Path Planning Algorithm Development for Autonomous Vacuum Cleaner Robots

Kazi Mahmud Hasan[1], Abdullah -Al-Nahid[1], Khondker Jahid Reza[2]
[1]Electronics and Communication Engineering Discipline
Khulna University, Khulna 9208, Bangladesh.

[2]School of Computer and Communication Engineering,
Universiti Malaysia Perlis (UniMAP), Pauh Putra main campus, Arau, Perlis 02600, Malaysia.

E-mail: shuvro_eceku07@yahoo.com, nahidku@yahoo.com, jahid_rifat@yahoo.com

***Abstract* - A vacuum cleaner robot, generally called a robovac, is an autonomous robot that is controlled by intelligent program. Autonomous vacuum cleaning robot will perform task like sweeping and vacuuming in a single pass. The DVR-1 vacuum cleaning robot consists of two DC motor operated wheels that allow 360 degree rotation, a castor wheel, side spinning brushes, a front bumper and a miniature vacuum pump. Sensors in the bumper are used for generating binary information of obstacle detection then they are processed by some controlling algorithms. These algorithms are used for path planning and navigation. The robot's bumper prevents them from bumping into walls and furniture by reversing or changing path accordingly.**

Index Terms - Bumper sensor, cliff sensor, navigation, vacuum cleaner, complete coverage, route map algorithm.

I. INTRODUCTION

Robots are emerged as a home appliance recently as the people demands are increasing. Home appliance robotics research is becoming active more than ever. So far several number of big bulky cleaning machines are available for domestic and industrial use. But their operations are non-autonomous type and these can perform only some specific functions of cleaning.

The two most widely known robot vacuum cleaning robots are the iRobot Roomba and Neato. Roomba cleans up very efficiently getting into hard-to-reach areas and often neglected areas like wall edges and beneath furniture. Roomba senses its surroundings and works intelligently around and under furniture, avoiding stairs and other drop offs with 'cliff detect' sensing. iRobot Roomba is a randomly moving robot. It does not build a map of the house and it has no clue where it is in the room rather it uses an algorithm called iAdapt. It goes from wall to wall, turns when it bumps into an obstacle and takes long enough time to cover the whole floor area [1]. But with the help of a laser range finder that scans around in a full 360° circle, the Neato robot uses SLAM algorithm that allows it to map the room for navigation and travels in straight line patterns partially overlapped [2][3]. Kinect sensor based Roomba robot is already been investigated but the costs are relatively higher [4]. But the proposed robot performs many of the functions of Roomba and Neato but it is very much cost effective costing only about US$ 55, far cheaper than any of the robots in the market today. If it is commercially produced, this robot will become a competitor. Although the cleaning efficiency of both the two robots is moderately good but the problem so far with the cost. The expenditure of making cost is pretty higher than the

The motto of this paper is to evaluate cost efficient, lightweight, less noisy and low maintenance robotic system. Simultaneously, having the facility of automatic avoidance of any obstacles and capable of finding its way around after the fall from a height. On the other hand, the robot may perform floor cleaning functions like vacuum cleaning and sweeping while moving around. Mapping technology is absent in this design due to cost effective issue. The idea make preference for the robot to clean the entire area of a room.

A couple of spinning side brushes are attached to the underneath of the cleaning machine to accumulate dirt, debris including pet-hair as well during the move along the way. Though the robot is round shaped but it can still clean along edges and into other hard-to-reach places. The robot can also be maneuvered using remote control to reach the destination.

In this paper section II describes the basic block diagram of the systems, where section III gives some brief idea about the sensors. Developed algorithms are discussed in section IV. Section V deals with design issues. This paper concludes with performance analysis of the robot.

II. BLOCK DIAGRAM

The proposed robot is battery powered electronic device having two different sensors which produces binary information. This information is then sent to the control unit which is regarded as the brain of the robot. This unit generates control signals automatically based on the information provided by the sensors. Finally these control signals are forwarded to the wheel motor drivers.

Currently, two modes, i.e., manual and autonomous are included to the robot. Manual mode allows the users to operate the robot by itself whether incase of autonomous it is driven by the operator. But the twice modes cannot be run simultaneously.

978-1-4799-5180-2/14/$31.00 ©2014 IEEE

The complete block diagram representation is given in the Fig. 1 below.

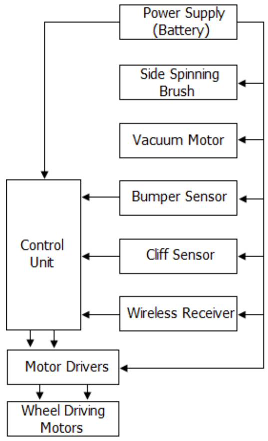

Fig. 1. Block diagram of the DVR-1 robot.

III. SENSORS

Basically, there are mainly two types of sensors to sense the environment for the robot . They are

A. Bumper sensor
B. Cliff sensor

A. Bumper sensor

Bumper sensor is used for sensing obstacle instead of IR proximity sensor. Because the IR LED's/receivers are quite directive and, thus, they cannot detect sharp obstacles as chair legs or sharp edges [5]. Since the robot has no built-in proximity sensors it can detect any obstacles using the bumper sensor only during collisions. The bumper sensor is basically a contact sensor switch. The switches are very reliable and do not need any filtering. When the robot knocks into something, its bumper retracts, activating mechanical object sensors that informed the machine about an obstacle. It then performs the sequential actions of backing up, rotating and moving forward until it finds a clear path. The information is set either to '0' or '1'. If the value is '0' it means no obstacle is encountered and '1' means a collision. The front bumper is 2cm above from the ground which means the robot can identify obstacle higher than 2cm.

B. Cliff sensor

While the robot is cleaning, it avoids steps or any other kind of drop-off using three infrared sensors on the front underside of the unit. The sensor has an IR transmitter and a receiver. The IR light is modulated at 38 KHz so that no interference is occurred due to daylight. These cliff sensors constantly send out infrared pulse train shown in Fig. 2 and the signal immediately bounces back. If it's approaching a cliff, the signals all of a sudden get lost. This is how the robot knows to head the other way. The output of the cliff sensors are binary equivalent where '0' means absence of cliff and '1' means presence of cliff.

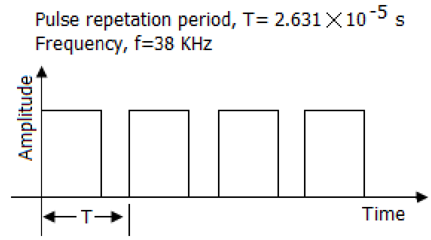

Fig..2 Modulated 38 KHz IR light waveform.

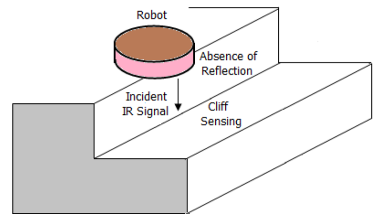

Fig. 3. Cliff sensing process where incident IR signal does not reflect back.

IV. ALGORITHM

It is stated earlier that the robot having two distinct operational modes and they are:

A. Autonomous mode
B. Manual mode

A. Autonomous mode

Usually, autonomous mode is guided by algorithms for path planning of the robot. Path planning is an important factor because the efficiency of cleaning robot is very much dependent on it [6]. Four different algorithms are focused in this paper for path planning. They are-

i.Random walk
ii.Spiral
iii.'S' shape pathway
iv. Wall follow

i. Random walk

The first algorithm has been based on the idea of random movement. A random walk does not require any precise realization of the route plan [7]. The robot moves in forward direction until an obstacle is sensed and then it stops if there is any barrier. Next, it turns by comparing sensor readings from the left or right direction in which to turn and finally by generating a random number it decides how much to turn. But there is a sophisticated condition called corner escape angle, that should be checked otherwise the robot will be trapped in corners [8][9]. The robot's front bumper having two bumper sensors and used to sense bumping into obstacle.

To avoid trapping in corners the turning time is calculated as follow.

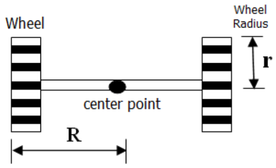

Fig. 4. Driving wheel joining axis and corresponding variables.

Assuming that,
Wheel radius= r
Wheel to centre point distance= R
Number of revolution of wheel per second=N
So, time for 360^0 rotation of robot = $\frac{R}{rN}$ second

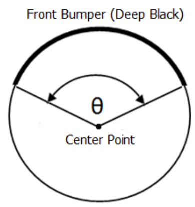

Fig. 5. Front bumper angle with respect to center point.

If the front bumper of the robot makes an angle 'θ' with the center point of driving wheel joining axis then to avoid corner trapping condition-

Minimum turning time after a collision= $\frac{R\theta}{720.rN}$ second

Now some randomness is added to this time to improve random walk performance.

Random time is set as-

Random time = 0 to $\frac{R\theta}{720.rN}$ second

A random number generator is used to generate this random time. So the total turning time is calculated as-

$$Turning\ time = [\frac{R\theta}{720.rN} + Random\ (0 \text{ to } \frac{R\theta}{720.rN})] \text{ second}$$

For the robot we implemented,
Wheel radius, r = 4cm
Wheel to centre point distance, R= 12cm
Number of revolution of wheel per second, N=1.2
Bumper angle, θ= 120^0

So, $Turning\ time = [0.41 + Random(0 \text{ to } 0.41)]$ second
The flowchart of the random walk algorithm is given below.

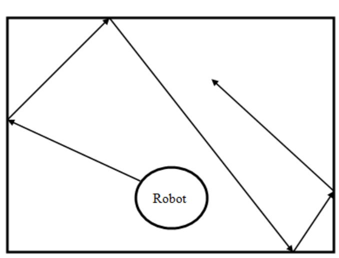

Fig. 6. Motion path of the DVR-1 robot in random walk algorithm.

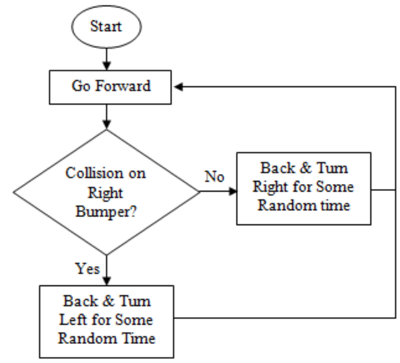

Fig. 7. Random walk algorithm flowchart.

ii. Spiral algorithm

This algorithm allows the robot to create an increased circle. First of all, the robot checks if there is enough place to start moving spirally. If yes, the robot convolutes in a LHS (left hand side) direction, increasing radius from centre point till an obstacle is sensed. When the obstacle is sensed the robot stops executing the algorithm. This algorithm helps in quick coverage of the room area [10].

The algorithm is exhibited in Fig. 8 using flowchart. For this algorithm, the speed of the right wheel driving motor is considered running in full speed while the speed of the left motor is gradually increasing with respect to time.

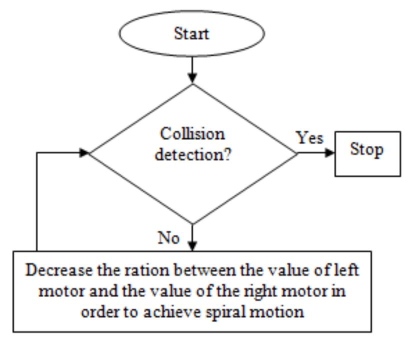

Fig.8. Flow chart of the spiral algorithm.

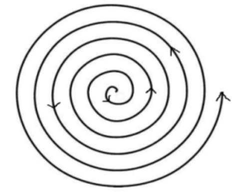

Fig. 9. Motion path produced by spiral algorithm.

iii. 'S' shaped pathway

The route map of this algorithm is like the letter 'S'. This algorithm is the fastest process to cover the entire room area [7][11]. With every collision with obstacle the turning direction of the robot continuously changes under this mode.

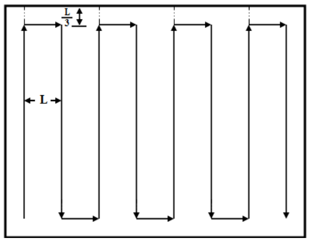

Fig. 10. 'S' shape pattern motion path.

The robot has circular body structure. For this algorithm, after every collision the robot has a sequence of movements. They are-

a. *Back*
b. *90^0 Turn (Right/Left)*
c. *Go*
d. *90^0 Turn (Left/Right)*

It is supposed that, the robot body diameter = L
The parallel lines should be such that the robot partially overlaps adjacent lines.
So maximum separation between parallel lines= L

Let, Back distance = $\frac{L}{3}$
Then various timings are
So, various required timings are (seconds)

Back = $\frac{L}{6\pi rN}$ second (1)

Turn = $\frac{R}{4rN}$ second (2)

Go = $\frac{L}{2\pi rN}$ second (3)

The robot has body diameter, L= 14.5cm
So, various required timings are calculated as

1) Back = 0.16 second
2) Turn = 0.625 second
3) Go = 0.48 second

The flowchart of the 'S' shape algorithm is given in Fig. 11.

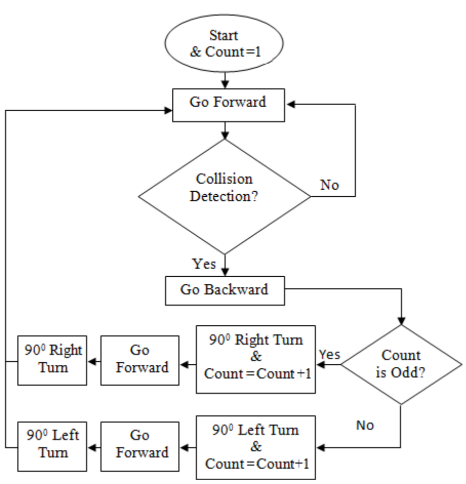

Fig. 11. Flow chart of the 'S' shape pattern algorithm.

iv. Wall follow

The wall follow algorithm allows the robot to move along wall. While moving along wall it uses side spinning brush to collect dirt from corners and walls. This algorithm also helps in efficient coverage of the whole area.

Expected motion path of the wall follow algorithm is given below.

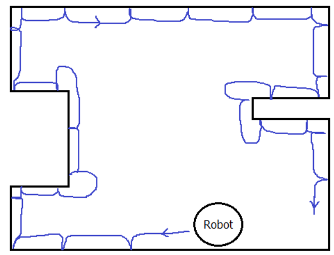

Fig.12. Robot moving along room boundary in wall follow algorithm.

The wall follow algorithm flowchart is give below.

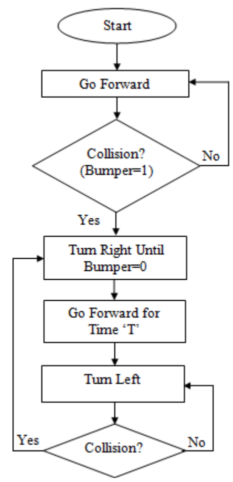

Fig. 13. Wall follow algorithm flowchart.

Cleaning Cycle

The robot continues executing those four algorithm one after another (combined mode) until the whole surface area is fully cleaned. The execution time for each algorithm is manually set by the user or if not default execution time for each algorithm is randomly set.

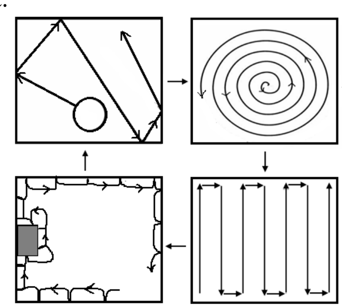

Fig.14. Different algorithms executing in cyclic order in the robot's cleaning cycle.

B. Manual mode

Manual mode allows the users to operate the robot hardly to reach places. The user has freedom to command the robot to create any pattern. In autonomous mode obstacles and cliffs are handled automatically by on board sensors and controllers. But as the user operates the robot by him in manual mode, some extra logics have to be developed. Without such programming the robot may bump an obstacle badly or fall from stairs. This may bring huge damage to the robot. The allowed movements associated with both the sensors are listed in Fig. 15.

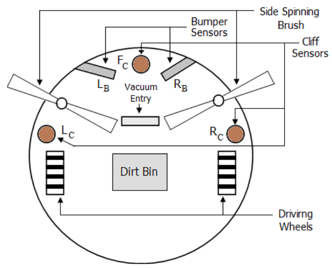

Fig. 15. The DVR-1 robot's bottom view showing various sensor arrangements.

The robot has two bumper sensors on left and right side namely L_B (left bumper) and R_B (right bumper) respectively. Table I includes the allowed commands regarding bumper sensors.

Table I Various allowed commands corresponding to bumper sensor.

Bumper Sensor		Featured Commands
L_B	R_B	
0	0	All (Forward, Right, Left, Backward)
0	1	Backward & Left
1	0	Backward & Right
1	1	Backward

There are three cliff sensors on the bottom of the robot named L_C, F_C and R_C. Featured commands of cliff sensor in manual mode are given below.

Table. II Allowed commands regarding cliff sensors.

Cliff Sensor			Featured Commands
L_C	F_C	R_C	
0	0	0	All
0	0	1	Backward & Left
0	1	0	All except Forward
0	1	1	Backward & Left
1	0	0	Backward & Right
1	0	1	Backward
1	1	0	Backward & Right
1	1	1	Backward

IV. Mechanical Design of Proposed Robot

Mechanical design consists of a three major parts. They will be discussed in this section.

A. The chassis

The body of the robot is circular and the driving wheels are placed along the center axis so that the robot can spin in place in any direction [8]. One free-spinning castor wheel is located at the rear of the robot to keep it balanced.

The body of the robotic vacuum is composed of a circular piece of plastic board, onto which all the components are attached. The proposed robot is made of plastic as it is relatively light and quite durable material for such a construction. The motor specification is 2Amp/6VDC, 5500 rpm with 70:1 gearbox.

B. Driving system

The robot has different sort of driving systems. With driving wheel placed on the center axis of the robot and differential drive, a robot can rotate 360^0 [12]. This facility is utilized in the robot.

C. Side brush

The robot has two side brushes on the bottom for collecting dirt along walls and corners. The brushes spin opposite to one another. The side brush controlling motor specification is 0.5Amp/6VDC with 30:1 gearbox.

Fig.16. Robot's side brush and vacuum filter.

D. Vacuum system

Cyclonic type dry vacuum is used for collection of dirt. Essentially the function of a cyclone separator is to remove the majority of the dust and debris that is sucked up, and separate it out from the air stream that carries it. It does this by introducing the dust filled air to the cyclone chamber. The cyclone causes the air and dust to circulate around the outside of the container, and as such centrifugal forces keep the particular matter to the outside edges. The vacuum motor is 5Amp/6VDC with 8000rpm.

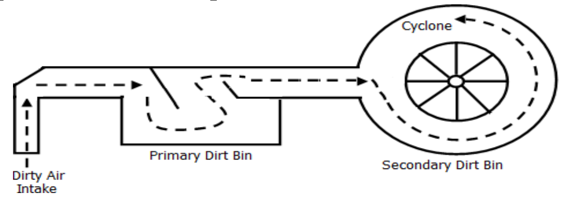

Fig.17. Dirty air suction and cyclonic filtration system.

E. Battery and electronics

The battery used is 6V/4.5Ah lead-acid. Lead-acid batteries are heavier than dry cell batteries but they have one big advantage- they are far more cheaper than li-ion or NI-MH batteries. The battery provides enough power to operate the robot for about an hour. Charging time for the battery is about 5-6 hours.

V. Performance Data

The performance was measured separately for cleaning, obstacle avoidance and for coverage. The test track for cleaning was on tiled floor covered with dust and sand. The obstacle avoidance was tested in normal room environment: chairs, tables on the room. It showed that simple bar with two micro-switches as a sensitive bumper is quite sufficient for given task. The room is 12×12 sq. ft and testing continues until entire room is cleaned.

Four different algorithms were tested separately. Finally a cleaning cycle was tested. The acquired data are then plotted in MATLAB. The efficiency was seen as area coverage in time. In theory, only random walk or spiral or 'S' shape algorithm can provide complete coverage of the area if required time is provided. Owing the fact that both algorithms involved partial or complete randomness, their outputs were highly unpredictable. Nevertheless, based on numerous tests the data shows that the robot covers the entire area faster when operated in combined mode.

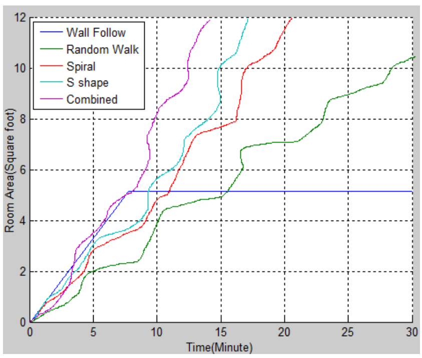

Fig.18. Performance graph of various algorithms in 12×12 foot room arena

VI. CONCLUSION

The aim of this work was to design and implement algorithms of movement of autonomous vacuum cleaner that would be able to work inside of a flat, but not free of obstacles. Though the robot has minimum number of sensors, yet developed algorithms fulfilled this goal. Future development of the robot includes mapping technology, automatic charging algorithm and virtual walls. Neural network and fuzzy logics based control will be developed for better control of robot in future. As the proposed robot performs several functions of Roomba and Neato, but the expenditure will be less than them and about 30 USD. If it is commercially produced, this robot will become a competitor.

REFERENCES

[1] Available at: http://www.irobot.com/EngineeringAwesome/images/iAdapt%20Fast%20Facts.pdf

[2] "Hands-On: Neato XV-11 Robotic Vacuum Review". Home Server Land. 2011-01-13. http://www.homeserverland.com/2011/01/hands-on-neato-xv-11-robotic-vacuum-review/. Retrieved 2011-01-13.

[3] "Neato Robotics XV-11". BotJunkie. 2010-05-18. http://www.botjunkie.com/2010/05/18/botjunkie-review-neato-robotics-xv-11/. Retrieved 2010-09-19.

[4] Ruiz, E.; Acuna, R.; Certad, N.; Terrones, A.; Cabrera, M.E., "Development of a Control Platform for the Mobile Robot Roomba Using ROS and a Kinect Sensor," *Robotics Symposium and Competition (LARS/LARC), 2013 Latin American* , vol., no., pp.55,60, 21-27 Oct. 2013.

[5] V. Colla, A.M. Sabatini, A composite proximity sensor for target location and color estimation, in: Proceedings of the IMEKO Sixth International Symposium on Measurement and Control in Robotics, Brussels, 1996, pp. 134–139.

[6] Chaomin Luo & Simon X. Yang Deborah A. Stacey: "Real-time Path Planning with Deadlock Avoidance of Cleaning Robot", Proceedings of the 2003 IEEE International Conference on Robotics & Automation. Taipei, Taiwan, September 14-19. 2003

[7] Stachniss and Cyrill, "Robotic Mapping and Exploration", in Springer Tracts in Advanced Robotics, Vol. 55, 2009, XVIII, 196 p. 89 illus.

[8] Y D Zhang, K L Fan, B L Luk, Y H Fung, S K Tso "Mechanical Design of A Cleaning Robot" in 8th IEEE Conference on Mechatronics and Machine Vision in Practice (2001)

[9] Ryo Kurazume, Shigeo Hirose, "Development of a Cleaning Robot System with Cooperative Positioning System" in Autonomous Robots (2000) Volume 9, Issue: 3, Publisher: Springer, Pages: 237-246

[10] Sewan Kim, "Autonomous cleaning robot: Roboking system integration and overview" in IEEE International Conference on Robotics and Automation 2004 Proceedings ICRA 04 2004 (2004) Pages: 4437-4441 Vol.5

[11] Chih-Hao Chen and Kai-Tai Song: "Complete Coverage Motion Control of a Cleaning Robot Using Infrared Sensors", Proceedings of the 2005 IEEE International Conference on Mechatronics July 10, 2005, Taipei, Taiwan.

[12] Charles A. Schuler, Willam L. Mcnamee, "Industrial Electronics and Robotics," Mcgraw-Hill International Edition, Industrial Electronics Series, 2003.