Report on an Experimental Run of Braitenberg Type-2 Vehicle Driven by a Homeostat Performed on February 21st, 2026

Overview

A Braitenberg type-2 vehicle controlled by a 6-unit homeostat was run for 2,000,000 simulation steps (~8 hours wall-clock time) in the Ashby fixed-topology mode. The robot started at position (4, 4), facing north, at a distance of 4.243 units from a light source at (7, 7) with intensity 100. By the end of the run, the robot had moved to approximately (16.03, 8.72), at a distance of 9.197 units from the light — further away.

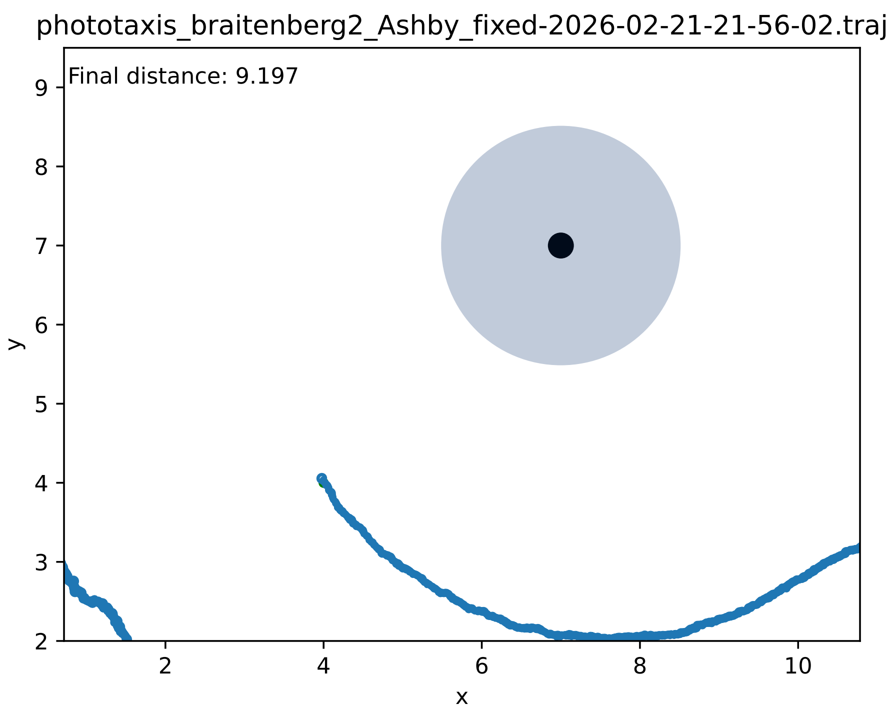

Figure 1: Vehicle trajectory

This result is consistent with the expected behaviour of a homeostat-driven vehicle. The homeostat is an equilibrium-seeking machine that strives to keep its essential variables within acceptable bounds. Since the internal variables are ultimately driven by light irradiance, moving away from the light reduces the disturbance and allows the internal variables to settle closer to equilibrium.

Experiment parameters

Parameter	Value
Experiment	`phototaxis_braitenberg2_Ashby_fixed`
Date	2026-02-21 21:56:03
Total steps	2,000,000
Topology	Ashby fixed (Braitenberg cross-wiring preserved)
Mass range	[1, 10] (randomised)
Motor max speed fraction	0.8
Motor switching rate	0.5
Uniselector time interval	100 ticks
Critical threshold	0.9
Light intensity	100
Light attenuation vector	(0, 0, 1) — pure quadratic decay
Light ambient ratio	0

Data files

- `phototaxis_braitenberg2_Ashby_fixed-2026-02-21-21-56-02.traj` — trajectory
- `phototaxis_braitenberg2_Ashby_fixed-2026-02-21-21-56-03.log` — initial and final conditions
- `phototaxis_braitenberg2_Ashby_fixed-2026-02-21-21-56-03.json` — experiment metadata

Network topology

In the diagrams above, **green** nodes are within normal bounds, **red** nodes are saturated at their deviation limits. Blue arcs are manual (protected) connections; orange arcs are uniselector-controlled. Labels show the effective weight (switch × weight).

The vehicle is a 6-unit homeostat wired as a Braitenberg type-2 (crossed) vehicle:

- **2 Sensor units** (HomeoUnitInput): Left Sensor, Right Sensor — pure input transducers, no uniselector, `always_pos = True`
- **2 Eye units** (HomeoUnitNewtonian): Left Eye, Right Eye — intermediate processing, uniselector active
- **2 Motor units** (HomeoUnitNewtonianActuator): Left Motor, Right Motor — drive the wheels via a sigmoid mapping, uniselector active

Active connections (Braitenberg cross-wiring):

```
Left Sensor   →  Left Eye    →  Right Motor    →  Right Wheel
Right Sensor  →  Right Eye   →  Left Motor     →  Left Wheel
```

Each unit also has a self-connection (state = `manual`, protected from uniselector). All other inter-unit connections exist but are disabled.

Phototaxis Braitenberg2 Ashby Fixed — INITIAL conditions

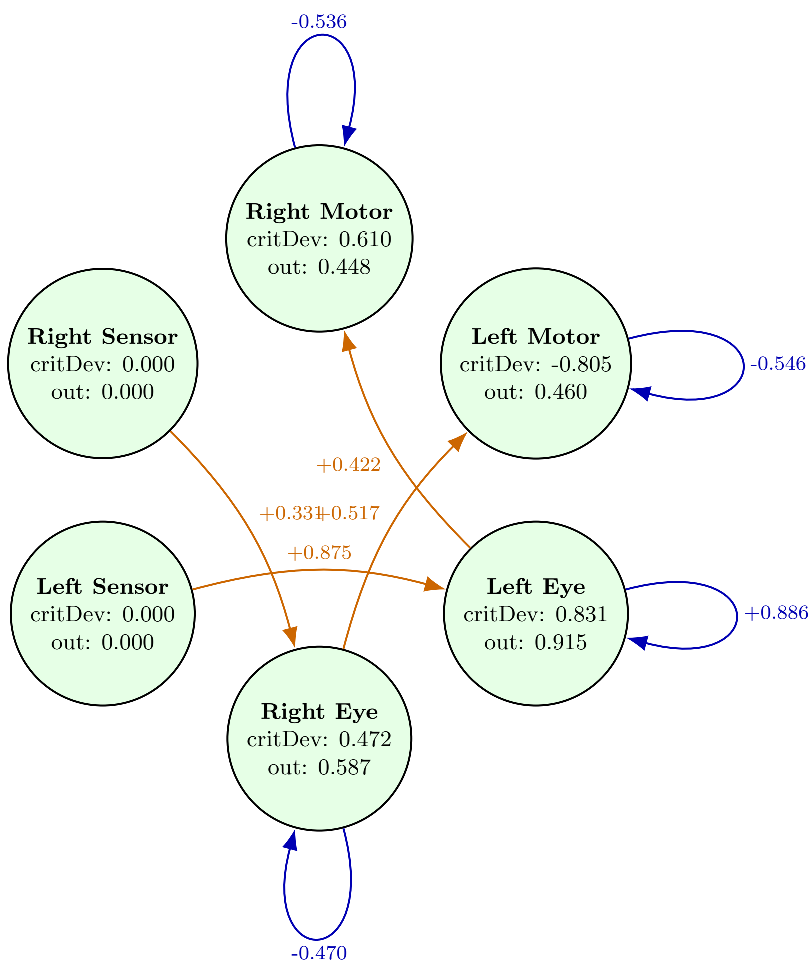

Figure 2: Initial topology

Phototaxis Braitenberg2 Ashby Fixed — FINAL conditions

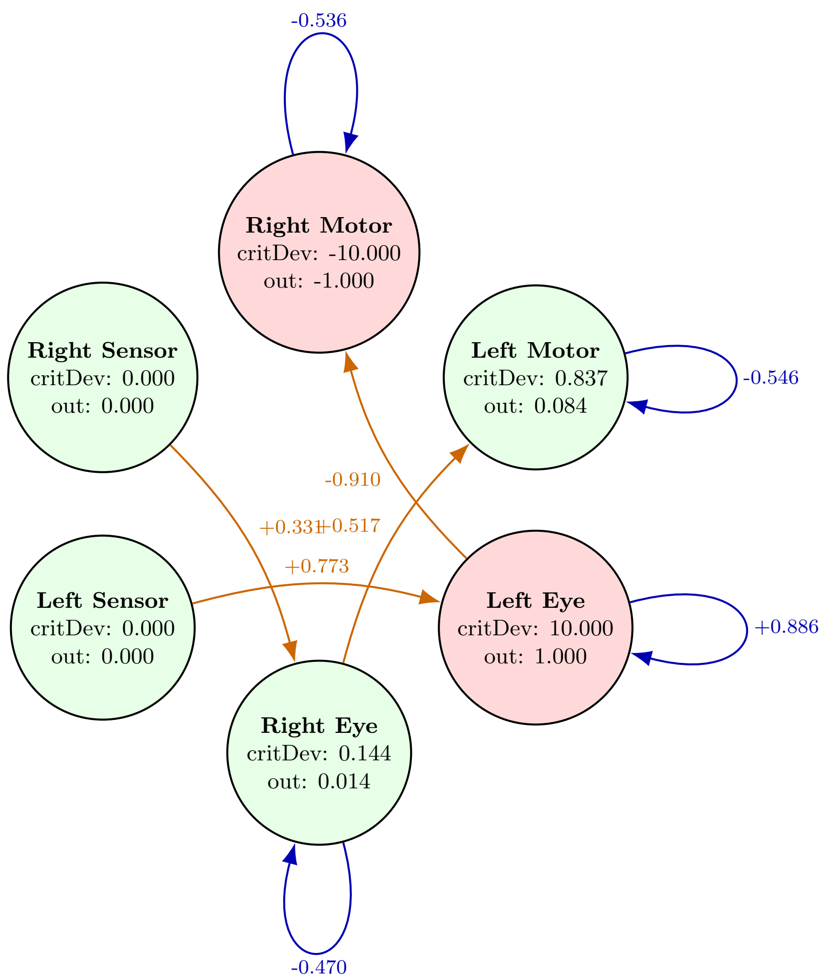

Figure 3: Final topology

1. Light irradiance at start and end

The irradiance at each sensor is computed by the Webots/V-REP model:

```
irradiance = (intensity * (1 - ambRatio) * cos(incidentAngle)) / (c0 + c1*d + c2*d^2)
```

With `ambRatio = 0` and `attenVec = (0, 0, 1)`, this simplifies to:

```
irradiance = 100 * cos(incidentAngle) / distance^2
```

	Distance to light	Approx. irradiance per sensor
Start	4.243	~5.6
End	9.197	~1.2

The sensor transducer (`HOMEO_LightSensor`) returns irradiance as a raw scalar with range `(0, maxRange)` = `(0, 10)`.

`HomeoUnitInput.selfUpdate()` scales this to the unit's deviation range. With `always_pos = True`:

```
critDev_sensor = scaleTo([0, 10], [0, maxDeviation], irradiance)
               = scaleTo([0, 10], [0, 10], irradiance)
               = irradiance
```

Then the output is computed by `computeOutput()`:

```
output = scaleTo([-maxDev, maxDev], [-1, 1], critDev) = critDev / 10
```

So each sensor's output was approximately **0.56** at start and **0.12** at end — a roughly 5-fold reduction in the driving signal.

2. Internal variable values at start and end

Unit parameters (unchanged during the run)

Unit	Mass	Viscosity	Noise	Potentiometer	Switch	Uniselector
Right Motor	1.053	7.472	0.076	0.246	-1	Active
Left Motor	7.519	8.024	0.090	0.896	-1	Active
Left Eye	9.844	2.549	0.048	0.231	+1	Active
Right Eye	5.695	3.046	0.056	0.490	-1	Active

Essential variables: initial vs final

Unit	Initial CritDev	Final CritDev	Initial Output	Final Output
Right Motor	0.61	**-10.0** (pinned at min)	0.448	**-1.000**
Left Motor	-0.805	0.837	0.460	0.084
Left Eye	0.831	**10.0** (pinned at max)	0.915	**1.000**
Right Eye	0.472	0.144	0.587	0.014

Two units (Right Motor, Left Eye) are saturated at their deviation limits. The other two (Left Motor, Right Eye) settled to moderate values near zero output.

Connection changes (initial vs final)

Only two of the active connections were modified by the uniselector during the run:

Connection	Initial Weight	Initial Switch	Final Weight	Final Switch	Changed?
Right Motor ← Right Motor (self)	0.536	-1	0.536	-1	No (manual)
Right Motor ← Left Eye	**0.422**	**+1**	**0.910**	**-1**	**Yes — sign flipped, weight doubled**
Left Motor ← Left Motor (self)	0.546	-1	0.546	-1	No (manual)
Left Motor ← Right Eye	0.517	+1	0.517	+1	No

Connection	Initial Weight	Initial Switch	Final Weight	Final Switch	Changed?
Left Eye ← Left Eye (self)	0.886	+1	0.886	+1	No (manual)
Left Eye ← Left Sensor	**0.875**	+1	**0.773**	+1	**Yes — weight reduced**
Right Eye ← Right Eye (self)	0.470	-1	0.470	-1	No (manual)
Right Eye ← Right Sensor	0.331	+1	0.331	+1	No

The critical change was to the **Right Motor ← Left Eye** connection: the uniselector flipped its sign from +1 to -1 and nearly doubled its weight (0.422 to 0.910).

Motor sigmoid output

The motor sigmoid maps critical deviation to wheel speed:

```
setSpeed = -maxSpeed + (2 * maxSpeed) / (1 + exp(-switchingRate * critDev))
```

With `switchingRate = 0.5` and `maxSpeedFraction = 0.8`:

Motor	Final CritDev	sigmoid(critDev)	Effective wheel command
Right Motor	-10.0	-0.787 * maxSpeed	Strong reverse
Left Motor	0.837	+0.066 * maxSpeed	Slight forward

This asymmetry (left wheel slightly forward, right wheel strongly in reverse) produces a slow, steady curve away from the light.

3. The function mapping light to internal variables

Signal path

The complete signal path from the environment to the wheels is:

```
Light source
      irradiance = 100 * cos(angle) / d^2

Sensor transducer (HOMEO_LightSensor)
```

```
        raw scalar, range [0, 10]

HomeoUnitInput (Left/Right Sensor)
        critDev = scaleTo([0,10], [0,10], irradiance) = irradiance
        output  = scaleTo([-10,10], [-1,1], critDev)  = critDev / 10

HomeoUnitNewtonian (Left/Right Eye)
        Newtonian needle dynamics (see below)

HomeoUnitNewtonianActuator (Left/Right Motor)
        Newtonian needle dynamics, then sigmoid
        setSpeed = -maxSpeed + 2*maxSpeed / (1 + exp(-0.5 * critDev))

Differential drive wheels
        forward_velocity = (vL + vR) / 2
        rotation_rate    = (vR - vL) / wheelSeparation

Robot position in the world
```

Newtonian needle equation

At each timestep, for each internal unit (eyes and motors):

1. **Apply noise** to current critical deviation: `critDev += noise_sample` (normally distributed, scaled by the unit's noise parameter)
2. **Compute torque** as the sum over active connections:

   ```
   torque = Σ_i (incoming_unit_i.output * conn_i.switch * conn_i.weight + conn_noise_i)
   ```

3. **Compute new needle position** (linear method):

   ```
   normalizedViscosity = viscosity / maxViscosity    (maxViscosity = 10)
   effectiveForce = torque * (1 - normalizedViscosity)
   velocity = effectiveForce / mass
   newCritDev = clip(critDev + velocity, -maxDev, maxDev)
   ```

4. **Compute output**:

   ```
   output = scaleTo([-maxDev, maxDev], [-1, 1], critDev) = critDev / 10
   ```

5. **Uniselector check** (every 100 ticks): if `|critDev| >= critThreshold * maxDev` (i.e. `|critDev| >= 9`), the uniselector fires and randomly reassigns weights on connections whose state is `'uniselector'`.

Tracing the final steady state

Using the final connection parameters to verify the equilibrium:

Right Eye (CritDev = 0.144, Output = 0.014): - Self-connection: 0.014 * (-1) * 0.470 = -0.007 - ← Right Sensor: ~0.12 * (+1) * 0.331 = +0.040 - Torque

+0.033 - Effective force = 0.033 * (1 - 3.046/10) / 5.695   0.004 - Very small acceleration — quasi-stable near zero. The reduced light at distance 9.2 keeps the driving input low enough that negative self-feedback can balance it.

Left Eye (CritDev = 10.0, Output = 1.0) — **saturated**: - Self-connection: 1.0 * **(+1)** * 0.886 = **+0.886** (positive feedback) - ← Left Sensor: ~0.12 * (+1) * 0.773 = +0.093 - Torque   +0.979 - The positive self-connection (switch = +1) creates runaway positive feedback. Once the deviation crossed the saturation threshold, the self-reinforcing loop locked the Left Eye at maximum deviation regardless of the (now-small) sensor input. This is a stable attractor of the dynamics.

Left Motor (CritDev = 0.837, Output = 0.084): - Self-connection: 0.084 * (-1) * 0.546 = -0.046 - ← Right Eye: 0.014 * (+1) * 0.517 = +0.007 - Torque -0.039 - Effective force = -0.039 * (1 - 8.024/10) / 7.519   -0.001 - Negligible — quasi-stable. The Right Eye's near-zero output means this motor is effectively self-damped.

Right Motor (CritDev = -10.0, Output = -1.0) — **saturated**: - Self-connection: (-1.0) * (-1) * 0.536 = **+0.536** (effectively positive feedback when output is negative and switch is negative) - ← Left Eye: 1.0 * **(-1)** * 0.910 = **-0.910** - Torque   -0.374 - The Left Eye (locked at +1) pushes through the flipped connection (switch = -1, weight = 0.910) to drive the Right Motor strongly negative. The self-connection, despite having negative switch, actually reinforces the negative deviation (negative output times negative switch = positive contribution, but that adds to a positive direction; however the Left Eye's -0.910 dominates). The net torque is negative, keeping the Right Motor pinned at -10.

The uniselector's decisive action

The single most important event during the entire run was the uniselector's modification of the **Right Motor ← Left Eye** connection:

	Before	After
Weight	0.422	0.910
Switch	+1	**-1**
Effective weight	+0.422	**-0.910**

This sign flip reversed the effect of the Left Eye on the Right Motor. Since the Left Eye was already locked at positive saturation (due to its positive self-feedback), this change permanently drove the Right Motor to negative saturation, producing the sustained asymmetric wheel command that steered the robot away from the light.

Interpretation

The homeostat achieved its objective of minimising disturbance to its essential variables, but not in the way a naive observer might expect. Rather than developing a coordinated phototactic behaviour (approaching or fleeing the light in a straight line), the system found a degenerate but effective solution:

1. **Two units saturated** (Left Eye, Right Motor) — these are no longer functioning as adaptive elements. They are locked into fixed output values by self-reinforcing feedback loops.
2. **Two units near equilibrium** (Right Eye, Left Motor) — these settled to low deviation values because the reduced light at greater distance provides only a weak driving input, easily balanced by their negative self-feedback.
3. **The robot curves away from the light**, reducing irradiance over time, which further reduces the driving input to the non-saturated units, stabilising them closer to zero.

This is a form of **ultrastable** behaviour in Ashby's sense: the system reorganised its parameters (via the uniselector) until it found a configuration where the essential variables remain within acceptable bounds. The fact that this was achieved by saturating half the units rather than by fine-tuned coordination is characteristic of the homeostat's indifference to elegance — it finds *any* stable configuration, not necessarily an optimal one.