Controlling Self-Landing Rockets Using CVXPY

Philipp Schiele Steven Diamond

Eric Luxenberg Stephen Boyd

Austin, TX
July 10, 2023

Outline

Anatomy of a CVXPY problem

```
import cvxpy as cp
x = cp.Variable(n)
objective = cp.sum_squares(A@x-b) + gamma*cp.norm(x,1)
constraints = [cp.norm(x,"inf") <= 1]
problem = cp.Problem(cp.Minimize(objective), constraints)
opt_val = problem.solve()
solution = x.value
```

- `x`: *n*-dimensional optimization variable
- `objective`: convex objective function
- `constraints`: constraints defining a convex set
- `problem`: optimization problem
- `problem.solve()`: solve the problem, return optimal value
- `solution`: optimal value of the variable `x`

Convexity analysis

- How can CVXPY determine if a problem is convex?
- Recall: A convex optimization problem has the form

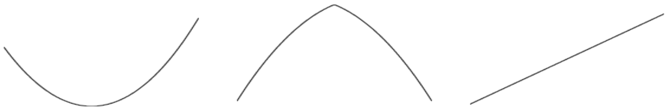
$$
\begin{array}{ll}
\text{minimize} & f_0(x) \\
\text{subject to} & f_i(x) \leq 0, \quad i = 1, \ldots, m \\
& Ax = b
\end{array}
$$

- variable $x \in \mathbf{R}^n$
- equality constraints are **linear**
- $f_0, \ldots, f_m$ are **convex**

Convexity analysis cont.

- How can CVXPY determine if a *function* is convex?
- Determining if an arbitrary function is convex is hard
- Yet, CVXPY can determine convexity efficiently in many cases
- Let us recap the definition of a convex function

Curvature: Convex, concave, and affine functions



- f is *convex* if for any x, y, $\theta \in [0, 1]$,

$$f(\theta x + (1-\theta)y) \leq \theta f(x) + (1-\theta)f(y)$$

- f is *concave* if $-f$ is convex
- f is *affine* if it is convex and concave, *i.e.*,

$$f(\theta x + (1-\theta)y) = \theta f(x) + (1-\theta)f(y)$$

for any x, y, $\theta \in [0, 1]$

A general composition rule

$h(f_1(x), \ldots, f_k(x))$ is convex when h is convex and for each i

- h is increasing in argument i, and f_i is convex, or
- h is decreasing in argument i, and f_i is concave, or
- f_i is affine

- there's a similar rule for concave compositions (just swap convex and concave above)
- allows us to compose convex functions from a small set of building blocks with known curvature
- This is the basis for Disciplined Convex Programming (DCP)

Convex functions: Basic examples

- x^p ($p \geq 1$ or $p \leq 0$), *e.g.*, x^2, $1/x$ ($x > 0$)
- e^x
- $x \log x$
- $a^T x + b$
- $x^T P x$ ($P \succeq 0$)
- $\|x\|$ (any norm)
- $\max(x_1, \ldots, x_n)$

Example

let's show that

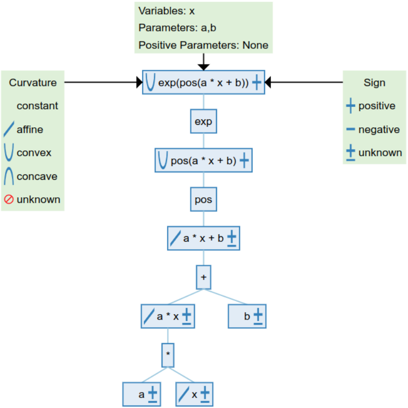
$$f(x) = \sum_{i=1}^{n} e^{(a_i^T x + b_i)_+}$$

is convex, where $(z)_+ = \max(z, 0)$

- $a_i^T x + b_i$ is affine
- $(z)_+$ is convex, so $(a_i^T x + b_i)_+$ is convex
- e^z is convex and increasing, so the composition is convex
- sum of convex functions is convex

DCP Analyzer

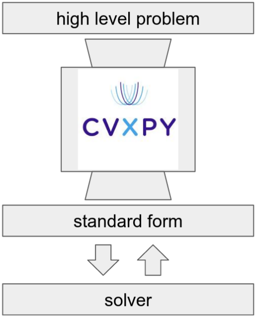
- analyzed by `dcp.stanford.edu` (*Diamond 2014*)



Exercise: DCP analysis

See notebook `DCP_analysis.ipynb`

CVXPY is not a solver



- best thought of as a compiler
- provides modeling language
- verifies problem convexity
- transforms problem into standard form
- includes interfaces to many solvers