NaviTop

Pypilot*

*** Open-source boat autopilot created by Sean D'Epagnier**

Profiles, Gain Adjustment, and Motor Controller Parameters

Summary

0. Preamble

Pypilot is a fantastic pilot that satisfies many users around the world. The various pre-installed settings profiles allow you to quickly use the autopilot with most actuators and boats. I have equipped sailboats of all sizes up to a tall ship with 1024m² of sail. Few skippers change the gains if their rudder actuator is not particularly slow. This manual is intended to show you how to use the preinstalled profiles. It will allow you to create your own if you are a demanding sailor wishing to improve the maneuverability of the autopilot response and perhaps further reduce its power consumption. The gain settings of the different pre-installed profiles change at most in a ratio of 1 to 3 between the different profiles. It is not recommended to modify these gains more significantly, especially without knowing their contribution to staying on course. Experimenting without making small changes may result in unstable driver operation until you return to the original or pre-installed setting.

1. Introduction to Profiles (Parameters and Gains)

With Pypilot, you can create, modify, and delete different profiles to quickly change parameters and gains depending on sailing conditions. Gains, as well as many actuator and tack parameters, can have different values depending on the profile. The advantage is that if you tune the boat for specific conditions, you can create a profile and then reuse it when sailing in identical conditions. Storing different profiles with parameters for different conditions improves efficiency, energy consumption, and even noise.

For example, in light winds, it's beneficial to reduce servo.speed_max to less than 100% to reduce motor noise. However, the efficiency of some motors drops below a minimum speed, which then increases power consumption. Therefore, you need to find a compromise that allows for quiet yet efficient actuator operation. In rough weather, it's advisable to ensure that servo.speed_min and servo.speed_max are set to 100% for optimal handling and energy efficiency.

Similarly, when sailing upwind, you can reduce the gains, especially the P gain, because many boats naturally balance themselves. This is because as the force exerted by the rigging on the hull increases, the boat tends to luff with the heel. However, when the boat luffs too much, the sails begin to "bounce," reducing their force on the hull, which causes the boat to bear away. For this reason, many boats can have the rudder locked upwind and maintain a correct course while oscillating around the mean heading. The autopilot is generally beneficial, especially in waves, to minimize and dampen this oscillating effect and improve overall speed. Therefore, it's wise to design a close-hauled profile with a higher D gain and a lower P gain.

2. How to find out the profile used by the pilot

The active profile is displayed in the bottom left corner of the calculator's LCD screen with the capital letter P, for "profile," followed by the first two letters of the profile name, for example, "Pde" for the "default" profile. The letter in the box above indicates the mode (Compass, etc.).

3. Gain adjustments

The basic autopilot uses an enhanced PID filter to create a control loop. Various gains can be adjusted to improve performance depending on the boat, sea state, and type of rudder actuator.

- P - proportional – action proportional to the heading error
- I - integral – action based on the cumulative error
- D – derivative – action proportional to the boat's rotation speed (rate of turn)
- DD – derivative of the derivative of the rate of turn, i.e., action proportional to the acceleration or deceleration of the boat's rotation speed
- PR - proportional root – action proportional to the square root of the heading error
- FF – anticipated gain – to adjust the boat's rotation speed during changes in heading setpoint

It is recommended to use the opencpn plugin or pypilot_control to adjust the gains, as a bar graph displays the feedback of each gain in green or red depending on the direction of action. A gain whose feedback is saturated, exceeding 100%, is indicated in blue.

To start from scratch (or on a new boat), set all gains to zero, except for the P and D gains. It is possible to have a usable autopilot with only these two gains, although it is less efficient.

Set the P gain to a low value (for example, 0.003) and the D gain to 0.01. Typically, on larger boats, you will need higher values, but this is usually because they have slower rudder actuators. It really depends on the speed at which the actuator turns the rudder and the sea conditions. The "hard over time" is the time it takes to turn the rudder from lock to lock, usually 30 degrees in either direction. If a smaller engine has a higher gear ratio and it takes, for example, 16 seconds to correct course, then these gains should be doubled to P .006 and D .02 as the starting setting.

If the boat takes too long to correct its heading and spends a lot of time on one side of the correct heading, increase these two gains. If the engine is working too hard and often overshoots the correct heading, decrease these gains.

- **P** - Proportional Gain - This value should normally be set low. If it is too high, the boat will constantly turn on either side of the desired heading. If it is too low, the boat may not reach the target heading. Using this gain alone will not provide a stable course. As it is increased, a higher D gain is needed to compensate (prevent overshoot).
- **I** - Integral Gain - This gain does not need to be used to maintain a heading, but it can compensate if the actual heading is different from the desired heading. If you are following routes and the boat tends to drift parallel to the route, this will compensate for that error. It's best to start at zero and gradually increase it until the results improve. If the value is too high, it will simply increase power consumption. Most users can use a value of 0 (off) with good results.
- **D** - Derivative Gain - This is the gain using the rotation speed determined by the gyroscopics. This is the primary gain used by the autopilot in rough seas. Most corrections should result from this gain. Once the optimal value is found, it can generally work in a wide range of conditions; however, in light air, it can be reduced (along with reducing other gains) to significantly decrease power consumption, especially if the boat is well-balanced.
- **DD** - Derivative Gain' - This gain is useful for improving reaction time and reducing oscillations. It can allow for corrections to occur sooner than would from gain D alone. Try gradually increasing this value, without changing the other gains and without exceeding 1.5 times the gain D value.
- **PR** - Root Proportional Gain - This gain can be really useful for avoiding oscillation, especially upwind. To use it, increase it until it takes effect, and gradually decrease the gain P. You will still need a gain P, but it can be halved if a sufficient PR gain is used. The 1/2 derivative of a linear function is actually 2x squared. So, in a sense, this gain mimics the PID fractional derivative filter by providing a useful fractional component to mix into the correction. It helps produce a damped response because it is often not good to scale the feedback linearly with the heading error, as this can produce overshoot. I generally set this gain to 1-2x the P gain, which allows for a lower P gain and reduces overshoot.
- **FF** - Anticipatory Gain - This gain is only used when changing the heading setpoint. It is not used for maintaining a heading. Following a course may involve course changes, and this gain **will**

then be used. The FF gain can be very useful for improving response time because a low P value is desirable for maintaining a heading. Significant heading changes can take a long time without the FF gain. This gain makes the main contribution when the setpoint is changed.

4. Tips for adjusting gains

- **Upwind:** less gain from D, more gain from P (or PR)
- **Downwind**: more gain from D, and possibly add a gain from DD
- **Light wind:** less gain - save energy
- **Strong wind:** more gain - more gain needed for proper operation

In flat seas, sailing a less straight course is a tuning error that will only increase power consumption.

In rough seas, tolerating a less straight course can decrease power consumption if you simply want to maintain the desired average heading with the sails sheeted in. This has always been the goal with a windvane self-steering system. It also helps reduce wear on the actuator motor.

5. Explanation of Gain Values

Using a P gain value of 0.003 is equivalent to moving the rudder actuator 0.3% of the time on average for each degree of heading error. So if the boat has a 10 degree off course, it would move the actuator 3% of the time on average. As the actuator moves in bursts, it will make a slight correction for 0.3 seconds every 10 seconds. We see that the correction therefore depends on the speed of the actuator. Except with a very slow actuator, P should not be increased significantly otherwise it will cause instability. The P gain typically contributes a small fraction of the overall correction, but combined with the PR gain (usually a bit higher), the effect is sufficient to maintain the course. By comparison, a D gain of 0.06 turns the rudder at full speed 6% of the time on average for each degree/second of boat rotation. So, if the boat is rotating 5 degrees per second, this could translate to the rudder moving at full speed 30% of the time (depending on the speed servo limits). The other earnings are calculated in the same way. The other gains are calculated in the same way.

6. Motor Controller Parameters

6.1 Global Motor Controller Parameters for All Profiles

Changing one of these global settings applies to all profiles.

servo.max_current

The first parameter to adjust is servo.max_current. This setting adjusts the controller's current limit, which is used for end-of-travel detection with most actuators. If the current limit is set too low, "OVERCURRENT_FAULT" error messages will appear constantly and block actuator movement. If the current value is too high, the controller may continue to supply power even with the actuator at its limit. For bar actuators, a setting of 4 to 7 amps is generally correct, and for hydraulic or high-power drives, higher values such as 15 to 20 amps should be used.

servo.slew_speed and servo.slew_slow

These essentially determine the maximum acceleration/deceleration of the actuator. Limiting acceleration or deceleration results in smoother movement with fewer current spikes, allowing you to lower the servo.max_current setting. However, if the acceleration/deceleration setting is too low, the response time will be limited. Typically, values between 15 and 30 should be used.

Servo.clutch_pwm

This parameter, accessible via "Configuration," "Pypilot Client," limits the clutch solenoid current once it is engaged. After 200-300ms, the current sent to the solenoid is pulse-width modulated (PWM) according to this adjustable coefficient, which can be set from 0-100%. In the case of an LS solenoid valve, the current can be reduced by a factor of 6.5 according to the manufacturer's documentation, which corresponds to a 16% reduction. This reduces the solenoid valve's power consumption from 30Ah per day to 5Ah.

5.2 Profile-Specific Motor Controller Parameters

These controller parameters have values for each profile. Changing them only affects the value in the currently active profile.

servo.period

This setting is the minimum time the motor can run or stop. Essentially, it prevents the motor from making too many short movements. Short movements can provide faster corrections and better steering, but at the expense of wear and power consumption. Typical values of 0.3 or 0.4 work well for most boats in most conditions, but larger/slower boats or lighter conditions can use higher values, and smaller/faster boats or harsher conditions can accommodate lower values.

servo.speed.min

This setting limits the minimum motor speed. This is useful because many electric actuators become less efficient if the motor spins too slowly due to excessive friction. For this reason, it's generally best to make short, quick corrections and spend the rest of the time with the rudder stationary. Setting a minimum speed allows you to either move at that speed or remain at a standstill.

servo.speed.max

This limits the maximum motor speed. In lighter conditions, slowing the rudder movement can be helpful to reduce noise or wear. Usually, the faster the drive motor, the better the performance, but if conditions are good, you can simply limit the drive speed by reducing this value.

solenoidservo.use_brake

When the actuator brake is engaged, the motor controller shorts the motor when the motor is not spinning and the autopilot is engaged. This can prevent the rudder from straining the motor in strong winds. The brake is particularly useful on monohulls sailing upwind with rudders on the keel or sternpost, which are not compensated. For other users, it is preferable not to have a brake because it allows you to quickly engage the actuator, without disengaging the autopilot or transmission, if you need to avoid an obstacle. The brake is also useful with high-speed electric actuators so that the position sensor stops the rudder more quickly before the mechanical stops.

7. Pre-installed Profiles in the Calculators

The pre-installed profiles in the calculators are listed in the table below, along with a profile by Yves, who sails extensively with Pypilot on a 42-footer.

Note: Servo.max-slew.slow and speed may have been different in some pre-installations

	Pre-installed profiles in Navitop computers					
Source	**Sean d'Epagnier**				**Stellian**	**Yves**
Profile	**default**	**light**	**downwind**	**upwind**	**stellian**	
Pilot	basic	basic	basic	basic	basic	basic
Gain P	0.0027	0.0056	0.0033	0.0055	0.0023	0.0026
Gain I	0	0	0	0	0	0.0288
Gain D	0.062	0.0431	0.0715	0.0473	0.1169	0.0888
Gain DD	0.0532	0.0288	0.0666	0.0288	0.1036	0.096
Gain PR	0.0048	0.0048	0.0048	0.0048	0.0051	0.0075
Gain FF	0.333	0.333	0.333	0.333	0.5	0.21
*Servo.max_current**	*6.37 A***	*6.37 A***	*6.37 A***	*6.37 A***	*6.37 A***	*4.50 A*
*Servo.max_slew_slow**	*26.378*	*26.378*	*26.378*	*26.378*	*26.378*	*18.1*
*Servo.max_slew_speed**	*19.0394*	*19.0394*	*19.0394*	*19.0394*	*19.0394*	*31.9*
truewind.offset	0	0	0	0	0	0
wind.offset	0	0	0	0	0	0
ap.tack.angle	100 deg	100 deg	100 deg	100 deg	100 deg	100 deg
ap.tack.delay	0 s	0 s	0 s	0 s	0 s	0 s
ap.tack.rate	5 deg/s	5 deg/s	5 deg/s	5 deg/s	20 deg/s	15 deg/s
ap.tack.threshold	55%	51%	49%	49%	50%	50%
servo.period	0.4 sec	0.4 sec	0.4 sec	0.3 sec	0.4 sec	0.1 sec
servo.speed.max	100%	100%	100%	100%	100%	90%
servo.speed.min	100%	100%	100%	100%	91%	47%

* These three parameter settings are common to all profiles

** The Servo.max_current setting must be adjusted according to the actuator