Analog Computation for Mobile Robotics Education

Bradley E. Bishop, Carl E. Wick

United States Naval Academy
Systems Engineering
105 Maryland Ave (Stop 14a)
Annapolis, MD 21401
bishop@usna.edu, wick@usna.edu

Abstract: In this paper, we present a mobile robot design exercise that relies on simple analog circuits to accomplish tasks that are typically carried out using microcontrollers. Students are challenged to use simple analog sensors and IC's to develop a mobile robot that is attracted to a light source while avoiding obstacles. The primary outcome of this exercise is a deeper understanding of computation for mobile robots, and a clearer view of possible alternatives to embedded processors for low-cost applications.

Background

Mobile robotics is a well-recognized motivational vehicle for engineering education. Not only is it an enjoyable topic for many students, but it has a broad appeal due to its wide scope, including aspects of electrical, mechanical and computer engineering. Further, the design of such systems is an excellent tool for reinforcing fundamental engineering concepts. It is important for instructors in robotics to understand, however, that robotics is not just a tool to teach other aspects of engineering. Rather, it is a robust and mature discipline with applications in a wide range of fields. The precept of this paper is that students are taking a course in mobile robot design in order to be better equipped to act as robotics designers and engineers.

In this paper, we address some key issues about computation that arise in robotic systems; namely, what is the appropriate type and level of computation for a task? In our present microprocessor-intense society, students often believe that digital computer control is the only way to affect intelligent-like behavior in robotic systems. We describe in this paper a set of experiments and classroom discussions that allow students to compare traditional (microprocessor-driven) mobile robotics computation with analog-only solutions using the principles of Valentino Braitenberg (from his famous text *Vehicles*[1]) and the BEAM concept of Mark Tilden[2].

In the end, our students construct a fully functional analog robot, and are able to draw valuable new insight into the fundamental nature of computation for mobile systems, as opposed to the device used for these computations. This new knowledge allows students to design mobile robot computational suites that are more appropriate for given sets of objectives, and gives them a more global perspective of robot control systems. The discussion and experiments can be integrated easily into course segments involving artificial intelligence, biomimetics, or behavior-based systems. Furthermore, because these innovative analog methods seem to mimic the behavior of living systems, they are often seen by engineering students to be a new and exciting topic of study and research.

The Set-Up

A common task for a mobile robot is to move toward some stimulus, whether it be a light source, chemical signature or radio beacon, while simultaneously avoiding collisions with the environment. Methods for accomplishing this range from traditional feedback control techniques[3] to behavior-based[4] and artificial intelligence[5] control schemes.

Given a challenge, stated as simply as "head toward light and avoid collisions," students will quickly select a vehicle frame and sensing modalities, locate appropriate hardware, and set about developing code for their favorite microprocessor. The most common solution to this challenge, across all curricula, would be the following:

- Chassis/locomotion: Two-wheel differential drive with slider or castor.
- Sensors:
  Light: photodiode, phototransistor or photoresistor
  Collision: acoustic ranging sensors
- Computation: microprocessor (Handy Board, Brain Stem, Basic Stamp, PIC, etc.)

There is nothing fundamentally wrong with the vehicle described above. It will, after some careful signal conditioning, computer interfacing and coding, perform quite well. The difficulty is that the system is over-designed.

Students in robotics courses often fall victim to the classic problem: when all you have is a hammer everything looks like a nail. In the case above, the hammer is a microprocessor, and the students will bang it onto the robot until it fits, no matter how many times they have to hit it.

The point of the exercise presented herein is that students must *design* (or select) the computational capabilities of a robot with the same care that they take designing the other components of the system. Microprocessors vary in complexity and capability, but students are rarely exposed to more than one standard system in a given curriculum. Therefore, they do not typically take recourse to other devices when faced with a challenge that looks to require computation. Unfortunately, this sort of shortsightedness often results in sub-optimal systems with a distinct lack of elegance. In this exercise, we attempt to demonstrate to the students that computational capability runs a full spectrum, on which the microprocessor is merely one band.

The Lead-In

To prepare for the challenge of building an analog robot, students are first introduced to the concepts laid out in Valentino Braitenberg's *Vehicles: Experiments in Synthetic Psychology*[1]. In this folio, Braitenberg investigates the nature of biological and neurological response through a series of simple thought experiments based 'vehicles' with sensor-motor connections of the most straightforward type. Students look at diagrams such as that shown in Figure 1 and determine the fundamental nature of the response of the vehicle to appropriate stimuli. In Figure 1, the simple differentially-driven vehicle responds to a single stimulus type in a straightforward manner. Each sensor is connected to one motor as shown. When the sensor reading increases, so does the power to the motor in question (due to the + sign on the connector). The vehicle shown in Figure 1 turns toward a stimulus and increases in speed as it approaches the source (assuming that the sensor response increases with increased stimulus). This and many other examples are described in Braitenberg's text[1].

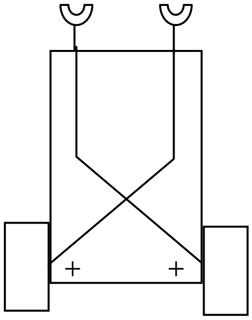


Figure 1: Simple Braitenberg-style vehicle

Having seen a variety of more advanced 'vehicles,' students generally adopt and accept the idea that complex (or seemingly complex) behaviors can result from simple sensor-motor connections. At this point, it is appropriate for the students to be challenged to make the most straightforward vehicle possible to accomplish a desired task of moving toward a light source while avoiding obstacles.

The Experiment

Students are provided with the following as building blocks for their project.

- LEGO Mindstorms hardware (blocks, wheels, etc.) and motors. (No sensors or RCX)
- APEX PA26 power operational amplifier
- LM324 low power quad operational amplifier chip
- Various CdS cells (photoresistors)
- Sharp GP2D12 analog ranging sensor
- Resistors and potentiometers as needed
- Battery packs: 4-cell "AA" size
- NiMH AA batteries
- Mini bread-boards

Students design and implement an analog controller and mobile robot using these devices. The system development (ideally) follows a logical progression:

Step 1) Build a power supply

Before any hardware testing or design may occur, students must build a power supply. Looking at the power requirements of the available IC's and devices, the best solution is a dual-sided supply of +/- ~5V[i], generated by connecting two 4-cell NiMH AA battery packs in series, with the connection point being ground. It is possible to achieve the tasks with the given devices using a single-sided supply, but the added benefit of the dual-sided version is that the vehicle can move both forward and backward.

Step 2) Characterize the sensor response

CdS cells have high resistance in low-light conditions, but the resistance drops dramatically when they are exposed to bright light. The range of values for the CdS cells used ranged from less than 100Ω in bright light to more than 100KΩ in complete darkness.

The Sharp GP2D12 ranging sensor is a relatively inexpensive, low-power, compact device that provides an output signal that is inversely proportional to distance over the range of ~4 – 48cm. Maximum output (at 4cm) reaches a value between 2.6V and 3.2V from a 5V supply. Closer than 4cm, the response dips back down rapidly.

Step 3) Build appropriate signal conditioning.

Students typically want the output of the light sensor to increase with increasing light. Since the resistance of the sensor *decreases* as light intensity increases, students must take care in development of the signal conditioning. The most common system is a voltage divider using a 5K potentiometer and the CdS cell as shown in Figure 2. Other solutions involve placing the CdS cell in an op-amp circuit to make a variable gain.

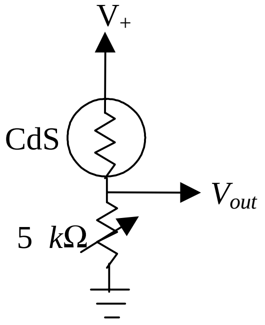


Figure 2: CdS setup

The output of the ranging sensor *increases* as distance *decreases*. This is appropriate, as the "avoidance" stimulus should be thought of as increasing as the system gets closer to collision. As such, there is no need for special signal conditioning, although it is very important to note that the GP2D12 is a fragile device and cannot be attached to a low impedance load. Thus, students are encouraged to run the GP2D12 output directly into a terminal of an op-amp, which will draw negligible current.

Step 4) Develop system topology

The students now have sensors that provide increasing readings as the stimulus increases. They quickly realize that a differential arrangement is needed to track a desired stimulus. As such, they develop two sets of sensors; one left and one right. The most straightforward vehicle design, based on these observations, is a simple differential drive. Drawing sensor-motor connections (with appropriate sign) is the next step in developing a system topology. There are two fundamental ways in which the system can be connected, as shown below.

---

[i] Approximate value. NiMH cells have a nominal full-charge voltage of 1.2V,so a 4-cell pack should provide 4.8V. In reality, fully charged cells have slightly higher voltage.

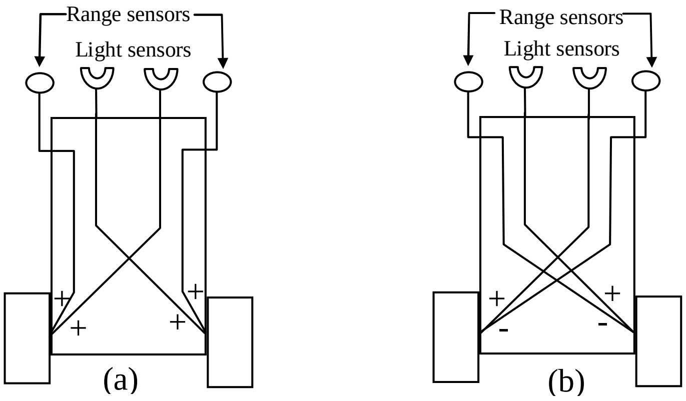


Figure 3: Basic robot architecture

In Figure 3 (a), we see all positive connections, which will cause the vehicle to speed up a wheel as it nears an obstacle. This seems like a good idea until the students realize what will occur when the vehicle has an obstacle on each side. This quickly results in a switch to the vehicle topology shown in Figure 3 (b), which works well.

Step 5) Perform signal conditioning / analog computation

At this point, students need to generate the signals required for analog control. Using the topology shown in Figure 3(b), there are a variety of ways to build the system. The important observation is that the inhibitory signal from the range sensor should be able to completely overcome any possible output from the light sensor. For simplicity, most of the students use a +5V supply for the GP2D12 and light sensors. Thus, the output of a light sensor has a maximum value near 5V while that of the GP2D12 is usually in the range of 2.6V.

As mentioned, it is not appropriate to connect a GP2D12 output directly to a load. Thus, students begin their design by buffering the GP2D12 outputs. A simple voltage follower proves effective in protecting the sensor, and students quickly develop the arrangement shown in Figure 4 to produce an output that is $V_{light} - 2*V_{range}$, using an LM324 quad operational amplifier. Note that the resistor values shown in Figure 4 are examples only. The values given reflect a 2:1 or 1:1 gain as required by the overall circuit function of $V_{light} - 2*V_{range}$.

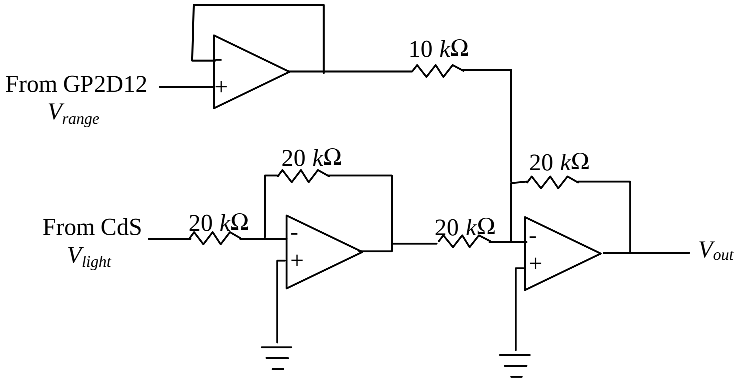


Figure 4: Basic circuit to provide $V_{light} - 2*V_{range}$

It is generally necessary to show the students that the circuit developed in Figure 4 is identical to the simplified circuit shown in Figure 5 except for a sign change. After some thought, students typically understand that the orientation of the output from the analog controller is irrelevant, so long as the magnitude increases and decreases properly and the sign can change when needed. The direction of motor rotation is entirely determined by the polarity of the connectors, as will be discussed next.

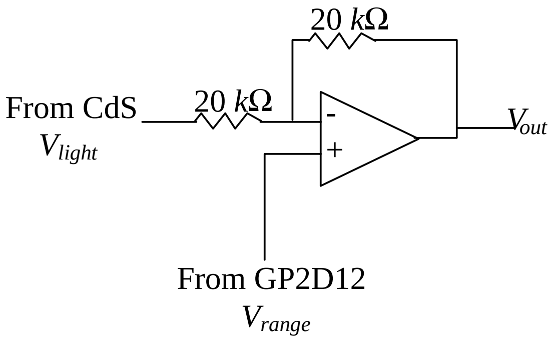
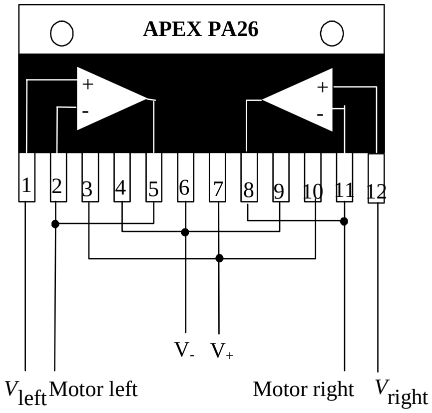


Figure 5: Simplified analog computation circuit, $V_{out} = -V_{light} + 2*V_{range}$

Step 6) Design the power system

The output of a low power op-amp IC such as the LM324 is not sufficient to drive a motor of even moderate size. Students must build a power interface using the APEX PA26 power op-amp chip in voltage follower mode as shown. The connection to a motor is made using a modified LEGO connector. Polarity is adjusted to achieve proper rotation when the motors are eventually mounted (a negative voltage turns the motors forward). Two copies of the circuit in Figure 5 are used to generate the V_{left} and V_{right} signals. One of the motor terminals from each motor is connected to the PA26 as shown, with the other attached to ground.



The circuit in Figure 5 and the PA26 both require dual-sided power supplies. The supply is made from two 4-cell AA battery packs with a single common lead (which is ground). NiMH AA cells are used.

Step 7) Design the robot chassis

Now that the analog controller has been designed and tested, the motors must be coupled to a chassis. The chassis is a differential drive system that must be carefully designed. The three main characteristics that are required are:

i) Low axel and contact friction. In order for the system to be flexible and maneuverable, small changes in wheel powering for each side must result in measurable turning. As such, skid steering (in a tank-like configuration) is inappropriate. The most common type of system has two driven wheels and a passive castor or slider, as seen in Figure 6. The allowable friction on the passive contact point depends on its distance from the wheel axis and the available torque, but it should be kept as low as possible for greatest agility.

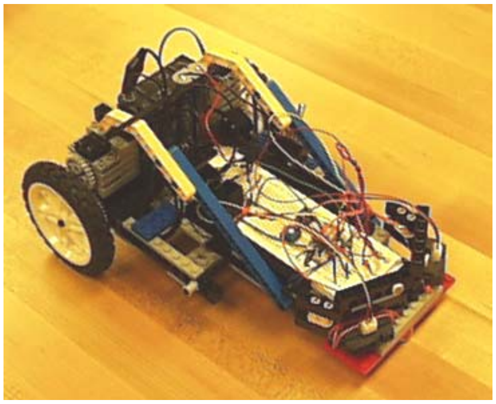
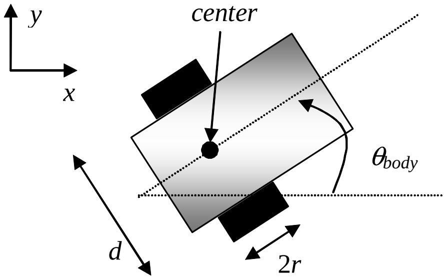


Figure 6: Example robot chassis

ii) Agility. The robot must be able to turn sharply given small differentials in wheel power. This relies on low friction as mentioned, but also on the wheelbase and the gear train. The equations for motion for a differentially-driven robot are:



$$\| v_c \| = \left(\dot{\theta}_r + \dot{\theta}_l\right) r / 2$$
$$\dot{\theta}_{body} = \left(\dot{\theta}_r - \dot{\theta}_l\right) r / d$$
$$\dot{x} = \| v_c \| \cos(\theta_{body})$$
$$\dot{y} = \| v_c \| \sin(\theta_{body})$$

where $\dot{\theta}_r$ is the angular velocity of the right wheel and $\dot{\theta}_l$ is that of the left wheel. Given constant wheel velocities, a decrease in wheelbase *d* results in an increase in the angular velocity of the body. This indicates that a vehicle should be narrow to achieve high agility. Students who use these equations to evaluate potential designs may note that an increase in wheel radius should result in an increase in body angular velocity as well, but they should be cautioned to consider the implications of increased wheel radius on required torque.

iii) Gearing/torque. Finally, students must carefully consider the level of gearing needed to achieve motion. There is one subtlety in this design that often escapes the students until a system is built. While they recognize that an increase in gearing to provide required torque results in a decrease in speed, students often overlook the effect of this gearing on the agility of the vehicle. Given two robots, identical except that one has high gearing and the other low, the highly geared system will react more slowly to an identical stimulus. Fortunately, the vehicle is also traveling more slowly and thus has a greater time to react.

The subtlety here involves the nonlinear and non-ideal characteristics of drivetrain dynamics, which actually aid in agility. As gearing increases, the effective differential between two wheels' speed decreases (as expected), but the actual results are more significant than would be anticipated from the gear ratio alone. This added effect is due not to the gearing itself, but to the reduced impact of friction and loading on the wheels. For systems with little gearing, friction actually aids in effecting quick heading changes as one wheel slows down.

Step 8) Sensor placement

Once the chassis and sensors are complete, it is necessary to integrate sensors into the robot. The CdS cells used are sensitive to light falling on their surface from any direction. Thus, initial placement of the sensors typically results in poor performance if the two sensor suites are not separated by a baffle or focused using lenses. These approaches are easily managed, and are typically done as a retrofit.

Placement of GP2D12 ranging sensors requires some attention and involves difficulties that are not entirely obvious. It is clear that the ranging sensors must be placed so that the device does not run into obstacles with some outboard component. As such, typical placements involve locating the sensors near the midline of the vehicle, angled outward. GP2D12 sensors are very directional, so students must take care to place them so that the vehicle cannot impact typical obstacles (such as table and chair legs). A narrow chassis is helpful. Placing the sensors slightly differently can also aid in this step by crossing the "beam" of each sensor across the midline as shown below.

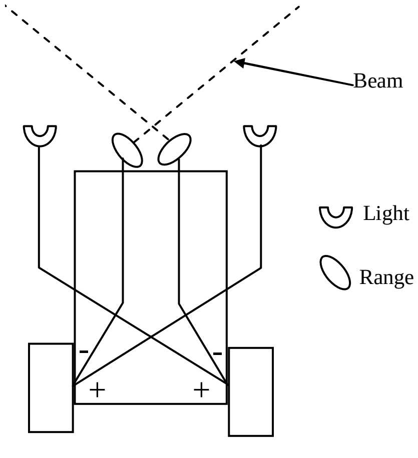


Figure 7: Crossed Ranging Sensor Setup

Step 9) Tuning

Once the entire vehicle is completed and functioning, it is necessary for students to make adjustments to enhance robot performance. Primary among these adjustments is to balance the light sensors so that the vehicle moves in a straight line when both sensors experience identical light intensity. This is accomplished through adjustment of the potentiometers shown in Figure 2. Additional tuning involves adjusting wheel type and size, modifying the passive floor contact (slider or castor) and tweaking the sensor placement.

Evaluation and Outcomes

Students are asked to demonstrate their robot and prepare a paper on their design, focusing on the novel aspects of using analog computation. Most of today's students have never seen the sort of analog computer that many professors used during their time as a student, so this exercise provides for them their first introduction to this concept. In their paper, students are asked to discuss the important aspects of analog computation: effectiveness, efficiency and significance. This exercise helps to crystallize the experiences of the exercise into a clear picture of analog computation. References are made to various analog circuits used in other courses to provide differentiation, integration, etc. Students finish the experience with a new tool and a new appreciation for selection of appropriate computational facilities for mobile robots. Later experiments focus on aspects of microprocessors that make them uniquely useful for tasks that analog systems are unable to accomplish.

Conclusions

In this paper we have described an exercise that focuses on the use of appropriate levels and type of computation for mobile robot systems. Through experimentation, students are exposed to the strengths and weaknesses of analog computation. They design and build robot systems to accomplish a common task: tracking a stimulus while avoiding obstacles. This experiment leaves them with a fresh insight into the usefulness of standard analog techniques in mobile robotics, and broadens their understanding of computation. The students who took the course in which the discussed experiment is offered were seniors (1/C midshipmen) in Systems Engineering at the United States Naval Academy. Approximately 45 students take this course each semester.

Bibliography

1 Braitenberg, V., *Vehicles: Experiments in Synthetic Psychology*, The MIT Press, Cambridge, MA., 1984.
2 Hrynkiw, D. and Tilden, M., *Junkbots, Bugbots & Bots on Wheels: Building Simple Robots with BEAM Technology*, McGraw-Hill, New York, 2002.
3 Dorf, R. C. and Bishop, R. H., *Modern Control Systems*, Eighth Edition, Addison-Wesley, Reading, MA, 1998.
4 Arkin, R., *Behavior-Based Robotics*, The MIT Press, Cambridge, MA., 1998.
5 Murphy, R., *Introduction to AI Robotics*, The MIT Press, Cambridge, MA., 2000.

Biographical Information

BRADLEY E. BISHOP is an Associate Professor in the Weapons and Systems Engineering Department at the United States Naval Academy (USNA). He received the B.S. degree in Electrical Engineering from Michigan State University in 1991, and the M.S. and Ph.D. degrees in EE from the University of Illinois at Urbana-Champaign in 1994 and 1997, respectively. He is the founder of the Mobile Robotics Laboratory at USNA. His research interests include robot swarm control, autonomous surface vessels, and nonlinear control.

CARL E. WICK is a Professor and Chairman of the Weapons and Systems Engineering Department at the United States Naval Academy (USNA). He received the B.S. degree from USNA in 1970, the M.S. degree from the Naval Postgraduate School in 1978, and the D.Sc. degree from The George Washington University in 1993. His research interests include autonomous underwater vehicles, sensor systems, and contemporary microprocessor systems