

```

mcdp {
# We need to fly for this duration
provides endurance [s]
# While carrying this extra payload
provides extra_payload [kg]
# And providing this extra power
provides extra_power [W]

# Sub-design problem: choose the battery
sub battery = mcdp {
# A battery provides capacity
provides capacity [J]
# and requires some mass to be transported
requires mass [kg]
# requires cost [$]

specific_energy_Li_Ion = 500 Wh / kg

mass >= capacity / specific_energy_Li_Ion
}

# Sub-design problem: actuation
sub actuation = mcdp {
# actuators need to provide this lift
provides lift [N]
# and will require power
requires power [W]
# simple model: quadratic
c = 10.0 W/N^2
power >= lift * lift * c
}

# Co-design constraint: battery must be large enough
power = actuation.power + extra_power
energy = power * endurance
battery.capacity >= energy

# Co-design constraint: actuators must be powerful enough
gravity = 9.81 m/s^2
weight = (battery.mass + extra_payload) * gravity
actuation.lift >= weight

# suppose we want to optimize for size of the battery
requires mass for battery
}

```