Using GPS for coordination of robots

Hubert Piontek

October 15, 2003

Contents

1 Goal

The main goal of our experiments is to find out whether GPS is a suitable means to coordinate autonomous robots with a size of about 30cm on a "virtual circuit" being about 10m in diameter.

2 Equipment

We use RFB 21 GPS receivers from Royaltek. They can provide their output through a serial line using the NMEA standard. The rate of messages, as well as the bit rate on the serial line can be set using special command messages. For all of our experiments, the serial line was set to "9600,8,n,1"[1], and we use a message rate of "one dataset per second"[2]. The receivers can keep track of 12 satellites at a time.

For our experiments, we use normal PCs running Linux and Windows, and some home–grown software for evaluating the receiver output. Most

[1] 9600 bits per second, 8 bits per character, no parity, 1 stop bit

[2] A complete dataset consists of multiple messages.

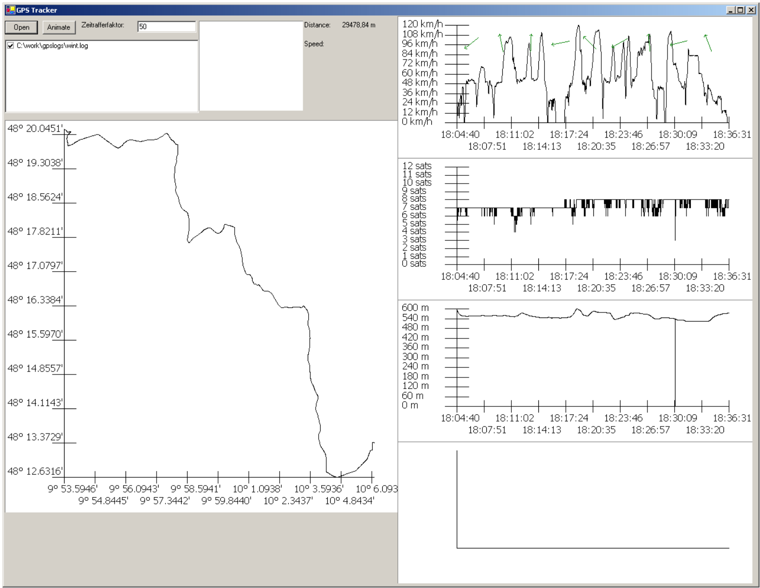

Figure 1: Program for plotting GPS data

importantly, we wrote a program that plots tracks "recorded"[3] previously in the NMEA format (see figure 1).

Besides some controls for operating, the most important parts are the 5 graphs. Below the "Open" button, there is a list of currently used log files. The ones that are checked, are plotted. The graph on the left shows a 2D plot of the track[4]. The top graph on the right side shows the speed over time, and a few direction indications[5]. Below, the number of visible satellites is plotted. The third graph shows the altitude. The last one is in action only when exactly two tracks are selected/plotted. It then shows the relative distance between the two tracks in meters over time.

3 Experiments

For all experiments, the receivers did not have to make a cold start in the beginning.

3.1 Difference between two GPS receivers

Setup: Two GPS receivers mounted on a car right next to each other.

[3]Well, logged to a file...
[4]Western/Southern coordinates are < 0
[5]up would be north

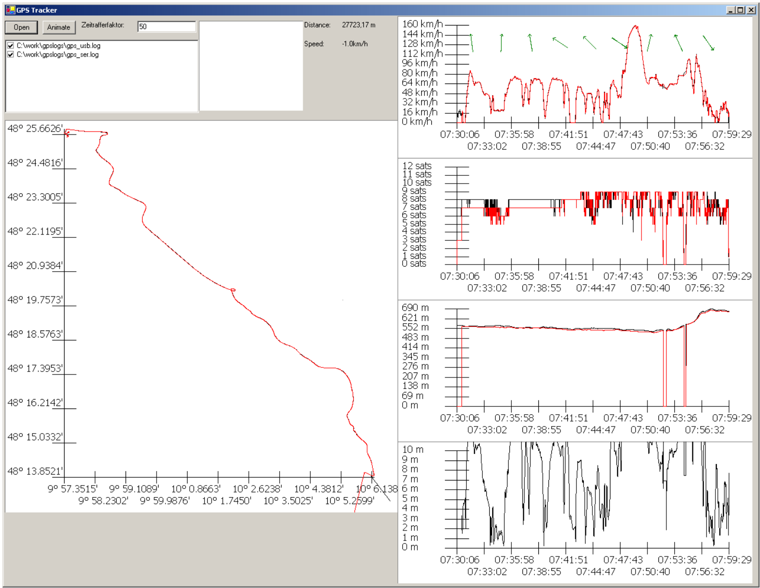

Figure 2: Two GPS receivers, driving

This experiment was done to find relative precision, and get an idea of the absolute errors of GPS, as well as getting an idea of how good GPS behaves in one of its (civil) main usages.

Besides seeing my every day drive to work in figure 2, one can see, that the two receivers match pretty good, if one looks at it with the scale of the whole way (plot of the track) for the scale of a car. Even though the two antennas were a mere 5cm apart, there is a rather large relative error that can be seen in the graph for relative distance.

Of great interest is the fact, that one of the receivers saw an extra satellite for quite some time; recall that both antennas were mounted right next to each other. I cannot give any explanation for this.

Most commercial navigation systems have features like "snap-to-road", and also include extra readings from the car's body, like speed, which becomes understandable looking at the magnitude of the observed errors.

3.2 GPS position stability over time

Setup: A single GPS receiver sitting in one spot.

In order to find out the precision over time, and the jitter in position, we mounted one GPS receiver on the balcony in front of our office, and logged its data for about 5 hours (see figure 3). The weather was cloudy with some rain showers in between. Under these conditions, the number of visible satellites decreases, probably enhancing the experienced error. The plot of

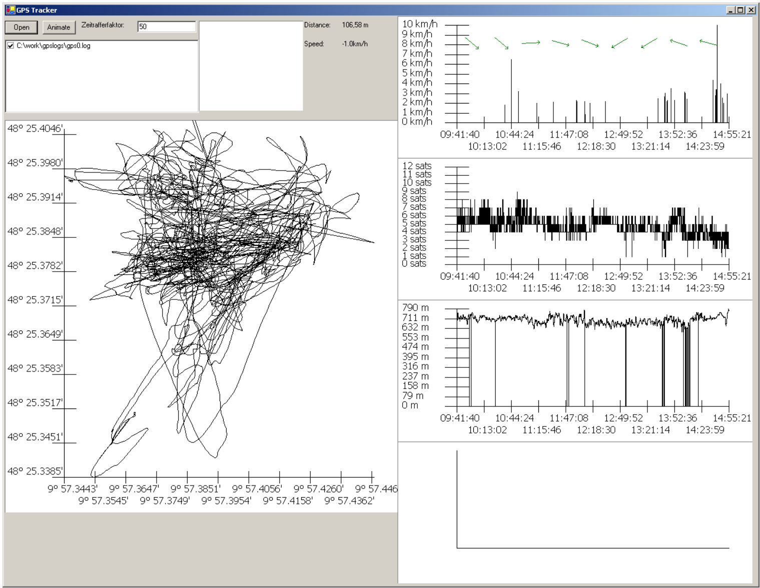

Figure 3: One GPS receiver, sitting still

the "track" would ideally be just a single spot, but as the plot shows, the track covers an area being about 125 by 122 meters. So much for precision information from datasheets…

It is very interesting to note, that even though the position readings change constantly, the GPS receivers constantly shows a speed of 0, only interrupted by a few "peaks of speed", i. e. the receiver actually showed a speed > 0 a few times. Comparing[6] at the GPS speed output and the car's speedometer while accelerating in a car from 0 to 100, one notices, that the GPS speed output lags behind about 2 seconds[7]. Both observations suggest, that the GPS receiver already does some internal filtering before calculating the speed.

What we can learn from this data (especially, if one looks at an animation of this data), is, that long term stability cannot be guaranteed, that is, if one takes a previously recorded path[8], and tries to have a robot follow it much later (e. g. a day). The error in measurements makes it useless. If e. g. the robot records a circle on a parking lot, and wants to do the same circle the next day, the robot would do the circle on the grass next to the parking lot. It is important to notice, that this behaviour is not a "must

[6] Well, having somebody else compare…

[7] Putting the car in cruise control and checking the GPS over some time, I now know the error of what the car tells me, and this is exactly 2km/h on a range from 30km/h to 190km/h, so one can really take a good guess about the lagging behind of the GPS output.

[8] A set of GPS coordinates

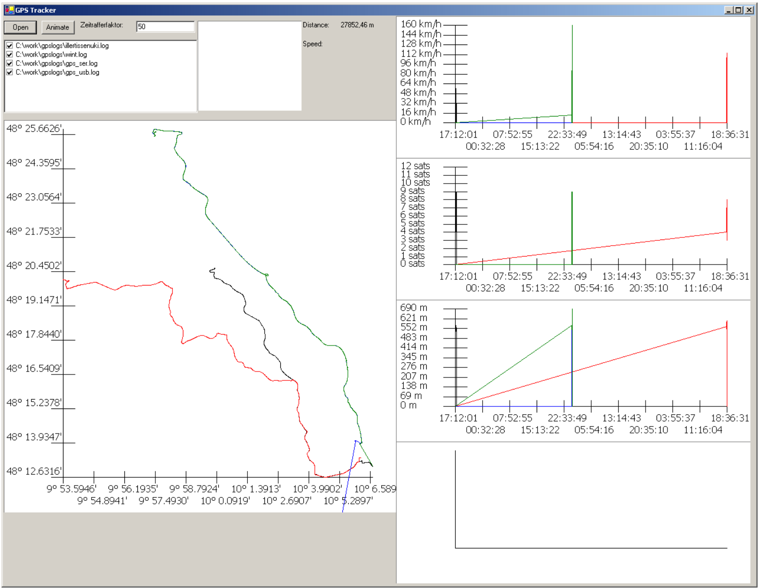

Figure 4: Errors over time

occur", but a "can occur".

A further indication of a large deviation of position can be seen looking at the starting points of the four tracks in figure 4. I park my car on a private parking space, so its position changes by less than 5m from day to day[9]. But now compare the starting points of the four tracks. While the black and the green match pretty good, the blue and the red one are off by several hundred meters. As a hint, the graph shows an area of about 16 by 24 kilometers. This data was collected on three different days[10].

3.3 Relative position of two GPS receivers over time

Setup: Two GPS receivers, distance 4m, no movement.

For this experiment, both receivers were set up on my balcony exactly 4 meters apart. This time, the weather was fine, and the "tracks" "only" cover an area of about 124 by 44 meters, and removing only the most extreme peaks, much less. Also interesting, the deviation in visible satellites between the two (identical) receivers. The worst part, however, is the relative distance between the two tracks over time: for too much time, it is not equal to the 4 meters they were placed apart, and the graph on the lower right certainly does not look like the straight line at 4m that would be ideal.

[9]There simply isn't more room.

[10]The blue and green ones are from the experiment with both receivers next to each other

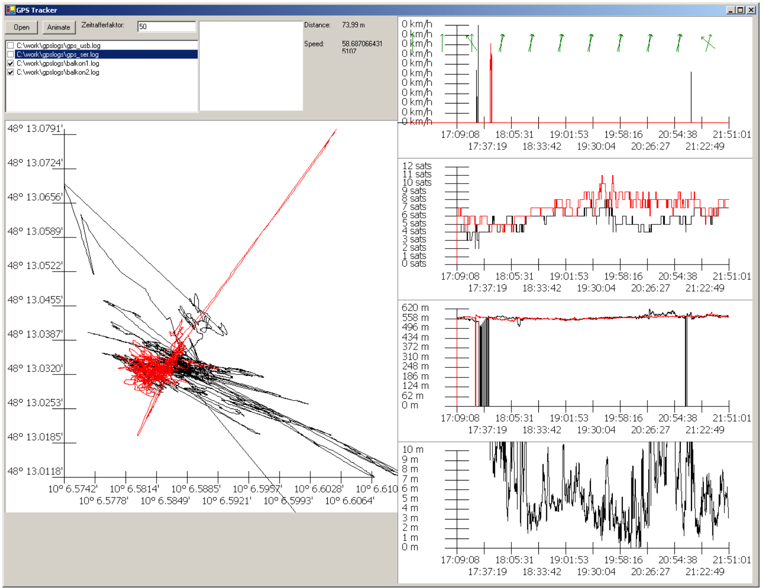

Figure 5: Two GPS receivers, sitting still

4 A basic algorithm for following a virtual circuit

```
start_driving_straight();
while( not_at_end_of_waypoint_list ) {
        get_gps_coords_to_go_to_from_list();
        get_curr_gps_reading();
        calculate_deviation_to_destination();
        if( deviation < threshold ) {
                destination_reached();
                remove_dest_from_waypoint_list();
        }
        calculate_heading_to_go();
        adjust_heading();
}
```

Of course, there must be an emergency stop whenever an obstacle ist detected.

5 Conclusion

Looking at the results from our GPS experiments, it's quite obvious, that this algorithm will not work well enough. GPS alone will not suffice to coordinate our robots in rather small areas. It's fine to destinguish whether the robot is on the university campus, or in downtown Ulm, but probably not even good enough to drive the robot from one northern entrance of our building to the southern one.

We are working on improving this by filtering the incoming GPS data, and fusing this data with odometry and motor control, e. g. if the GPS tells us, that our position just changed about 100m, and our speed was/is 0, and the motors are switched off, then we know that this GPS reading is rather useless.