# A bit of propaganda on Augmented Lagrangians

Wilson Jallet[1,3] Antoine Bambade[1,2,4]

[1]Inria WILLOW [2]Inria SIERRA [3]LAAS-CNRS [4]ENPC

# Intro

- what is prox-point?
- applying to constrained optimization?

# Proximal-point algorithms

# Prox-point

Let $f : \mathcal{X} \to \mathbb{R}$ be a lsc convex function. Given $\lambda > 0$, $\bar{x} \in \mathcal{X}$, the proximal operator is defined as

$$\mathrm{prox}_{\lambda f}(\bar{x}) = \operatorname*{argmin}_{x \in \mathcal{X}} f(x) + \frac{1}{2\lambda}\|x - \bar{x}\|_2^2. \tag{1}$$

# Prox-point

Let $f : \mathcal{X} \to \mathbb{R}$ be a lsc convex function. Given $\lambda > 0$, $\bar{x} \in \mathcal{X}$, the proximal operator is defined as

$$\operatorname{prox}_{\lambda f}(\bar{x}) = \operatorname*{argmin}_{x \in \mathcal{X}} f(x) + \frac{1}{2\lambda}\|x - \bar{x}\|_2^2. \tag{1}$$

## Facts

1. if $f$ is convex, the subproblem is *strongly* convex (the smaller $\lambda$ the stronger, but the further away from the initial problem)
2. $x^*$ is a minimizer of $f \iff x^* = \operatorname{prox}_{\lambda f}(x^*)$.

The value of the minimum

$$M_{\lambda f}(x) = f(\operatorname{prox}_{\lambda f}(x)) + \frac{1}{2\lambda}\|\operatorname{prox}_{\lambda f}(x) - x\|^2 \tag{2}$$

is called the *Moreau envelope* – and $M_{\lambda f} \in \mathscr{C}^1(\mathcal{X})$.

Prox-point iteration:

$$x^{k+1} = \mathrm{prox}_{\lambda f}(x^k) \tag{3}$$

Under some assumptions, the sequence $(x^k)_k$ converges to a $\bar{x} \in \operatorname{argmin} f$.

There are a lot of known closed-form prox operators[1].

[1] *http://proximity-operator.net/scalarfunctions.html*

# Dual prox-point: equality constraints

Consider a (convex) problem, where $c : \mathcal{X} \to \mathbb{R}^m$

$$\min_{x \in \mathcal{X}} f(x) \text{ s.t. } c(x) = 0 \tag{4}$$

and its dual

$$\max_{y} \underbrace{\min_{x \in \mathcal{X}} f(x) + y^\top c(x)}_{\stackrel{\text{def}}{=} g(y)} \tag{5}$$

The prox-point on this is

$$y^{k+1} = \operatorname*{argmax}_{y} g(y) - \frac{\mu_k}{2} \|y - y^k\|_2^2 \tag{6}$$

where the steplength is $\mu_k$. The problem becomes $\mu_k$-*strongly concave*.

## Prox-point: convergence bound

On the dual (or the primal!) problem,

$$|g(y^k) - g(y^*)| \leqslant \frac{\|y^* - y_0\|_2^2}{\sum_{i=0}^{k-1} \frac{1}{\mu_k}}. \tag{7}$$

Stationarity conditions for the dual prox-point read

$$\begin{bmatrix} \nabla f(x) + \nabla c(x)^\top y \\ c(x) + \mu_k(y^k - y) \end{bmatrix} = 0. \tag{8}$$

# Dual prox-point: inequality constraints

The problem:

$$\min_{x \in \mathcal{X}} f(x) \text{ s.t. } c(x) \leqslant 0 \tag{9}$$

Following the same process:

$$y^{k+1} = \operatorname*{argmax}_{y \geqslant 0} g(y) - \frac{\mu_k}{2} \|y - y^k\|_2^2. \tag{10}$$

The definition of $g$ and stationarity condition read

$$\begin{bmatrix} \nabla f(x) + \nabla c(x)^\top y \\ c(x) + \mu_k(y^k - y) - \partial \mathcal{I}_{\mathbb{R}^m_+}(y) \end{bmatrix} \ni 0. \tag{11}$$

# Dual prox-point: inequality constraints

The problem:

$$\min_{x\in\mathcal{X}} f(x) \text{ s.t. } c(x) \leqslant 0 \tag{9}$$

Following the same process:

$$y^{k+1} = \operatorname*{argmax}_{y\geqslant 0} g(y) - \frac{\mu_k}{2}\|y - y^k\|_2^2. \tag{10}$$

The definition of $g$ and stationarity condition read

$$\begin{bmatrix} \nabla f(x) + \nabla c(x)^\top y \\ [c(x) + \mu_k y^k]_+ - \mu_k y \end{bmatrix} = 0. \tag{11}$$

Equivalently:

1. get $x$ using minimize step

$$x^{k+1} = \operatorname*{argmin}_x \max_{y \geqslant 0} f(x) + y^\top c(x) - \frac{\mu_k}{2} \|y - y^k\|^2$$
$$= \operatorname*{argmin}_x \underbrace{f(x) + \frac{1}{2\mu_k} \|[c(x) + \mu_k y^k]_+\|^2 - \frac{\mu_k}{2} \|y^k\|^2}_{\stackrel{\text{def}}{=} \mathcal{L}_{\mu_k}(x; y^k)} \qquad (12)$$

where $\mathcal{L}_{\mu_k}$ is called the *augmented Lagrangian* function.

2. update $y^{k+1} = \left[ y^k + \frac{1}{\mu_k} c(x^{k+1}) \right]_+$

...this is called the *augmented Lagrangian method* (ALM).

Sometimes (LASSO, splitting...), we *assume that the prox operator has a closed form.*

Sometimes (LASSO, splitting...), we *assume that the prox operator has a closed form.*

What if we **don't**?

# Practical algorithms

# Practical algorithms

Problem setting/*Convex Quadratic Programs*

# Convex QP problem

$$\min_{x\in\mathbb{R}^n} \underbrace{\frac{1}{2}x^\top Hx + q^\top x}_{:=f(x)}$$

$$\text{s.t. } Cx - u \leqslant 0.$$

$H \in \mathcal{S}_+(\mathbb{R}^n)$, $C \in \mathbb{R}^{n_i \times n}$

$$\mathcal{L}(x,z) := f(x) + z^\top (Cx - u) \tag{13}$$

$$\begin{cases} \nabla_x \mathcal{L}(x,z) = 0, \\ Cx - u \leqslant 0, \\ z \odot [Cx - u] = 0, \\ z \geqslant 0 \end{cases} \tag{KKT}$$

# Convex QP problem

$$\mathcal{L}_A(x, z; \mu) \stackrel{\text{def}}{=} f(x) + \frac{1}{2\mu}\left( \|[Cx - u + \mu z]_+\|_2^2 - \|\mu z\|_2^2 \right) \tag{14}$$

$$\begin{cases} x^{k+1} \approx_{\varepsilon^k} & \operatorname{argmin}_x \mathcal{L}_A(x, z^k; \mu), \\ z^{k+1} = & [z^k + \frac{1}{\mu}(Cx^{k+1} - u)]_+. \end{cases} \tag{15}$$

$$\begin{cases} x^{k+1} \approx_{\varepsilon^k} & \operatorname{argmin}_x \Phi^k_{\mu,\rho}(x), \\ z^{k+1} = & [z^k + \frac{1}{\mu}(Cx^{k+1} - u)]_+. \end{cases} \tag{16}$$

$$\Phi^k_{\mu,\rho}(x) \stackrel{\text{def}}{=} \mathcal{L}_A(x, z^k; \mu) + \frac{\rho}{2}\|x - x^k\|_2^2$$

# Convex QP problem

$$x^{k+1}, z^{k+1} \approx_{\varepsilon^k} \underset{x,z}{\operatorname{argmin}} \mathcal{M}^k_{\mu,\rho}(x, z). \tag{17}$$

$$\begin{aligned} \mathcal{M}^k_{\mu,\rho}(x, z) &:= \Phi^k_{\mu,\rho}(x) \\ &+ \frac{1}{2\mu} \| [Cx - u + z^k \mu]_+ - z\mu \|_2^2. \end{aligned} \tag{18}$$

# Practical algorithms

## Technical specifications/*Cahier des charges*

| *QP solvers* | Method used | Limitations |
|---|---|---|
| Mosek, Gurobi | Interior Point | No warm-start |
| OSQP | ADMM | Accuracy to low threshold |
| quadprog, qpOASES | Active set | Robustness |

# Practical algorithms

## Implementations of PAL-type methods

# A recap

## ALM

$z^{k+1} = \mathrm{argmax}_{z \geqslant 0}\, g(z) - \frac{\mu_k}{2}\|z - z^k\|_2^2 = \mathrm{prox}_{\frac{1}{\mu_k}T_g}(z^k)$ is equivalent to

$$\begin{aligned} x^{k+1} &\in \underset{x}{\mathrm{argmin}}\, \mathcal{L}_\mu(x; z^k) \\ z^{k+1} &= [z^k + \frac{1}{\mu}(Cx^{k+1} - u)]_+. \end{aligned} \tag{19}$$

## Resolvant operator

$$\mathrm{prox}_{\frac{1}{\mu_k}T_g} \overset{\mathrm{def}}{=} (I + \frac{1}{\mu_k}T_g)^{-1}, \tag{20}$$

with $T_g = -\partial g$.

# The PAL

## Proximal Augmented Lagrangian

$x^{k+1}, z^{k+1} = \text{prox}_{\Sigma T}(x^k, z^k)$ is equivalent to

$$
\begin{aligned}
x^{k+1} &= \underset{x}{\text{argmin}} \underbrace{\mathcal{L}_\mu(x; z^k) + \frac{\rho}{2}\|x - x^k\|_2^2}_{\stackrel{\text{def}}{=} \varphi^k_{\mu,\rho}(x)} \\
z^{k+1} &= [z^k + \frac{1}{\mu}(Cx^{k+1} - u)]_+.
\end{aligned} \tag{21}
$$

## Resolvant operator

$$
\text{prox}_{\Sigma T} \stackrel{\text{def}}{=} (I + \Sigma T)^{-1}, \tag{22}
$$

with $\Sigma \in \mathcal{S}_{++}(\mathbb{R}^n)$, $T$ KKT conditions (12).

# QPALM [1]

**Idea:** Approximate $\operatorname{prox}$ with **semi-smooth Newton**.

Merit function: $\varphi^k_{\mu,\rho}(x) \overset{\text{def}}{=} f(x) + \frac{1}{2\mu}\|[\mu z^k + Cx - u]_+\|_2^2 + \frac{\rho}{2}\|x - x^k\|_2^2.$

$$\begin{cases} x^{k+1} & \approx_{\varepsilon^k} \operatorname{argmin}_x \varphi^k_{\mu,\rho}(x), \\ z^{k+1} & = [z^k + \frac{1}{\mu}(Cx^{k+1} - u)]_+, \\ y^{k+1} & = y^k + \frac{1}{\mu}(Ax^{k+1} - b). \end{cases} \tag{23}$$

with $\|\nabla \varphi^k_{\mu,\rho}(x^{k+1})\|_\infty \leqslant \varepsilon^k, \sum_k \varepsilon^k < +\infty.$

## Stopping criterion

$$\|\nabla \varphi^k_{\mu,\rho}(x^{k+1})\|_\infty \leqslant \varepsilon^k, \sum_k \varepsilon^k < +\infty, \tag{24}$$

With (26): $\|(x^{k+1}, z^{k+1}) - \operatorname{prox}_{\Sigma T}(x^k, z^k)\| \leqslant M\|\nabla \varphi^k_{\mu,\rho}(x^{k+1})\|_\infty.$

# QPALM [1]

**Idea:** Approximate $\operatorname{prox}$ with **semi-smooth Newton**.

Merit function: $\varphi^k_{\mu,\rho}(x) \stackrel{\text{def}}{=} f(x) + \frac{1}{2\mu}\|[\mu z^k + Cx - u]_+\|_2^2 + \frac{\rho}{2}\|x - x^k\|_2^2$.

$$x^{k+1}, z^{k+1} \approx_{\varepsilon^k} \operatorname*{argmin}_{x,z} \mathcal{M}^k_{\mu,\rho}(x,z), \tag{25}$$

with $\|\nabla \mathcal{M}^k_{\mu,\rho}(x^{k+1}, z^{k+1})\|_\infty \leqslant \varepsilon^k, \sum_k \varepsilon^k < +\infty$.

## Stopping criterion

$$\|\nabla \mathcal{M}^k_{\mu,\rho}(x^{k+1}, y^{k+1}, z^{k+1})\|_\infty \leqslant \varepsilon^k, \sum_k \varepsilon^k < +\infty, \tag{26}$$

With (26): $\|(x^{k+1}, z^{k+1}) - \operatorname{prox}_{\Sigma T}(x^k, z^k)\| \leqslant M \|\nabla \varphi^k_{\mu,\rho}(x^{k+1})\|_\infty$.

# QPALM [1]

## Semismooth Newton step

$\hat{x}^{(0)} := x^k$. Generate $\hat{x}^{(l+1)} = \hat{x}^{(l)} + \alpha^* \delta x$ with:

- $\alpha^* := \operatorname{argmin}_{\alpha \geqslant 0} \varphi^k_{\mu,\rho}(\hat{x}^{(l)} + \alpha \delta x)$
- $\delta x$ solution of (1).

$$\begin{bmatrix} H + \rho I & \hat{C}^{(k)}(\hat{x}^l)^\top \\ \hat{C}^{(k)}(\hat{x}^l) & -\mu I \end{bmatrix} \begin{bmatrix} \delta x \\ \delta z \end{bmatrix} = \begin{bmatrix} -\nabla \varphi^k_{\mu,\rho}(\hat{x}^l) \\ 0 \end{bmatrix}, \tag{27}$$

where $\hat{C}^{(k)}(x)$ is a short-hand for the generalized Jacobian of $[Cx - u + \mu z^k]_+$ at $x$ and

$$\nabla \varphi^k_{\mu,\rho}(x) = \nabla f(x) + C^\top [z^k + \frac{1}{\mu}(Cx - u)]_+ + \rho(x - x^k). \tag{28}$$

# QPALM [1]

$\hat{x}^{(0)} = x^k$, $\hat{z}^{(0)} = z^k$.

Generate $\hat{x}^{(l+1)} = \hat{x}^{(l)} + \alpha^*\delta x$, $\hat{z}^{(l+1)} = \hat{z}^{(l)} + \alpha^*\delta z$ with:

- $\alpha^* := \operatorname{argmin}_{\alpha \geqslant 0} \mathcal{M}^k_{\mu,\rho}(\hat{x}^{(l)} + \alpha\delta x, \hat{z}^{(l)} + \alpha\delta z)$
- $\delta x, \delta z$ solutions of (1).

$$\begin{bmatrix} H + \rho I & \hat{C}^{(k)}(\hat{x}^l)^\top \\ \hat{C}^{(k)}(\hat{x}^l) & -\mu I \end{bmatrix} \begin{bmatrix} \delta x \\ \delta z \end{bmatrix} = - \begin{bmatrix} \nabla f(\hat{x}^{(l)}) + \rho(\hat{x}^{(l)} - x^k) + C^\top \hat{z}^{(l)} \\ ([C\hat{x}^{(l)} - u + \hat{z}^{(l)}\mu]_+ - \hat{z}^{(l)}\mu) \end{bmatrix}, \quad (29)$$

where $\hat{C}^{(k)}(\hat{x}^l)$ stands for the generalized Jacobian of $[C\hat{x}^l - u + \mu z^k]_+$.

# Frame Title

$\hat{x}^{(0)} = x^k, \hat{z}^{(0)} = z^k.$

while $\|r(\hat{x}^l, \hat{z}^l)\|_\infty > \varepsilon^k$ do

$$\begin{bmatrix} H + \rho I & \hat{C}^{(k)}(\hat{x}^l)^\top \\ \hat{C}^{(k)}(\hat{x}^l) & -\mu I \end{bmatrix} \begin{bmatrix} \delta x \\ \delta z \end{bmatrix} = -\underbrace{\begin{bmatrix} \nabla f(\hat{x}^{(l)}) + \rho(\hat{x}^{(l)} - x^k) + C^\top \hat{z}^{(l)} \\ ([C\hat{x}^{(l)} - u + \hat{z}^{(l)}\mu]_+ - \hat{z}^{(l)}\mu) \end{bmatrix}}_{\overset{\text{def}}{=} r(\hat{x}^l, \hat{z}^l)}$$

$\alpha^* := \operatorname{argmin}_{\alpha \geqslant 0} \mathcal{M}^k_{\mu,\rho}(\hat{x}^{(l)} + \alpha\delta x, \hat{z}^{(l)} + \alpha\delta z);$

$\hat{x}^{l+1} = \hat{x}^l + \alpha^*\delta x;$

$\hat{z}^{l+1} = \hat{z}^l + \alpha^*\delta z;$

end

$\hat{C}^{(k)}(\hat{x}^l)$ stands for the generalized Jacobian of $[C\hat{x}^l - u + \mu z^k]_+$.

# Enhancements in [5]

1. New merit function, extending [2, 3]
2. Calibrate $\mu$ using *bound-constrained Lagrangian* (BCL) [4]
3. *Good* implementation

# Enhancements in [5]

## Changing merit functions

KKT conditions in $(x, z)$:

$$\begin{aligned} \nabla f(x) + C^\top z + \rho(x - x^k) &= 0, \\ \mu z - [Cx - d + \mu z^k]_+ &= 0. \end{aligned} \tag{30}$$

are equivalent to:

$$\begin{cases} \nabla f(x) + \rho(x - x^k) + C^\top [\frac{1}{\mu}(Cx - u) + z^k]_+ \\ \quad + \frac{1}{\mu} C^\top ([Cx - u + z^k \mu]_+ - z\mu) = 0, \\ z\mu - [Cx - u + z^k \mu]_+ = 0. \end{cases} \tag{31}$$

# Enhancements in [5]

## New merit function

$$\mathcal{M}^k_{\mu,\rho}(x,z) := \varphi^k_{\mu,\rho}(x) + \frac{1}{2\mu}\|[Cx-u+z^k\mu]_+ - z\mu\|_2^2. \tag{32}$$

## Linear systems involved

$$\left\{\begin{array}{l} \begin{bmatrix} H+\rho I & C^\top_{I_k(\hat{x}^{(l)})} \\ C_{I_k(\hat{x}^{(l)})} & -\mu I_{I_k(\hat{x}^{(l)})} \end{bmatrix} \begin{bmatrix} \delta x \\ \delta z_{I_k(\hat{x}^{(l)})} \end{bmatrix} = \\ -\begin{bmatrix} \nabla f(\hat{x}^{(l)}) + \rho(\hat{x}^{(l)} - x^k) + C^\top_{I_k(\hat{x}^{(l)})}\hat{z}^{(l)}_{I_k(\hat{x}^{(l)})} \\ ([C\hat{x}^{(l)} - u + \hat{z}^{(l)}\mu]_+ - \hat{z}^{(l)}\mu)_{I_k(\hat{x}^{(l)})} \end{bmatrix}, \\ \delta z_{I_k^c(\hat{x}^{(l)}))} = -\hat{z}^{(l)}_{I_k^c(\hat{x}^{(l)}))}, \end{array}\right. \tag{33}$$

where $I_k(x) = \{i \in [1, n_i] | Cx - u + z^k\mu \geqslant 0\}$.

## Comparing iterative residual errors

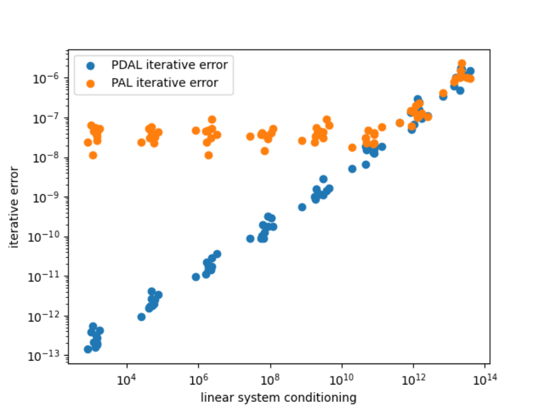



Figure 1: Iterative error after one iteration of semi-smooth Newton (with PAL or PDAL merit functions) for QPs with increasing condition numbers.

# Enhancements in [5]

**while** *Stopping criterion not satisfied* **do**

- $\hat{x}, \hat{z} \approx_{\varepsilon^k} \operatorname{argmin}_{x,z} \mathcal{M}^k_{\mu,\rho}(x,z)$;
- $x^{k+1} = \hat{x}$;
- **if** $\|[Cx^{k+1} - u]_+\|_\infty \leqslant \eta_k$ **then**
  - Accept multiplier $\hat{z}$;
  - Strictly decrease $\varepsilon^k$, $\eta^k$;
- **else**
  - Keep previous multiplier $z^k$;
  - Low decrease of $\varepsilon^k$, $\eta^k$;
  - Strictly decrease $\mu$;
- **end**

**end**

# Enhancements in [5]

$$\text{sparsity} = \frac{\text{number of non zeros}}{\text{number of matrix elements}} \tag{34}$$

$$\text{sparsity} \leqslant 10\% \tag{35}$$

$$\text{matrix dimensions} \leqslant 1000 \tag{36}$$

# Some benchmarks

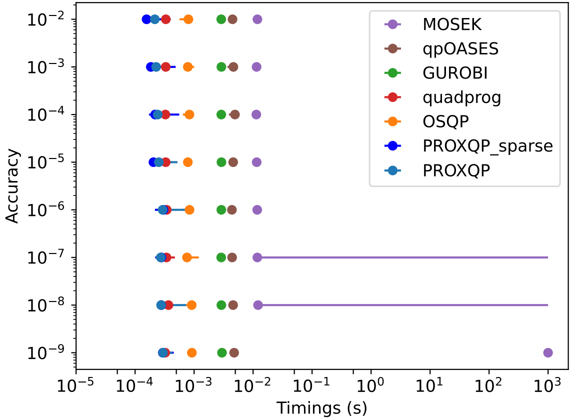



Figure 2: Run times for random sparse equality and inequality constrained QPs (sparsity of $H, A, C \approx 15\%$) with fixed dimension ($d = 50$).

# Some benchmarks

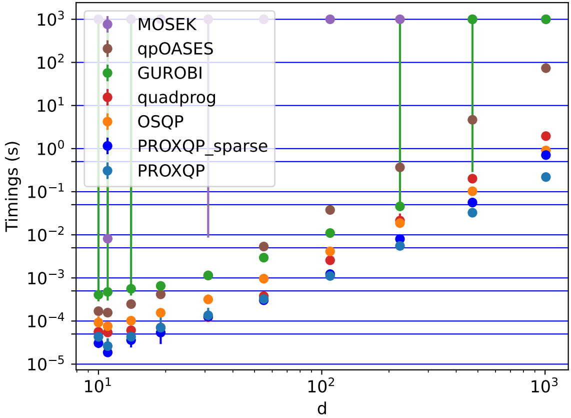



Figure 3: Run times for random sparse equality and inequality constrained QPs (sparsity of $H, A, C \approx 15\%$) with increasing dimension.

# Some benchmarks

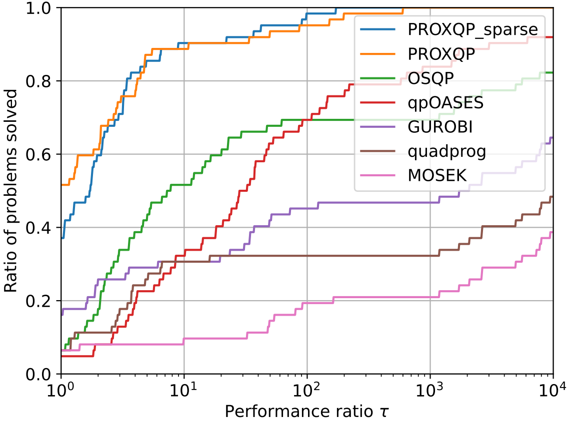



Figure 4: Performance profiles on small to medium-sized Maros-Mészàros problems.

# Applications

# Extension I: Nonlinear control

Work from Jallet et al. [6, 7]

- solve sparse problems
- derive first-order sensibilities from solution
- often nonlinear & nonconvex structure
- software coming up: *`proxddp`*

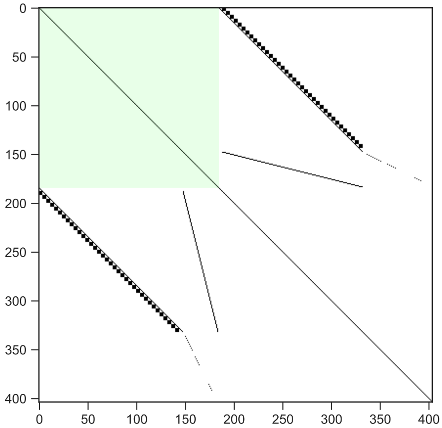



Figure 5: KKT matrix for a Newton step on a control problem (acrobot w/ control bounds).

Basic building block: general **nonlinear programming** (NLP).

Basic building block: general **nonlinear programming** (NLP).

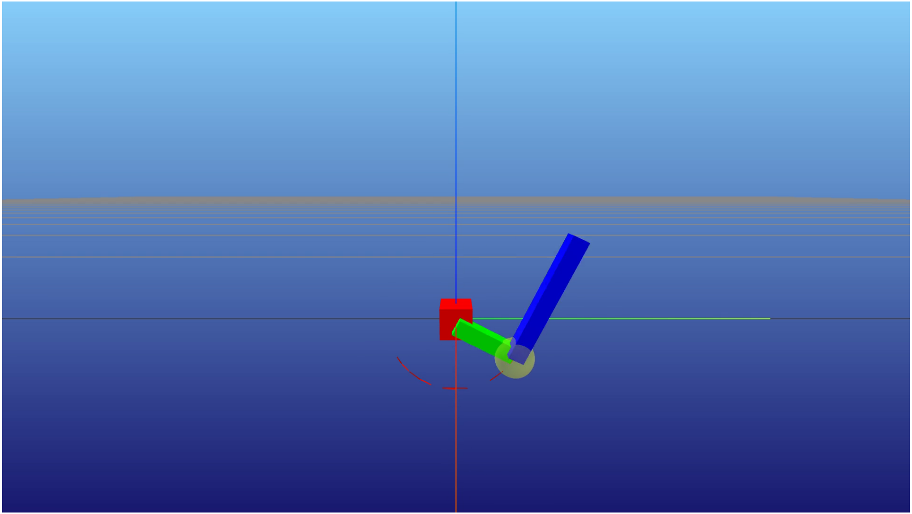
Same principle: approximate the $\mathrm{prox}$ with Newton steps, but:

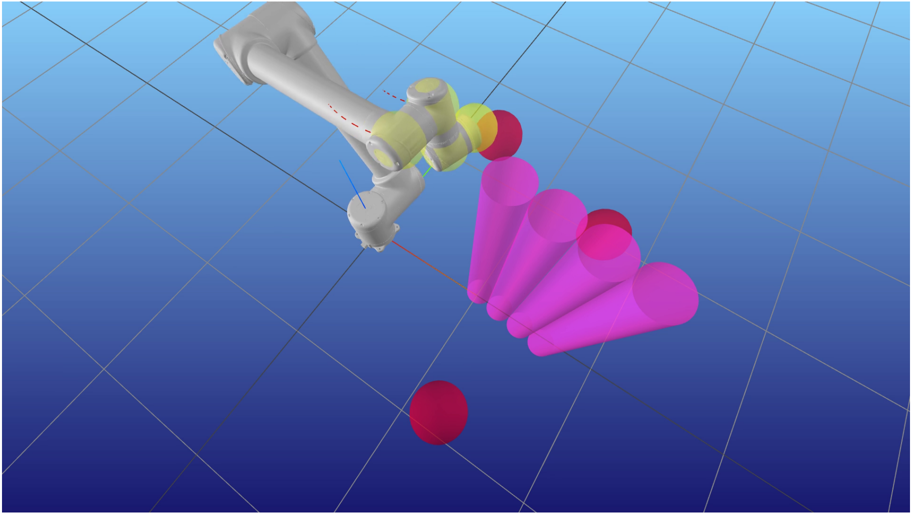
- Hessian matrix changes all the time
- no analytical formula for linesearch

Basic building block: general **nonlinear programming** (NLP).

Same principle: approximate the $\mathrm{prox}$ with Newton steps, but:

- Hessian matrix changes all the time
- no analytical formula for linesearch
- compute Newton steps with dynamic programming (Riccati-type recursion)





# Extension II: sparse coding?

Consider a nonsmooth penalty problem

$$\min_{x \in \mathbb{R}^n} f(x) + \|Lx\|_1. \tag{37}$$

Traditionally: if we know $\operatorname{prox}_f$ then use ADMM.

# Extension II: sparse coding?

Consider a nonsmooth penalty problem

$$\min_{x \in \mathbb{R}^n} f(x) + \|Lx\|_1. \tag{37}$$

Traditionally: if we know $\operatorname{prox}_f$ then use ADMM.

Idea from [8]: build augmented Lagrangian

$$\mathcal{L}_\mu(x; y_e) = f(x) + M_{\mu\ell_1}(Lx + \mu y_e) - \frac{\mu}{2}\|y_e\|^2 \tag{38}$$

where $M_{\mu\ell^1}(v) = \min_z \frac{1}{2\mu}\|z - v\|^2 + \|z\|_1$ is the Moreau envelope.

To investigate:

- semi-smooth Newton steps
- convergence rates

# References

[1] Ben Hermans, Andreas Themelis, and Panagiotis Patrinos. "QPALM: A Newton-type Proximal Augmented Lagrangian Method for Quadratic Programs". In: *2019 IEEE 58th Conference on Decision and Control (CDC)* (Dec. 2019), pp. 4325–4330. DOI: `10.1109/CDC40024.2019.9030211`. arXiv: `1911.02934`.

[2] Philip E. Gill and Daniel P. Robinson. "A Primal-Dual Augmented Lagrangian". In: *Computational Optimization and Applications* 51.1 (Jan. 2012), pp. 1–25. ISSN: 0926-6003, 1573-2894. DOI: `10.1007/s10589-010-9339-1`.

[3] Paul Armand and Riadh Omheni. "A Globally and Quadratically Convergent Primal-Dual Augmented Lagrangian Algorithm for Equality Constrained Optimization". In: *Optimization Methods and Software* (May 5, 2015). DOI: `10/gng3kw`.

[4] Andrew Conn, Nicholas Gould, and Philippe Toint. "A Globally Convergent Augmented Lagrangian Algorithm for Optimization with General Constraints and Simple Bounds". In: *SIAM Journal on Numerical Analysis* 28 (Apr. 1, 1991). DOI: `10.1137/0728030`.

[5] Antoine Bambade et al. *ProxQP: Yet Another Quadratic Programming Solver for Robotics and Beyond*. URL: https://www.semanticscholar.org/paper/ProxQP%3A-Yet-another-Quadratic-Programming-Solver-Bambade-El-Kazdadi/f15cd243a6a397447030d9a0ba98b643313f5877 (visited on 05/31/2022).

[6] Wilson Jallet, Nicolas Mansard, and Justin Carpentier. "Implicit Differential Dynamic Programming". In: *International Conference on Robotics and Automation (ICRA 2022)*. Philadelphia, United States: IEEE Robotics and Automation Society, May 2022.

[7] Wilson Jallet et al. "Constrained Differential Dynamic Programming: A Primal-Dual Augmented Lagrangian Approach". Mar. 2022.

[8] Neil K. Dhingra, Sei Zhen Khong, and Mihailo R. Jovanović. "The Proximal Augmented Lagrangian Method for Nonsmooth Composite Optimization". In: *IEEE Transactions on Automatic Control* 64.7 (July 2019), pp. 2861–2868. ISSN: 1558-2523. DOI: 10/gnzpjn.