

CM30229 - Circumnavigation using Lejos

Dominic Hauton

March 3, 2017

0.1 Introduction

The goal of this research is to see the effect of changing the rate of sensing on the navigation capabilities of a rover. The rover is built using parts from the LEGO Mindstorm NXJ kit. Four sensors are utilised for sensing; a *Sonar*, *Light Sensor* and *Bump Sensor* (x2), and Three motors provide movement.

The sonar is mounted on a pivot to allow for measurements in multiple directions, as shown in figure 1. Two driven wheels are used for drive and a trolley wheel is placed at the back for stability.

During development a reactive approach is taken, using the subsumption architecture [Wooldridge, 2009] with emphasis on sensing using multiple layers of perception to allow for informed movement decisions. The use of the subsumption architecture is based on Brooks' thesis that Intelligence is an emergent property of certain complex systems. [Brooks, 1991] During development this subsumption approach is added into the perception mechanism, to help the rover understand it's current situation.



Figure 1: Rover Head Sensor

0.2 Approach

The initial design of the rover is a simple reflex agent as described by Russell et al. [1995]. This takes numerical sensor readings, and uses transduction converting them to an qualitative proximity label in every direction, making immediate decisions based on these readings. This allows the rover to react to a crash but gives no context for recovery.

The final design uses Russell et al. [1995]'s model-based reflex agent. This allows the sensors to modify the state at individual rates. There are two sensing thread loops as shown in figure 2. This is done to prevent the sonar slowing down other sensors.

As the front sensor uses three¹ separate sensors for proximity detection, the closest, most important result is taken. To improve sensor quality 15 readings are taken every measurement and any reading more than two standard deviations away from the mean is thrown out. The remainder of the readings are averaged.

To account for arena variances, a calibration routine is used. This requests certain rover placements a certain distance from the arena wall on each side.

A decision loop reads the state and determines a meta-state² from the current state using a subsumption architecture. If the rover is *NEAR* something to the front, it has crashed and it does not matter that it is *NEAR* an object on the left too. This meta-state is then used in the action planner, which selects the best action given a

¹Bump, light and sonar sensor

²Such as *TOO_CLOSE_TO_WALL* or *WALL_NOT_FOUND*.

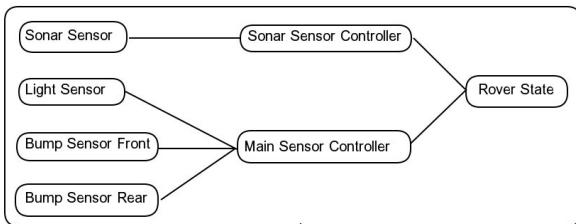


Figure 2: Final Perception Architecture

meta-state. If the rover is crashed and it is *NEAR* an object to the left, it should not try to reverse to the left.

The meta-state also directs the sonar. If it is following a wall on the left and has found the wall, it skips the reading on the right as it is irrelevant to the navigation. This system describes the platform for experimentation. The goal is to find if there is an optimal sensing rate.

The rover is run at a tick rate (measurements per second) of 5, 25 and 45. During a run the rover performs 10 laps of the test track (fig. 3), and the number of bump sensor activations was recorded programmatically.

Each run is done twice. The hypothesis is that the rover will perform better with a higher tick rate, but with diminishing returns. If the rover is stuck in a corner, the experiment is restarted. During each run, the rover is started in the bottom left hand corner of the arena facing the left wall. The experiment is conducted with a variety of obstacle types and sizes to produce a generalised result.

The rover was shared with Ryan Cullen. I built the rover itself, optimising its design and created the perception system in the code. The software continuously modified proximity readings in the state, indi-

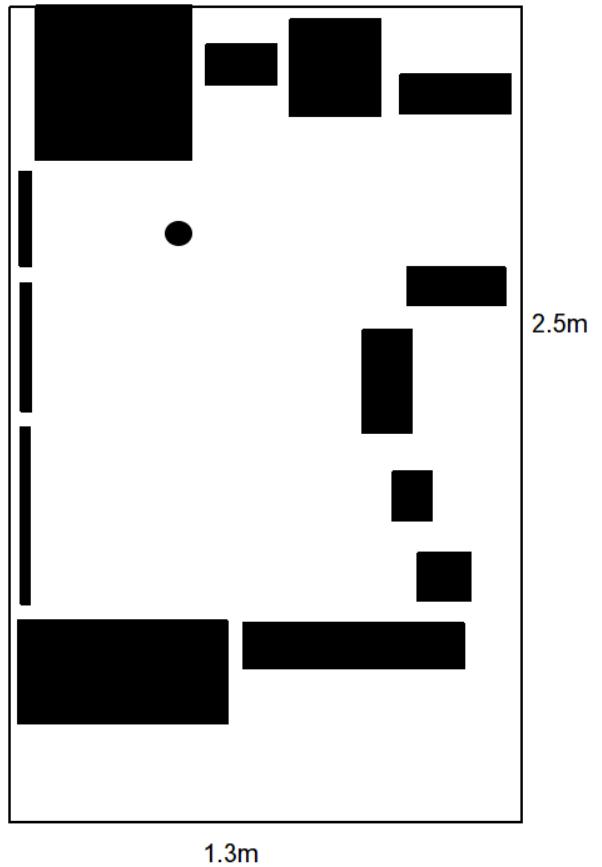


Figure 3: Test Area Layout

cating how close the rover was to an object in all four cardinal directions. At this point the code was then forked by Ryan and we each developed our own action selection code based on the rover state.

0.3 Results

The results of the experiment are consistent with the hypothesis. With this positive result the experiment is also conducted after doubling the rover movement speed. A higher tick rate is found to help the rover at higher speeds, leading to the conclusion

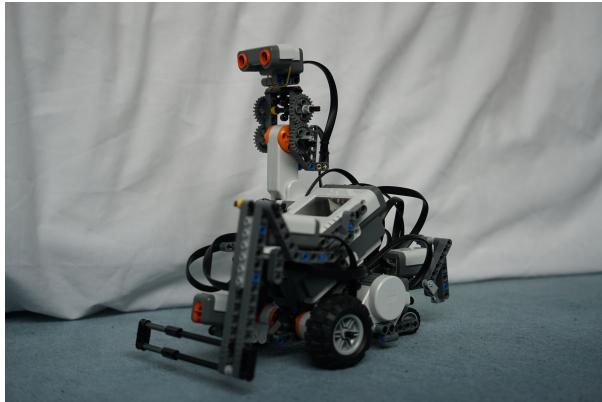


Figure 4: Stanley Rover

Table 1: Bumps per lap at two speeds

Tickrate	5	25	45
Slow	23 ± 2	14 ± 0	16 ± 1
Fast	53 ± 3	46 ± 2	26 ± 2

that the faster the rover moves, the faster it needs to sense the environment around it.

As the results for one lap are inconsistent, the rover performs two runs of 10 laps. Both 10 lap runs were within 2 bumps per lap of each other on average, which was within the required 0.10 p-value for statistical significance.

The video found at <https://youtu.be/LOPd00w1Uec> depicts the start of a trail run.

0.4 Discussion

The results closely match the hypothesis, however, the sensing speed of the sonar, which is the key part of the navigation logic, is limited due to physical speed. A second sonar that does not require physical movement for sensing may improve the navigation of the rover.

The arena environment was kept as con-

sistent as possible. The light sensor is very susceptible to sunlight and closing curtains in the room makes runs at different times of day more consistent. As the rover also occasionally detects the room light as a light sensor reading, a lamp shade was used. To make sonar readings consistent the arena is covered with hard surfaces.

To improve navigation; sensing and decision loops should be increased as far as they add a tangible benefit to navigation, and the faster the movement the more important this is. In applications such as quadcopters loop times affect flight characteristics, but enabling faster loop times might mean disabling other features so this left adjustable in drone software for users. [Betaflight, 2017]

0.5 Conclusion

This research set out to find the impact of sensing speed on the navigation of rovers. The results show that a faster sensing loop results in more accurate navigation with diminishing returns.

When deciding on sensing loop speeds, this level should be reached and not surpassed, as going too high might take compute cycles away from other important parts of the planning loop.

Bibliography

Betaflight. Betaflight FAQ. <https://github.com/betaflight/betaflight/wiki/Frequently-Asked-Questions#what-is-2khz-mode->, 2017. [Online; accessed 28-Feb-2017].

Rodney A Brooks. Intelligence without representation. *Artificial intelligence*, 47(1-3):139–159, 1991.

Stuart Russell, Peter Norvig, and Artificial Intelligence. A modern approach. *Artificial Intelligence. Prentice-Hall, Englewood Cliffs*, 25:27, 1995.

Michael Wooldridge. *An introduction to multiagent systems*. John Wiley & Sons, 2009.