Faculty of Design

2022

Ontological Design for Robotics

Battle, Steve

Relating Systems Thinking and Design
2022 Symposium
University of Brighton, Brighton, UK,
October 13-16, 2022

Ontological Design for Robotics

Steve Battle

Dept. Computer Science and Creative Technologies | University of the West of England, Bristol, UK

Ontological design is a way of defining, or constructing, a world of possibilities through design practice. Robotics is often taught in a very rationalistic and didactic fashion, focusing on the kinematics of robot motion. Ontological design is fundamentally enactive, allowing cognition to arise through a dynamic interaction between an individual and their environment. With the ontological design of robots, the focus is on learning robotic design through practice and serendipity. This paper reports on the practice of running hands-on robot workshops for children and adults, the philosophical background to this, and future direction using tangible programming tools. We learn about robots controlled, not through conventional programming, but using simple neural circuits. The inspiration for this is Braitenberg's *Vehicles: Experiments in Synthetic Psychology* (1984), which describes a cybernetic approach to modelling brain function. To realise learning as a fully shared experience, the 'coding' of the robot should be experienced as a tangible activity. We explore a sequence of robot designs that can be built using tools for augmented reality and consider the possibilities for ontological learning at each stage.

KEYWORDS: ontological design, robotics

RSD TOPIC: Learning & Education

Presentation summary

Ontological design is a way of defining, or constructing, a world of possibilities through design practice that shapes the learner (Escobar, 2018). This space of possibilities is created by the tools and learning experiences available. The idea is rooted in the work of Martin Heidegger (Willis, 2006), where the ontological character of “things” is not pre-given but emerges from our experiences with them. Consider “Heidegger’s hammer”, which we learn about through use and manipulation, by which it acquires its “Thingly”, or ontological character (Heidegger et al., 1967, 69f). The use of tools engenders new ways of being (Winograd & Flores, 1987). The design of robots is an interesting case as they can be both tools and autonomous agents in their own right. It is tempting to see robots in our own image; we are naturally drawn to anthropomorphise them. Robots make us question our own human nature as we observe them engaged in seemingly purposeful animalistic behaviours. Robots often occupy a place in society that may challenge our values, for example, over the displacement of workers and, potentially, lovers.

Robotics is often taught in a very rationalistic and didactic fashion, focusing from the start on the kinematics of robot motion and then compensating for the resulting complexity with ever-smarter algorithms. But this approach doesn’t scale well, as fine theories rarely survive contact with reality. According to Escobar, “the rationalistic tradition traps our imagination through constraining metaphors.” With the ontological design of robots, the focus is on approaching robotic design through practice and creative serendipity. In 1984 Valentino Braitenberg, cyberneticist, neuro-anatomist and musician, published a landmark series of thought experiments. His book, *Vehicles: Experiments in Synthetic Psychology* (Braitenberg, 1984), explores the principles of cognition by defining a series of successively more complicated creatures. One way to learn about the ideas introduced in his book is to create physical robots that reproduce the behaviours it describes. I have followed this approach for many years, and I report here on the practice of running hands-on robot workshops for children and adults of all ages, recreating Braitenberg’s Vehicles.

Robotics is fundamentally tangible and exploratory in a way that will often throw up surprises. To enable the best learning experience, we make the 'coding' of the robot a tangible activity as well, with the aim of getting the students out from behind the screen. In previous work, we've adapted Logo-like ideas to create simple pseudo-code for robot control. The Logo language (Abelson et al., 1974) has simple commands for moving the turtle forwards or backwards and turning left or right. These ideas were first explored by Seymour Papert in *Mindstorms* (Papert, 1980), where he discusses physical enactment using TURTLE TALK, the precursor to Logo, “To make the Turtle trace a square you walk in a square yourself and describe what you are doing in TURTLE TALK. And so, working with the Turtle mobilises the child's expertise and pleasure in motion.” This encourages what Papert calls “body-syntonic reasoning,” where students understand the turtle's motion by putting themselves in the place of the turtle.

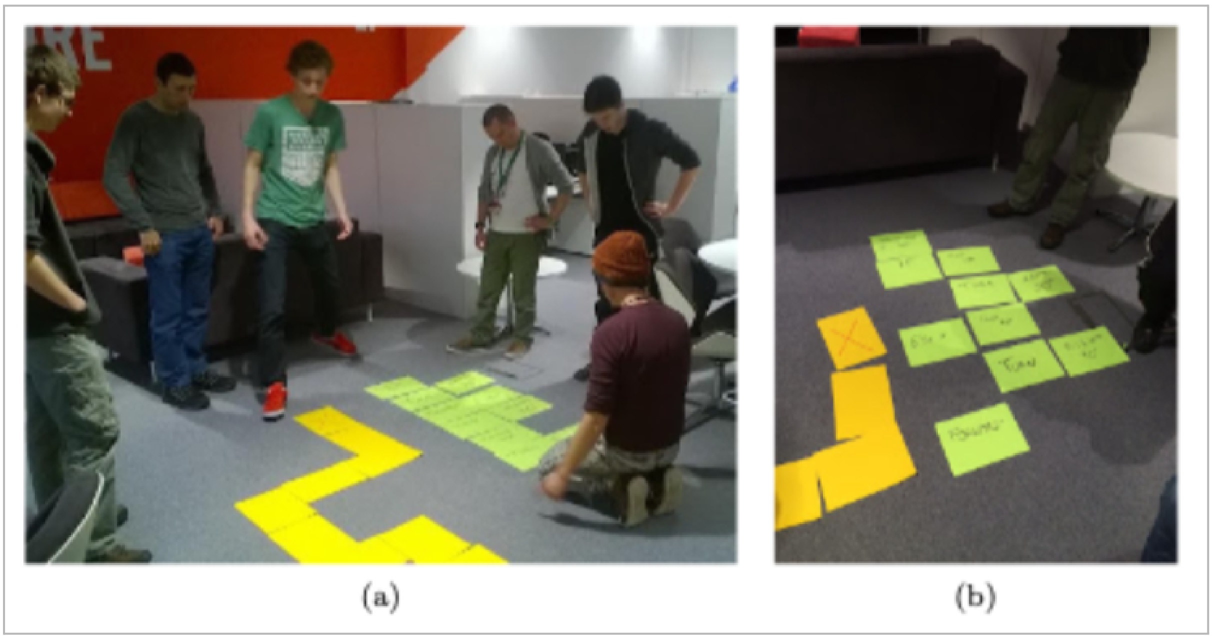


Figure 1. Tangible programming using code cards. The target track to follow is in yellow, and the code cards are in green.

Rather than writing code on-screen, we write these instructions on A4 sheets of paper and lay them out on the floor or a table to form a program. This approach encourages groups of students to work and solve problems together. Program sequences are read from top to bottom, and loops or conditional code are indented (by one sheet), much like the Python programming language. To make the concepts more tangible, the students enact the robot movements themselves. A target track is first laid out on the floor in yellow sheets, with a big red ‘X’ on the target square; they simply "follow the yellow brick road." These sheets also provide a useful ‘pixel’ size, so an instruction like “forward 2” would mean “step forward two sheets.” Students then devise code to follow the path. As the teacher or guide, in this territory, we would then vary the target path in some small way and encourage the students to find a solution that generalises over all the seen examples rather than producing isolated point solutions. For example, changing the direction of a bend encourages students to write conditional code that checks which path is open. Following this example with a simple “T” shaped maze helps them think about how to backtrack along the same path.

This present work builds on these ideas by allowing students to create robot programs using tangible code cards that can be scanned using an ordinary smartphone camera and transferred directly to the robot for execution. Each code-card is identified by a machine-readable fiducial code alongside a human-readable description. In addition, we look at robots that are controlled, not through sequential programming, but using simple neural circuits. These are not the enormous artificial neural networks popularised by machine learning but simple circuits that can, nevertheless, generate complex and often surprising behaviours using a tiny number of neurons. The model for this is Braitenberg's *Vehicles: Experiments in Synthetic Psychology*. A typical workshop progresses through the robots introduced in *Vehicles,* building piece by piece on the ontological scaffolding it provides while allowing space for discovery.

Enactive: being in the world

Ontological design is fundamentally practice-led and *enactive* (Varela et al., 1992), where cognition arises through a dynamic interaction between an individual and their environment. This is as true for the learner as it is for the robot. As we learn how the robot perceives its world, this shapes our own understanding of behavioural possibilities. We will discover that the goals of the robot are not necessarily to be found in the code but emerge from the system as a whole; the robot, its environment, and especially the control surface of the robot that divides one from the other. We must also consider the physical affordances of the environment (Gibson, 1966) and how these shape the phenomenology of the robot, if we may call it that.

Braitenberg's Vehicle 1 introduces the basic idea of motility, or "Getting Around," as he describes it. It has one sensor and one motor. The term "motor," means not just electric motors but anything that can provide a propulsive force. For example, the E. Coli bacterium is one of nature's Vehicles. At its base is a long hollow flagellum that spins 100 times per second. This can either propel the bacterium forwards or cause it to tumble and change direction, depending upon the direction of rotation.

Vehicle 1's sensor can be of any kind of analogue detector, reflecting the analogue nature of the world at this scale. The computing platform we use is the BBC micro:bit with a light-emitting diode (LED) array that can be used in reverse as a light sensor. This provides the robot with a measure of the analogue light level upon its face. Signals are transmitted within the 'brain' of the Vehicle by nerve fibres that carry levels of activation. A single nerve fibre connects the eye directly to the motor in Vehicle 1, causing the motor to vary continuously with the level of activation of the eye. As Braitenberg puts it, "The more there is of the quality to which the sensor is tuned, the faster the motor goes."

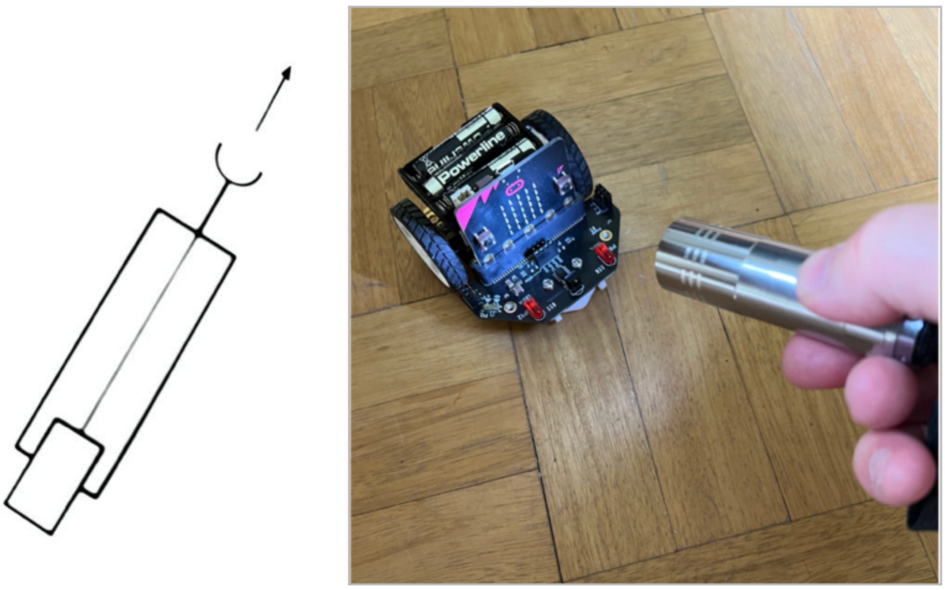


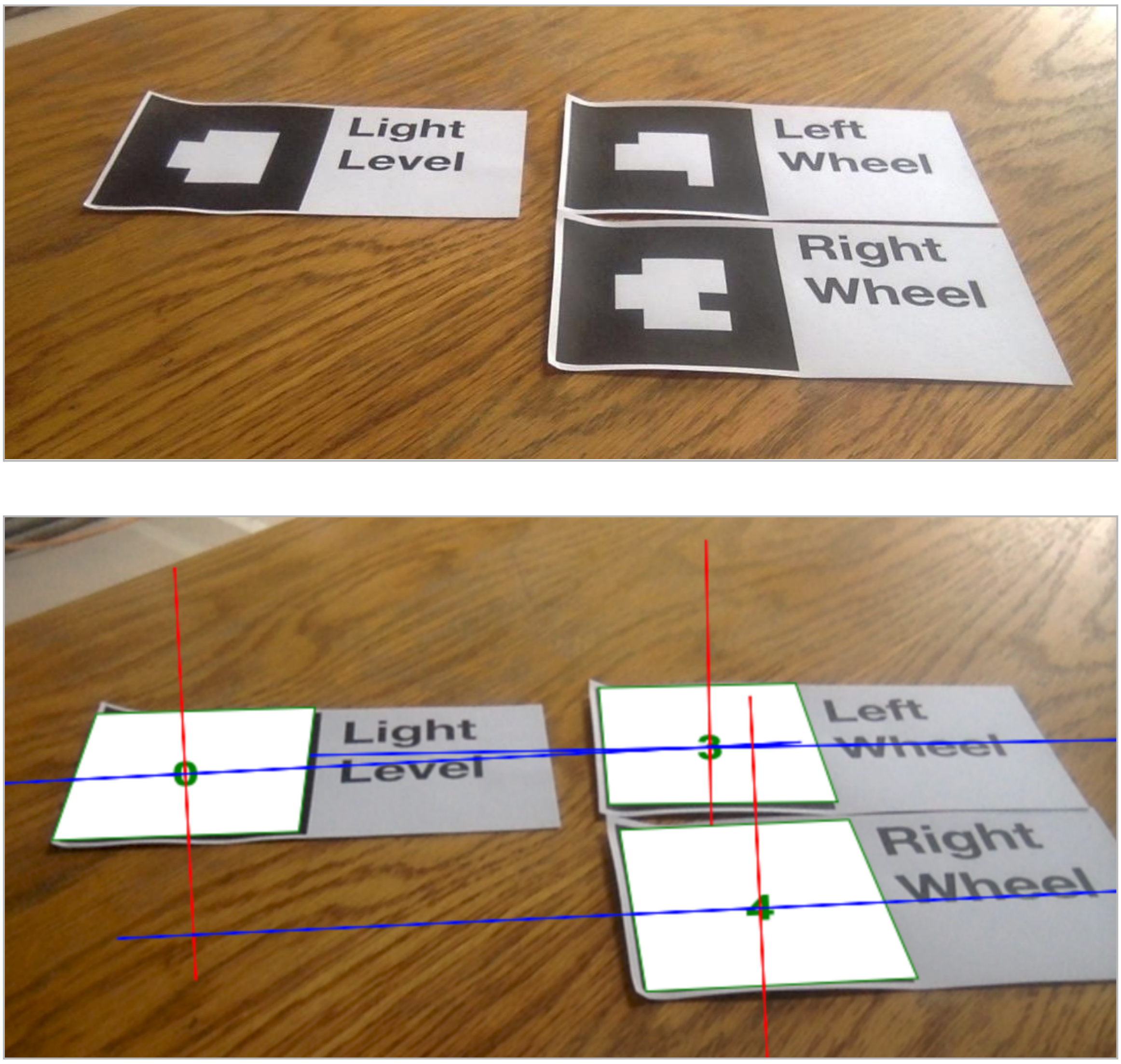
Figure 2. Vehicle 1 with a single light sensor. The brighter the light, the faster it goes.

The layout of neural circuits for each Vehicle is created using code cards that can be laid out on a convenient surface. The circuits run from left to right, with inputs to the left and outputs to the right. The robot we're using has two wheels, so to produce the behaviour of Vehicle 1, the signal from the light-level sensor must be sent to both wheels (otherwise, it would drive around in a circle). We could represent this as two connections, one per wheel, but to indicate this grouping, we allow repeated elements to the left to be eliminated. In other words, the indentation can be read as "ditto", but it really indicates a logical grouping, and the resulting neural circuits can be organised as a left-to-right tree structure. The image in Figure 3a shows how the code cards are laid out, while Figure Figure 3b shows the view through the application, providing feedback to the user about which code cards have been detected. This is an augmented reality application running on an ordinary smartphone that recognises two-dimensional fiducial codes and overlays them in the image with a box indicating the identifier (an integer) for the code card. Each kind of code card has a unique identifier. We use ARToolKit (Kato & Billinghurst, 1999), an open-source library for augmented reality applications.

The application constructs an array in memory representing the arrangement of the code cards. This is achieved geometrically by extending horizontal and vertical line segments from the centres of each code card. The horizontal line segments are twice the length of the verticals in proportion to the overall shape of the cards. Where these line segments intersect another fiducial code. The relative positions are recorded, and the array representing any given frame can be constructed. The completed structure may be read out of the array as a tree. The micro:bit has good wireless communications, so the code tree can be transmitted to the robot, where it is interpreted directly on the robot platform.

The brain of Vehicle 1 could hardly be simpler, but something magic happens when learners get the robot working. I see the surprise and even joy on learners' faces as the robot advances when they shine a torch on its face. This comes about through the student engaging with the narrow world of the robot, with the torch being essential equipment, as it has an easy affordance for both the learner and the robot. The loop, from robot to learner, through the torch back to the robot, is closed. After Maturana

(Maturana & Varela, 1980, p. 30), we might describe this as a new consensual domain—a new ontological domain generated through a recurrent conversation between the learner and robot.



(a)

(b)

Figure 3. A neural circuit for Vehicle 1 was created with tangible code cards, with fiducial codes visible (a) and the AR view through the app (b).

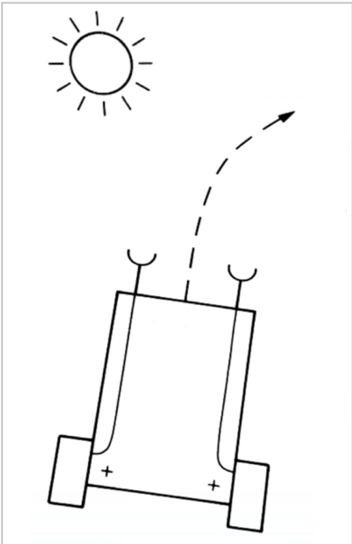
Relational: expanding the cognitive domain

More sophisticated perception relies on forming relational perceptions among lower-order absolute sensor values. According to Maturana (Maturana & Varela, 1980, p. 13), “the animal is modified through its interactions with the relations that hold between the activated sensors” and “The nervous system expands the cognitive domain of the living system by making possible interactions with 'pure relations.”

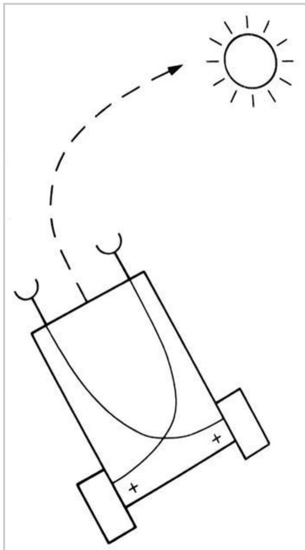
Vehicle 2 equips the robot with a fight or flight response to light, or “Fear & Aggression” as Braitenberg puts it, “It flees from light in fear, or heads towards it in a way that might be considered aggressive.” Vehicle 2a connects each sensor to the motor on the same side. If the light is brighter on one side of the Vehicle, the motor on that side runs faster, causing it to turn away from it. Vehicle 2a is averse to light, veering away from it, “escaping until it safely reaches a place where the influence of the source is scarcely felt.” According to Braitenberg, “Vehicle 2a is a coward.” Unlike Vehicle 1, it is not simply the light level that Vehicle 2 responds to, but the relations between its sensory input.

The image in Figure 4a shows a representation of Braitenberg's Vehicle 2a. The ‘circuits’ are read from left to right, then top to bottom. This circuit connects the left eye directly to the left wheel and the right eye directly to the right wheel. These Vehicles introduce the concept of excitatory connections that were implicit in Vehicle 1 but are here explicitly labelled. This new addition will be reflected in the code.

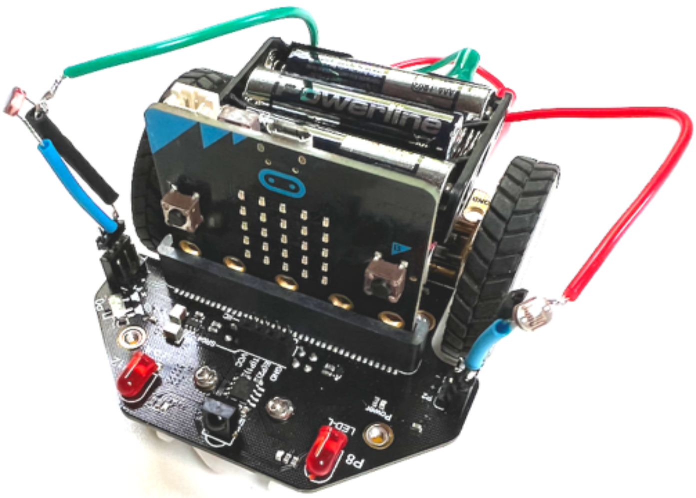
Vehicle 2b simply swaps over the connections from the eyes to the motors. If the light is brighter on one side of the Vehicle, the motor on that side runs faster, causing it to head towards the light. Braitenberg expresses this more emotively, “It, too, is excited by the presence of sources, but resolutely turns toward them and hits them with high velocity, as if it wanted to destroy them. Vehicle 2b is aggressive, obviously.”



(a)



(b)

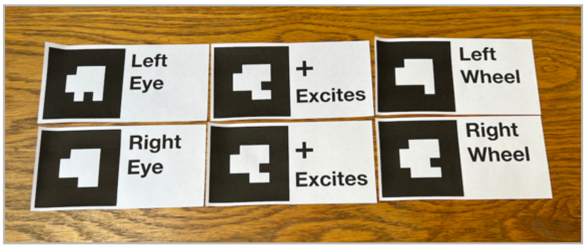


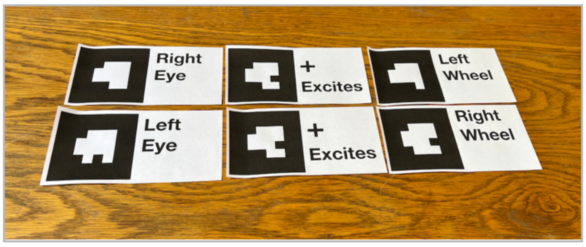
(c)

Figure 4. Vehicle 2 has two eyes and two motors, `+' indicates an excitatory connection. V2a is negatively phototactic, avoiding light (a). The connections in V2b cross over to make it positively phototactic (b). A pair of photocell 'eyes' on flexible 'stalks' (blue) are added to the robot port & starboard (c). These carry the analogue light level in the direction they are facing.

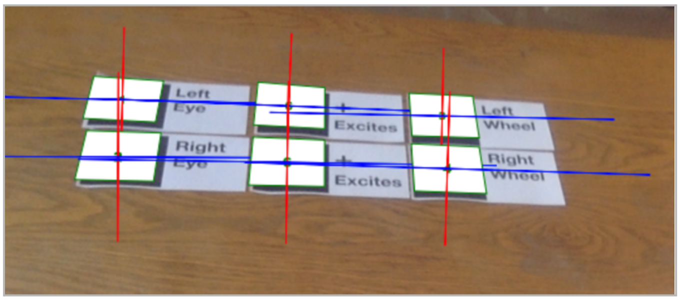
There is a biological precedent for this. In 1899, Spanish neuroanatomist Ramón y Cajal observed optic nerve fibres from the half of the eye closest to the nose cross over to the opposite side of the brain. However, where each human optic nerve carries a million or so separate nerve fibres, the humble Vehicle 2 has just two. Crossed connections are common in vertebrates, but nobody really knows why.

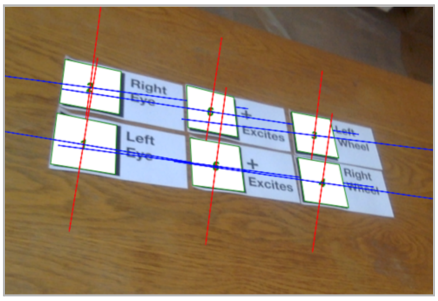
The swapping around of the circuits is achieved very simply by re-arranging the code cards, as seen in Figure 5b. The left eye is on the same line as the right wheel, and vice versa.





(a) **Vehicle 2a** **(b) Vehicle 2b**





(c) **(d)**

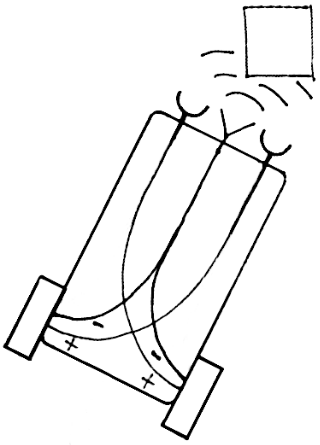
Figure 5. Vehicle 2a neural circuit with explicit excitatory connections(a), with eyes swapped over in Vehicle 2b (b). AR views for (a) & (b) are shown in (c) & (d), respectively.

The simple act of crossing the neural pathways from the eyes turns the robot from a reactive light-avoiding machine into a machine that appears determined to home in on the light from a torch—to become a machine with a goal. Vehicle 3 is now positively phototactic. The physical orientation of the sensors—the robot's eyes—can make all the difference for the emergence of goal-directed behaviour; they need to be tilted forwards and out to exhibit torch-following behaviour. We make the sensor mountings pliable to encourage this physical exploration of functionality. Once given permission and encouraged to ‘bend’ the physical components of the robot, learners soon discover the optimal orientation of the eyes. Without any further changes to the code, the Vehicle 2b robot now actively follows a torch—turning towards the learner, following them if they move. A new ontological level of purpose reveals itself, emerging as if from nowhere in the eye of the observer. This goal cannot be reduced to, nor deduced from, the code alone; this is whole-system behaviour that emerges from tinkering. It's no longer just reactive to light, in the sense of being a simple stimulus-response system, but appears to have a goal. These simple Vehicles are incapable of learning or adapting; that’s not what’s happening here. The learner observes the purpose of the Vehicle and the Vehicle behaving as if it has a purpose. Within the ontology of the consensual domain, maybe that’s enough.

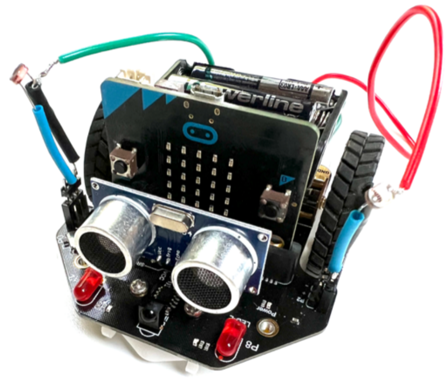
Contextual

Vehicle 2 may act purposefully, but it has no freedom to choose. Being able to produce a variety of behaviours requires the ability to select appropriate behaviour depending on a repeatable context (Bateson, 1972, p. 288). The ability to act, or not to act, requires inhibition as well as excitation.

Vehicle 3 introduces the idea of inhibition. Cells in the brain extend long tendrils towards each other called neurites (axons and dendrites) which allow them to communicate with each other. Where these neurites—almost—touch, they form bulbous extrusions known as synapses through which they exchange electrochemical messages. These synapses can be either excitatory or inhibitory; a signal crossing an excitatory synapse stimulates activity in the receiving brain cell, while a signal crossing an inhibitory synapse suppresses activity in the receiver.



(a)



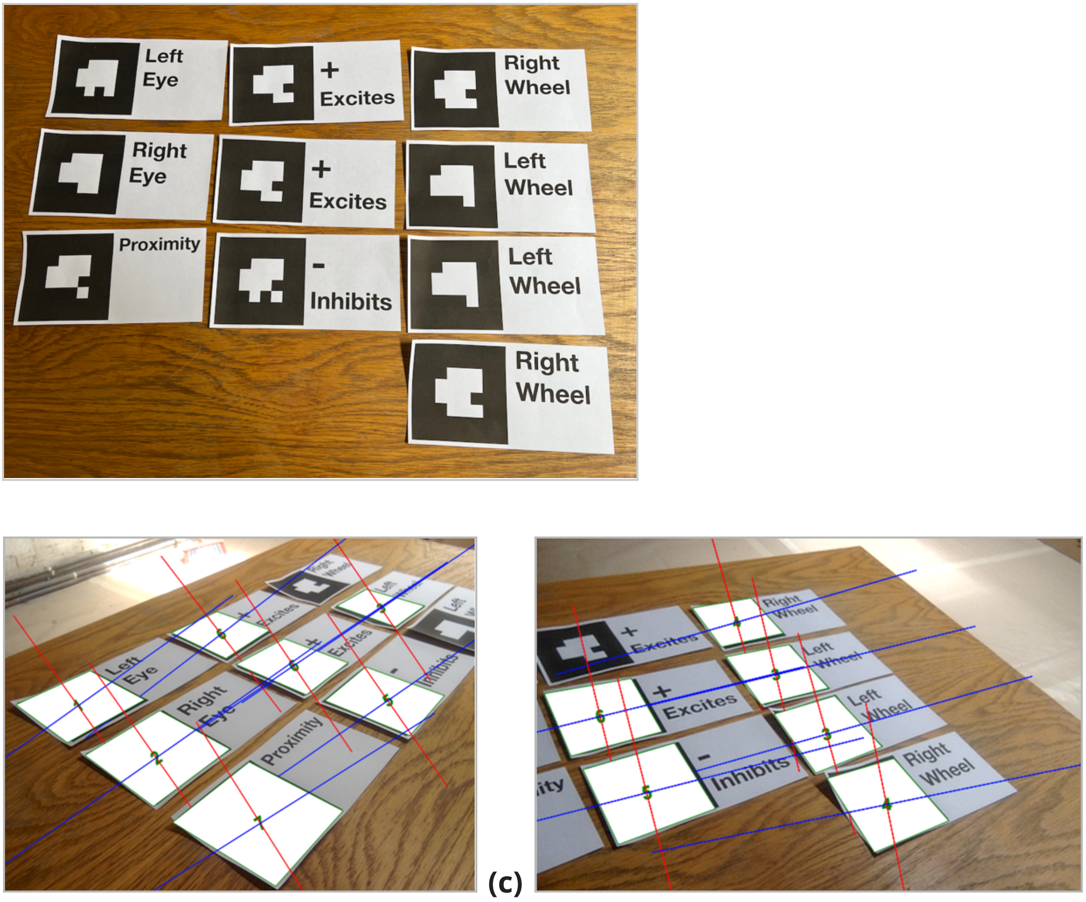
(b)

Figure 6. Vehicle 3 is multi-sensorial (a). We add a proximity sensor, with an ultrasonic transmitter (left) and receiver (right) pair, plugs into a socket at the front of the robot (b).

Braitenberg describes Vehicle 3 more poetically in terms of *love*. It introduces multisensorial variety, adding new sensors that inhibit the motors, causing it to slow down and bask in the light that it loves so much. “If you consider the possibility of strong and weak influences from the sensors to the motors, you realize that the variety becomes even greater.”

Inputs arriving at the same time and place are summed together. Excitatory and inhibitory inputs are balanced against each other. Bernard Katz first observed the summation of signals in nerve cells in the 1950s in the squid giant synapse (not the same thing as a giant squid synapse).

Adding a proximity sensor inhibits the motors, causing Vehicle 3 to slow down whenever it perceives an object in front of it. This bat-like sonar gives it obstacle avoidance behaviour. The code for Vehicle 2b can be extended by adding the inhibitory connection from the new proximity sensor to both wheels.



(a) (b) (c)

Figure 7. Tangible code cards can't always be scanned in one frame. The code for Vehicle 3 (a) is scanned in two frames, (b) and (c), with three common fiducials. These enable the frames to be stitched together to complete the program.

The code example is now too large to fit easily into a single frame of captured video. The fiducial code scanning works in real-time, so the user can simply scan the camera across the code. They receive feedback on the codes scanned so they can see when the scan is complete. This involves stitching together the outputs from a number of different frames. The process is simplified by making an assumption of continuity between successive frames; the stitching algorithm expects to find overlap between frames. In the example in Figure 7, there are three fiducials common to the separate frames shown in Figure 7b and Figure 7c.

Breakdown

Ontological design provides a different way of thinking about the process we ordinarily call "debugging." Inevitably mistakes are made, and the robot fails to produce the expected behaviour. I recall one example with Vehicle 3, where the inhibitory signal was only sent to one wheel rather than both. This serendipitous mistake caused the robot to steer around the obstacle rather than slow down and stop. Many interesting mistakes produce unintended and unanticipated behaviours. Heidegger talked about breakdowns in our habitual use of a tool when encountering a situation where, for example, "the hammer is too heavy." This makes us look at the tool in a new way (present-to-hand) and study it as a thing in itself. The same is true of buggy circuits, where we switch from observing the behaviour to thinking about how that behaviour is produced, considering the circuit as a thing in itself. A new design is an interpretation of the breakdown, and serendipitous mistakes can emerge as new designs follow the breakdown.

Conclusion

We have applied ontological design principles to the pedagogy of teaching and learning about robotics. The approach is enactive on multiple levels; firstly, the learning environment is enactivated through the use of tangible programming, and secondly, the robots we create are themselves enactive—world-building in their own right. The learner engages with the robot, forming a new consensual domain if they find a way to control its behaviour by successfully interacting with it. Relational properties greatly expand the phenomenological domain of the robot and enable the emergence of seemingly purposeful behaviour within the consensual domain. Braitenberg introduces us to this ontological domain with the poetic nomenclature for his Vehicles, such as "Fear," "Aggression," and "Love." Contextual awareness unleashes a greater variety of potential behaviours through inhibition and, therefore, selection of a given behaviour. Breakdown occurs when buggy code fails to act as expected, triggering an ontological shift back and forth between a performative view and theoretical consideration of the code, an opportunity for new ontological designs to emerge from the breakdown.

References

1. Abelson, H., Goodman, N., & Rudolph, L. (1974). *Logo Manual*. Artificial Intelligence Lab, Massachusetts Institute of Technology.
2. Bateson, G. (1972). *Steps to an Ecology of Mind*. Chandler Publishing Company.
3. Braitenberg, V. (1984). *Vehicles: Experiments in Synthetic Psychology*. MIT Press.
4. Escobar, A. (2018). *Designs for the Pluriverse: Radical Interdependence, Autonomy, and the Making of Worlds*. Duke University Press.
5. Gibson, J. J. (1966). *The Senses Considered as Perceptual Systems*. Houghton Mifflin.
6. Heidegger, M., Macquarrie, J., & Robinson, E. (1967). *Being and Time*. Blackwell.
7. Kato, H., & Billinghurst, M. (1999). Marker tracking and HMD calibration for a video-based augmented reality conferencing system. *The 2nd International Workshop on Augmented Reality (IWAR 99)*, 85–94. https://doi.org/10.1109/IWAR.1999.803809
8. Maturana, H. R., & Varela, F. J. (1980). *Autopoiesis and Cognition: The Realization of the Living*. D. Reidel Publishing Company.
9. Papert, S. A. (1980). *Mindstorms: Children, Computers, And Powerful Ideas*. Basic Books.
10. Varela, F. J., Rosch, E., & Thompson, E. (1992). *The Embodied Mind: Cognitive Science and Human Experience*. MIT Press.
11. Willis, A.-M. (2006). Ontological Designing. *Design Philosophy Papers*, *4*(2), 69–92.
12. Winograd, T., & Flores, F. (1987). *Understanding Computers and Cognition: A New Foundation for Design*. Addison-Wesley.