Controlling Self-Landing Rockets Using CVXPY

Philipp Schiele Steven Diamond

Eric Luxenberg Stephen Boyd

Austin, TX
July 10, 2023

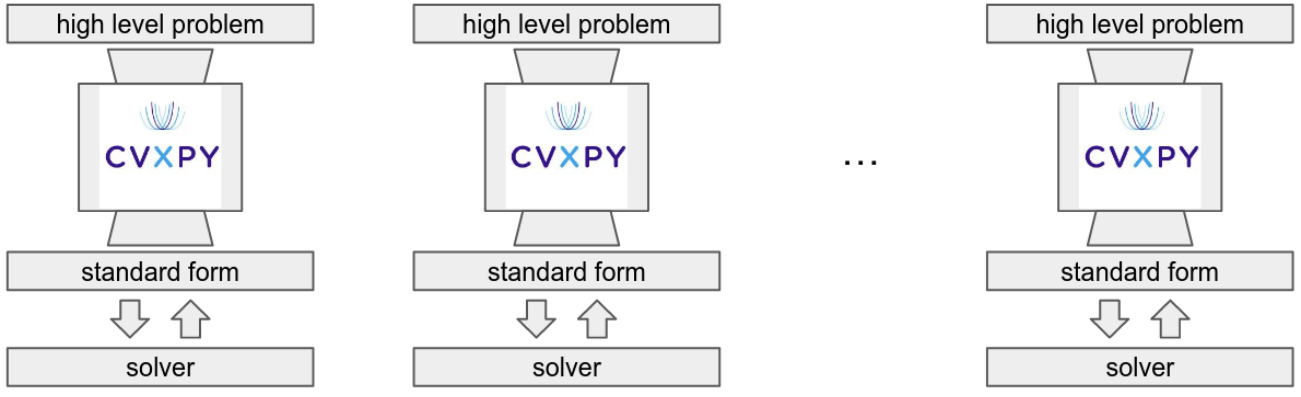
Outline

Disciplined Parametrized Programming (DPP)

- often we want to solve a family of problems of the same form
- we can replace constants with parameters
- this can speed up the canonicalization process for successive solves
- requires that the problem is DPP-compliant

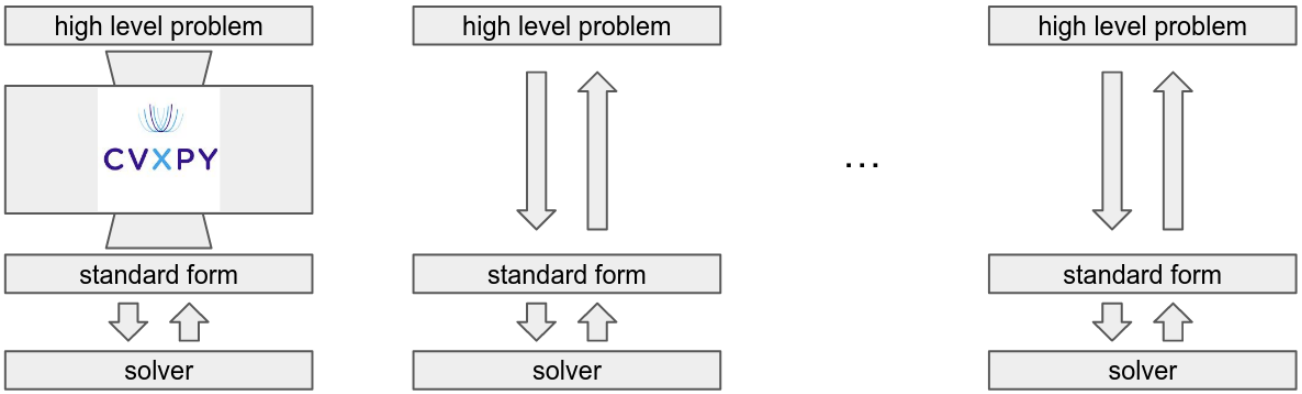
Without DPP

- want to solve P_θ for different values of $\theta \in \mathbf{R}^p$
- without DPP, we have to recompile the problem for each value of θ



With DPP

- can compile problem once, then obtain instances via affine mapping
- initial compilation is slower

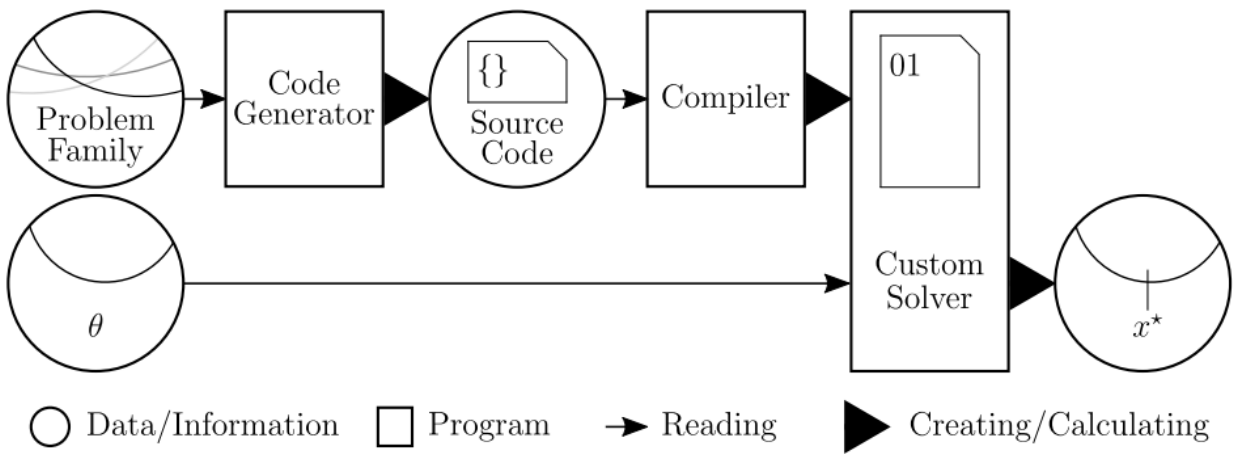


Exercise: DPP

See notebook `DPP.ipynb`

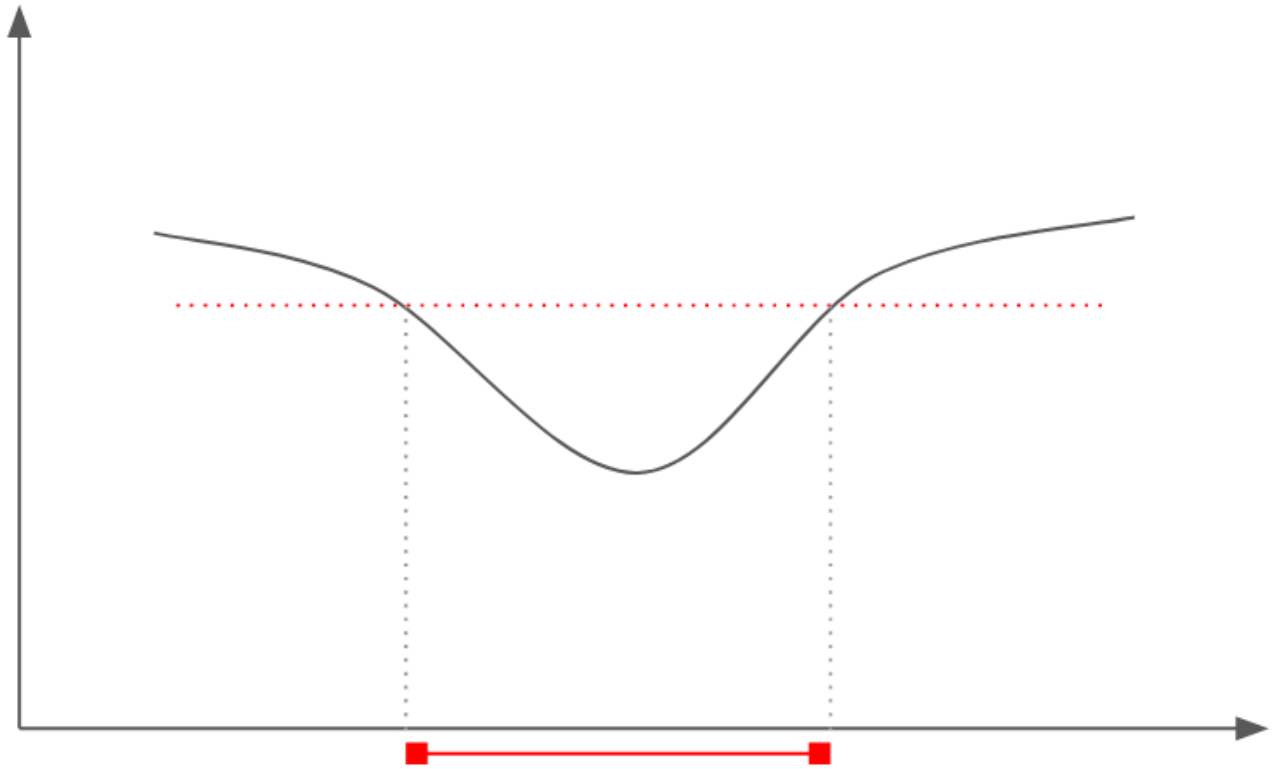
Code generation via CVXPYgen

- DPP provides simple mapping from parameters to standard form
- CVXPYgen (*Schaller et al. 2022*) generates C code that solve the problem given parameters
- can be used to solve problems on embedded devices



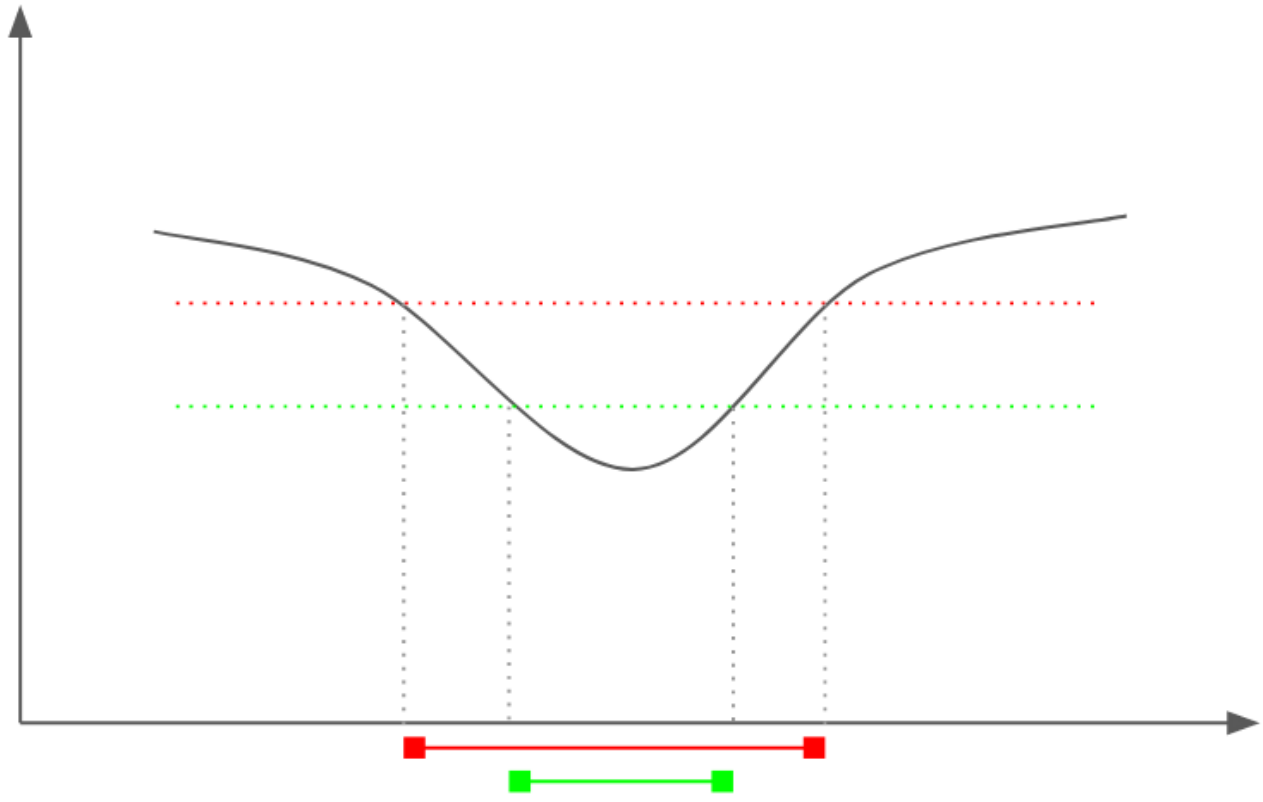
Disciplined Quasiconvex Programming (DQCP)

- quasiconvex functions have convex sublevel sets
- can be solved as a sequence of convex problems



Disciplined Quasiconvex Programming (DQCP)

- quasiconvex functions have convex sublevel sets
- can be solved as a sequence of convex problems



Some quasiconvex functions

- distance ratio $\frac{\|x-a\|}{\|x-b\|}$ with $a, b \in \mathbf{R}^n$
- ceiling and floor, $\lceil x \rceil$ and $\lfloor x \rfloor$
- sign function $sgn(x)$
- **length of a vector**, *i.e.*, index of last nonzero entry

- Can used to solve fastest descent problem

Fastest descent problem

- instead of minimizing fuel consumption, minimize time to reach target
- can be solved using DQCP[1]

```
objective = cp.length(cp.sum(cp.abs(P), axis=1))
constraints = ...
problem = cp.Problem(cp.Minimize(objective), constraints)
problem.solve(qcp=True)
```

[1] requires CVXPY 1.4