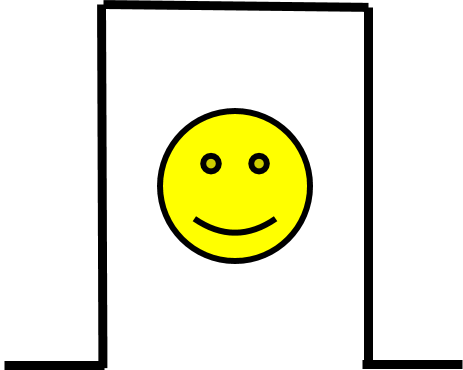


Figure : An emergency situation which may cause collisions despite wall following or obstacle avoidance behavior

## Output behavior control

When multiple behaviors are used by a robot to achieve some navigation functionality, the actions dictated by each behavior will not necessarily agree. For example, goal seeking behavior might direct the robot forward, but an obstacle in the path with cause obstacle avoiding behavior to try to take the robot in another direction. Without some means of obtaining a single actionable command from multiple behaviors, the robot may not perform in a desirable manner. Rosenblatt and Payton [11] recognize this need and have multiple behaviors vote on the preferred action from the robot, in a non-fuzzy manner, reminiscent of neural networks.

### Fuzzy context behavior supervision

For end goal navigation functionality, Al Yahmedi and Fatmi [12] coordinated wall following and goal reaching behaviors with a fuzzy rule context supervisor layer. The supervisor layer consisted of a set of fuzzy rules using fuzzified range sensor readings from directions all around the robot and end-goal heading and distance errors. Table 5 shows several examples of such rules, and although they also implemented obstacle avoiding and emergency behavior, no examples of their context priority rules are given. The fuzzy sets used as predicates in the rules in Table 5 are Near (N), Medium (M), and Far (F) for the six range sensors and end-goal distance, Drg. The angular fuzzy sets for the heading error θerror, are Positive (P), Small Positive (SP), and Zero (Z), and similarly for negative heading errors.

Table : Priority context fuzzy rules acting on sensor readings to determine which behavior(s) to use

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Cases | RD | RU | FR | FL | LU | LD | θerror | Drg | Behavior |
|  |  | F | F | F | F |  |  |  | Goal reaching |
|  |  |  |  |  |  |  | N |
|  |  |  |  | F | F | F | SP |  | Goal reaching |
|  |  |  | F | F | F | P |  |
|  |  |  | F | F | F | Z |  |
|  |  |  | N | N |  |  | SN |  | Fusion |
|  |  | N | N |  |  | N |  |
|  |  | M | M |  |  | SN |  |
|  |  | M | M |  |  | N |  |
|  | F | F | F |  |  |  | SP |  | Goal reaching |
| F | F | F |  |  |  | P |  |
|  | M | M |  | F |  |  |  |  | Wall following |
|  |  |  | F |  | M | M |  |  | Wall following |

# Summary

This work has identified three principal processes of mobile robot navigation and analyzed these processes into components. The components were identified in a way intended to allow for their modular implementation through various fuzzy-logic approaches explored in this work, or through the designers choice of non-fuzzy means, outside the scope of this guide. The proposed navigation topic organization scheme is shown in Table 2.

Table : Summary of proposed topic organization scheme with authors' approaches to different navigation problems.

|  |  |  |  |
| --- | --- | --- | --- |
| **Process** | **Component** | **Approach** | **References** |
| Perception | Measurement | Angular heading fuzzification | [5, 7] |
| Linear range fuzzification | [5, 6] |
| Modelling | Measurement to features method | Obstacles from range disjunction | [7] |
| Desired area from GPS heading | [7] |
| Terrain roughness from vision | [8] |
| Features to model method | Obstacles and goal-direction to desired direction | [7] |
| Roughness, tilt, and discontinuity to traversability | [8] |
| Behavior | Behaviors for functionality | Goal seeking and obstacle avoidance for end-goal navigation | [9, 10] |
| End-goal navigation with wall following and emergency situation behaviors | [10] |
| Output behavior control | Fuzzy context behavior supervision | [12] |

The approaches listed and categorized in Table 2 provide the designer with a set of tools suitable to modular implementation of mobile robot navigation using fuzzy logic. Using these approaches as they are described throughout the work may permit the designer to incorporate end-user programmability into the robots. Fuzzy sets for any of the measurement perceptions could be introduced or removed as new sensors are added to a robot, for example. Or the geometric parameters of fuzzy sets used to relate behavior to environmental context could be tuned for custom performance. The designer may choose to make none of the navigation modules user-programmable, if it is not expected to profitable or safe. This tool set is intended only to enhance design creativity and productivity, not to impose constraints on either.

# References

[1] R. Arkin, “Motor Schema Based Navigation for a Mobile Robot: An Approach to Programming by Behavior”, IEEE International Conference on Robotics and Automation, 1989.

[2] R. Brooks, “A Robust Layered Control System for a Mobile Robot”, IEEE Journal of Robotics and Automation, vol. RA-2. pp. 14-23, 1986.

[3] H. Xu, H. Chen, C. Cai, X. Guo, J. Fang, and Z. Sun, “Design and Implementation of Mobile Robot Remote Fire Alarm System”, International Conference on Intelligence Science and Information Engineering, 2011.

[4] S.F. Memon, I.H. Kalwar, I. Grout, E. Lewis, and Y.N. Panhwar, “Prototype for Localization of Multiple Fire Detecting Robots in a Dynamic Environment”, International Conference on Computing for Sustainable Global Development, 2016.

[5] M. Sugeno, and M. Nisida, “Fuzzy Control of Model Car”, Fuzzy Sets and Systems, vol. 16 pp. 103-113, 1985.

[6] X. Li, and D. Wang, “Behavior-based Mamdani Fuzzy Controller for Mobile Robot Wall- following”, International Conference on Control, Automation and Robotics, 2015.

[7] X. Yang, M. Moallem, and R.V. Patel, “A Layered Goal-Oriented Fuzzy Motion Planning Strategy for Mobile Robot Navigation”, IEEE Transactions on Systems Man and Cybernetics, vol. 25(6) pp. 1214-1224, 2005.

[8] H. Seraji, and A. Howard, “Behavior-Based Robot Navigation on Challenging Terrain: A Fuzzy Logic Approach”, IEEE Transactions on Robotics and Automation, vol. 18(3) pp. 308-321.

[9] M. Ravangard, “Fuzzy Behavior Based Mobile Robot Navigation”, Iranian Joint Conference on Fuzzy and Intelligent Systems, 2015.

[10] W. Dongshu, Z. Yusheng, and S. Wenjie, “Behavior-Based Hierarchical Fuzzy Control for Mobile Robot Navigation in Dynamic Environment”, Chinese Control and Decision   
 Conference, 2011.

[11] J. Rosenblatt, and D. Payton, “A Fine-Grained Alternative to the Subsumption Architecture for Mobile Robot Control”, IEEE International Joint Conference on Neural Networks, 1989.

[12] A.S. Al Yahmedi, and M.A. Fatmi, “Fuzzy Logic Based Navigation of Mobile Robots”, Recent Advances in Mobile Robotics, InTech, pp.287-310, 2011.