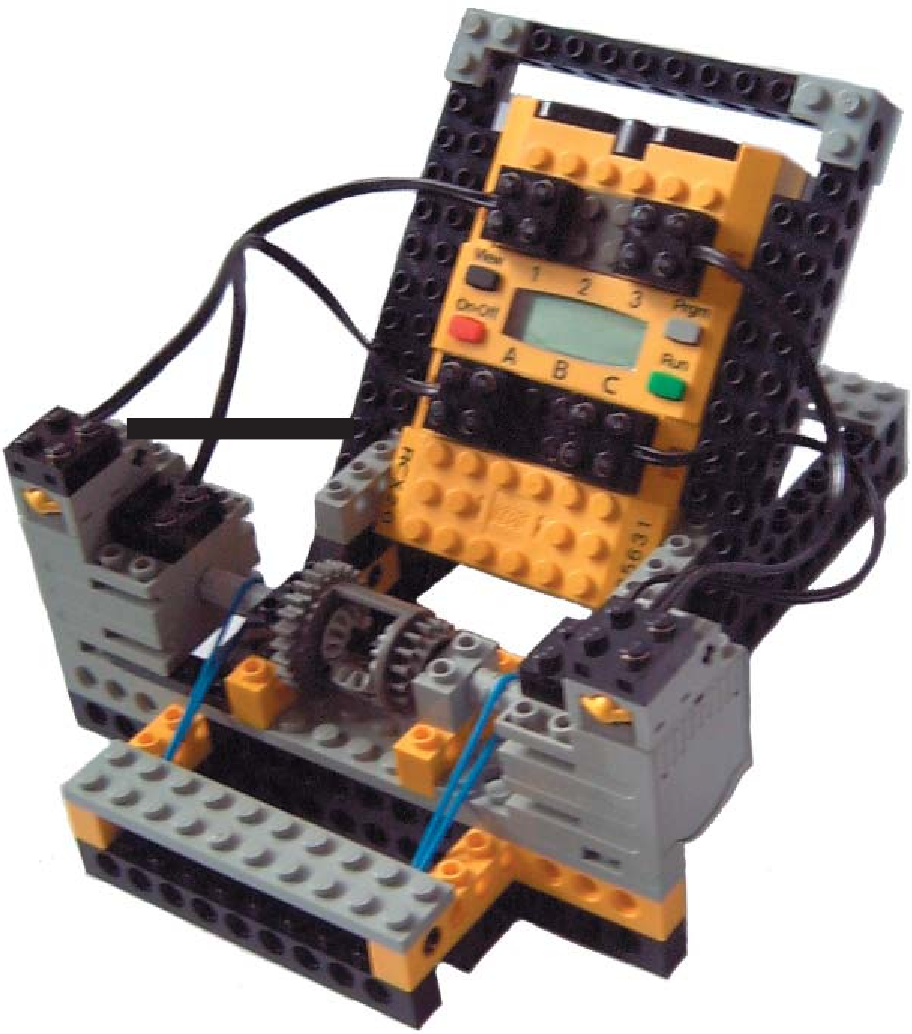
Testing the Lego
servomotors



Wouter Walmink
s020336
P0206

Introduction

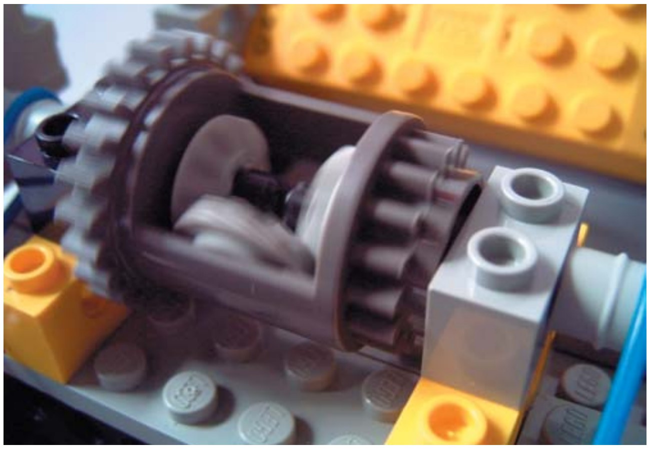


Fig. 1: the differential system to compare two motors

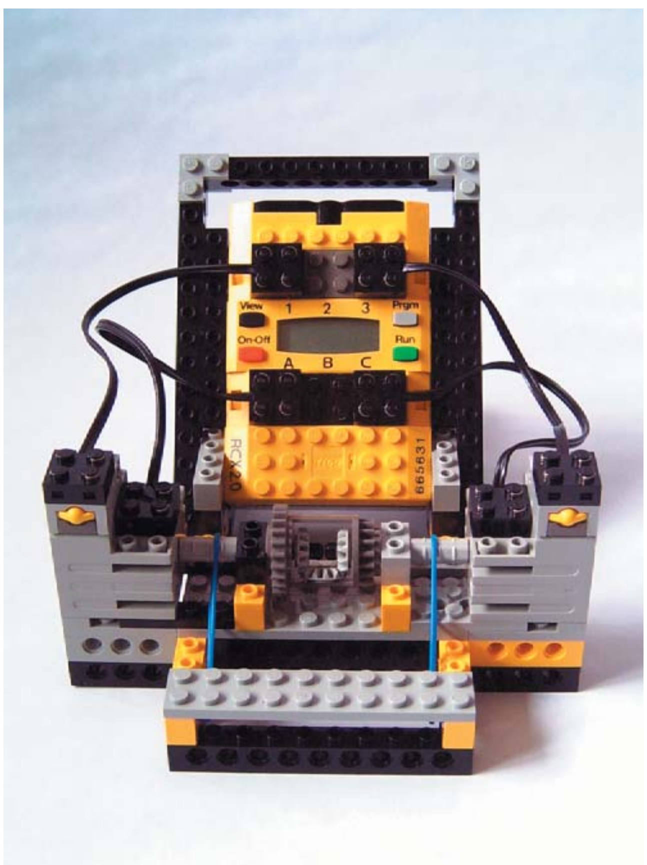


Fig. 2: Front view of the test situation

What do we want to test?

Joost Pluijms and I are working on a project called "Lego Java Butler". Our goals is to create a robot which can collect a can of coke and deliver it to one of two persons.

Our robot will probably drive around by two independently controlled Lego motors. One problem we face here is that these motors are not exactly the same, just as the transmission between the motors and the wheels. To limit deviation for the motors, I am going to test all three of them. The results will be useful to minimize the deviation in general.

How are we going to test this?

Our competency coach, Christoph Bartneck, helped me creating a construction that was fairly simple, but extremely useful for testing the relative power of the Lego motors. This contruction is based on a differential flavoured piece of technique: it is a combination of three tooth wheels and part that contains them and has a tooth wheel itself *(fig. 1)*.

Usually this system is used to power to wheels with different speeds, but with one transmission. However, we are going to connect two motors to both ends where normally the two wheels are located. These two motors will be powered independently and will spin in reversed direction of eachother. Due to the working of the tooth wheels, the holder of the tooth wheels will stay still when both motors rotate in exact opposite proportions. When the two motors are out of balance, the holder will spin around with a speed that is in proportion with the deviation resulting from the leak of balance.

The purpose of this investigation is not to find our the precise deviations in strength between the three motors. We want to know the rough differences between the motors and test if we can calibrate the motors to reduce deviation.

The software

The assignment

The test situation requested a program which could change the strength of the motors when the user instructed to do so and which was reflective: the program should display the current strengths of the motors. As I had to write the final program in Java, this was a good moment to get a real start with the language (I already read much about it, but practice was limited to some small tests). I structured the functions of the program and summarized the elements I would have to use *(Fig 3)*. Because it was a very linear program, the OOP (object orientated programming) was of little use *(Fig. 4)*.

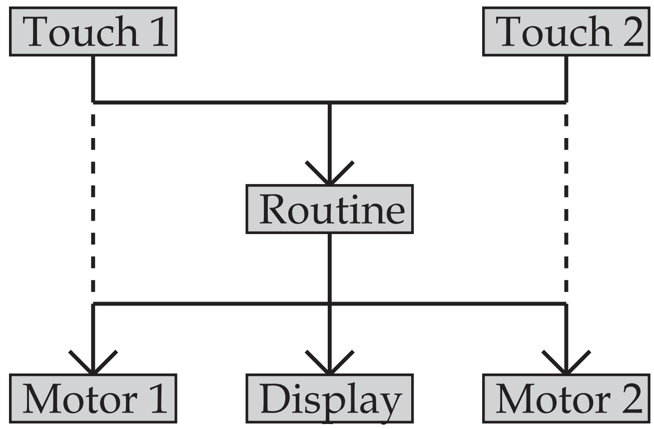


Fig 3: Diagram of the test program

```
import josx.platform.rcx.*;
class TestMotorSpeed
{
   public static void main(String[] args)
   {
      int power1 = 0;
      int power2 = 0;
      boolean sensor1on = false;
      boolean sensor2on = false;

      Sensor.S1.setTypeAndMode(0,0x20);
      Sensor.S3.setTypeAndMode(0,0x20);

      Motor.A.setPower(power1);
      Motor.C.setPower(power2);

      Motor.A.forward();
      Motor.C.forward();

      LCD.showNumber(power1*1000+power2);
      LCD.refresh();

      while(true)
      {
         sensor1on =
            Sensor.S1.readBooleanValue();
        sensor2on =
            Sensor.S3.readBooleanValue();
        if(sensor1on)
        {
           if(power1 == 7)
           {
              power1 = 0;
           }
           else
           {
              power1++;
           }
           Motor.A.setPower(power1);
           try
           {
              Thread.sleep(400);
           }
           catch(InterruptedException e)  {}
        }

        if(sensor2on)
        {
           if(power2 == 7)
           {
              power2 = 0;
           }
           else
           {
              power2++;
           }
           Motor.C.setPower(power2);
           try
           {
              Thread.sleep(400);
           }
           catch(InterruptedException e)  {}
        }

        if(sensor1on || sensor2on)
        {
LCD.showNumber(power1*1000+power2);
           LCD.refresh();
        }
      }
   }
}
```

Fig 4: The testing program, writen in Java

Test structure

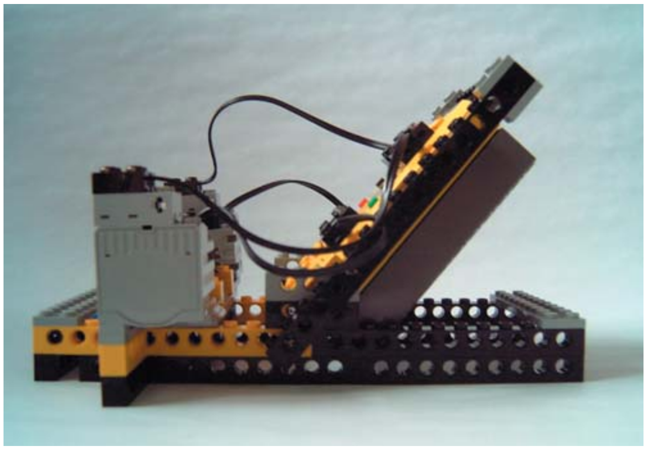


Fig. 5: a broad base to ensure stability

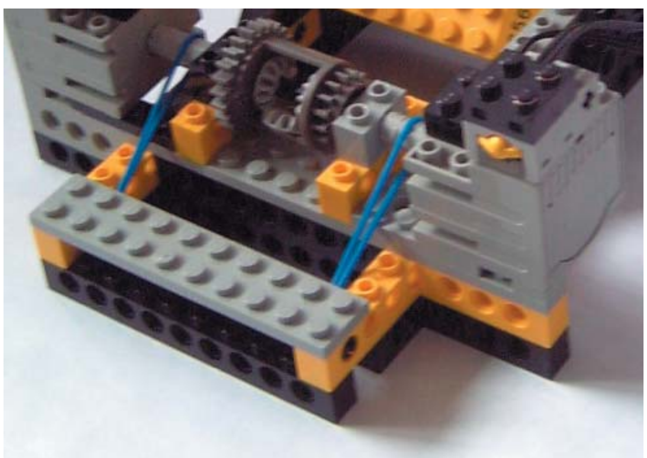


Fig. 6: Supporting parts to keep the motors horizontal

Tests to be done

I know (based on conclusions of a previous successful group who did this project) that only the higher power-levels of the motors would be used in the final robot, because this reduces the inaccuracy. Therefore I decided to test first of all the equal power-level four, five, six and seven.

I also knew (because of the deviation of our build cars) that the motors are not exactly similar. It would therefore also be interesting to test which effect differences in power would have. I choose the divergent power-levels 5-4, 6-5, 7-6, 6-4 and 7-5. This should provide enough information in draw dependable conclusions.

To support reliable results

Though our tests were not expected to result in highly precise measurements, I had to make sure that there was no unacceptable distortion of the results by factors that were not tested. The most crucial factor would in this case indefinitely be the friction of construction (i.e. tooth wheels, rotating bars. To make sure that the construction caused little friction, I made it, in cooperation with our coach, as stable as possible by adding a large base *(fig 5)* and extra supporting parts to keep everything in fixed in their right positions *(fig 6)*. The construction requires a flat surface.

To truly ensure reliable results

Of course the stabilization would make the results more reliable, but is it enough the make the friction of the construction not cancel out the differences between the motors? To be absolutely sure, I did some logical reasoning:

When the difference between the motors is the only factor that has effect on the test results, then the switching of the just the motors would cause the test results to invert, since they are the differential results of the two motors. (This is anyhow ensured for the equal power- levels, so these tests could be used as proof)

Test results

Testing with equal power-levels

This test was done to see how the motors function at equal power-levels and what effect the increasement of the power had on the deviation. Also it could be used as prove that the test was valid, if there was a horizontal symmetry in the chart show the outcomes.

The chart below *(fig 7)* shows that this global symmetry is preserved, because the x-axisfunctions as a symmetry axis. This means that the friction of the rest of the contruction did not previal the differences between the motors. Some points differ from the overall picture, but this is because each tested point was only done once.

Another thing that can be concluded from the chart is that the deviation is reduced when the equal power-level is higher. This makes sense if the deviation between the motors is more or less static. For example:

If Motor1 is slowed down by 1/5 of its power, then the relation between Motor1 and Motor2 is *Motor1*(4/5) : Motor2*. In this way, the deviation should be constant. However, when Motor1 has a constant delay, say one rotation per minute, then the relation will be like this: *(Motor1 - 1) : Motor 2*. When the values of Motor1 and Motor2 increase, the proportional difference between the motors will decrease.

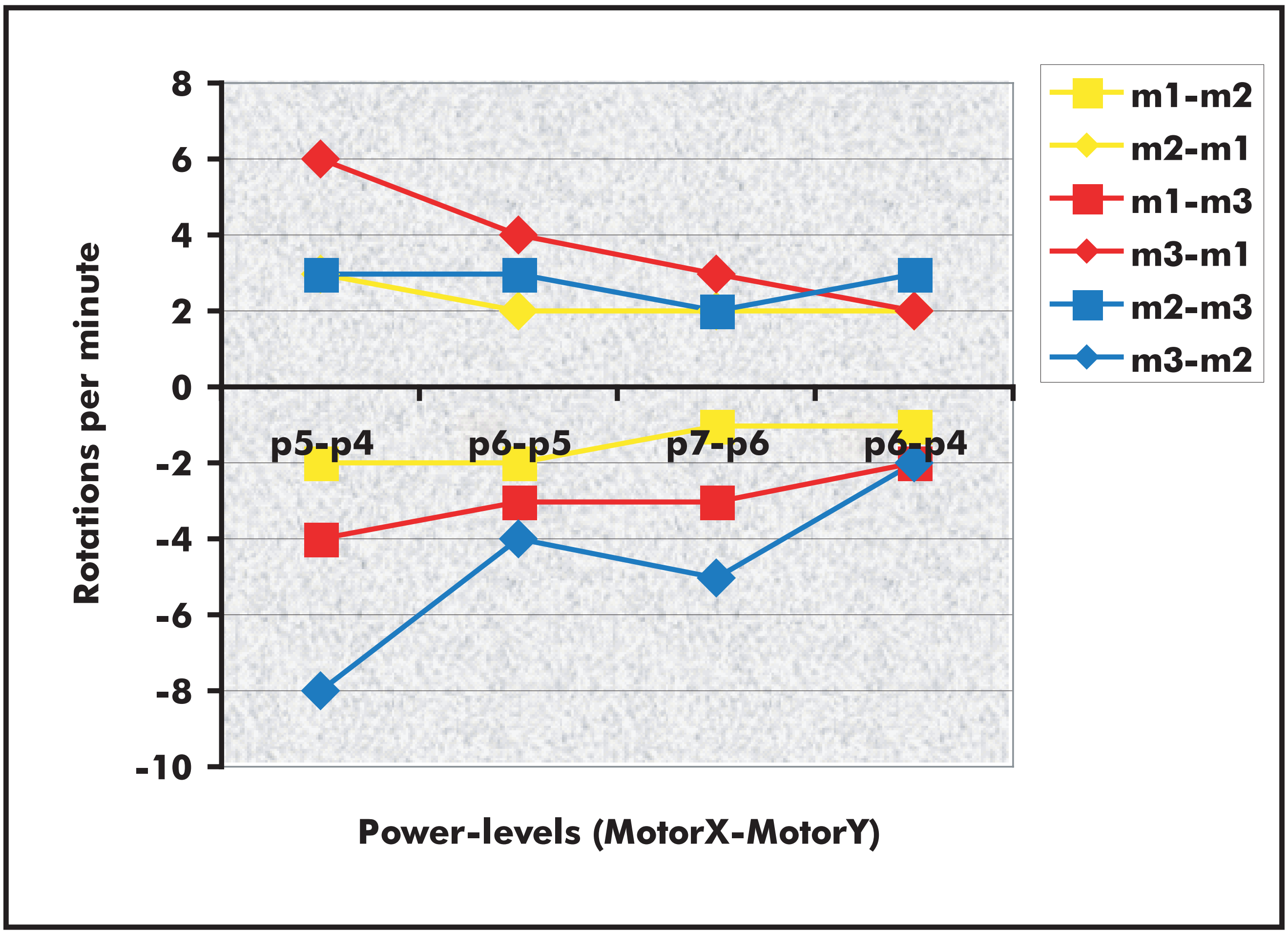


Fig 7: A chart showing the activity of the toothwheel holder for equal power-levels (in rotations per minute)

Test results

Testing with differing power-levels

This test was done to see the effect of different power-levels on the motors' deviation.

As the chart below *(fig. 8)* shows, there is no motor that is slower than another one when its power is one unit higher. This means that the figures in the chart show the extra rotations that the toothwheel holder makes, caused by the first motor named.

The relation between each colour pair of graphs is that the is that *higher graph - lower graph* results in the difference in power in between the two motors.

Conclusion

The results have proven to be valid and though the number of tests was restricted to one for each setting (nevertheless resulting in 2 hours of spinning tooth wheels), the global figures are quite consistent with eachother. The most important points:

* *(Strength: Motor1 < Motor3 < Motor2)*
* *(PowerLevel1 > PowerLevel2 => Deviation1 < Deviation2)*

This document will now be reference for me to the deviation between my three motors and might be used as a guide for those how wish to do test their own Lego servomotors.

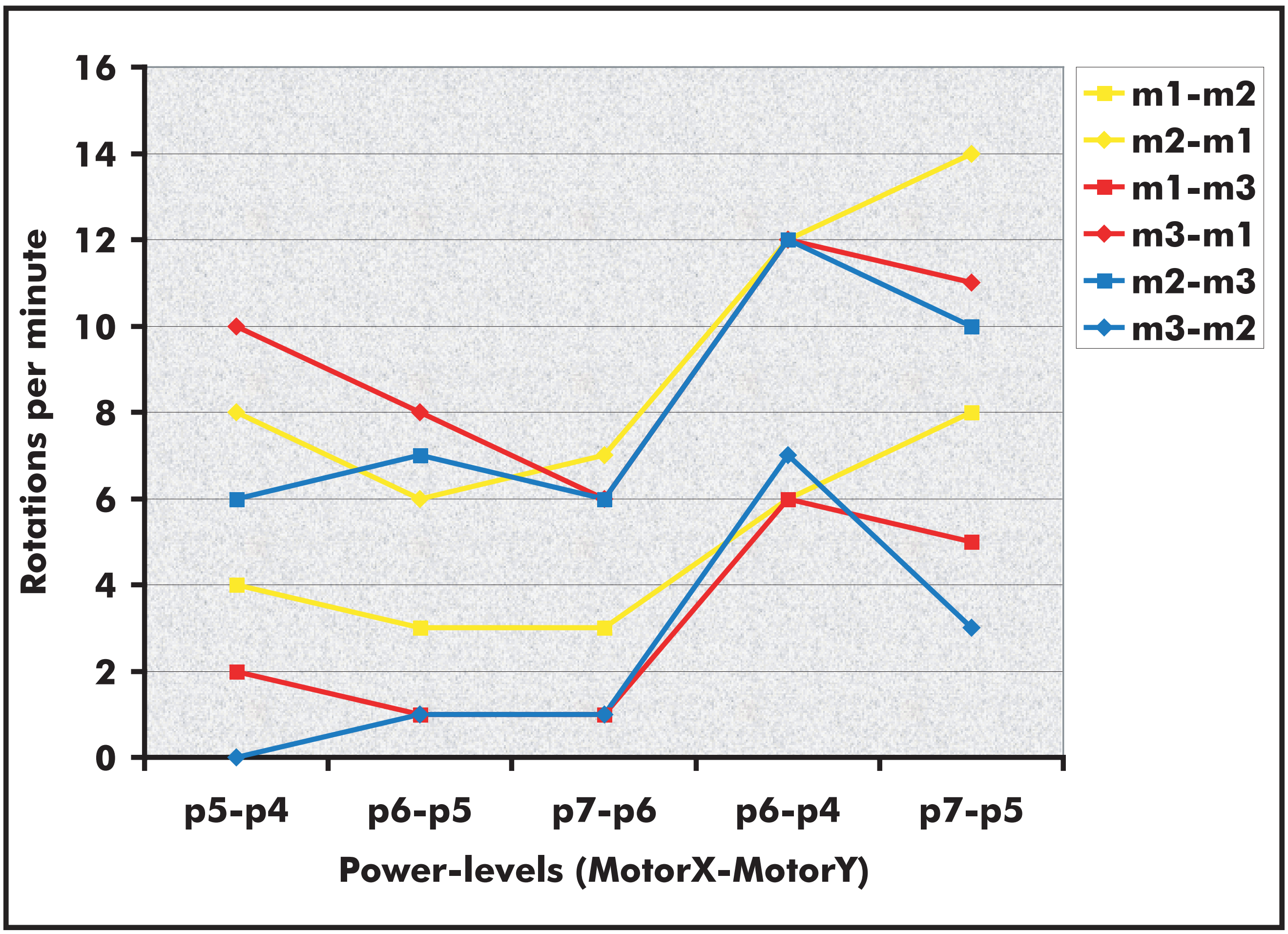


Fig 8: A chart showing the activity of the toothwheel holder for differing power-levels (in rotations per minute)