Obstacle Avoidance as a Consequence of Suppressing Irreversible Actions

Adam Eppendahl
Department of Design and Manufacturing
University of Malaya, Malaysia
a.eppendahl@mac.com

Maarja Kruusmaa
Institute of Technology
University of Tartu, Estonia
maarja.kruusmaa@ut.ee

The observation that abstract principles can produce useful concrete behaviours is not new. In (Kaplan and Oudeyer, 2003) a number of basic visual behaviours are seen to emerge from abstract motivational principles based on prediction errors. The general idea is to identify principles that can be expressed without reference to the ground meaning of sensor-motor values, with the expectation that code based on such principles will function reliably in a broad range of environments and on different robots.

We began, rather ambitiously, with the idea of capturing, in a single principle, absolutely everything a domestic robot should avoid doing: do not do what you cannot undo! We then discovered that, when this principle is implemented on a simple robot with differential drive and proximity sensors, obstacle avoidance behaviour emerges.

The implementation is entirely ungrounded: it is written with no information about the meaning of sensor-motor values—which makes the code easy to write!—and runs with no external source of information about its behaviour (such as a reward or training signal). Externally, the robot comes to exhibit obstacle avoidance, but internally it is simply learning about and suppressing irreversible actions. We suggest that this is a significant advance on the idea of obstacle avoidance as a 'basic' behaviour. We would argue that our program captures *why* the robot should avoid obstacles and that this makes our approach much more robust.

Before we describe experiments with a real robot, we explain how obstacle avoidance results from the maxim of avoiding irreversible actions. The sensors on the basic Khepera robot, roughly speaking, indicate the robots position and the distances to nearby objects. Suppose the robot starts at position p and distance d from an object in front of the robot. Suppose the robot moves forward far enough to displace the object. The position is now $p + a$ and the distance to the object is 0. Now, because the object has moved, the robot can move backward to position p and distance a, or to position $p+d-a$ and distance d, but not to the original position p and distance d. To a first approximation, the action appears to be irreversible. Note the a similar argument applies when, instead of moving the obstacle, the robot's wheels slip.

Here are the details of our code. We program the robot to alternate between testing the reversibility of an action and simply performing a random action. Each action is a pair of wheel displacements, between -500 and 500 in native wheel counter units. When testing the reversibility of an action, the program begins by reading the eight proximity sensors, which return values between 0 and 1023. It then performs a randomly chosen action and, when the action is complete, it performs the reverse action obtained by negating the pair of wheel displacements. It then reads the eight proximity sensors a second time and calculates the Euclidean distance from the first reading. Finally it records the action together with the initial sensor reading as an *action-situation* pair and the distance to the second reading as the associated *reversal error*.

While running, the program checks each action for reversibility in the context of the current sensor readings. An action is deemed irreversible in the current sensor situation if the error associated with the nearest recorded action-situation pair (with respect to Euclidean distance) is larger than half the largest recorded error. When an irreversible action is proposed, a red light on the robot is lit until a reversible action is proposed. When the program is running with suppression turned on, irreversible actions are not performed. When suppression is off, all actions are performed and only the light indicates which are considered irreversible.

For the experiments, the robot is allowed to move about among a collection of ordinary desk-top objects. At the start of an experiment, the program has suppression turned off. (If it starts with suppression on, the first few bad experiences tend to paralyse the robot.) As the robot learns about more and more irreversible actions, the red light begins to come on each time the robot is pushing an object. As it learns about more and more reversible actions, it becomes possible to run the program with suppression on so that the robot stops and moves away from objects.

If actions are restricted to forward and backward movements, the robot develops reliable obstacle avoidance in a few minutes. Here is an action-situation pair deemed irreversible in an actual run.

```
a 466 s 5 893 1023 1023 1023 0 0 0 e 1923
```

The action was a forward movement of 466 units, the eight sensor values show an object to front right of the robot and the error was the largest of the run. If the robot is allowed to move in any direction, it takes several hours before the robot begins to display reliable obstacle avoidance. This is to be expected, however, given that the full set of irreversible action-situation pairs is much more complex than the linear set and is being explored by a random walk. We expect that standard search techniques can be used to explore of the action-situation space more efficiently.

More importantly, our program assumes the robot has already learned that the effect of the wheel command (x, y) on the wheel position counters can be reversed by the wheel command $(-x, -y)$. We suggest that this sort of confined but highly reliable reversibility space typically corresponds to the body of the robot. As long as the Khepera is allowed to move freely, the motors and wheel counters are certainly part of the robot's body. Perhaps the repetitive movements observed in infants help establish this fundamental reversibility space. We intend to add to our program an initial phase in which the robot learns the wheel-command/wheel-counter space from scratch.

Our current program then corresponds to a later step in the development of the robot's model of what is reversible: the extension of a known reversibility space to new sensors. The additional information from the new sensors may render some of the old reversals invalid. For example, in the presence of objects, the addition of the Khepera's proximity sensors to the situation vector, means that simply backing-up is no longer a reliable method of restoring the sensor state. The strategy our program adopts for this developmental step, consists in trying known reversals to see which ones still work and to associate any failures to the sensor situation (including, vitally, information from the new sensors).

An alternative strategy, would consist in extending the set of actions to include actions that allow the apparent irreversibilities to be reversed. For example, after pushing a block forward the robot might go around behind the block and push it back to its original position. Discovering such actions would appear to be a much greater challenge than simply suppressing problematic actions.

Khepera code and video are available online from

`http://homepage.mac.com/a.eppendahl/work/`

Acknowledgements

We would like to thank Professor Marzuki Khalid and everyone at the Centre for Artificial Intelligence and Robotics, Universiti Teknologi Malaysia, for providing a friendly and productive working environment during the first author's visit.

References

Kaplan, F. and Oudeyer, P.-Y. (2003). Motivational principles for visual know-how development. In *Proceedings of the Third International Workshop on Epigenetic Robotics*.