Robotic Systems II

Lecture 3: Constrained Optimization Methods

Konstantinos Chatzilygeroudis - costashatz@upatras.gr

Department of Electrical and Computer Engineering
University of Patras

Template made by Panagiotis Papagiannopoulos

Logistics

- **Lectures**:
  - 10 Lectures
  - 1 Recitation Lecture
  - 1-2 Seminars
  - 10 Lab Exercises
- **Examination**:
  - 4 Homeworks **(40% of total grade)**
  - Oral Exam **(60% of total grade)**
- **Office Hours**:
  - **Tue & Thu** *(09:00-11:00)*
  - 24/7 by email (`costashatz@upatras.gr`, subject: *ECE_RSII_AM*)
- **Material and Announcements**

Laboratory of Automation & Robotics

What is this course about?

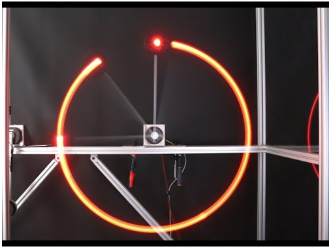
We could name the course “Applied Optimal Control”:

- Optimization
- Control of Linear Systems (LQR)
- Model-Predictive Control
- Linearization of Non-Linear Systems
- Trajectory Optimization (Direct Collocation)
- Optimal Control on Manifolds
- Differential Dynamic Programming
- Locomotion, Contacts, Complex Robots
- **We focus on implementation/robotics applications. You are required to write code!** Not a theory course!

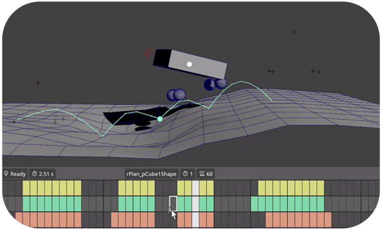
Lectures/Lab Format

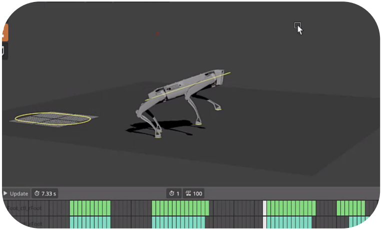
- **3-hour lectures:**
  - A lot of theory! Hints for personal reading!
  - Live code examples + analyses!
  - Ask questions please!
- **2-hour lab exercises:**
  - Do not count for final grade! **You need to participate in 60%**!
  - Small code examples that you need to fill
  - You need to deliver code even if it doesn't run
  - Ask questions, experiment!

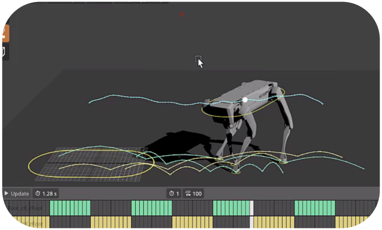
Why study optimal control?

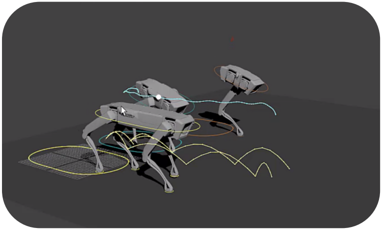


What will you be able to achieve?









Credits: Ragdoll Dynamics (Imbalance Ltd)

Duality

$$\min_{\boldsymbol{x}} f(\boldsymbol{x})$$
$$\text{s.t. } h(\boldsymbol{x}) = 0$$

We could write the problem as:

$$\min_{\boldsymbol{x}} f(\boldsymbol{x}) + P_\infty(h(\boldsymbol{x}))$$

where $P_\infty(h(\boldsymbol{x})) = \begin{cases} 0 & \text{for } h(\boldsymbol{x}) = 0 \\ \infty & \text{otherwise} \end{cases}.$

Duality (2)

Of course, this is "useless" in practical applications, but the above is equivalent to:

$$\min_{\boldsymbol{x}} \max_{\boldsymbol{\lambda}} \underbrace{f(\boldsymbol{x}) + \boldsymbol{\lambda}^T h(\boldsymbol{x})}_{\mathcal{L}(\boldsymbol{x},\boldsymbol{\lambda})}$$

Why?

Of course, this is "useless" in practical applications, but the above is equivalent to:

$$\min_{\boldsymbol{x}} \max_{\boldsymbol{\lambda}} \underbrace{f(\boldsymbol{x}) + \boldsymbol{\lambda}^T h(\boldsymbol{x})}_{\mathcal{L}(\boldsymbol{x},\boldsymbol{\lambda})}$$

Why?

KKT Conditions (equivalent to the above) define a saddle point in $(\boldsymbol{x}, \boldsymbol{\lambda})$: the KKT system should have N (dimension of $\boldsymbol{x}$) positive eigenvalues and M (dimension of $\boldsymbol{\lambda}$) negative eigenvalues at the optimum **("quasi-definite")**.

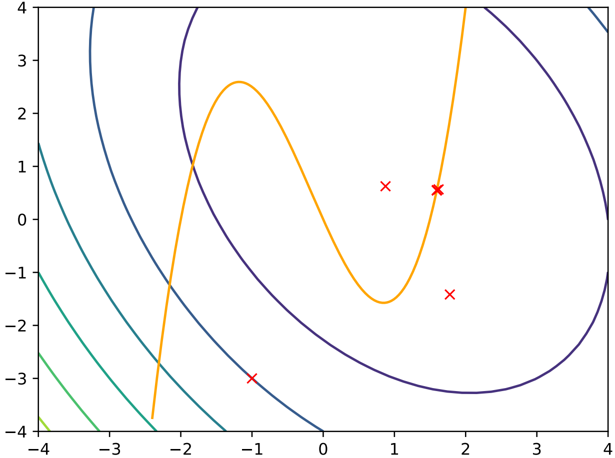
Strong duality: We can swap the min and max. Only when the problem is **strictly convex**.

We now know how to regularize a constrained KKT system:

$$\begin{bmatrix} \left.\frac{\partial^2 \mathcal{L}}{\partial \boldsymbol{x}^2}\right|_{\boldsymbol{x},\boldsymbol{\lambda}} + \beta \boldsymbol{I} & \left(\left.\frac{\partial h}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}}\right)^T \\ \left.\frac{\partial h}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}} & -\beta \boldsymbol{I} \end{bmatrix} \begin{bmatrix} \Delta \boldsymbol{x} \\ \Delta \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} -\nabla_{\boldsymbol{x}} \mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda}) \\ -h(\boldsymbol{x}) \end{bmatrix}$$

for $\beta > 0$. **We apply this while the KKT system is not "quasi-definite!"**

Regularized Newton's Method - Code Example



$$\min_{\boldsymbol{x}} \; f(\boldsymbol{x})$$
$$\text{s.t. } \; g(\boldsymbol{x}) \leqslant 0$$

where $f(\boldsymbol{x}) : \mathbb{R}^N \rightarrow \mathbb{R}, g(\boldsymbol{x}) : \mathbb{R}^N \rightarrow \mathbb{R}^M.$ We define the **Lagrangian** as follows:

$$\mathcal{L}(\boldsymbol{x}, \boldsymbol{\mu}) = f(\boldsymbol{x}) + \boldsymbol{\mu}^T g(\boldsymbol{x})$$

First Order Necessary (KKT) Conditions:

$$1) \; \nabla_{\boldsymbol{x}} f(\boldsymbol{x}) + \left(\frac{\partial g}{\partial \boldsymbol{x}}\right)^T \boldsymbol{\mu} = 0, \qquad \text{“stationarity”}$$
$$2) \; g(\boldsymbol{x}) \leqslant 0, \qquad \text{“primal feasibility”}$$
$$3) \; \boldsymbol{\mu} \geqslant 0, \qquad \text{“dual feasibility”}$$
$$4) \; \boldsymbol{\mu}^T g(\boldsymbol{x}) = 0, \qquad \text{“complementarity”}$$

Augmented Lagrangian Method

We define the **Augmented Lagrangian** as:

$$\mathcal{L}_\rho(\boldsymbol{x}, \boldsymbol{\mu}) = f(\boldsymbol{x}) + \boldsymbol{\mu}^T g(\boldsymbol{x}) + \frac{\rho}{2}\|\max(0, g(\boldsymbol{x}))\|^2$$

$\frac{\rho}{2}\|\max(0, g(\boldsymbol{x}))\|^2$ is called the **penalty** term.

$$\begin{aligned}\nabla_{\boldsymbol{x}}\mathcal{L}_\rho &= \Big(\frac{\partial f}{\partial \boldsymbol{x}} + \boldsymbol{\mu}^T\frac{\partial g}{\partial \boldsymbol{x}} + \rho g(\boldsymbol{x})^T\frac{\partial g}{\partial \boldsymbol{x}}\Big)^T \\ &= \Big(\frac{\partial f}{\partial \boldsymbol{x}} + \Big(\boldsymbol{\mu} + \rho g(\boldsymbol{x})\Big)^T\frac{\partial g}{\partial \boldsymbol{x}}\Big)^T = 0\end{aligned}$$

So we update $\boldsymbol{\mu}$ at each iteration as follows:

$$\boldsymbol{\mu} = \boldsymbol{\mu} + \rho g(\boldsymbol{x})$$

Augmented Lagrangian Method (2)

$$
\begin{aligned}
&1)\ \boldsymbol{x}_{k+1} = \min_{\boldsymbol{x}} \mathcal{L}_\rho(\boldsymbol{x}, \boldsymbol{\mu}_k)\\
&2)\ \boldsymbol{\mu}_{k+1} = \max\Big(\mathbf{0}, \boldsymbol{\mu}_k + \rho g(\boldsymbol{x}_{k+1})\Big)\\
&3)\ \rho = \alpha\rho,\ \text{with } \alpha \in \mathbb{R}^+
\end{aligned}
$$

Usually $\alpha \approx 10$ and we use (2) only when the constraints are violated. The method comes with strong convergence guarantees.

How can we solve (1)?

- We need to solve $\min_{\boldsymbol{x}} \mathcal{L}_\rho(\boldsymbol{x}, \boldsymbol{\mu})$
- Important to note that $\boldsymbol{\mu}$ and ρ are constants!
- Let's apply Newton's Method:

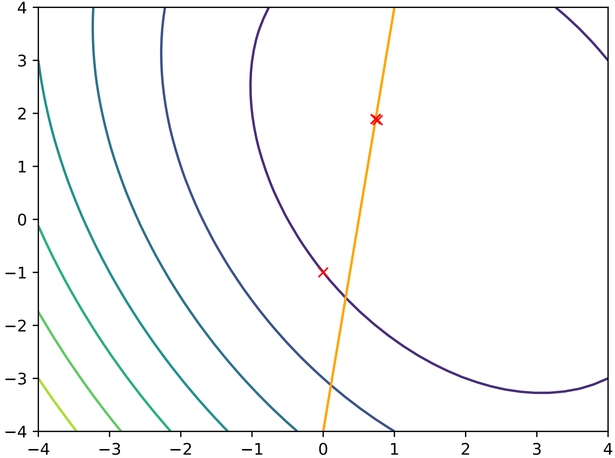
$$\nabla_{\boldsymbol{x}}\mathcal{L}_\rho(\boldsymbol{x}_k + \Delta\boldsymbol{x}, \boldsymbol{\mu}_k) \approx \nabla_{\boldsymbol{x}}\mathcal{L}_\rho(\boldsymbol{x}_k, \boldsymbol{\mu}_k) + \frac{\partial}{\partial \boldsymbol{x}}\Big(\nabla_{\boldsymbol{x}}\mathcal{L}_\rho(\boldsymbol{x}_k, \boldsymbol{\mu}_k)\Big)\Delta\boldsymbol{x} = 0$$

We can use the Gauss-Newton version and:

$$\frac{\partial}{\partial \boldsymbol{x}}\Big(\nabla_{\boldsymbol{x}}\mathcal{L}_\rho(\boldsymbol{x}_k, \boldsymbol{\mu}_k)\Big) \approx \nabla^2_{\boldsymbol{x}\boldsymbol{x}}f(\boldsymbol{x}_k) + \rho\Big(\frac{\partial g}{\partial \boldsymbol{x}}\Big|_{\boldsymbol{x}_k}\Big)^T\frac{\partial g}{\partial \boldsymbol{x}}\Big|_{\boldsymbol{x}_k}$$

$$\Delta\boldsymbol{x} = -\Big(\nabla^2_{\boldsymbol{x}\boldsymbol{x}}f(\boldsymbol{x}_k) + \rho\Big(\frac{\partial g}{\partial \boldsymbol{x}}\Big|_{\boldsymbol{x}_k}\Big)^T\frac{\partial g}{\partial \boldsymbol{x}}\Big|_{\boldsymbol{x}_k}\Big)^{-1}\nabla_{\boldsymbol{x}}\mathcal{L}_\rho(\boldsymbol{x}_k, \boldsymbol{\mu}_k)$$

Augmented Lagrangian Method - Code Example



Full constrained minimization:

$$\begin{aligned}\min_{\boldsymbol{x}}\ & f(\boldsymbol{x})\\ \text{s.t. } & g(\boldsymbol{x}) \leqslant 0\\ & h(\boldsymbol{x}) = 0\end{aligned}$$

The **Lagrangian** is given:

$$\mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}) = f(\boldsymbol{x}) + \boldsymbol{\lambda}^T h(\boldsymbol{x}) + \boldsymbol{\mu}^T g(\boldsymbol{x})$$

First Order Necessary (KKT) Conditions:

1) $\nabla_{\boldsymbol{x}} f(\boldsymbol{x}) + \left(\frac{\partial h}{\partial \boldsymbol{x}}\right)^T \boldsymbol{\lambda} + \left(\frac{\partial g}{\partial \boldsymbol{x}}\right)^T \boldsymbol{\mu} = 0,$ "stationarity"
2) $g(\boldsymbol{x}) \leqslant 0$ and $h(\boldsymbol{x}) = 0,$ "primal feasibility"
3) $\boldsymbol{\mu} \geqslant 0,$ "dual feasibility"
4) $\boldsymbol{\mu}^T g(\boldsymbol{x}) = 0,$ "complementarity"

Merit Functions

- **How can we apply line search in the constrained case?**

Merit Functions

- **How can we apply line search in the constrained case?**
- We define a **merit function**, $P(\boldsymbol{x})$
- Then we apply the Armijo rule to it

$$
\begin{aligned}
&\alpha = 1\\
&\text{while } P(\boldsymbol{x}_k + \alpha\Delta\boldsymbol{x}) > P(\boldsymbol{x}_k) + b\alpha\nabla_{\boldsymbol{x}}P(\boldsymbol{x}_k)^T\Delta\boldsymbol{x} :\\
&\qquad \alpha = c\alpha\\
&\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \alpha\Delta\boldsymbol{x}
\end{aligned}
$$

Usual Merit Functions

- KKT residual

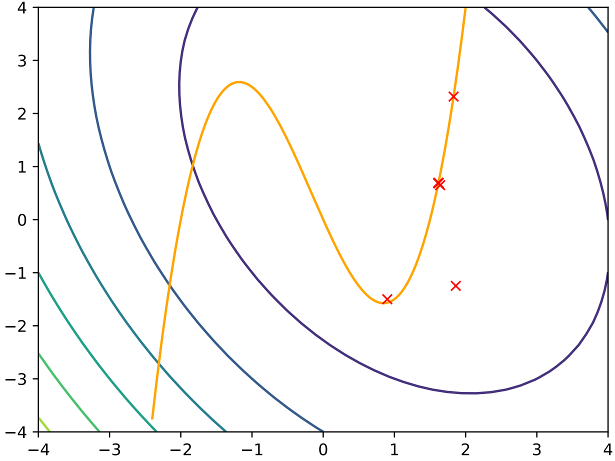
$$P(\boldsymbol{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}) = \frac{1}{2}\left\| \begin{bmatrix} \nabla_{\boldsymbol{x}}\mathcal{L} \\ h(\boldsymbol{x}) \\ \min(0, g(\boldsymbol{x})) \end{bmatrix} \right\|^2$$

- Weighted Sum

$$P(\boldsymbol{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}) = f(\boldsymbol{x}) + \frac{1}{2} w \left\| \begin{bmatrix} h(\boldsymbol{x}) \\ \min(0, g(\boldsymbol{x})) \end{bmatrix} \right\|^2$$

- Augmented Lagrangian

$$P(\boldsymbol{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}) = \mathcal{L}_\rho(\boldsymbol{x}, \boldsymbol{\lambda}, \boldsymbol{\mu})$$


4
3
2
1
0
−1
−2
−3
−4
−4
−3
−2
−1
0
1
2
3
4

Input: Initial $\boldsymbol{x}_0$, $\boldsymbol{\lambda}_0$ (equality), $\boldsymbol{\mu}_0 \geqslant \mathbf{0}$ (inequality); penalties $\rho_{h,0} > 0$, $\rho_{g,0} > 0$; growth factors $\alpha_h, \alpha_g > 1$; tolerances $\varepsilon_{\text{stat}}, \varepsilon_{\text{fe,h}}, \varepsilon_{\text{fe,g}}$; max outer K

Output: Approximate solution $\boldsymbol{x}_k$ to $\min f(\boldsymbol{x})$ s.t. $\boldsymbol{h}(\boldsymbol{x}) = \mathbf{0}$, $\boldsymbol{g}(\boldsymbol{x}) \leqslant \mathbf{0}$

for $k = 0, 1, \ldots, K-1$ **do**

    `// Define the augmented Lagrangian (your notation)`

    $\mathcal{L}_{\rho_h,\rho_g}(\boldsymbol{x}, \boldsymbol{\lambda}_k, \boldsymbol{\mu}_k) = f(\boldsymbol{x}) + \boldsymbol{\lambda}_k^\top \boldsymbol{h}(\boldsymbol{x}) + \frac{\rho_h}{2}\|\boldsymbol{h}(\boldsymbol{x})\|^2 + \boldsymbol{\mu}_k^\top \boldsymbol{g}(\boldsymbol{x}) + \frac{\rho_g}{2}\|[\,\boldsymbol{g}(\boldsymbol{x})\,]_+\|^2$

    `// (1) Primal update: approximately minimize w.r.t.` $\boldsymbol{x}$

    Solve (approximately) $\boldsymbol{x}_{k+1} \approx \arg\min_{\boldsymbol{x}} \mathcal{L}_{\rho_h,\rho_g}(\boldsymbol{x}, \boldsymbol{\lambda}_k, \boldsymbol{\mu}_k)$

    `// Evaluate constraint residuals at` $\boldsymbol{x}_{k+1}$

    $\boldsymbol{h}_{k+1} \leftarrow h(\boldsymbol{x}_{k+1}), \quad \boldsymbol{g}_{k+1} \leftarrow g(\boldsymbol{x}_{k+1})$

    `// (2) Dual updates (project` $\boldsymbol{\mu}$ `onto` $\mathbb{R}_+^m$`)`

    $\boldsymbol{\lambda}_{k+1} \leftarrow \boldsymbol{\lambda}_k + \rho_h\, \boldsymbol{h}_{k+1}$

    $\boldsymbol{\mu}_{k+1} \leftarrow \max\big(\mathbf{0},\ \boldsymbol{\mu}_k + \rho_g\, \boldsymbol{g}_{k+1}\big)$

    `// (3) Penalty updates (e.g. simple multiplicative growth)`

    $\rho_h \leftarrow \alpha_h \rho_h; \quad \rho_g \leftarrow \alpha_g \rho_g$

    `// Convergence (feasibility + (approx.) stationarity)`

    if $\|\boldsymbol{h}_{k+1}\| \leqslant \varepsilon_{fe,h}$ **and** $\|[\,\boldsymbol{g}_{k+1}\,]_+\| \leqslant \varepsilon_{fe,g}$ **and** $\left\|\nabla f(\boldsymbol{x}_{k+1}) + \left(\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{x}}\right)^\top \boldsymbol{\lambda}_{k+1} + \left(\frac{\partial \boldsymbol{g}}{\partial \boldsymbol{x}}\right)^\top \boldsymbol{\mu}_{k+1}\right\| \leqslant \varepsilon_{stat}$ **then**

        break

return $\boldsymbol{x}_k$

Equality-Constrained Convex Optimization

$$\min_{\boldsymbol{x}} \; f(\boldsymbol{x})$$
$$\text{s.t. } \boldsymbol{A}\boldsymbol{x} - \boldsymbol{b} = \boldsymbol{0}$$

where $\boldsymbol{x} \in \mathbb{R}^N$, $\boldsymbol{A} \in \mathbb{R}^{M\times N}$, $\boldsymbol{b} \in \mathbb{R}^M$ and $f : \mathbb{R}^N \rightarrow \mathbb{R}$ is convex.
How do we solve this?

Equality-Constrained Convex Optimization

$$\min_{\boldsymbol{x}} \; f(\boldsymbol{x})$$
$$\text{s.t.} \;\; \boldsymbol{A}\boldsymbol{x} - \boldsymbol{b} = \boldsymbol{0}$$

where $\boldsymbol{x} \in \mathbb{R}^N$, $\boldsymbol{A} \in \mathbb{R}^{M\times N}$, $\boldsymbol{b} \in \mathbb{R}^M$ and $f : \mathbb{R}^N \rightarrow \mathbb{R}$ is convex.
How do we solve this?

We can use the **Augmented Lagrangian:**

$$\mathcal{L}_\rho(\boldsymbol{x}, \boldsymbol{\lambda}) = f(\boldsymbol{x}) + \lambda^T(\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}) + \frac{\rho}{2}\|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\|^2$$

Can we do better?

Equality-Constrained Convex Optimization (2)

It turns out we can if we can split the problem into subparts:

$$\min_{\boldsymbol{y},\boldsymbol{z}} \; f(\boldsymbol{y}) + g(\boldsymbol{z})$$
$$\text{s.t. } \boldsymbol{A}\boldsymbol{y} + \boldsymbol{B}\boldsymbol{z} - \boldsymbol{b} = \boldsymbol{0}$$

where $\boldsymbol{y} \in \mathbb{R}^Q$, $\boldsymbol{z} \in \mathbb{R}^P$, $\boldsymbol{A} \in \mathbb{R}^{M\times Q}$, $\boldsymbol{B} \in \mathbb{R}^{M\times P}$, $\boldsymbol{b} \in \mathbb{R}^M$, $f : \mathbb{R}^Q \rightarrow \mathbb{R}$ is convex and $g : \mathbb{R}^P \rightarrow \mathbb{R}$ is convex. This is possible in many ways, for example one usual case is:

$$\min_{\boldsymbol{y},\boldsymbol{z}} \; f(\boldsymbol{y}) + g(\boldsymbol{z})$$
$$\text{s.t. } \boldsymbol{y} = \boldsymbol{z}$$

Alternating Direction Method of Multipliers:

$$
\begin{aligned}
&1)\ \boldsymbol{y}_{k+1} = \min_{\boldsymbol{y}} \mathcal{L}_\rho(\boldsymbol{y}, \boldsymbol{z}_k, \boldsymbol{\lambda}_k)\\
&2)\ \boldsymbol{z}_{k+1} = \min_{\boldsymbol{z}} \mathcal{L}_\rho(\boldsymbol{y}_{k+1}, \boldsymbol{z}, \boldsymbol{\lambda}_k)\\
&3)\ \boldsymbol{\lambda}_{k+1} = \boldsymbol{\lambda}_k + \rho(\boldsymbol{A}\boldsymbol{y}_{k+1} + \boldsymbol{B}\boldsymbol{z}_{k+1} - \boldsymbol{b})\\
&4)\ \rho = \alpha\rho,\ \text{with } \alpha \in \mathbb{R}^+
\end{aligned}
$$

What is the difference with solving directly with Augmented Lagrangian Method?

Quadratic Programming (QP)

$$\begin{aligned} \min_{\boldsymbol{x}} f(\boldsymbol{x}) &= \frac{1}{2}\boldsymbol{x}^T\boldsymbol{Q}\boldsymbol{x} + \boldsymbol{q}^T\boldsymbol{x} \\ \text{s.t. } \boldsymbol{A}\boldsymbol{x} - \boldsymbol{b} &= \boldsymbol{0} \\ \boldsymbol{C}\boldsymbol{x} - \boldsymbol{d} &\leqslant \boldsymbol{0} \end{aligned}$$

where $\boldsymbol{x}, \boldsymbol{q} \in \mathbb{R}^N$, $\boldsymbol{Q} \succ 0 \in \mathbb{R}^{N\times N}$. Let's solve this with the **Augmented Lagrangian Method:**

Quadratic Programming (QP)

$$\min_{\boldsymbol{x}} f(\boldsymbol{x}) = \frac{1}{2}\boldsymbol{x}^T\boldsymbol{Q}\boldsymbol{x} + \boldsymbol{q}^T\boldsymbol{x}$$
$$\text{s.t. } \boldsymbol{A}\boldsymbol{x} - \boldsymbol{b} = \boldsymbol{0}$$
$$\boldsymbol{C}\boldsymbol{x} - \boldsymbol{d} \leqslant \boldsymbol{0}$$

where $\boldsymbol{x}, \boldsymbol{q} \in \mathbb{R}^N$, $\boldsymbol{Q} \succ 0 \in \mathbb{R}^{N\times N}$. Let's solve this with the **Augmented Lagrangian Method:**

$$\mathcal{L}_\rho(\boldsymbol{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}) = f(\boldsymbol{x}) + \boldsymbol{\lambda}^T(\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}) + \boldsymbol{\mu}^T(\boldsymbol{C}\boldsymbol{x} - \boldsymbol{d}) + \frac{\rho}{2}\|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\|^2 + \frac{\rho}{2}\|\max(\boldsymbol{0}, \boldsymbol{C}\boldsymbol{x} - \boldsymbol{d})\|^2$$

KKT Conditions:

1) $\boldsymbol{Q}\boldsymbol{x} + \boldsymbol{q} + \boldsymbol{A}^T\boldsymbol{\lambda} + \boldsymbol{C}^T\boldsymbol{\mu} = 0$
2) $\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b} = \boldsymbol{0}$ and $\boldsymbol{C}\boldsymbol{x} - \boldsymbol{d} \leqslant \boldsymbol{0}$
3) $\boldsymbol{\mu} \geqslant \boldsymbol{0}$
4) $\boldsymbol{\mu}^T(\boldsymbol{C}\boldsymbol{x} - \boldsymbol{d}) = \boldsymbol{0}$

What other methods are there?

- **Active Set Methods**:
  - Try to guess which constraints are active (equal to zero) or not
  - Solve equality constrained optimization only with active constraints
  - Need good heuristic for identifying potentially active constraints
- **Barrier/Interior Point (IP) Methods:**
  - Add inequality constraints as "barriers"/penalties in the cost function. For example, $f'(\boldsymbol{x}) = f(\boldsymbol{x}) - \frac{1}{\rho}\log(-g(\boldsymbol{x}))$
  - Very good for convex problems

General Constrained Optimization (2)

What other methods are there?

- **Penalty Methods:**
  - Add constraint violation into the cost function. For example, $f'(\boldsymbol{x}) = f(\boldsymbol{x}) + \frac{\rho}{2}\|\max(\mathbf{0}, g(\boldsymbol{x}))\|^2$
  - Similar to IP methods, but there is a key difference: IP methods blow up at the boundary, penalty methods start pushing only if there is a violation!
  - Can easily get ill-conditions
  - Cannot get good accuracy
- **Sequential Quadratic Programming (SQP):**
  - Sequential QP problems
  - General idea:
    - Fit a quadratic approximation of the cost function
    - Linearize the constraints
    - Solve the produced QP with slack variables for feasibility
    - Iterate until convergence

Final Verdict on Optimization

- Newton's Method is very effective; Gauss-Newton the one used mostly in practice
- Line Search helps a lot
- Interior Point Methods are very effective in practice
- Transforming your problem into well known forms (e.g. QP) can greatly help
- A lot of good libraries

Thank you

- **Any Questions?**
- **Office Hours**:
  - **Tue & Thu** *(09:00-11:00)*
  - 24/7 by email (`costashatz@upatras.gr`, subject: *ECE_RSII_AM*)
- **Material and Announcements**

Laboratory of Automation & Robotics