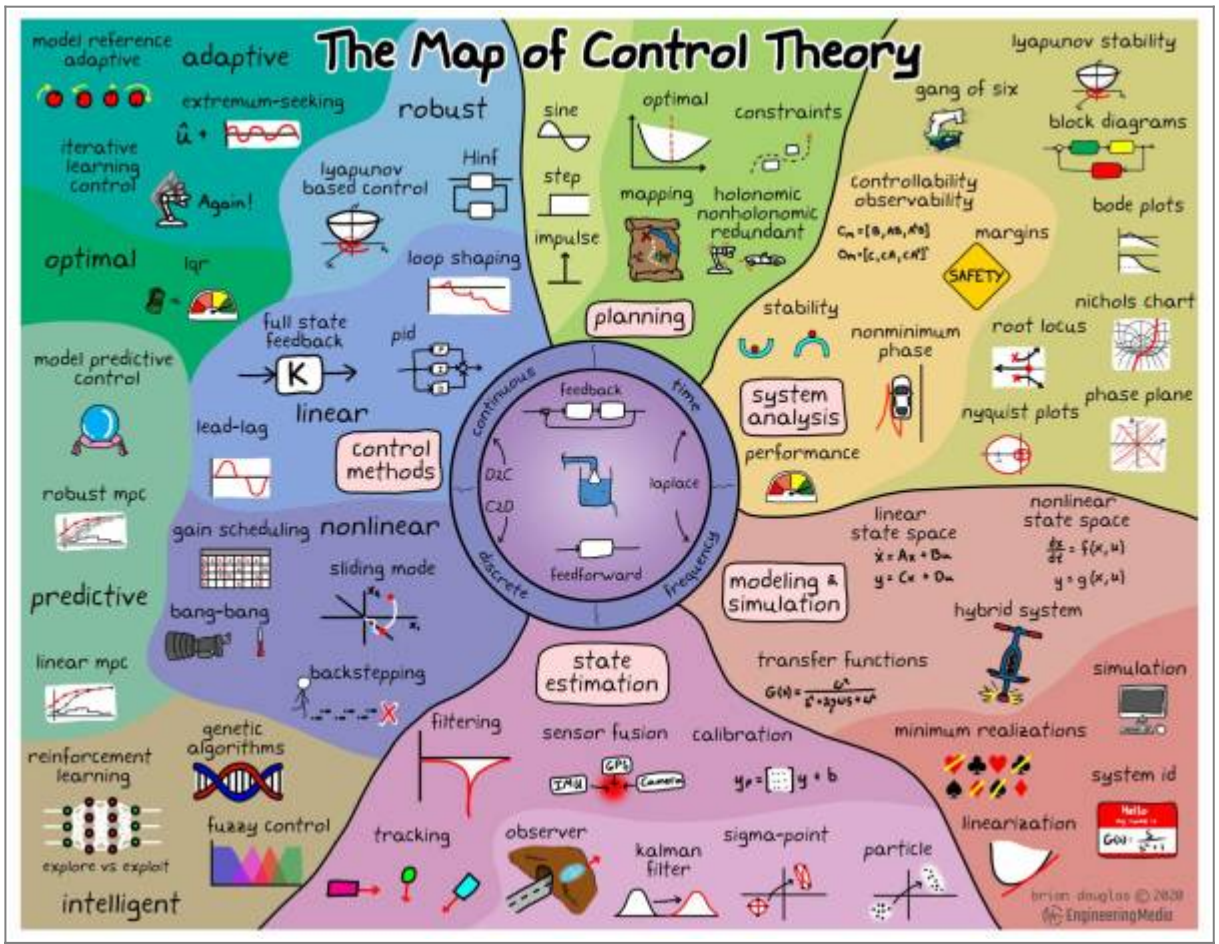
EMRP2022

Topics: Wireless sensor networks to collect environmental data, Interactive web dashboards and databases to display data, Drone technology, sensor integration, and image analysis

Closed loop control

PID Control Theory

https://www.youtube.com/watch?v=wkfEZmsQqiA&list=PLn8PRpmsu08pQBgjxYFXSsODEF3Jqmm-y&ab_channel=MATLAB



https://engineeringmedia.com/map-of-controlFig. 1

Inverted Pendulum with PID

Optimal control of inverted pendulum system using PID controller, LQR and MPC

https://iopscience.iop.org/article/10.1088/1757-899X/263/5/052007/pdf

Stabilising an Inverted Pendulum Controller with PID controller

https://www.matec-conferences.org/articles/matecconf/pdf/2018/11/matecconf_eureca2018_02009.pdf

Control the Ryze Tello Drone from Python

- **tello-pathon** code by Harley Lara: https://github.com/harleylara/tello-python
- **RyzeTelloHSRW** code by Ilgar Rasulov (EligoSoftware): https://github.com/eligosoftware/ryzetellohsrw

git

git for dummies (eli5): https://www.youtube.com/watch?v=mJ-qvsxPHpY&ab_channel=NickWhite

git for professionals: https://www.youtube.com/watch?v=Uszj_k0DGsg&ab_channel=freeCodeCamp.org

useful resources

ardupilot: https://ardupilot.org/copter/index.html#

PX4: https://docs.px4.io/main/en/

Quadcopter construction guide: https://docs.px4.io/main/en/frames_multicopter/dji_f450_cuav_5plus.html

OpenDroneMap: https://opendronemap.org/

MAVlink: https://mavlink.io/en/

Dronekit: https://dronekit.io/

API: https://dronekit-python.readthedocs.io/en/latest/automodule.html

ROS: http://wiki.ros.org/Documentation

Jetson TX2: https://elinux.org/Jetson_TX2

ZED ROS wrapper: https://github.com/stereolabs/zed-ros-wrapper

Understanding 3-axis flight movement: https://emissarydrones.com/what-is-roll-pitch-and-yaw

IMU: https://www.ceva-dsp.com/ourblog/what-is-an-imu-sensor/

MAVROS: https://dev.px4.io/v1.11_noredirect/en/ros/mavros_installation.html

https://404warehouse.net/2015/12/20/autopilot-offboard-control-using-mavros-package-on-ros/

Companion computers:
https://ardupilot.org/dev/docs/companion-computers.html#companion-computers

IntelRealSense camera: https://github.com/IntelRealSense/librealsense

(Notes from env monitoring research projects 02.11.22) Drones need a flight controller (FMC), there are two software options for a flight controller: ardupilot (Copter is an advanced open-source autopilot system for multicopters, helicopters, and other rotor vehicles) and px4. The receiver should match the transceiver, also you need a gps module, a ground control station like QGroundControl (multiple platforms, used by px4), or missionplanner (only for windows, used by ardupilot), like the receiver/transceiver approach. The software autopilot should be compatible with the hardware, there is a standard called Pixhawk that does that. If it is compatible with pixhawk, we can use ardupilot. Another option for autopilot is px4. We can interface the drone by sending MAVlink commands to the drone. Dronekit is a interface. All the complex tasks must be done from source or with a different framework. ROS (robot operating system) is a bunch of frameworks which is modular. GPS, laser finders can be directly connected to the flight controller. In ROS, Sensor is a publisher who publishes data to the subscribers. MavROS is a direct connection interface which pushes all the info from MAVlink into a ROS format. For image processing, you can't directly connect the camera to the flight controller, for that you need a companion computer. ROS and ROS 2 are different systems with the same concept. Offboard mode needs to be enabled when you link to a raspberry pie or jetson.