random search, maze mapping, and right or left wall following methods.

“Wall following algorithm” and

“Flood fill algorithm”

“Decision Making Algorithm”.

To solve the maze, this robot will apply wall

following algorithms such as left or right hand rule. It will also follow the *Flood fill* algorithm for

finding the shortest path.

It is a small self-reliant robot that can solve a maze from a known starting

position to the center area of the maze in the shortest possible time

A maze solving robot make

multiple runs in a maze, first it create a map of the maze layout and store it in its memory, then

run through a shortest path [3].

small microprocessor controlled vehicle with self-intelligence and capability to

navigate a critical maze

They used a non-graph theory algorithm, **Wall Following Algorithm**

But their robot did not move intelligent in the map and it could not solve maze with loop

Flood Fill Algorithm is able to find the shortest path but more memory is required

for execution.

For controlling the motors speed, pulse width modulation (PWM) is used.

Pulse width modulation

is a simple method of controlling analogue devices via a digital signal through changing or

modulating the pulse width [11].

With 50% duty cycle the

average value is 2.5V, and if the duty cycle is 75%, the average voltage is 3.75V and so on. So, by

varying the average voltage, the motor speed can be controlled.

The power loss in PWM switching devices is very low. While being used, resistors will tend to loss more power because of

its heat dissipation. So, PWM is efficient in controlling motors.

All program of this project is stored in arduino microprocessor.

**3.2 ATmega328P**

ATmega328P is an 8-bit AVR RISC-based microcontroller. It has 32KB ISP flash memory, 1024B

EEPROM, 2KB SRAM, 23 general purpose I/O lines, 6-channel 10-bit A/D converter, 32 general

purpose working registers, 3 flexible timers, serial programmable USART, a byte-oriented 2-wire

serial interface and SPI serial port [16].

“Bellman Ford

Algorithm [23]”

The algorithm works by assigning value for all cells in the maze, where these

values indicate the steps from any cell to the destination cell [24]. The first array is holding the

walls map values, while the other one is storing the distance values [25].

**Figure-4.1: Fill flood algorithm using one-dimensional arrays** [25]**.**

In every cell, robot will follow the following steps:

1. Update the wall map.

2. Flood the maze with the new distance values.

3. Decide which neighboring cell has the lowest distance value.

4. Move to the neighboring cell with the lowest distance value.

The first section of the code, if the right wall and left wall distance is equal or less than 5cm than

both motors are in forward mode. So, the robot move straight. If, the right wall is more than 5cm,

it is considered as open wall, than the robot turn into right. Same as, if the left wall is more than

5cm, it is considered as open wall, than the robot turn into left. And this function is continued in a

loop till the maze is solved.

Ultrasonic sonar is used to measure the distance of wall. The values are measured by using serial

monitor of Arduino IDE.

The flood fill algorithm based Arduino code is used to find the shortest path of the maze. When

the robot pass a wall, it set a value for every cell. It calculates the values by itself and drive the

robot in shortest path.

This project is based

on decision making algorithms. So, it can be used in various intelligent fields. It can be used

as a rescue operation, navigation problems, search operation, medical attention, military

search and rescue, etc



