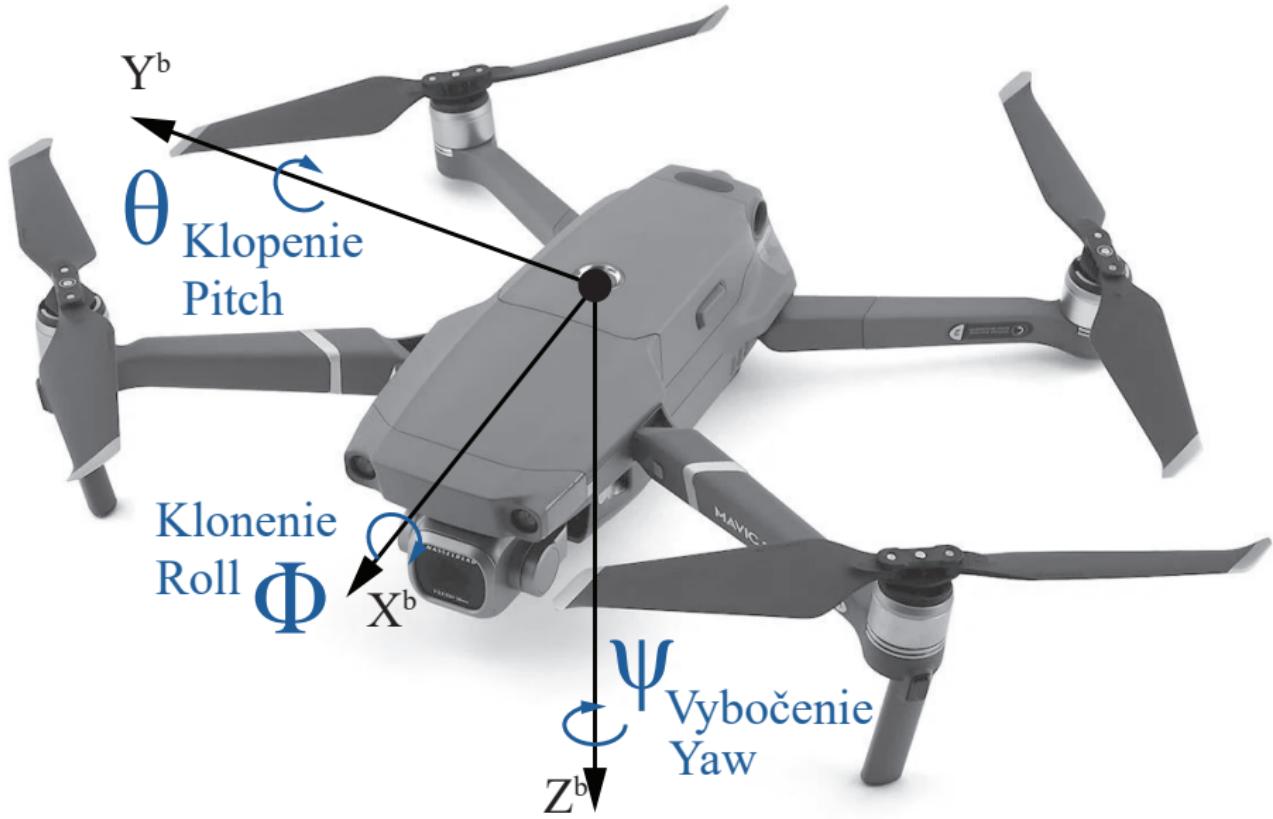


Working title drone control

Riadenie dronov

prof. Ing. Gergely Takács, PhD.

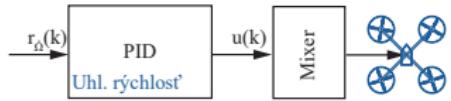




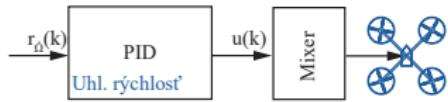
DCM -> Direction Cosine Matrix (pre 3.2) Transformation to inertial to body axes. Quaternion 4 by 1

- Najnižšia úroveň je riadenie orientácie, tj. uhly $\Omega = \Phi, \Theta, \Sigma$ a uhlové rýchlosťi $\dot{\Omega}$ v "body frame"

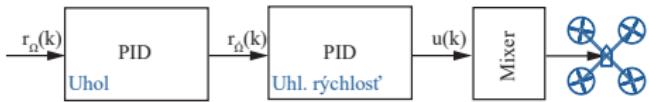
- Riadiť orientáciu (*angl.*: attitude) len na základe zmeny uhl. rýchlosťi (*angl.*: rate) by bolo dosť neintuitívne, potrebujeme prepočítať $r_\Theta \rightarrow r_{\dot{\Theta}}$
- Tvárme sa, že poznáme žiadane orientácie, napr. klopenie r_Θ a chceme riadiť y_Θ . V skutočnosti riešené s kvaterniónmy aby sme obišli singularitu Eulerových uhlov pri odhade [Erasmus 2020].
- Riadenie orientácie môže byť riešené ďalšou, nadradenou regulačnou slučkou - hovoríme o tzv. kaskádom riadení (*angl.*: nested, cascaded).



- Riadiť orientáciu (*angl.*: attitude) len na základe zmeny uhl. rýchlosťi (*angl.*: rate) by bolo dosť neintuitívne, potrebujeme prepočítať $r_\Theta \rightarrow r_{\dot{\Theta}}$
- Tvárme sa, že poznáme žiadane orientácie, napr. klopenie r_Θ a chceme riadiť y_Θ . V skutočnosti riešené s kvaterniónmi aby sme obišli singularitu Eulerových uhlov pri odhade [Erasmus 2020].
- Riadenie orientácie môže byť riešené ďalšou, nadradenou regulačnou slučkou - hovoríme o tzv. kaskádnom riadení (*angl.*: nested, cascaded).



- Riadiť orientáciu (*angl.*: attitude) len na základe zmeny uhl. rýchlosťi (*angl.*: rate) by bolo dosť neintuitívne, potrebujeme prepočítať $r_\Theta \rightarrow r_{\dot{\Theta}}$
- Tvárme sa, že poznáme žiadane orientácie, napr. klopenie r_Θ a chceme riadiť y_Θ . V skutočnosti riešené s kvaterniónmi aby sme obišli singularitu Eulerových uhlov pri odhade [\[Erasmus 2020\]](#).
- Riadenie orientácie môže byť riešené ďalšou, nadradenou regulačnou slučkou - hovoríme o tzv. kaskádom riadení (*angl.*: nested, cascaded).

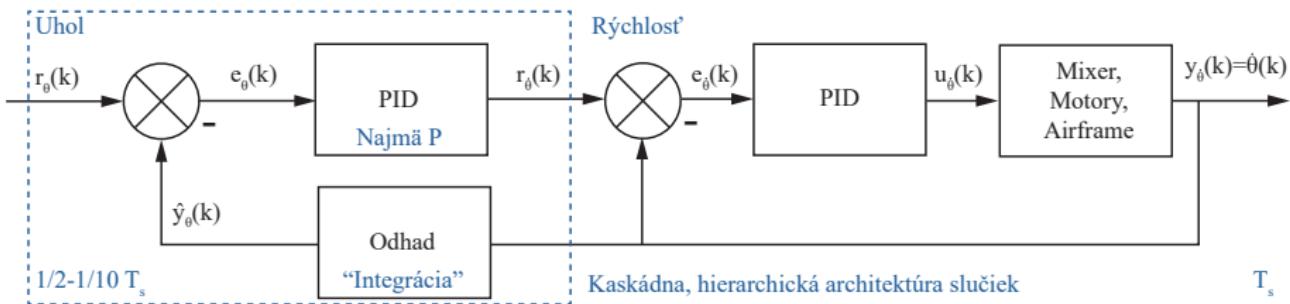


- Nadradené slučky sú pomalšie, vytvára to istý "filter", t.j. nemôžeme rýchlejšie ovládať rýchlosť ako polohu (cca. o rád, min polovicu pomalšie¹) [Hall 2018; PX4 Autopilot 2021]
- Pri PID skôr P², lebo reaguje príliš agresívne na šum.
- Do slučky dopracujeme doprednú väzbu (*angl.:* feedforward) ktorá zrýchli odozvu regulácie. "Whatever works" - netreba mystifikovať.

¹ArduCopter 40 Hz vs 400 Hz, PX4 250 vs. 1000 Hz [Hall 2018; PX4 Autopilot 2021]

²ArduCopter a PX4 Autopilot používa P regulátor [PX4 Autopilot 2021; ArduPilot 2021]

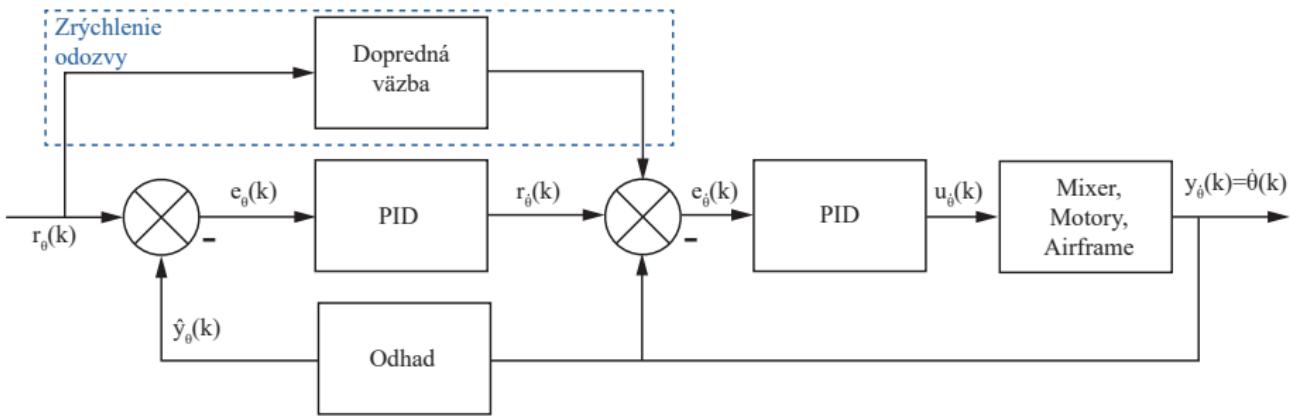
- Nadradené slučky sú pomalšie, vytvára to istý "filter", t.j. nemôžeme rýchlejšie ovládať rýchlosť ako polohu (cca. o rád, min polovicu pomalšie¹) [Hall 2018; PX4 Autopilot 2021]
- Pri PID skôr P², lebo reaguje príliš agresívne na šum.
- Do slučky dopracujeme doprednú väzbu (*angl.*: feedforward) ktorá zrýchli odozvu regulácie. "Whatever works" - netreba mystifikovať.



¹ArduCopter 40 Hz vs 400 Hz, PX4 250 vs. 1000 Hz [Hall 2018; PX4 Autopilot 2021]

²ArduCopter a PX4 Autopilot používa P regulátor [PX4 Autopilot 2021; ArduPilot 2021]

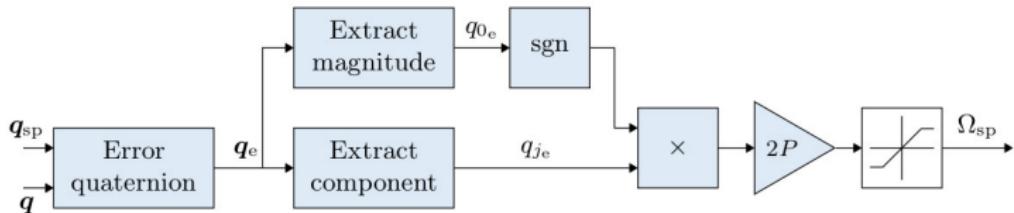
- Nadradené slučky sú pomalšie, vytvára to istý "filter", t.j. nemôžeme rýchlejšie ovládať rýchlosť ako polohu (cca. o rád, min polovicu pomalšie¹) [Hall 2018; PX4 Autopilot 2021]
- Pri PID skôr P², lebo reaguje príliš agresívne na šum.
- Do slučky dopracujeme doprednú väzbu (*angl.*: feedforward) ktorá zrýchli odozvu regulácie. "Whatever works" - netreba mystifikovať.



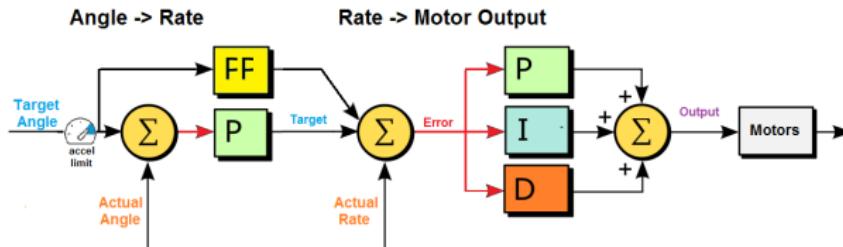
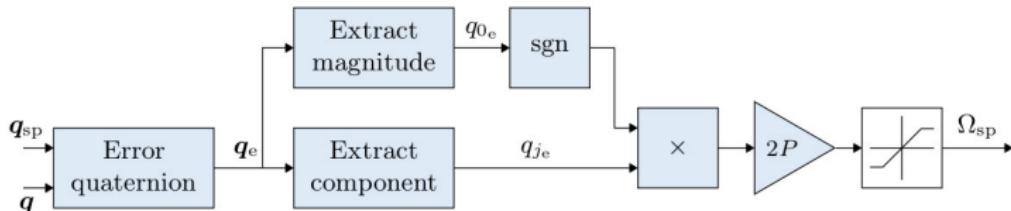
¹ArduCopter 40 Hz vs 400 Hz, PX4 250 vs. 1000 Hz [Hall 2018; PX4 Autopilot 2021]

²ArduCopter a PX4 Autopilot používa P regulátor [PX4 Autopilot 2021; ArduPilot

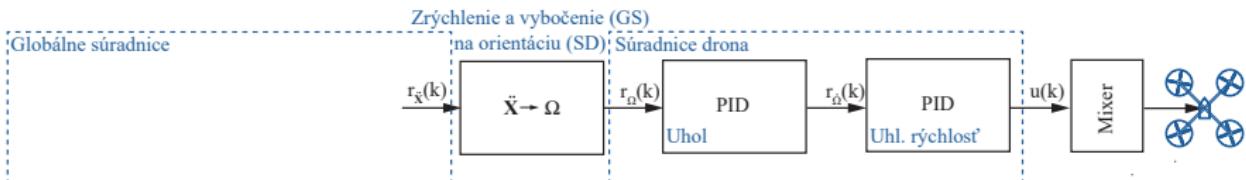
- PX4 používa kvaternióny, ale je to iba P regulátor
- ArduCopter v podstate taktiež tam má P regulátor + FF



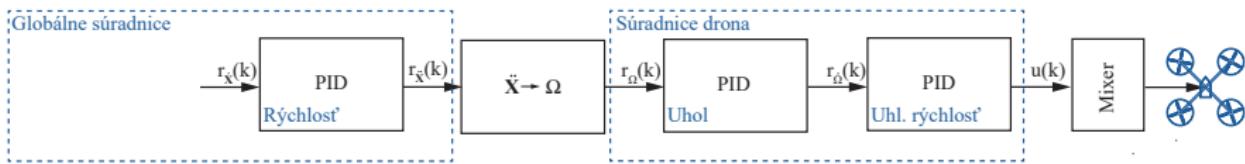
- PX4 používa kvaternióny, ale je to iba P regulátor
- ArduCopter v podstate taktiež tam má P regulátor + FF



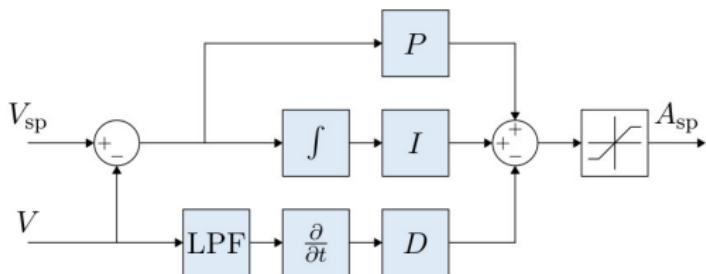
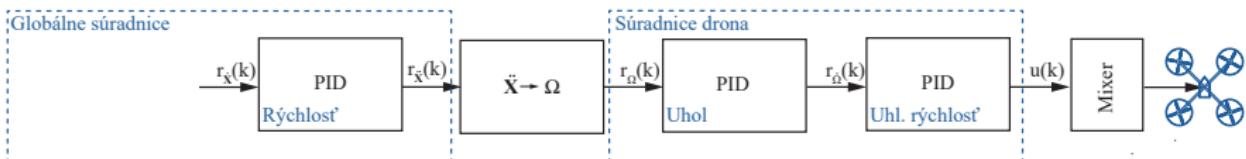
• aaaa



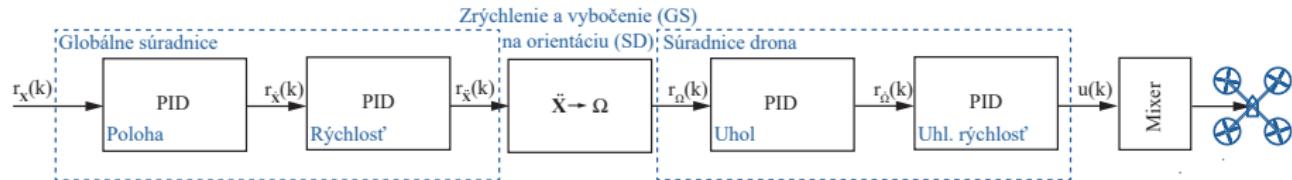
• aaaa



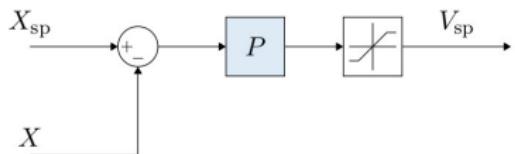
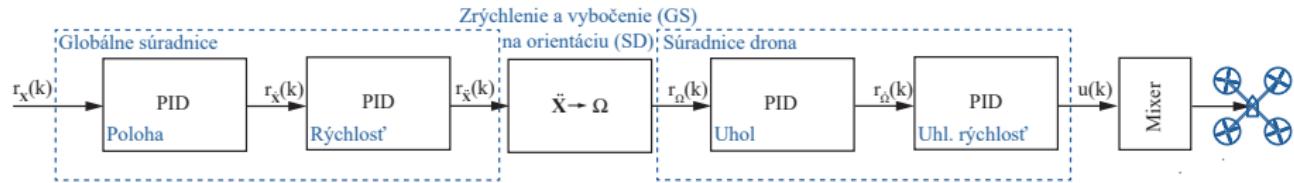
- aaaa



● aaaa



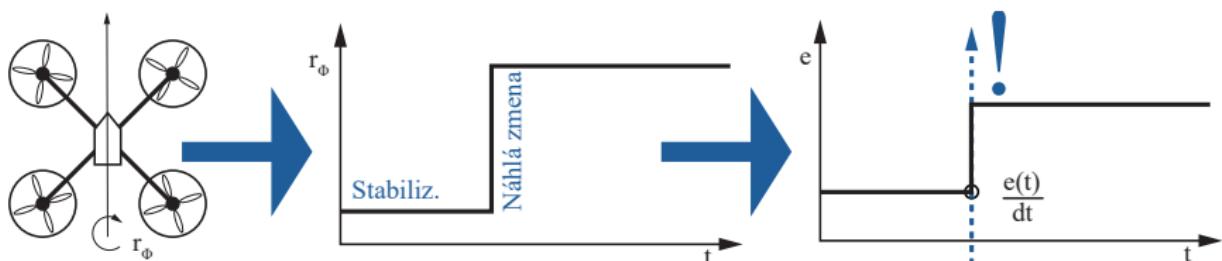
• aaaa



- Dron je stabilizovaný, pilot/ROS náhle chce $r_\phi = +30^\circ$ klonenie. Čo sa stane s riadením?
- Žiadana hodnota je skokový signál. Derivácia skoku je... D zložka a tým aj vstup do akčných členov vystrelí!
- Potrebujeme tvarovať vstupy do regulácie (*angl.: input shaping*), t.j. tvarovať žiadane hodnoty:
 - ▶ Pomalšie vzorkovanie na rýchlejšie (interpolácia) ³
 - ▶ Vyhladenie filtráciou, saturácie
 - ▶ Ak hovoríme o manuálnom pilotovaní, tvarovanie určuje aký "pocit" je riadiť stroj

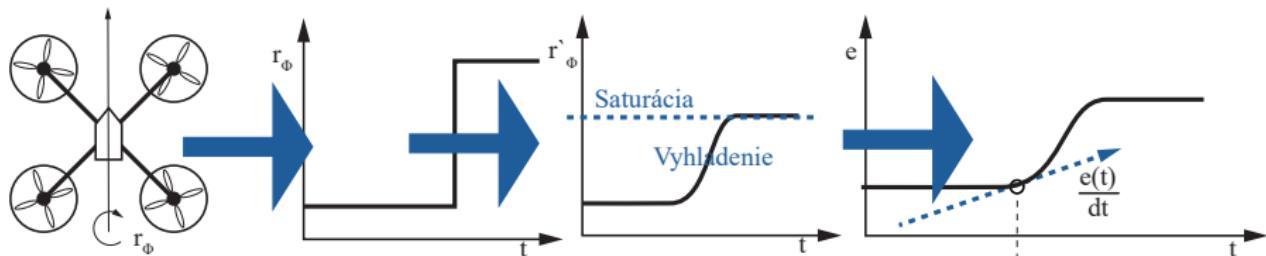
³ArduCopter 50 Hz → 400 Hz [Hall 2018]

- Dron je stabilizovaný, pilot/ROS náhle chce $r_\phi = +30^\circ$ klonenie. Čo sa stane s riadením?
- Žiadana hodnota je skokový signál. Derivácia skoku je... D zložka a tým aj vstup do akčných členov vystrelí!
- Potrebujeme tvarovať vstupy do regulácie (angl.: input shaping), t.j. tvarovať žiadane hodnoty:
 - ▶ Pomalšie vzorkovanie na rýchlejšie (interpolácia)³
 - ▶ Vyhľadenie filtráciou, saturácie
 - ▶ Ak hovoríme o manuálnom pilotovaní, tvarovanie určuje aký "počit" je riadiť stroj



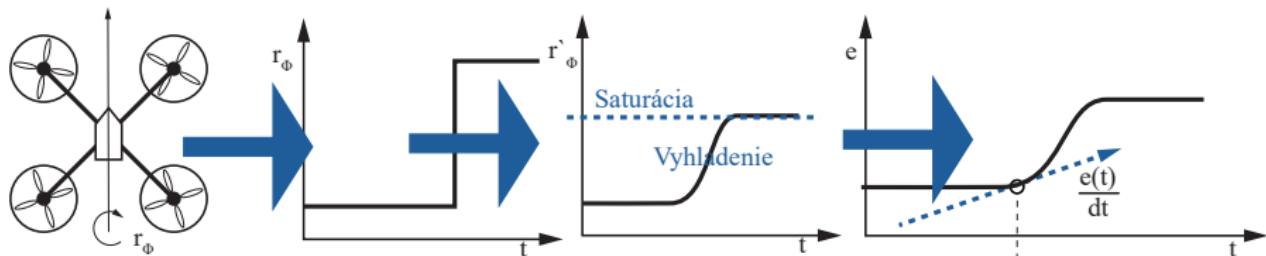
³ArduCopter 50 Hz → 400 Hz [Hall 2018]

- Dron je stabilizovaný, pilot/ROS náhle chce $r_\phi = +30^\circ$ klonenie. Čo sa stane s riadením?
- Žiadana hodnota je skokový signál. Derivácia skoku je... D zložka a tým aj vstup do akčných členov vystrelí!
- Potrebujeme tvarovať vstupy do regulácie (angl.: input shaping), t.j. tvarovať žiadane hodnoty:
 - ▶ Pomalšie vzorkovanie na rýchlejšie (interpolácia) ³
 - ▶ Vyhladenie filtráciou, saturácie
 - ▶ Ak hovoríme o manuálnom pilotovaní, tvarovanie určuje aký "pocit" je riadiť stroj



³ArduCopter 50 Hz → 400 Hz [Hall 2018]

- Dron je stabilizovaný, pilot/ROS náhle chce $r_\phi = +30^\circ$ klonenie. Čo sa stane s riadením?
- Žiadana hodnota je skokový signál. Derivácia skoku je... D zložka a tým aj vstup do akčných členov vystrelí!
- Potrebujeme tvarovať vstupy do regulácie (angl.: input shaping), t.j. tvarovať žiadane hodnoty:
 - ▶ Pomalšie vzorkovanie na rýchlejšie (interpolácia) ³
 - ▶ Vyhľadenie filtráciou, saturácie
 - ▶ Ak hovoríme o manuálnom pilotovaní, tvarovanie určuje aký "pocit" je riadiť stroj



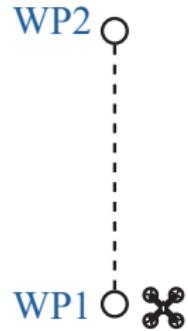
³ArduCopter 50 Hz → 400 Hz [Hall 2018]

- Ako naštartujeme riadenie? Aká je odchýlka $e(0)$ pri štarte?
- Kritické pri zmene letových módov.
- Pre ArduCopter Hall 2018
 - ▶ Nadradené riadenie poloha vs. rýchlosť (P) — tak aby vstup bol konštantný
 - ▶ Rýchlosť (PID) — $e(0) = 0$, I zložka s konštantným vstupom, zrátať D pri nastavení žiadanej hodnoty

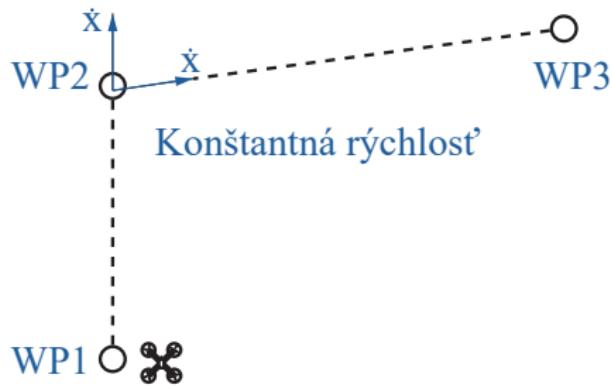
- Ako naštartujeme riadenie? Aká je odchýlka $e(0)$ pri štarte?
- Kritické pri zmene letových módov.
- Pre ArduCopter Hall 2018
 - ▶ Nadradené riadenie poloha vs. rýchlosť (P) — tak aby vstup bol konštantný
 - ▶ Rýchlosť (PID) — $e(0) = 0$, I zložka s konštantným vstupom, zrátať D pri nastavení žiadanej hodnoty

- Ako naštartujeme riadenie? Aká je odchýlka $e(0)$ pri štarte?
- Kritické pri zmene letových módov.
- Pre ArduCopter [Hall 2018](#)
 - ▶ Nadradené riadenie poloha vs. rýchlosť (P) — tak aby vstup bol konštantný
 - ▶ Rýchlosť (PID) — $e(0) = 0$, I zložka s konštantným vstupom, zrátať D pri nastavení žiadanej hodnoty

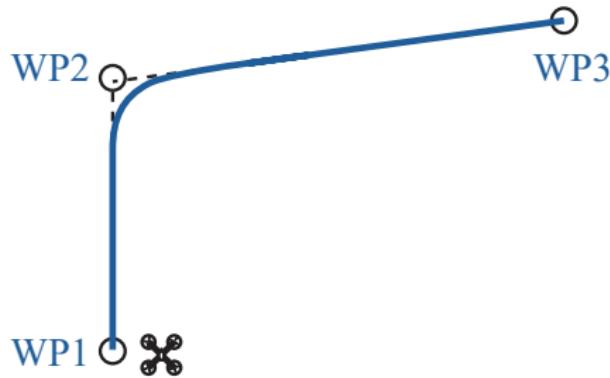
- Majme trasu WP1 do WP2, všetko je v poriadku.
- Pridajme WP3 a rozmýšľajme čo sa deje pri WP2. Je možné preletieť nad WP2?
- Bud' musíme úplne sa zastaviť alebo nemôžeme priamo preletieť — ináč by sme potrebovali nekonečne veľké zrýchlenia



- Majme trasu WP1 do WP2, všetko je v poriadku.
- Pridajme WP3 a rozmýšľajme čo sa deje pri WP2. Je možné preletieť nad WP2?
- Bud' musíme úplne sa zastaviť alebo nemôžeme priamo preletieť — ináč by sme potrebovali nekonečne veľké zrýchlenia



- Majme trasu WP1 do WP2, všetko je v poriadku.
- Pridajme WP3 a rozmýšľajme čo sa deje pri WP2. Je možné preletieť nad WP2?
- Bud' musíme úplne sa zastaviť alebo nemôžeme priamo preletieť — ináč by sme potrebovali nekonečne veľké zrýchlenia



- ArduPilot (ArduCopter) — Napodobňuje heuristiku [Tridgell 2021]
 - ▶ Zvyšuje D parameter pomaly
 - ▶ Akonáhle deteguje osciláciu, zníži D parameter
 - ▶ Opakuje pre iné zložky/slučky
- Intuitívne ale konzervatívne

Notes: [Hall 2018](#)

User (ROS) → Shaping → PID → Actuators Yaw je prioritizovanych nad Pitch Roll, lebo to drzi dron v lufte [\[Erasmus 2020\]](#) 50 Hz -> 400 Hz Min 24 tazsie uchopytelne koncepty pre prezentaciu, dava menej konkretnosti

Ďakujem za Vašu pozornosť.

- [1] **ArduPilot.** *Copter Attitude Control*. Online. [cited 29.11.2021]; Available from <https://ardupilot.org/dev/docs/apmcopter-programming-attitude-control-2.html>. 2021.
- [2] **Anton Erasmus.** *An In-depth Look at the Multicopter Control System Architecture*. Online. PX4 Developer Summit Virtual 2020. [cited 1.12.2021]; Available from <https://www.youtube.com/watch?v=nEo4WG14Lgc>. 2020.
- [3] **Leonard Hall.** *Pratical PID implementation and the new Position Controller*. [online]. ArduPilot UnConference 2018, uploaded Feb 22, 2018 [cited 24.11.2021]; Available from <https://www.youtube.com/watch?v=-PC69jcMizA>. 2018.
- [4] **PX4 Autopilot.** *Controller Diagrams — Multicopter Control Architecture*. Online. [cited 29.11.2021]; Available from https://docs.px4.io/master/en/flight_stack/controller_diagrams.html. 2021.
- [5] **Andrew Tridgell.** *ArduPilot Log Analysis Seminar*. Online. [cited 10.11.2021]; Available from https://www.youtube.com/watch?v=WcfLTW_qZ08&list=PLC8WVaJJhN4ya_HDxh6qGBT6VbjXR6L5p. 2021.