**The different ways of determining position of a robot using a signal:**

There is no truly elegant solution for this. The many partial solutions can roughly be categorized into two groups: relative and absolute position measurements. Because of the lack of a single, generally good method, developers of automated guided vehicles (AGVs) and mobile robots usually combine two methods, one from each category [2].

Relative Position Measurements:

1. Odometry

This method uses encoders to measure wheel rotation and/or steering orientation.

The disadvantage of odometry is that the position error grows without bound unless an independent reference is used periodically to reduce the error. In our experiment we have slip involved

1. Inertial Navigation

This method uses gyroscopes and sometimes accelerometers to measure rate of rotation and acceleration.

Problem with inertial navigation is the high equipment cost. For example, highly accurate gyros, used in airplanes, are inhibitively expensive.

Absolute Position Measurements:

1. Active Beacons

This method computes the absolute position of the robot from measuring the direction of incidence of three or more actively transmitted beacons. The transmitters, usually using light or radio frequencies, must be located at known sites in the environment.

1. Artificial Landmark Recognition

In this method distinctive artificial landmarks are placed at known locations in the environment. Unlike the usually point shaped beacons, artificial landmarks may be defined as a set of features, e.g., a shape or an area. Additional information, for example distance, can be derived from measuring the geometric properties of the landmark, but this approach is computationally intensive and not very accurate.

1. Natural Landmark Recognition

Here the landmarks are distinctive features in the environment.

1. Model Matching

In this method information acquired from the robot's onboard sensors is compared to a map or world model of the environment. If features from the sensor-based map and the world model map match, then the vehicle's absolute location can be estimated.

**Our Proposed model:**

In our proposed model we planned to use a simplified version of the active beacons. Because we want to measure linear speed only, we can work with 2 sensors.

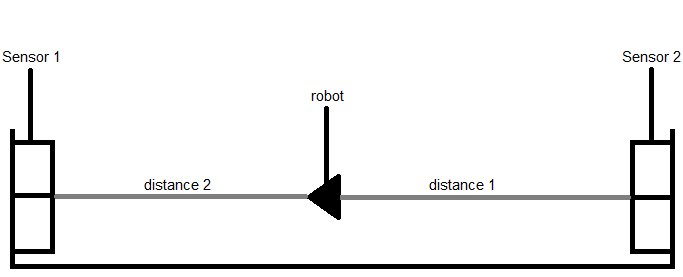


Image 1: Our proposed model

We align two range sensors at two ends of the track and measure the distance of the robot from each sensor. We can also get dynamic distance of the robot from each sensor whilst the robot moves between the sensors, which would allow us to measure the speed and hence the acceleration. Inaccuracy may occur from the robot deviating to a side. The task can be accomplished with only one sensor, but using two sensors at either end allows us to get a more accurate measurement.

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Ultrasonic sensor** | **Infrared** | **Laser** |
| **Range** | From 1 to 250 cm | From 5 to 80 cm | Several metres to tens of metres depending on the model. |
| **Directivity** | Cone of approx. 30° | Cone of approx. 5° | The most directional (around one degree, or even half a degree). |
| **Accuracy** | Relatively accurate but accuracy lessens with distance, with the measurement angle and with temperature and pressure conditions. | Relatively accurate but accuracy lessens over distance. | Accurate to within a few centimetres over measurements of several metres. |
| **Cost** | Inexpensive | Inexpensive | Relatively expensive |
| **Sensibility to interference** | Sensitive to temperature and pressure. Also sensitive to other robots using the same frequency, which can cause problems in competition. | Sensitive to strong light sources containing a high level of infrared radiation, and also to the colour and type of obstacles. | Cannot detect objects that reflect lasers (windows, chrome-plated objects, etc.). |

Table 1: Comparison between three different range sensors [1]

As we can see from the table above that the two feasible sensors are ultrasonic and infrared sensors. Lasers, although accurate but can be relatively expensive. So we propose to use one ultrasonic sensor and one infrared sensor, which would be connected to the same Arduino microcontroller. This would allow us to ensure that signals from each sensor don't interfere with the other. Also comparing readings of distance one and distance two at every instant of time would give us an understanding whether our readings are accurate. Ultrasound waves travel at 330 m/s and infrared travels at 300,000,000 meters per second in a vacuum and our robot travels at few cm/s.

Proposed model 2:

Using two Ldrs and laser beams:

An object that passes by will "cut" the laser beams, this means the LDR sensor will detect this sudden drop of light intensity. First I defined a threshold value under which the sensor is considered triggered, once the value is under threshold for the first sensor then Arduino waits for the second one to be triggered. During this waiting time it counts the elapsed time between the two events. When the second beam is interrupted, the timer stops and now is just simple math. The distance between the 2 sensors is known, the time between the two events is known, and speed can be computed as speed = distance/time [3].

1. <http://www.generationrobots.com/en/content/65-ultrasonic-sonar-sensors-for-robots>

2. <http://www-personal.umich.edu/~johannb/Papers/pos96rep.pdf>

3. http://kumuya.blogspot.com/2012/02/speed-measurement-with-arduino.html