Emerging Behaviors in Braitenberg-type Robotic Vehicles

I planned and built robots with simple internal structures which can produce behaviors similar to those of primitive animals: My robots like hill climbing and light.

1 Introduction

In his book, Braitenberg [1] describes how robots with simple internal structures can produce behaviors similar to those of primitive animals. One of the early vehicles Braitenberg describes has a light sensor on both sides connected to a motor on the same side. As the light sensor is activated more strongly, the connected motor speeds up. This vehicle fears light; as it approaches light it turns and speeds away. By adding more sensors in different configurations it is possible to create a vehicle that fears light and likes warm places. The goal of my project was to create several robots similar to the ones Braitenberg described and to observe and explain the robots' behaviors. It was also necessary to create an arena in which the robots could navigate.

2 Methods and Materials

2.1 Lego Mindstorm and Technic

The robots were created using Lego Mindstorm and Technic parts. Each robot was controlled by a Lego RCX programmable brick and was powered by two motors configured in a differential drive (the left and right wheels were controlled separately). The RCX was connected to two light sensors and a "gravity sensor," which measured the roll of the robot using a pendulum and a rotation sensor. The RCXs ran the LegOS operating system, which allowed complete control of the robots using the C programming language.

2.2 Closed-loop System

Before beginning work on the main part of the project, I built a two-wheel differential drive robot to familiarize myself with the Lego pieces and the LegOS libraries (see figure 1). Both wheels of the robot were connected to rotation sensors to measure the distance each wheel rolled. The right motor was then set to 75% speed, while the left motor was set to full speed. The RCX was programmed so that it turned the left motor off when the left wheel had traveled 3/16th of a revolution farther than the right wheel, and then turned the motor back on when the distance traveled

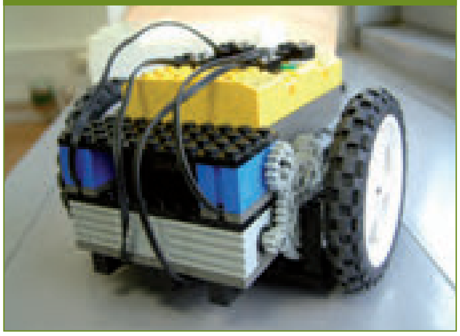
Autor

Samuel Gross, *1988
Fairfax, Virginia

Scitech 2005 des Technion,
Haifa

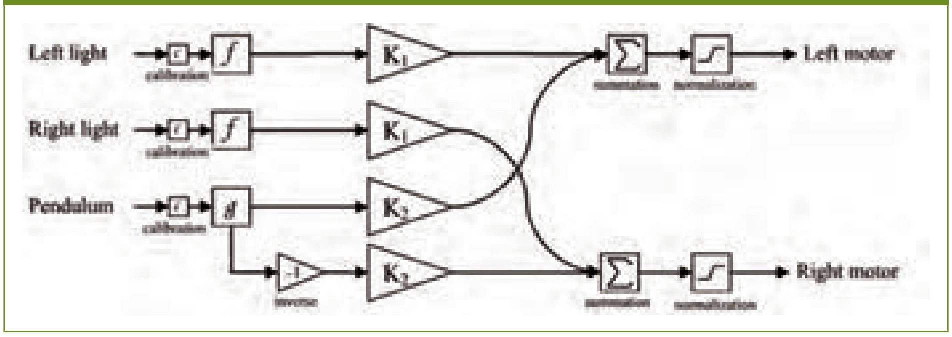
Eingang der Arbeit:
März 2007

Figure 1



The closed-loop demonstration robot

Figure 2



Control flow diagram

was equal. A test of the robot showed that by quickly switching on and off the left motor, it was able to travel along a straight line. The robot is considered a closed-loop system because feedback from adjusting the motors (by means of the rotation sensors) was used to correct the motor speed.

2.3 Structural Design

The robot created for the closed-loop demonstration was unsuitable for the main part of the project because its high center of gravity caused it to be unstable, and there were no obvious places to mount the light sensors. A new robot was created with these problems in mind. The new robot had a much lower center of gravity, and allowed for light sensors to be mounted on both sides near to the ground. Later in the project another robot was created using a similar design with more efficient motors and a stronger structure.

2.4 Control Flow Design

Before programming the robot, a schematic diagram was drawn to depict how the values from the sensors affect the speed of the motors (see figure 2). On the left side of the diagram there are the three inputs: the left light sensor, the right light sensor, and the "gravity" sensor. After calibration, the values from the light and gravity sensors are transformed by the light and gravity transfer functions. For most of the experiments, the transfer functions were either linear or absolute value functions. The constants K_1 and K_2 control the relative effects of the light and gravity sensors; when K_1 is larger than K_2, the light sensors affect the robots movement more than the gravity sensor does. The results from the gravity and light sensors are summed and then normalized to within a range of speeds that the motor supports.

2.5 Gravity Sensor

The gravity sensor consisted of a pendulum connected to a series of gears that spun a rotation sensor. The pendulum was placed so that it swung perpendicularly to the direction of the robot. The placement of the pendulum allowed the gravity sensor to detect the path of steepest ascent or descent. The gravity sensor had a precision of 1.5 degrees because it had a net gear ratio of 15:1 and the rotation sensor measured 16 positions per revolution (360° / 15 * 16 = 1.5°). One of the problems encountered with the gravity sensor was that when the robot made sudden turns, the pendulum swung back and forth, which gave incorrect values for the roll of the robot. To mitigate this problem, the average value over the period of the pendulum was taken from the rotation sensor. In order to determine the period of the pendulum, a short program that measured the time between pendulum swings was created for the robot.

2.6 Environment

In order to observe the full effects of using the gravity sensor and light sensors together, it

Figure 3

The arena under construction

was necessary to create an environment with varying heights and to create a floor pattern with many different shades of gray. The varying heights were achieved by using sheets of plastic supported by cardboard boxes and newspaper (see figure 3). A floor pattern was created in Adobe Photoshop and printed on 72 A3-sized tiles (see figure 4). After the tiles were trimmed, they were glued and taped together in 4x2 and 5x2 blankets, which were then attached to the plastic sheets.

2.7 Turning

The first robot created had difficult turning because its motors did not produce enough torque to overcome the friction between its wheels and the ground. The second robot was able to turn in place by moving one set of wheels forward and the other backwards because its motors were more efficient. When one motor was turned off or ran at a slower speed than the second motor, however, the robot did not turn as was expected. The robot moved in a straight line because the force required to turn the robot was greater than the force required to accelerate the slow running motor (see figure 5). To solve this problem, the program was changed so that the motors always ran at full speed or were set to "brake." When a slower speed was desired, the motor pulses of "brake" and "full speed" were sent to the motor about every 20 ms. For example, repeating the series "brake, brake, full speed" would cause the motor to run at 33 % of its maximum speed.

Figure 4

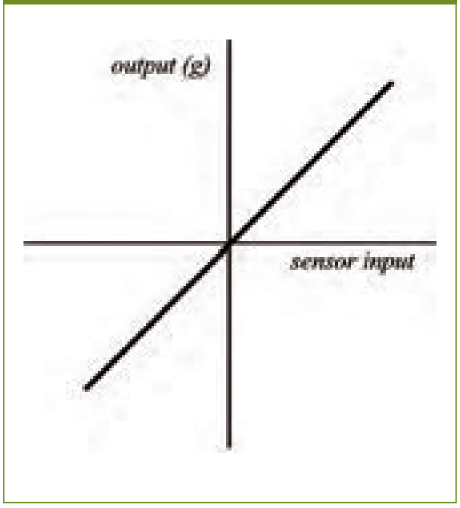
The floor pattern

3 Experimental Results

3.1 Hill Climbing

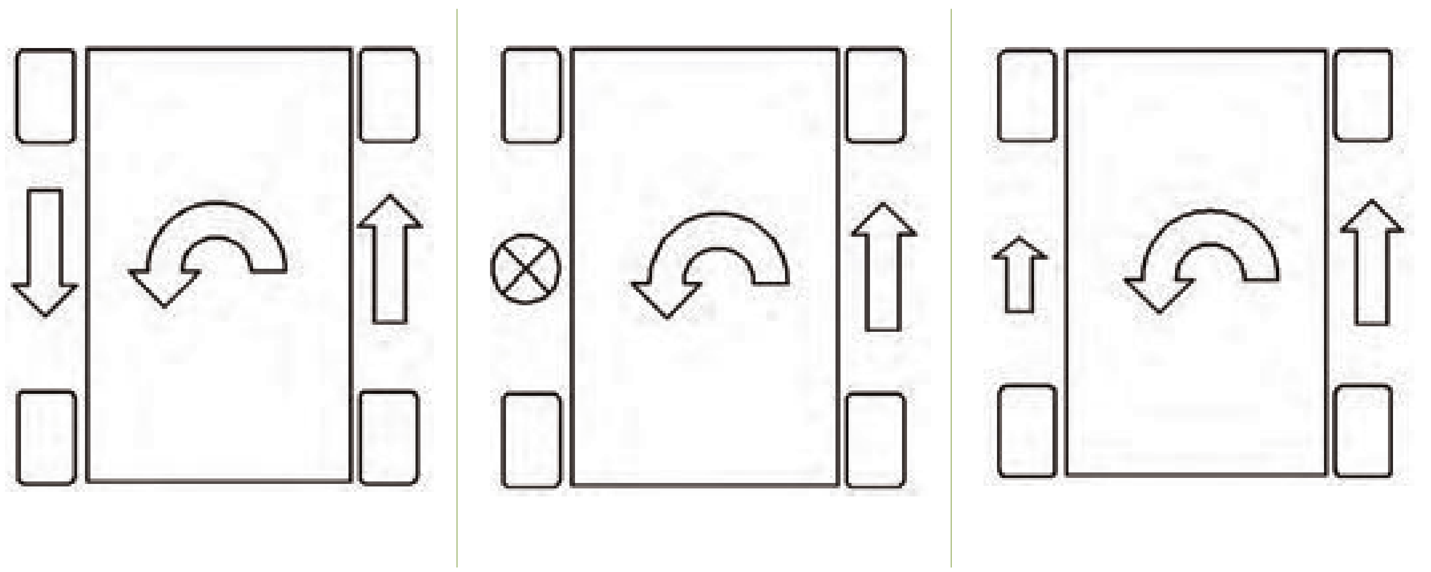
Most of the experiments with the gravity sensor involved a robot that "liked" higher altitudes. This was achieved by using a simple linear function to transform the value from the pendulum (see figure 6). This value is added to one motor and subtracted from the other. A small constant is added to both motors, so that the robot would move forward. As the pendulum swings to the left, the robot rotates clockwise, and as the pendulum swings to the right, the robot rotates counterclockwise. If the robot is heading uphill and deviates slightly to the right, the pendulum will swing to the right, causing the robot to

Figure 6



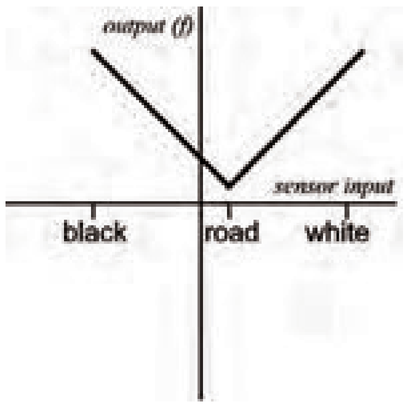
Transfer function for the gravity sensor

Figure 5a, b, c



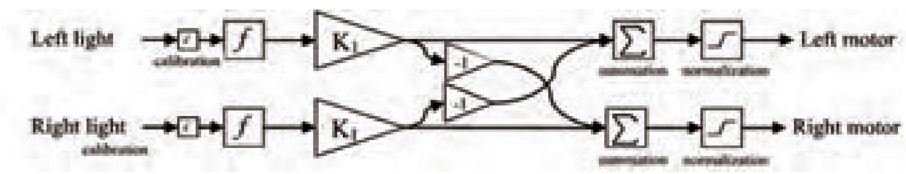
The robot can turn in different kinds: a) the robot turns in place b) the robot turns around stopped wheels c) the robot turns in wide circle

Figure 7



Absolute-value transfer function

Figure 8



Control flow and motor speed for road tracking

turn towards the uphill. Because the pendulum swings perpendicularly to the direction of the robot, there is no way for the robot to determine if it is heading uphill or downhill. However, the robot's journey uphill is stable (any deviation from the path of steepest ascent will be corrected by the pendulum), while its journey downhill is unstable (a slight deviation from the path of steepest descent will cause the robot to turn around and head back uphill).

3.2 Road Tracking

Two types of roads were used to test the robot's tracking ability. The first type of road (seen near the bottom of the floor pattern in figure 4) consists of a gray area that fades to white at the edges. A robot that likes light-gray would follow the light-gray road from either end, by turning away from the lighter colored edges. Similarly, a robot that likes dark-gray would also follow the light-gray road, because it would be repelled by the lighter edges. If one sensor of the robot were over the darker road, the robot would turn onto that road, because it prefers the darker color.

An absolute-value transfer function with its minimum at the road color was used for this experiment (see figure 7). The transfer function causes the robot to speed up on a side that touches the lighter color edge or a darker colored road, causing it to turn back onto the road it prefers.

Experimentation with the robot showed that it did not make sharp enough turns to stay on the road. This problem was solved by subtracting the speed of the left motor from the right motor, and the speed of the right motor from the left motor. A small constant is added to both motors to move the robot forward:

newleftmotor = leftmotor - rightmotor + c

and

newrightmotor = rightmotor - leftmotor + c

(see figure 8).

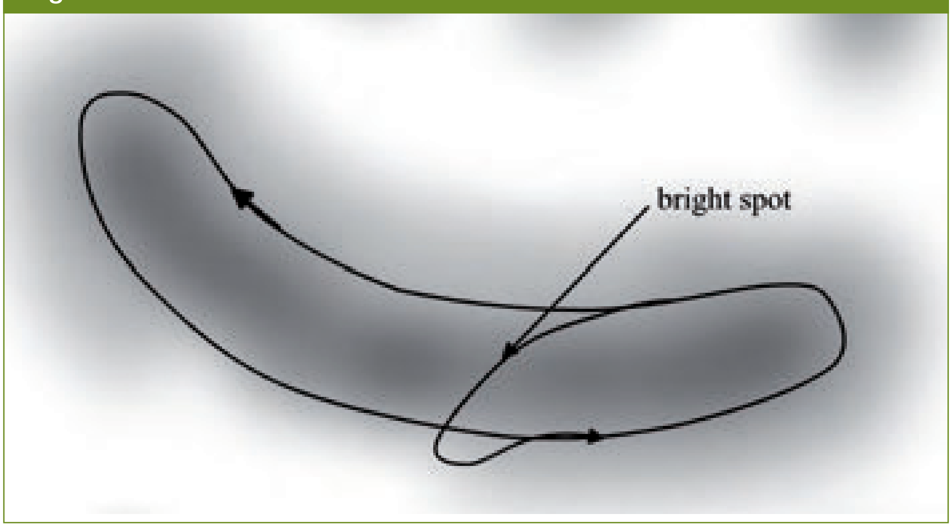
Subtracting the speeds from each other keeps the overall speed of the robot low so that there can be a large difference in speed between the left and right motors.

The second type of road, the "banana," consisted of a dark inner area that fades to light shades of gray (see figure 9). The goal of this experiment was for the robot to follow the banana in either a clockwise or counter-clockwise direction. The transfer function for the other road tracking experiment is not acceptable because, as the robot moves towards moves toward darker or lighter areas, both sensors will read values that are off by similar amounts from their preferred values, so the robot would not turn as desired. The reason for this behavior is that one side of the road fades to a darker color near its edge, while the other side fades to a lighter color.

In order to achieve the desired behavior, reverse transfer functions were used for the left and right motors. If the robot travels clockwise (the right light sensor is over darker colors), then the right motor would speed up near dark colors and slow down near light colors, while the left motor would act the opposite way.

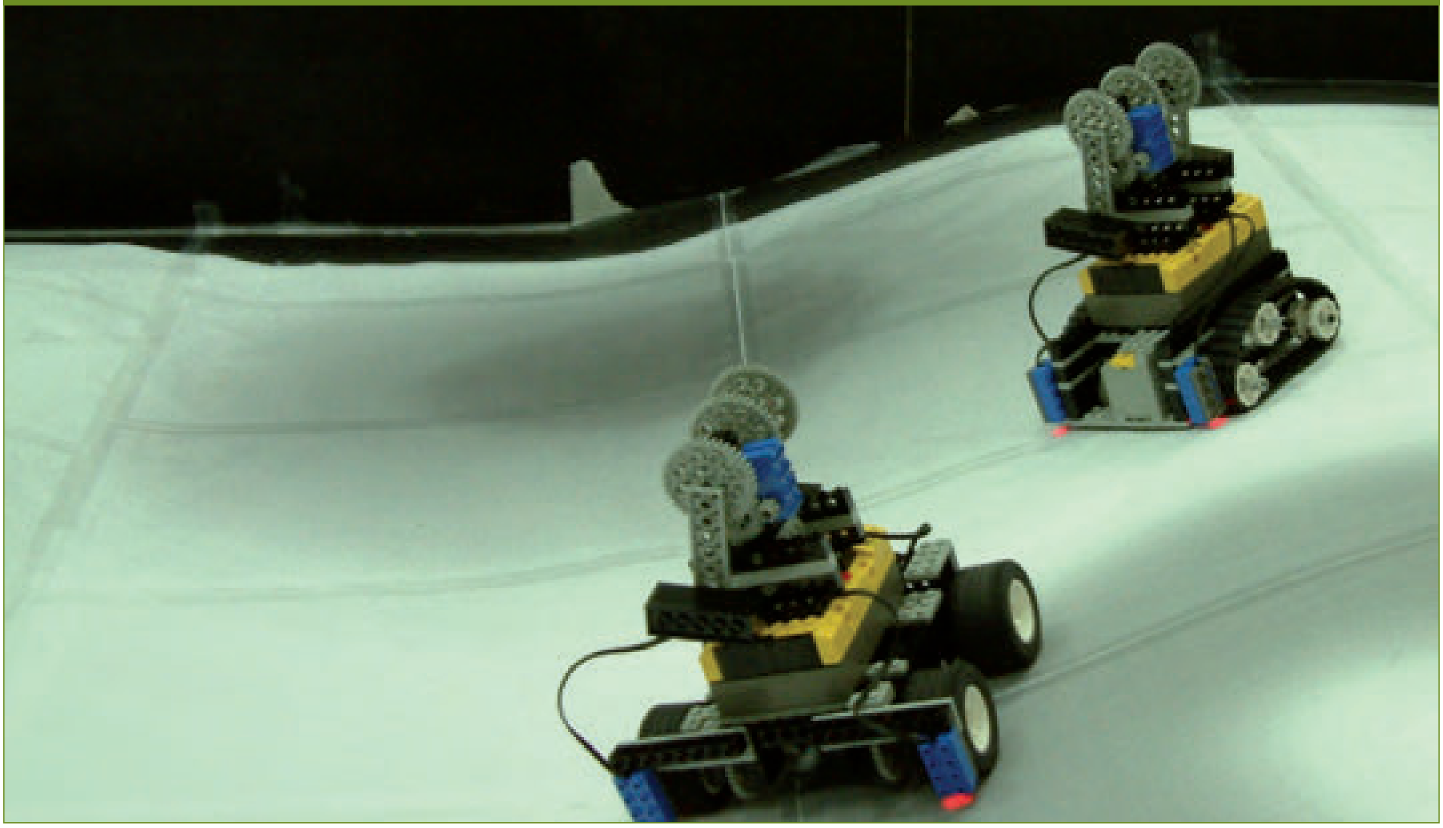
The reverse transfer functions combined with subtracting the speeds of the motors from

Figure 9



Banana with desired path and shortcut marked

Figure 10



The closer robot, which has just come down the hill, begins to turn towards its left because its right light sensor is on the very edge of the road. Once both sensors are on the road, the robot continues to turn due to its gravity sensor. The farther robot is running the hill climbing program without light sensors.

each other allowed the robot to follow the banana. A bright spot near the center of the banana, however, caused the robot to sometimes take a shortcut through the banana. This problem was never completely solved – different starting positions determined whether the robot followed the shortcut.

3.3 Combined Sensors

One of the major goals of this project was to create a robot that combined both types of sensors. To demonstrate this ability, the robot was programmed to dislike dark areas and to like inclines. The light sensors were weighted four times a strongly as the gravity sensor, so that the robot would dislike the dark spots much more than it liked the inclines.

Depending on the initial position, the robot was sometimes able to find the summit while avoiding the dark areas. Other times, a dark spot would force the robot to turn downhill, causing the gravity sensor to have little effect on the path of the robot. In these cases, the robot might have reached the summit eventually, but the effect of the gravity sensor would not have been obvious.

The road-tracking robot also used a gravity sensor combined with light sensors that were weighted more than the gravity sensor. The robot stayed on the road for most of the time, sometimes taking shortcuts. When the road traveled parallel to the hill, the robot began to turn because of the light sensors but continued to turn around the whole way because of the gravity sensor. On the other side of the hill, the robot again turned around for the same reasons. The robot continued traveling back and forth along the segment of the road (see figure 10).

4 Conclusion

The robots, running simple programs and using only a few sensors, successfully demonstrated Dr. Braitenberg's ideas. With only simple control rules and a few sensors, the robots demonstrated interesting and useful behaviors – heading towards the goal, while avoiding obstacles. During this project I created a new Lego sensor, the gravity sensor, as well as several new robot designs.

Acknowledgements

I would like to thank my mentors Noam Gordon and Ronen Keidar of the Technion – Israel Institute of Technology for their time and effort and for teaching me so much through the course of this project.

References

[1] Braitenberg, Valentino, Vehicles, The MIT Press (1984)