

2. Oscillation

The oscillation process is the first basic behavior which is needed for the cooperative height control. Everything starts with a defined take off phase. In this phase the copter will start flying and after it leaves the ground it will fly upwards to 1 meter. After it reaches this height and stays there for 10 seconds, it will switch to the search phase.

In the search phase the copter will automatically oscillate between the upper and the lower search boundary as shown in figure 1. Since we use an infrared sensor for measuring the height we get only correct values for the height between 0.4 and 0.9 meters. That is the only reason why we have chosen this values for the boundaries.

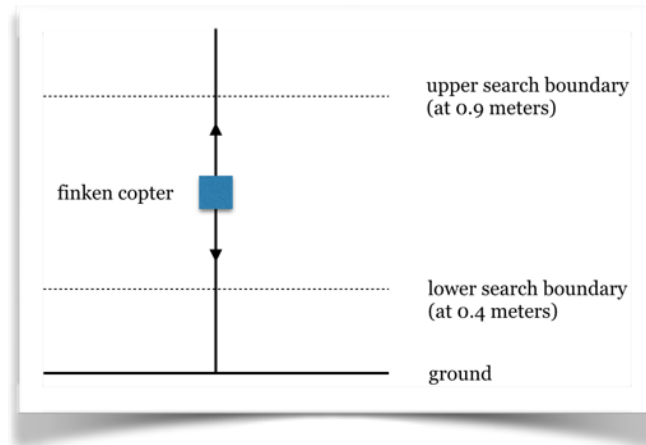


Figure 1: Oscillation process for one copter.
(Own illustration.)

The function in which we have implemented the oscillation process is a periodic one. Therefore we have to care about how often the height increases or decreases. The problem is solved using the stage time. The stage time increases by one every second and so we change the height also every second. The „search_neighbor“ variable indicates whether to oscillate or not. One step deeper the decision of moving upwards or downwards is made by the variable „go_down“. Finally if the upper or lower boundary is not reached we need to change the height by the variable „height_changing_rate“ which is manually defined to 0.05 meters. If one boundary is reached the direction of searching is changed to the opposite.

In figure 2 you can see the code snippet where the oscillation process is programmed. The variable declarations and definitions are excluded since they are easy.

```

if ( (finken_oscillating_last_time + 1) <= stage_time ){
    finken_oscillating_last_time = stage_time;
    if( search_neighbor ){
        if ( go_down ){
            if ( finken_system_set_point.distance_z > height_oscillating_down ){
                finken_system_set_point.distance_z -= height_changing_rate;
            } else {
                finken_system_set_point.distance_z = height_oscillating_down;
                go_down = false;
            }
        } else {
            if ( finken_system_set_point.distance_z < height_oscillating_up ){
                finken_system_set_point.distance_z += height_changing_rate;
            } else {
                finken_system_set_point.distance_z = height_oscillating_up;
                go_down = true;
            }
        }
    }
}

```

Figure 2: Code of the oscillation process.
(Own illustration.)

The only problem that occurs during this procedure was the infrared sensor. It measures only correct values in a really short range (between 0.4 and 0.9 meters). Fortunately it is enough to get a useful oscillation process. Thus the copters have a smaller searching area so they should converge faster but in future the sensors should be replaced by better ones.