Unit 3 - Introduction - Robotic Movement 2 – Affectors

October 17, 2024

8:07 PM

Affectors are the key to allowing a robot to interact with its environment. Affectors affect the environment by touching, grabbing, and other motions. In this unit you will examine different affectors in robotics.

**Learning Objectives**

After completing this unit, you should be able to

* describe various types of affectors, both how they are created and how they work.
* discuss the creation of robots that employ affectors to interact with their environment.

**Readings**

Please read the following chapters in the textbook:

* Grasping at Straws (Chapter 6)

**Questions to Ponder**

At the end of each chapter in the assigned readings there are questions labelled “Food for Thought”. Please answer these questions as best you can in your weblog, which will become part of your portfolio of competence submitted for marking during this course.

**Exercises**

Exercises for this unit can be found in the *Instructor’s Weblog* on the Landing. Please follow along with the exercises and programs using your own Arduino kit, and keep a record of your explorations in your own weblog.

**Further Readings**

At the end of each chapter in the assigned readings you will find a section titled “Looking for More.” While the links and readings mentioned in this section are not assigned, please feel free to examine them if you are interested, or ask questions on the Landing.

From <[*https://comp.athabascau.ca/444/r1/unit03.html*](https://comp.athabascau.ca/444/r1/unit03.html)>

Unit 3 - Instructor Blog Entry

October 17, 2024

8:27 PM

This unit completes the section on robotic movement. By now you should have completed all SIK Guide tutorials and circuits from Circuit #1 through Circuit #12, inclusive.

**Design Question**: How many motors, and of what type, would you require to make a fully functional robotic arm that had a working elbow, wrist, and end affector (i.e. a simple clamp)? What components would you add if you wanted the clamp to be able to tell how hard it was grabbing an object such as an egg (i.e. to avoid crushing it)? Discuss your design in your weblog in detail, especially describing the choice of motor for each joint, the degrees of freedom and the range of motion.

(See: page 63 of textbook)

**Programming/Circuit Task**: Since we don't have all the hardware to build a robotic arm, imagine you have been given the task of creating the elbow joint. Select the appropriate motor for this task, and then create a program and circuit using your Arduino which can demonstrate your motor performing the correct elbow movement. It may help if you tape an object such as a popsicle stick, drinking straw, or long skinny piece of paper to your motor to demonstrate the movement of the lower portion of the arm under control of your program. Your program should take as input a number of degrees to move the elbow from an arbitrary starting position. For example, if you choose 'fully straight' as the starting position, this will be designated 0 degrees (start). The arm could then bend about 170 degrees, indicating 'fully bent' (check the amount of bend on your own elbow from hand straight out to hand near your shoulder for reference).

As always, keep detailed notes in your weblog of the entire development, testing and debugging process. If you have video capabilities, upload a video of your completed project. You should also include a listing of your code in your weblog, as well as a description of the final wiring of your project.

Design: Robot Arm

October 17, 2024

11:32 PM

**How many motors, and of what type, would you require to make a fully functional robotic arm that had a working elbow, wrist, and end affecter (i.e. a simple clamp)? What components would you add if you wanted the clamp to be able to tell how hard it was grabbing an object such as an egg (i.e. to avoid crushing it)? Discuss your design in your weblog in detail, especially describing the choice of motor for each joint, the degrees of freedom and the range of motion.**

**(See: page 63 of textbook)**

I can think of three possible ways for a robot to be able to pick up an object without crushing it, with a minor variance only at the end. The primary issues to over come is a term we learned in this chapter called manipulation problems and how to identify grasp points.

If a robot has an immersive AI, then a simple image recognition of the object, could allow the robot to search up typical physical attributes of an object (like an egg for example) including it's weight, ability to estimate center of gravity, the shells tensile strength etc. A properly programmed robots internal state should allow the robot to know how strong it is and how to a range of force. Ie: If the robot applies 12 watts to an actuator, it impose 12 ft/lb if the robot applies 5 watts, its only 5 ft/lb. So please be gentle!

Another option would be to rely on trial and error. Have the robot attempt to pickup the object to only a very short height, with increasingly greater grip strength, or lower grasp point, until the robot can successfully pick it up without dropping. This is a dumb robot, so it never remembers that, just keeps trying.

Finally, I suggest a specifically designed endeffector for the task. If we know that the task of the robot is to pickup eggs, lets design it that way. Specifically, an endeffector that matches the shape of an egg, is designed with firm but gentle method of grasping it - perhaps some malleable or elastic medium is what's in contact with the egg, but that is mounted on a more firm portion which is moved by the actuator.

**Goal:**

To design a fully functional robotic arm with elbow, wrist and clamp

Able to pickup an egg without breaking it

Simple design minimal components, maximum reliability

**Design:**

**3D Model:**

My drawing and artistic skills are dismal at best, so I'll use my previous Sketchup experience to draft a design of the robot arm. I'll do my best, but my experience is drafting decks and building plans with dimensional lumber, not robots with servos and links!

[STL Format](https://github.com/malcolmsdad/COMP444/blob/main/Excercises/Egg%20Picker%20Upper.stl)

[SKP Format](https://github.com/malcolmsdad/COMP444/blob/main/Excercises/Egg%20Picker%20Upper.skp)

**Overview:**

[Complete Diagram](https://github.com/malcolmsdad/COMP444/blob/main/Excercises/U3%20-%20Arm%20-%20Complete%20Assembly.png?raw=true)

[Complete Diagram (alt. view)](https://github.com/malcolmsdad/COMP444/blob/main/Excercises/U3%20-%20Arm%20Complete%20Assembly%20(alt%20view).png?raw=true)

[Claw Details](https://github.com/malcolmsdad/COMP444/blob/main/Excercises/U3%20-%20Claw%20-%20Entire.png?raw=true)

**Components, Joints and Motors:**

[Labelled Diagram](https://github.com/malcolmsdad/COMP444/blob/dev/Excercises/U3%20-%20Arm%20Complete%20Assembly%20(Labelled).png?raw=true)

Use this diagram to identify the letter below:

* A, G: Shoulder and Wrist Joint
  + 180 degrees rotational
  + Servo
* B, D, F: Shoulder, Elbow and Writs Joints
  + 90 degrees prismatic
  + Hydraulic pump
* C, E: Arm Link
* H: Slider prismatic joint
  + Slider causes clamps to open or close grip
* I: Cylinder link
  + A central cylinder a for claw prismatic joint, and a ring that slides along the cylinder (prismatic) moved by hydraulic pump
* J: Claw connecting link
  + Static link transfers prismatic to rotational
* K: Claw mount
  + 45 degree rotational
  + Moved by sliding action, hydraulic pump
* L: Claw link
* M: Elastic Band link
  + The bands are the only part that comes in contact with the egg, and provide extra grip through friction.

**Joints**:

[Claw - Rotational](https://github.com/malcolmsdad/COMP444/blob/dev/Excercises/U3%20-%20Claw%20-%20Rotational.png?raw=true)

[Elbow - Prismatic](https://github.com/malcolmsdad/COMP444/blob/dev/Excercises/U3%20-%20Elbow%20-%20Prismatic.png?raw=true)

[Ring - Prismatic](https://github.com/malcolmsdad/COMP444/blob/dev/Excercises/U3%20-%20Ring%20-%20Prismatic.png?raw=true)

[Ring - Rotational](https://github.com/malcolmsdad/COMP444/blob/dev/Excercises/U3%20-%20Ring%20-%20Rotational.png?raw=true)

[Shoulder - Prismatic and Rotational](https://github.com/malcolmsdad/COMP444/blob/dev/Excercises/U3%20-%20Shoulder%20-%20Prismatic%20and%20Rotational.png?raw=true)

[Wrist - Prismatic](https://github.com/malcolmsdad/COMP444/blob/dev/Excercises/U3%20-%20Wrist%20-%20Prismatic.png?raw=true)

[Wrist - Rotational](https://github.com/malcolmsdad/COMP444/blob/dev/Excercises/U3%20-%20Wrist%20-%20Rotational.png?raw=true)

**Degrees of Freedom:**

[**Calculation Worksheet**](https://github.com/malcolmsdad/COMP444/blob/dev/Excercises/U3%20-%20DOF%20Calculation.png?raw=true)

Total DOF: 8

Circuit: Elbow Circuit

October 17, 2024

11:34 PM

For my robots elbow, I decided to forgo the rotational joint, as the two other rotational motors (wrist and shoulder) should allow for enough range of motion and freedom of movement.

Reviewing the elbow again:

[Elbow - Prismatic](https://github.com/malcolmsdad/COMP444/blob/dev/Excercises/U3%20-%20Elbow%20-%20Prismatic.png?raw=true)

I've selected a hydraulic motor (piston) to affect upon the arms in relation to the shoulder. The primary action of upper arm and shoulder is to lift, meaning the motor selection must be very strong.

Hydraulic motors are very powerful and very reliable. They have limited range of motion, as the pivot points and piston impost limits. The elbow will allow rotation of 0 to 170 degrees.

Not displayed is the hydraulic pump, or hydraulic lines.

Goal: Write a program that inputs a number of degrees and moves bends the elbow that much.

Problem: Converting degrees (rotational) to distance.

<https://github.com/malcolmsdad/COMP444/blob/main/Excercises/U3%20-%20Elbow%20-%20Trigonometry.png?raw=true>

We want to be able to move the arm by specifying degrees "D", but since we're using a piston along "C" then we must calculate the length of C to move. Unfortunately, that also means that we need to know the values of A and B. I'll assign const values for those in the code.

A quick google, and I was able to find this calculation " Law of Cosines " to calculate the length:

c2 = b2 + a2 – 2ab cos γ

I used an example circuit where a servo was controlled by a potentiometer. I created a new class to represent a piston controller, it allows for reading current length of the piston, and a function to move the piston. The code reads the value of the potentiometer, converts to degrees, then using the trig function above, to calculate the desired length of the piston. The code then reads the current length of the piston, and move it by the difference of desired and current.

Here is the code here:

<https://github.com/malcolmsdad/COMP444/blob/fde853753d40ff637935de238e4d858a6c029d60/Unit%203/Instructor_Excercise____Elbow_Circuit/Instructor_Excercise____Elbow_Circuit.ino>

Chapter 6

October 17, 2024

8:12 PM

Chapter 6 is entitled Gasping at Straws / Manipulation. I suspect this will be a complicated one.

Immediately we're introduced to the term manipulator, and it being a effector. It can be any type of gripper, hand, arm or body part that is used to move things. Manipulators typically consisted of 1+ links connected by joints, and the endeffector. Endeffector is like the fingers on a hand - does the actual touching of objects in world.

Endeffectors relies on manipulator to move itself into position for the endeffector. In this case, manipulation involves entire body. Arm must move to make up for its own joint limit. We then learn that the endeffector is limited by the arm, that is limited by the joint limit and its free space.

Then we move to Teleoperation (introduced in Chapter 1) and the difference between teleoperation and remote control. We also learned about the challenge puppetering (a form of teleoperation), and its inherent difficulties: complexity, interface constraints and limitation of sensing.

Teleoperation has been used in surgery with great success. Finer, smaller controls, smaller incisions, reduced heal time and risk to patient. One suggestion to overcome some of these difficulties are exoskeletons, which are more of shells on top of a body, however they are not considered robots.

And now on to why manipulation…

First we learn about kinematics (correspondence between actuator motion and resulting effector motion), then joints (connections between links), as well as types of joints (rotary or prismatic).

Robotic manipulators have one or more type of joint, and each join typically provides on controllable DOF, requiring multiple actuators. Materic contrasts the many required components for a robotic arm vs a human arm with seven DOF. Additionally, ball and socket joints are very difficult to recreate in artificial systems.

Mimicking the appearance of a humanlike shapes is called anthropomorphic and is extremely complicated and associated manipulation problems.

From there we moved to defining what is required to control a robot manipulator is called kinematics, including describing manipulation problems (what is attached, how many joints are there, how many DOF) and how to accomplish to move the manipulator to accomplish the goal (inverse kinematics). Further to that are dynamics which describe of an object that has motion or energy.

Managing kinematics, dynamics, compliance within an environment and grasping an object are the reason that manipulation is so hard.

Note:

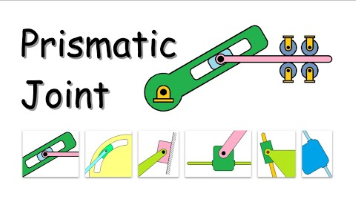
While doing the textbook exercises, I found that my understanding of Degrees of Freedom was lacking, especially while trying to calculate DOF. I re-read the textbook but unhappy with my comprehension so I watched this video:

[Understand prismatic joint and how it affects the degrees of freedom in a system](https://www.youtube.com/watch?v=oYx-nBuejTA&ab_channel=mecademie)

I found it very helpful, and I sure hope it's accurate, because I intend on using the algorithm:

DOF = 3xB - 2xJ

(B = # of Bodies, J = # of Joints)



Definition

October 17, 2024

8:13 PM

**Manipulator**: An affector. Any appendage that is used to affect and move objects in the environment.

**Manipulation**: A goal driven movement of any type of manipulator.

**Manipulator Links**: individual components of a manipulator that are controlled independently. Ie: upper arm and lower arm.

**Endeffector:**  The endeffector is the part of the manipulator that affects the world. Fingers on a hand, foot of the leg etc.

**Joint Limit:**  The extreme limit for how far a joint can move.

**Free Space:** The space in which a movement is possible

**Teleoperation:** Controlling a machine at a distance

**Remote Control:** Controller simple mechanisms

**Robot Assisted Surgery:** A surgeon remotely controls a robot that moves inside the body of the patient to cute and suture.

**Exoskeleton:** A structure that a human wears and controls. It provides additional strength, or mimics controls of a robot for teleoperation.

**Kinematics:** Correspondence between actuator motion and resulting effector motion

**Joints:** The various links of a manipulator are connected by joints.

**Rotary Joint:** Ball and socker join, providing rotational movement around a fixed axis.

**Prismatic Joint:** A piston: providing linear movement

**Anthropomorphic:** Human like shape

**Manipulation Problems:** Problems about where the endpoint is relative to the rest of the arm, and how to generate paths for the manipulator to follow in order to get the job done.

**Inverse Kinematics:** The conversion from Cartesian position to an endpoint and the angles of the whole manipulator

**Dynamics:** Properties of motions and energy of a moving object

**Grasp Points:** The point on an object where fingers need to grasp relative to the center of gravity, friction, obstacles etc.

**Compliance:** Yielding to the environment required for tasks with close contacts such as sliding along a surface.

Food for Thought

October 17, 2024

8:12 PM

**How many DOF are there in the human hand? Can you control each of them independently?**

I will attempt to identify each joint within the hand, describe and count their degrees of freedom.

Each finger and thumb has three joints:

1. Where digit is attached to hand has 2 DOF: up, down, left, right
2. Two knuckles on each digit with 1 DOF: up and down. All the joints within a digit allow for the entire digit to translate forward and back, adding another DOF

The total DOF of a human hand is 25 - although.

Not all DOF are independent however, simply due to the anatomy of a hand, some joints or ligaments can be tight or tied closely to another. For example, its extremely difficult if not impossible for me to make a fist, while leaving the ring finger extended.

**Which of the two joint types we discussed, rotational and prismatic, is more commonly found in biological bodies? Can you think of examples of specific animals?**

Rotational joint type is most commonly found in biological bodies, in the form of a ball and socket. Typically found in complex joints where large links are attached to bodies, such as shoulders, hips, knees. They're extremely difficult to reproduce in robots.

**Are astronaut suits exoskeletons? What about a lever-controlled backhoe?**

Yes, astronaut suits are exoskeletons. An exoskeleton, in bioloigcal systems is the form of a hard outer structure, such as s shell, that provides protection or support. This is exactly what an astronaut suit does. It provides heat, or cooling, oxygen to breath, communications with other astronauts and support team. Some space suits, when equipped, provide locomotion in the form of a thruster pack.

A back hoe is an interesting question. It does provide some protection, and ofcourse increases strength. It provides locomotion as well as comfort (such as AC or heat). Many backhoes have a feedback option, where you can feel the substrate that the bucket is cutting into, and they definitely have sensors, including modern GPS, temperature, etc. The only question that I have issue with is the part of the definition where the backhoe provides a hard outer surface. It definitely does that, but in robots the definition states they must 'wear'. I don't consider the operator of a backhoe wearing the machine, so no. I don't believe a backhoe is an exoskeleton. The machine Ripley used in Aliens is definitely an exoskeleton though!

**Imagine an exoskeleton which has its own sensors, makes its own decisions, and acts on them. This is definitely a robot, yet it is also controlled by a human and attached to the human’s body. Can you imagine where this could be useful?**

Absolutely, and I can provide an example of where that could possible come in handy. During early days of exploration, there was an intense race to achieving the hundreds of 'firsts' in space. The USSR beat the US to almost all of these achievements, however they did with much greater rush and risk. One such event was the very first space walk. Performed by Alexei Leonov, he successfully left the Voskhod 3KD space craft and entered empty space for approximately 12 minutes. The whole event was filmed and touted as a propaghanda win for the USSR. However, it was nearly a complete disaster, Leonov found that as soon as he left the capsule, his suit ballooned up much larger than the crafts door. He had to vent his suit to open space to be able to fit in, resulting in suffering the bends. Now imagine if something went wrong with Bruce McCandless during the first untethered space walk? He achieved untethered EVA by relying on the space suits backpack. Now imagine if a fault occurred that left him unconscious or disabled in another way. If that space suit had the ability to recognize that fact, it could return him safely to the shuttle. Additionally, I could see drone exoskeletons used to extract personnel from dangerous or war situations.

How about disabled individuals, exoskeletons could allow someone to walk, while the system manages all balancing efforts with internal sensors, decisions and actions.

**Robotic dogs have been used to play soccer. Because they are not designed for that task, they move slowly and have trouble aiming and kicking the ball (not to mention finding it, but that’s not part of manipulation; it belongs to sensing, discussed in the next few chapters). Some of the best robot soccer teams have come up with a clever idea: instead of having the dogs walk as they normally do, they lower them so that the two rear legs are driving and steering the robot, while the two front legs are on the ground, used to grasp the ball and kick it, which is much easier. This turns out to work very well, even if it looks rather silly. What are the endeffectors of the dogs in that case? Are the dogs now purely mobile robots?**

Endeffectors are the part of the manipulator that affects the world. In this case the front legs are endeffectors. Since the wheels do not interact with the ball, or anything else, and are more related to movement than manipulation, they are not endeffectors.

Yes, these are robots. They have a goal (to find the ball and kick the ball), they are aware of their environment, they exist, and can do so by itself. Bend it like the little bot!