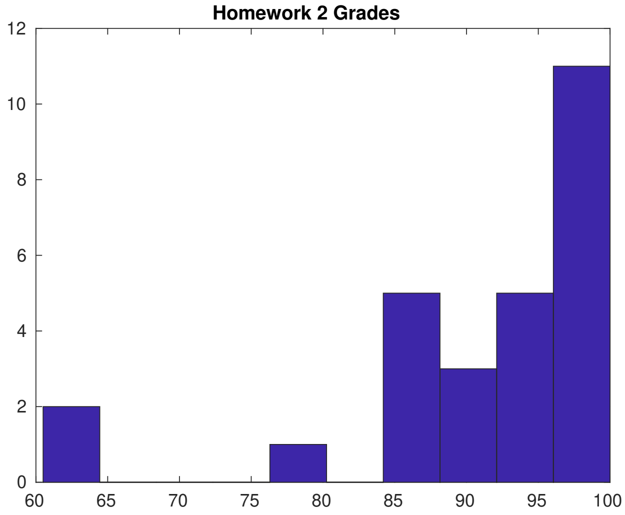
Distributed Optimization I

Convex Optimization

Mathias Hudoba de Badyn

Advanced Topics in Control
May 23, 2022

Announcements



relative state-based control – double-integrator case

Relative state-based control – Double Integrator

Consider the double-integrator control $\ddot{x}_i(t) = u_i(t)$, for all $i \in \mathcal{N}$.

Relative state-based control – Double Integrator

Consider the double-integrator control $\ddot{x}_i(t) = u_i(t)$, for all $i \in \mathcal{N}$.

Suppose that desired relative positions and velocities are defined via a spanning tree digraph $\mathcal{D}$ as,

$$\begin{bmatrix} z_{\text{ref}} \\ \dot{z}_{\text{ref}} \end{bmatrix}.$$

.

Relative state-based control – Double Integrator

Consider the double-integrator control $\ddot{x}_i(t) = u_i(t)$, for all $i \in \mathcal{N}$.

Suppose that desired relative positions and velocities are defined via a spanning tree digraph $\mathcal{D}$ as,

$$\begin{bmatrix} z_{\text{ref}} \\ \dot{z}_{\text{ref}} \end{bmatrix}.$$

We further assume that $\ddot{z}_{\text{ref}} = 0$, i.e. the reference velocity does not change with time.

Relative state-based control – Double Integrator

Again, we set the error as $e(t) = z_{\text{ref}} - D(\mathcal{D})^T x(t)$.

Again, we set the error as $e(t) = z_{\text{ref}} - D(\mathcal{D})^T x(t)$.

We can show that,

$$\ddot{e} = -D(\mathcal{D})^T u(t).$$

Proof:

Define the (proportional-derivative) feedback controller,

$$u(t) = k \begin{bmatrix} D(\mathcal{D}) & D(\mathcal{D}) \end{bmatrix} \begin{bmatrix} e(t) \\ \dot{e}(t) \end{bmatrix}.$$

Relative state-based control – Double Integrator

Define the (proportional-derivative) feedback controller,

$$u(t) = k \begin{bmatrix} D(\mathcal{D}) & D(\mathcal{D}) \end{bmatrix} \begin{bmatrix} e(t) \\ \dot{e}(t) \end{bmatrix}.$$

Then, the closed-loop system is

$$\begin{bmatrix} \dot{e}(t) \\ \ddot{e}(t) \end{bmatrix} = \begin{bmatrix} 0 & I \\ -kL_e(\mathcal{D}) & -kL_e(\mathcal{D}) \end{bmatrix} \begin{bmatrix} e(t) \\ \dot{e}(t) \end{bmatrix}$$

Proof:

Stability of the closed-loop system

Stability of the closed-loop system

continued:

Dynamics of the closed-loop system

Theorem

$$\begin{bmatrix} \dot{x}(t) \\ \ddot{x}(t) \end{bmatrix} = \begin{bmatrix} 0 & I \\ -kL(\tilde{\mathcal{D}}) & -kL(\tilde{\mathcal{D}}) \end{bmatrix} \begin{bmatrix} x(t) \\ \dot{x}(t) \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ kD(\mathcal{D}) & kD(\mathcal{D}) \end{bmatrix} \begin{bmatrix} z_{\mathrm{ref}}(t) \\ \dot{z}_{\mathrm{ref}} \end{bmatrix}$$

Conclusion

1. Shape-based control (Take points Ξ and control towards them)

Conclusion

1. Shape-based control (Take points Ξ and control towards them)
2. Relative state-based control (Define relative states $x_i - x_j$ and control them so $\|x_i - x_j\| = d_{ij}$.

Conclusion

1. Shape-based control (Take points Ξ and control towards them)
2. Relative state-based control (Define relative states $x_i - x_j$ and control them so $\|x_i - x_j\| = d_{ij}$.
3. Single/double integrator versions

recap: convergence of nonlinear systems via LaSalle invariance principle

LaSalle Invariance Principle

Consider a dynamical system $\dot{x} = f(x)$ with differentiable f. Assume that

1. there exists a compact set $W \subset \mathbb{R}^n$ that is invariant for $\dot{x} = f(x)$,

LaSalle Invariance Principle

Consider a dynamical system $\dot{x} = f(x)$ with differentiable f. Assume that

1. there exists a compact set $W \subset \mathbb{R}^n$ that is invariant for $\dot{x} = f(x)$,
2. there exists a continuously-differentiable function $V : \mathbb{R}^n \to \mathbb{R}$ satisfying

$$\frac{d}{dt} V(x(t)) = \frac{\partial V(x)}{\partial x}^T f(x) \leq 0.$$

LaSalle Invariance Principle

Consider a dynamical system $\dot{x} = f(x)$ with differentiable f. Assume that

1. there exists a compact set $W \subset \mathbb{R}^n$ that is invariant for $\dot{x} = f(x)$,
2. there exists a continuously-differentiable function $V : \mathbb{R}^n \to \mathbb{R}$ satisfying

$$\frac{d}{dt}V(x(t)) = \frac{\partial V(x)}{\partial x}^T f(x) \leq 0.$$

Then each solution $t \mapsto x(t)$ starting in W converges to the largest invariant set contained in

$$\left\{x \in W : \frac{\partial V(x)}{\partial x}^T f(x) = 0\right\}$$

LaSalle Invariance Principle

Consider a dynamical system $\dot{x} = f(x)$ with differentiable f. Assume that

1. there exists a compact set $W \subset \mathbb{R}^n$ that is invariant for $\dot{x} = f(x)$,
2. there exists a continuously-differentiable function $V : \mathbb{R}^n \to \mathbb{R}$ satisfying

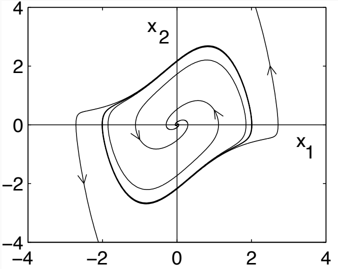
$$\frac{d}{dt} V(x(t)) = \frac{\partial V(x)}{\partial x}^T f(x) \leq 0.$$

Then each solution $t \mapsto x(t)$ starting in W converges to the largest invariant set contained in

$$\left\{ x \in W : \frac{\partial V(x)}{\partial x}^T f(x) = 0 \right\}$$

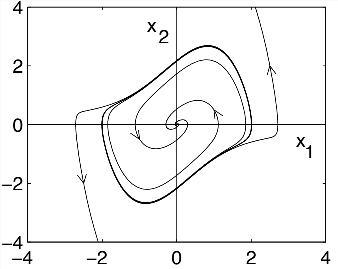
Note: compact and invariant set can often be constructed as a sublevel set of V: $\{x \in \mathbb{R}^n : V(x) \leq \ell\}$ where $\ell = V(x_0)$

$$\dot{x}_1 = -x_2$$
$$\dot{x}_2 = x_1 + (x_1^2 - 1)x_2$$



Example: reverse-time van der Pol oscillator

$$\dot{x}_1 = -x_2$$
$$\dot{x}_2 = x_1 + (x_1^2 - 1)x_2$$

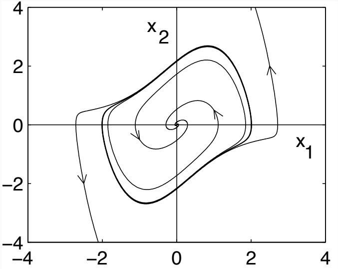


origin **0** is unique **equilibrium** & **linearization** is exp. stable:

$$\dot{x} = \begin{bmatrix} 0 & -1 \\ 1 & -1 \end{bmatrix} x$$

Example: reverse-time van der Pol oscillator

$$\dot{x}_1 = -x_2$$
$$\dot{x}_2 = x_1 + (x_1^2 - 1)x_2$$



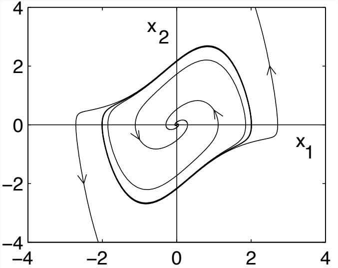
origin **0** is unique **equilibrium** & **linearization** is exp. stable:

$$\dot{x} = \begin{bmatrix} 0 & -1 \\ 1 & -1 \end{bmatrix} x$$

constructing an estimate of the **region of attraction** via LaSalle:

Example: reverse-time van der Pol oscillator

$$\dot{x}_1 = -x_2$$
$$\dot{x}_2 = x_1 + (x_1^2 - 1)x_2$$



origin **0** is unique **equilibrium** & **linearization** is exp. stable:

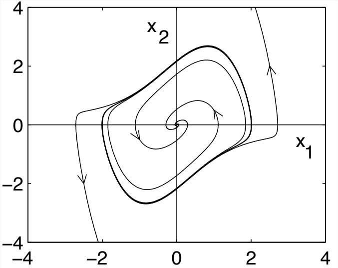
$$\dot{x} = \begin{bmatrix} 0 & -1 \\ 1 & -1 \end{bmatrix} x$$

constructing an estimate of the **region of attraction** via LaSalle:

- choose $V(x) = x_1^2 + x_2^2 = \|x\|^2$

Example: reverse-time van der Pol oscillator

$$\begin{aligned}\dot{x}_1 &= -x_2 \\ \dot{x}_2 &= x_1 + (x_1^2 - 1)x_2\end{aligned}$$



origin **0** is unique **equilibrium** & **linearization** is exp. stable:

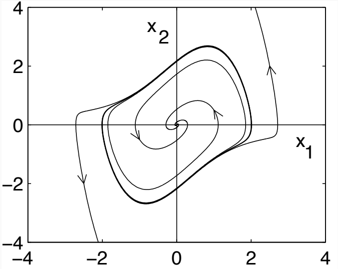
$$\dot{x} = \begin{bmatrix} 0 & -1 \\ 1 & -1 \end{bmatrix} x$$

constructing an estimate of the **region of attraction** via LaSalle:

- choose $V(x) = x_1^2 + x_2^2 = \|x\|^2$
- derivative $\dot{V}(x) = (x_1^2 - 1) \cdot x_2^2$

Example: reverse-time van der Pol oscillator

$$\dot{x}_1 = -x_2$$
$$\dot{x}_2 = x_1 + (x_1^2 - 1)x_2$$



origin **0** is unique **equilibrium** & **linearization** is exp. stable:

$$\dot{x} = \begin{bmatrix} 0 & -1 \\ 1 & -1 \end{bmatrix} x$$

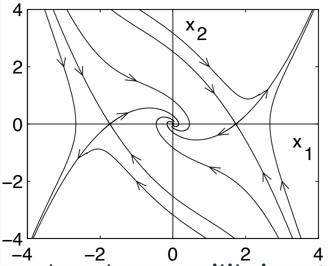
constructing an estimate of the **region of attraction** via LaSalle:

- choose $V(x) = x_1^2 + x_2^2 = \|x\|^2$
- derivative $\dot{V}(x) = (x_1^2 - 1) \cdot x_2^2$

$$= \begin{cases} \leq 0 & \text{if } \|x\| \leq 1 \\ > 0 & \text{else} \end{cases}$$

Example: reverse-time van der Pol oscillator

$$\dot{x}_1 = -x_2$$
$$\dot{x}_2 = x_1 + (x_1^2 - 1)x_2$$



origin **0** is unique **equilibrium** & **linearization** is exp. stable:

$$\dot{x} = \begin{bmatrix} 0 & -1 \\ 1 & -1 \end{bmatrix} x$$

constructing an estimate of the **region of attraction** via LaSalle:

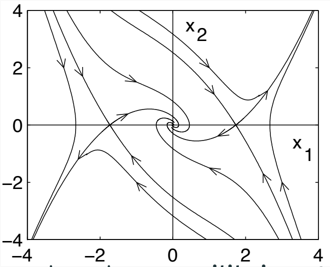
- choose $V(x) = x_1^2 + x_2^2 = \|x\|^2$
- derivative $\dot{V}(x) = (x_1^2 - 1) \cdot x_2^2$

$$= \begin{cases} \leq 0 & \text{if } \|x\| \leq 1 \\ > 0 & \text{else} \end{cases}$$

$\Rightarrow$ $W_\epsilon = \{x : V(x) \leq 1 - \epsilon\}$ for $\epsilon \in]0, 1]$ is compact & invariant

Example: reverse-time van der Pol oscillator

$$\dot{x}_1 = -x_2$$
$$\dot{x}_2 = x_1 + (x_1^2 - 1)x_2$$



origin $\mathbf{0}$ is unique **equilibrium** & **linearization** is exp. stable:

$$\dot{x} = \begin{bmatrix} 0 & -1 \\ 1 & -1 \end{bmatrix} x$$

constructing an estimate of the **region of attraction** via LaSalle:

- choose $V(x) = x_1^2 + x_2^2 = \|x\|^2$
- derivative $\dot{V}(x) = (x_1^2 - 1) \cdot x_2^2$

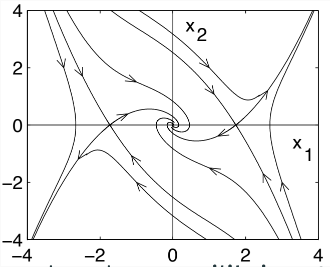
$$= \begin{cases} \leq 0 & \text{if } \|x\| \leq 1 \\ > 0 & \text{else} \end{cases}$$

$\Rightarrow$ $W_\epsilon = \{x : V(x) \leq 1 - \epsilon\}$ for $\epsilon \in]0, 1]$ is compact & invariant

- largest invariant set in $\{x \in W_\epsilon : \dot{V}(x) = 0\}$ is $\mathbf{0}$

Example: reverse-time van der Pol oscillator

$$\dot{x}_1 = -x_2$$
$$\dot{x}_2 = x_1 + (x_1^2 - 1)x_2$$



origin $\mathbf{0}$ is unique **equilibrium** & **linearization** is exp. stable:

$$\dot{x} = \begin{bmatrix} 0 & -1 \\ 1 & -1 \end{bmatrix} x$$

constructing an estimate of the **region of attraction** via LaSalle:

- choose $V(x) = x_1^2 + x_2^2 = \|x\|^2$
- derivative $\dot{V}(x) = (x_1^2 - 1) \cdot x_2^2$

$$= \begin{cases} \leq 0 & \text{if } \|x\| \leq 1 \\ > 0 & \text{else} \end{cases}$$

$\Rightarrow$ $W_\epsilon = \{x : V(x) \leq 1 - \epsilon\}$ for $\epsilon \in]0, 1]$ is compact & invariant

- largest invariant set in $\{x \in W_\epsilon : \dot{V}(x) = 0\}$ is $\mathbf{0}$

$\Rightarrow$ $\mathbf{0}$ is exp. stable with W_ϵ as guaranteed region of attraction

basics of (un)constrained optimization

Unconstrained optimization

- unconstrained optimization:

$$\underset{x \in \mathbb{R}^n}{\text{minimize}}\, f(x)$$

where $f : \mathbb{R}^n \to \mathbb{R}$ is a continuously differentiable *objective* function

Unconstrained optimization

- unconstrained optimization:

$$\underset{x \in \mathbb{R}^n}{\text{minimize}}\, f(x)$$

  where $f : \mathbb{R}^n \to \mathbb{R}$ is a continuously differentiable *objective* function
- $x^\star \in \mathbb{R}^n$ is said to be a **minimizer** and $f(x^\star)$ is a **minimum** if

$$f(x^\star) \leq f(x) \qquad \text{for all } x \in \mathbb{R}^n$$

Unconstrained optimization

- **unconstrained optimization**:

$$\underset{x \in \mathbb{R}^n}{\text{minimize}}\, f(x)$$

  where $f : \mathbb{R}^n \to \mathbb{R}$ is a continuously differentiable *objective* function
- $x^\star \in \mathbb{R}^n$ is said to be a **minimizer** and $f(x^\star)$ is a **minimum** if

$$f(x^\star) \leq f(x) \qquad \text{for all } x \in \mathbb{R}^n$$

- a minimizer and minimum are said to be **strict** if

$$f(x^\star) < f(x) \qquad \text{for all } x \in \mathbb{R}^n \setminus \{x^\star\}\,.$$

Note: all concepts are defined globally, but there are also local versions

Unconstrained optimization

- **unconstrained optimization**:

$$\operatorname*{minimize}_{x \in \mathbb{R}^n} f(x)$$

  where $f : \mathbb{R}^n \to \mathbb{R}$ is a continuously differentiable *objective* function

- $x^\star \in \mathbb{R}^n$ is said to be a **minimizer** and $f(x^\star)$ is a **minimum** if

$$f(x^\star) \leq f(x) \qquad \text{for all } x \in \mathbb{R}^n$$

- a minimizer and minimum are said to be **strict** if

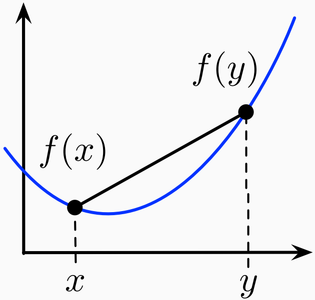
$$f(x^\star) < f(x) \qquad \text{for all } x \in \mathbb{R}^n \setminus \{x^\star\}\,.$$

  Note: all concepts are defined globally, but there are also local versions

- on **gradients**: $\frac{\partial f}{\partial x} : \mathbb{R}^n \to \mathbb{R}^n$ is the *column vector* of partial derivatives

Convexity

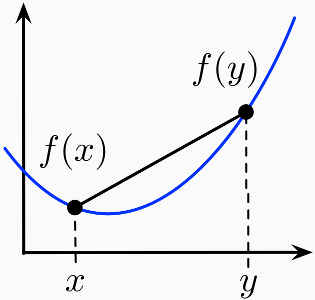
- f is said to be **convex** if "function is below line segment"



$$\forall x, y \in \mathbb{R}^n \text{ and } \forall \alpha, \beta \geq 0 \text{ with } \alpha + \beta = 1$$

$$f(\alpha x + \beta y) \leq \alpha f(x) + \beta f(y)$$

- f is said to be **convex** if "function is below line segment"



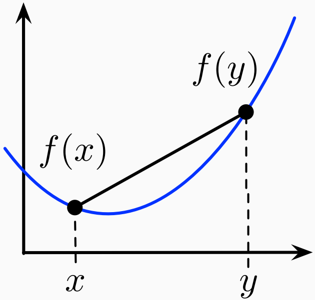
$$\forall x, y \in \mathbb{R}^n \text{ and } \forall \alpha, \beta \geq 0 \text{ with } \alpha + \beta = 1$$

$$f(\alpha x + \beta y) \leq \alpha f(x) + \beta f(y)$$

$\Rightarrow$ f is **strictly convex** if inequality is strict

Convexity

- f is said to be **convex** if "function is below line segment"



$$\forall x, y \in \mathbb{R}^n \text{ and } \forall \alpha, \beta \geq 0 \text{ with } \alpha + \beta = 1$$

$$f(\alpha x + \beta y) \leq \alpha f(x) + \beta f(y)$$

$\Rightarrow$ f is **strictly convex** if inequality is strict

$\Rightarrow$ f is **concave** if $-f$ is convex

Convexity

- **underestimator property**: "convex iff function above its tangents"

a cont. diff. function f is convex
if and only if for all $x, y \in \mathbb{R}^n$

$$f(y) \geq f(x) + (y - x)^\top \frac{\partial f(x)}{\partial x}$$

(strictly convex if inequality is strict for $x \neq y$)

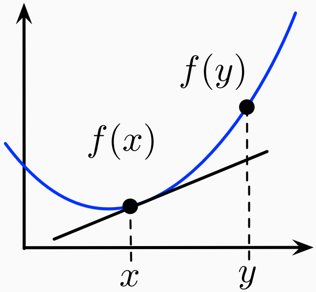
Convexity

- **underestimator property**: "convex iff function above its tangents"

a cont. diff. function f is convex
if and only if for all $x, y \in \mathbb{R}^n$

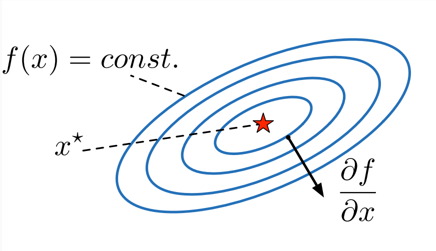
$$f(y) \geq f(x) + (y - x)^\top \frac{\partial f(x)}{\partial x}$$

(strictly convex if inequality is strict for $x \neq y$)

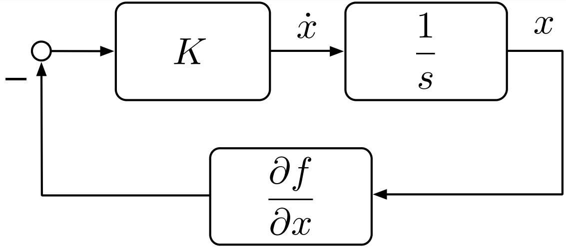


Unconstrained optimization & gradient flows

$$\dot{x} = -K\frac{\partial f(x)}{\partial x}$$



geometry of an unconstrained optimization problem



block-diagram of negative gradient flow with positive definite gain $K \in \mathbb{R}^{n\times n}$

Constrained optimization

- linearly-constrained optimization:

$$\begin{array}{ll} \mathrm{minimize}_{x\in\mathbb{R}^n} & f(x) \\ \text{s.t.} & g(x) = \mathbf{0} \end{array}$$

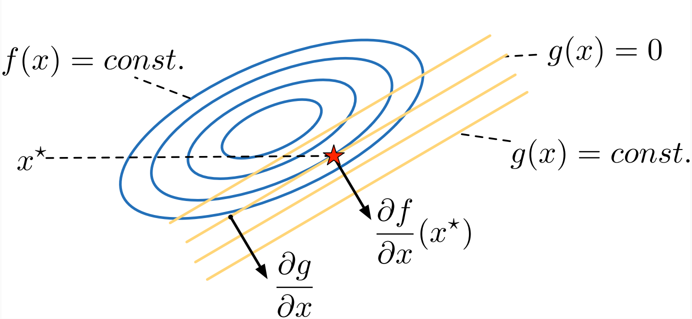
  where f is continuously differentiable & $g(x) = Ax - b$ is **linear-affine**

Constrained optimization

- linearly-constrained optimization:

$$\begin{array}{ll} \text{minimize}_{x \in \mathbb{R}^n} & f(x) \\ \text{s.t.} & g(x) = \mathbf{0} \end{array}$$

where f is continuously differentiable & $g(x) = Ax - b$ is **linear-affine**

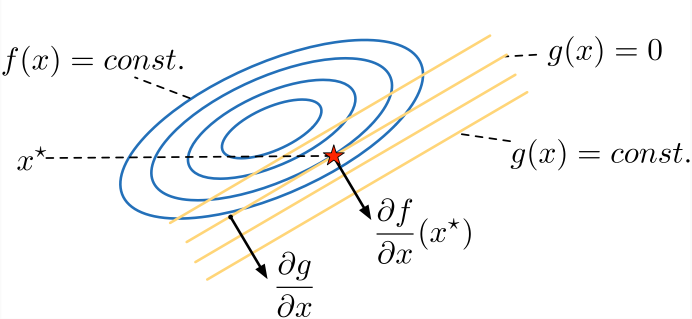


Constrained optimization

- linearly-constrained optimization:

$$\begin{aligned} \text{minimize}_{x\in\mathbb{R}^n} \quad & f(x) \\ \text{s.t.} \quad & g(x) = 0 \end{aligned}$$

where f is continuously differentiable & $g(x) = Ax - b$ is **linear-affine**



$\Rightarrow$ minimizer $x^\star$ when $\{g(x) = 0\}$ **tangent** to $\{f(x) = const.\}$ $\Leftrightarrow$ $\frac{\partial f}{\partial x} \parallel \frac{\partial g}{\partial x}$

- **linearly-constrained optimization:**

$$\begin{aligned} \mathrm{minimize}_{x\in\mathbb{R}^n} \quad & f(x) \\ \text{s.t.} \quad & g(x) = \mathbf{0} \end{aligned}$$

where f is continuously differentiable & $g(x) = Ax - b$ is linear-affine

$\Rightarrow$ minimizer $x^\star$ satisfies **KKT conditions** (after Karush, Kuhn, Tucker):

KKT Conditions

$$(i)\ \ \text{constraint equation: } \mathbf{0} = g(x) = Ax - b$$

$$(ii)\ \ \text{tangency condition: } \mathbf{0} = \frac{\partial f(x)}{\partial x} + \frac{\partial g(x)}{\partial x}^\top \lambda,$$

where λ is Lagrange multiplier or *dual variable* (analogous: x is *primal variable*)

Constrained Optimization

- linearly-constrained optimization:

$$\begin{aligned} \text{minimize}_{x\in\mathbb{R}^n} \quad & f(x) \\ \text{s.t.} \quad & g(x) := Ax - b = \mathbf{0} \end{aligned}$$

KKT Conditions

$$\begin{aligned} (i) \quad & \text{constraint equation: } \mathbf{0} = g(x) = Ax - b \\ (ii) \quad & \text{tangency condition: } \mathbf{0} = \frac{\partial f(x)}{\partial x} + \frac{\partial g(x)}{\partial x}^\top \lambda \end{aligned}$$

1. the KKT conditions are necessary conditions for optimality of $x^\star$

Constrained Optimization

- **linearly-constrained optimization:**

$$\begin{aligned} \text{minimize}_{x \in \mathbb{R}^n} \quad & f(x) \\ \text{s.t.} \quad & g(x) := Ax - b = \mathbf{0} \end{aligned}$$

KKT Conditions

(i) constraint equation: $\mathbf{0} = g(x) = Ax - b$

(ii) tangency condition: $\mathbf{0} = \frac{\partial f(x)}{\partial x} + \frac{\partial g(x)}{\partial x}^{\top} \lambda$

1. the KKT conditions are necessary conditions for optimality of $x^\star$
2. if f is convex, then the KKT conditions are sufficient for optimality, and their solutions $(x^\star, \lambda^\star)$ specify all minimizers and optimal dual variables

Constrained Optimization

- linearly-constrained optimization:

$$\begin{aligned} \operatorname{minimize}_{x \in \mathbb{R}^n} \quad & f(x) \\ \text{s.t.} \quad & g(x) := Ax - b = \mathbf{0} \end{aligned}$$

KKT Conditions

$$\begin{aligned} &(i) \;\; \text{constraint equation: } \mathbf{0} = g(x) = Ax - b \\ &(ii) \;\; \text{tangency condition: } \mathbf{0} = \frac{\partial f(x)}{\partial x} + \frac{\partial g(x)}{\partial x}^{\top} \lambda \end{aligned}$$

1. the KKT conditions are necessary conditions for optimality of $x^\star$
2. if f is convex, then the KKT conditions are sufficient for optimality, and their solutions $(x^\star, \lambda^\star)$ specify all minimizers and optimal dual variables
3. if f is strictly convex and A has full rank, then the KKT conditions admit only a single solution $(x^\star, \lambda^\star)$

Constrained Optimization

- **linearly-constrained optimization:**

$$\begin{aligned} \mathrm{minimize}_{x\in\mathbb{R}^n} \quad & f(x) \\ \text{s.t.} \quad & g(x) := Ax - b = \mathbf{0} \end{aligned}$$

where f is cont. diff. & **convex** and $g(x) = Ax - b$ is linear-affine

$\Rightarrow$ minimizer $x^\star$ and multiplier $\lambda^\star$ from **KKT conditions:**

$$(i) \ \text{constraint equation: } \mathbf{0} = g(x) = Ax - b$$

$$(ii) \ \text{tangency condition: } \mathbf{0} = \frac{\partial f(x)}{\partial x} + \frac{\partial g(x)}{\partial x}^\top \lambda$$

Constrained Optimization

- linearly-constrained optimization:

$$\begin{array}{ll} \mathrm{minimize}_{x\in\mathbb{R}^n} & f(x) \\ \text{s.t.} & g(x) := Ax - b = \mathbf{0} \end{array}$$

where f is cont. diff. & **convex** and $g(x) = Ax - b$ is linear-affine

$\Rightarrow$ minimizer $x^\star$ and multiplier $\lambda^\star$ from **KKT conditions:**

$$(i)\ \ \text{constraint equation: } \mathbf{0} = g(x) = Ax - b$$

$$(ii)\ \ \text{tangency condition: } \mathbf{0} = \frac{\partial f(x)}{\partial x} + \frac{\partial g(x)}{\partial x}^\top \lambda$$

- **Lagrangian:** $\mathcal{L}(x, \lambda) = f(x) + \lambda^\top g(x)$ is convex in x and concave in λ

Constrained Optimization

- **linearly-constrained optimization:**

$$\begin{array}{ll} \operatorname{minimize}_{x\in\mathbb{R}^n} & f(x) \\ \text{s.t.} & g(x) := Ax - b = \mathbf{0} \end{array}$$

where f is cont. diff. & **convex** and $g(x) = Ax - b$ is linear-affine

$\Rightarrow$ minimizer $x^\star$ and multiplier $\lambda^\star$ from **KKT conditions:**

$$(i)\ \text{ constraint equation: } \mathbf{0} = g(x) = Ax - b$$

$$(ii)\ \text{ tangency condition: } \mathbf{0} = \frac{\partial f(x)}{\partial x} + \frac{\partial g(x)}{\partial x}^\top \lambda$$

- **Lagrangian:** $\mathcal{L}(x, \lambda) = f(x) + \lambda^\top g(x)$ is convex in x and concave in λ

$\Rightarrow$ $\mathcal{L}(x, \lambda)$ has **saddle points**: $\mathcal{L}(x^\star, \lambda) \le \mathcal{L}(x^\star, \lambda^\star) \le \mathcal{L}(x, \lambda^\star)$

Constrained Optimization

- **linearly-constrained optimization:**

$$\begin{aligned} &\text{minimize}_{x\in\mathbb{R}^n} && f(x) \\ &\text{s.t.} && g(x) := Ax - b = \mathbf{0} \end{aligned}$$

where f is cont. diff. & **convex** and $g(x) = Ax - b$ is linear-affine

$\Rightarrow$ minimizer $x^\star$ and multiplier $\lambda^\star$ from **KKT conditions:**

$$(i)\ \text{constraint equation: } \mathbf{0} = g(x) = Ax - b$$

$$(ii)\ \text{tangency condition: } \mathbf{0} = \frac{\partial f(x)}{\partial x} + \frac{\partial g(x)}{\partial x}^\top \lambda$$

- **Lagrangian:** $\mathcal{L}(x, \lambda) = f(x) + \lambda^\top g(x)$ is convex in x and concave in λ

$\Rightarrow$ $\mathcal{L}(x, \lambda)$ has **saddle points**: $\mathcal{L}(x^\star, \lambda) \leq \mathcal{L}(x^\star, \lambda^\star) \leq \mathcal{L}(x, \lambda^\star)$

$\Rightarrow$ saddle points = { minimizer $x^\star$ and multiplier $\lambda^\star$ } from KKT:

Constrained Optimization

- **linearly-constrained optimization:**

$$\begin{aligned} \text{minimize}_{x\in\mathbb{R}^n} \quad & f(x) \\ \text{s.t.} \quad & g(x) := Ax - b = \mathbf{0} \end{aligned}$$

where f is cont. diff. & **convex** and $g(x) = Ax - b$ is linear-affine

$\Rightarrow$ minimizer $x^\star$ and multiplier $\lambda^\star$ from **KKT conditions:**

$$(i) \;\; \text{constraint equation: } \mathbf{0} = g(x) = Ax - b$$

$$(ii) \;\; \text{tangency condition: } \mathbf{0} = \frac{\partial f(x)}{\partial x} + \frac{\partial g(x)}{\partial x}^\top \lambda$$

- **Lagrangian:** $\mathcal{L}(x,\lambda) = f(x) + \lambda^\top g(x)$ is convex in x and concave in λ

$\Rightarrow$ $\mathcal{L}(x,\lambda)$ has **saddle points**: $\mathcal{L}(x^\star,\lambda) \leq \mathcal{L}(x^\star,\lambda^\star) \leq \mathcal{L}(x,\lambda^\star)$

$\Rightarrow$ saddle points = { minimizer $x^\star$ and multiplier $\lambda^\star$ } from KKT:

$$\mathbf{0} = \frac{\partial \mathcal{L}(x,\lambda)}{\partial x} = \frac{\partial f(x)}{\partial x} + \frac{\partial g(x)}{\partial x}^\top \lambda \qquad \mathbf{0} = \frac{\partial \mathcal{L}(x,\lambda)}{\partial \lambda} = g(x)$$

Linear Quadratic Case

Recall useful matrix lemma

Lemma: Consider a positive semidefinite matrix $P \in \mathbb{R}^{n\times n}$ and a matrix $A \in \mathbb{R}^{m\times n}$ with $n \geq m$ forming the composite **saddle matrix**

$$\mathcal{A} = \begin{bmatrix} -P & -A^\top \\ A & \mathbf{0}_{m\times m} \end{bmatrix}.$$

The matrix $\mathcal{A}$ has the following properties:

1) all eigenvalues are in the closed left half-plane: $\mathrm{spec}(\mathcal{A}) = \{\lambda \in \mathbb{C} \mid \mathcal{R}(\lambda) \leq 0\}$. Moreover, all eigenvalues on the imaginary axis have equal algebraic and geometric multiplicity;

Recall useful matrix lemma

Lemma: Consider a positive semidefinite matrix $P \in \mathbb{R}^{n\times n}$ and a matrix $A \in \mathbb{R}^{m\times n}$ with $n \geq m$ forming the composite **saddle matrix**

$$\mathcal{A} = \begin{bmatrix} -P & -A^\top \\ A & \mathbf{0}_{m\times m} \end{bmatrix}.$$

The matrix $\mathcal{A}$ has the following properties:

1) all eigenvalues are in the closed left half-plane: $\mathrm{spec}(\mathcal{A}) = \{\lambda \in \mathbb{C} \mid \mathcal{R}(\lambda) \leq 0\}$. Moreover, all eigenvalues on the imaginary axis have equal algebraic and geometric multiplicity;
2) if $\mathrm{kernel}(P) \cap \mathrm{image}(A^\top) \subseteq \{\mathbf{0}_n\}$, then $\mathcal{A}$ has no eigenvalues on the imaginary axis except for 0; and

Recall useful matrix lemma

Lemma: Consider a positive semidefinite matrix $P \in \mathbb{R}^{n \times n}$ and a matrix $A \in \mathbb{R}^{m \times n}$ with $n \geq m$ forming the composite **saddle matrix**

$$\mathcal{A} = \begin{bmatrix} -P & -A^\top \\ A & \mathbf{0}_{m \times m} \end{bmatrix}.$$

The matrix $\mathcal{A}$ has the following properties:

1) all eigenvalues are in the closed left half-plane: $\text{spec}(\mathcal{A}) = \{\lambda \in \mathbb{C} \mid \mathcal{R}(\lambda) \leq 0\}$. Moreover, all eigenvalues on the imaginary axis have equal algebraic and geometric multiplicity;
2) if $\mathrm{kernel}(P) \cap \mathrm{image}(A^\top) \subseteq \{\mathbf{0}_n\}$, then $\mathcal{A}$ has no eigenvalues on the imaginary axis except for 0; and
3) if P is positive definite and A has full rank, then $\mathcal{A}$ has no eigenvalues on the imaginary axis.

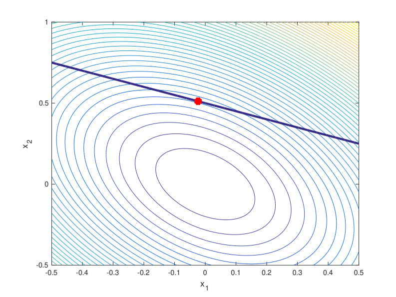
Running example for linear quadratic (LQ) case

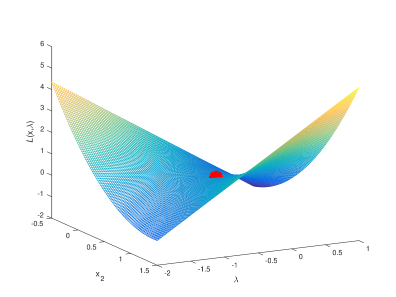
$$\underset{x\in\mathbb{R}^n}{\text{minimize}}\ f(x) = \frac{1}{2}(x-\underline{x})^\top P(x-\underline{x}) \quad \text{subject to} \quad g(x) = Ax - b = 0$$

data: $P = \begin{bmatrix} 2.6 & 0.8 \\ 0.8 & 1.4 \end{bmatrix}$, $\underline{x} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$, $A = \begin{bmatrix} 1 & 2 \end{bmatrix}$, $b = 1$

facts: P is positive definite with eigenvalues $\{1, 3\}$, optimizer

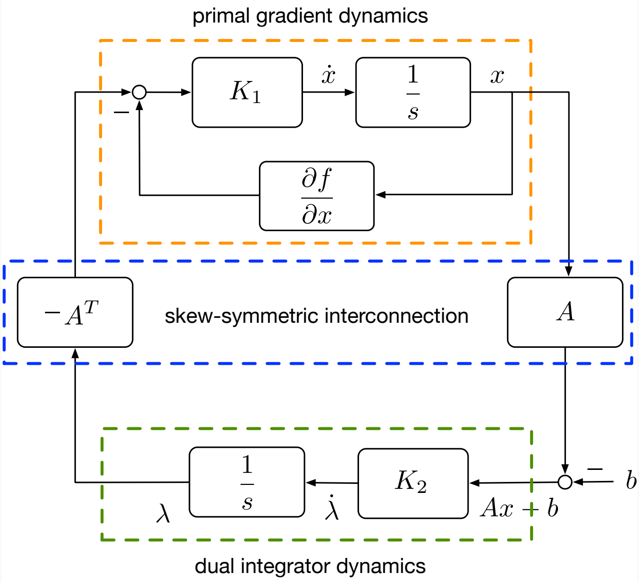
$x^\star = [-0.0233\ 0.5116]^\top$, and optimal multiplier $\lambda^\star = -0.3488$





Primal-Dual Saddle-Point Flow

Block-diagram of primal-dual saddle-point flow

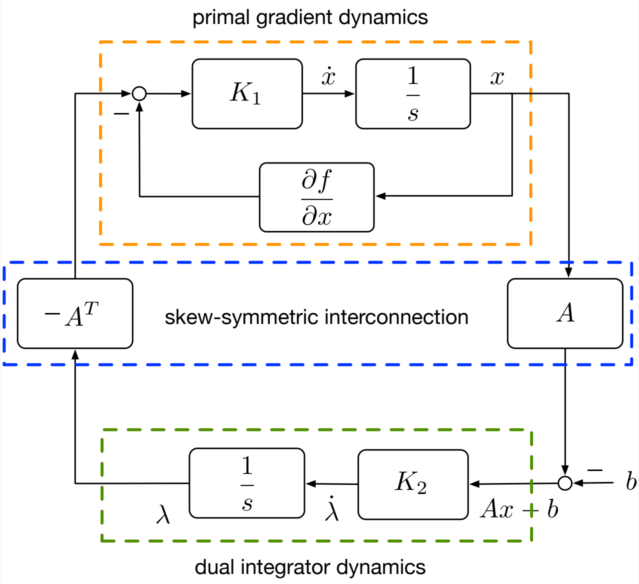


$$\dot{x} = -K_1 \frac{\partial f(x)}{\partial x} - K_1 A^\top \lambda$$

$$\dot{\lambda} = K_2 \, (Ax - b)$$

1. primal gradient descent

Block-diagram of primal-dual saddle-point flow

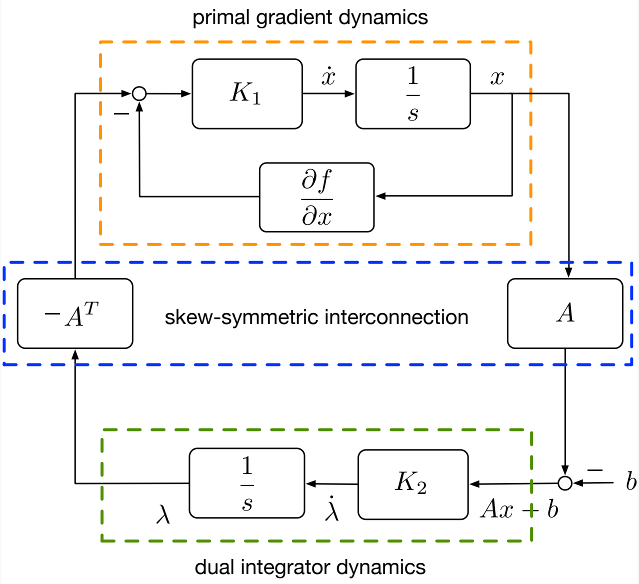


$$\dot{x} = -K_1 \frac{\partial f(x)}{\partial x} - K_1 A^\top \lambda$$

$$\dot{\lambda} = K_2 (Ax - b)$$

1. primal gradient descent
2. dual integral control penalizing constraint violation

Block-diagram of primal-dual saddle-point flow



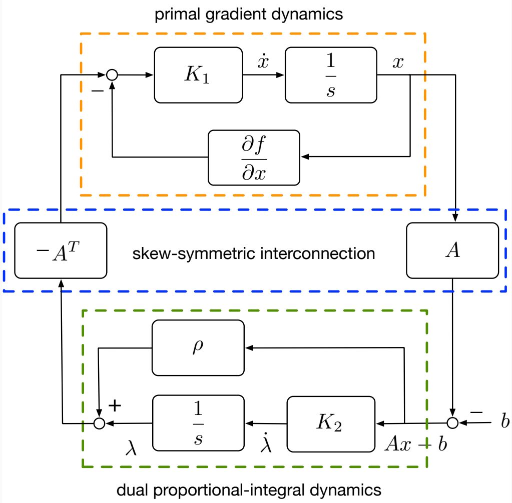
$$\dot{x} = -K_1 \frac{\partial f(x)}{\partial x} - K_1 A^\top \lambda$$

$$\dot{\lambda} = K_2 (Ax - b)$$

1. primal gradient descent
2. dual integral control penalizing constraint violation
3. skew-symmetric interconnection

Augmented Primal-Dual Saddle-Point Flow

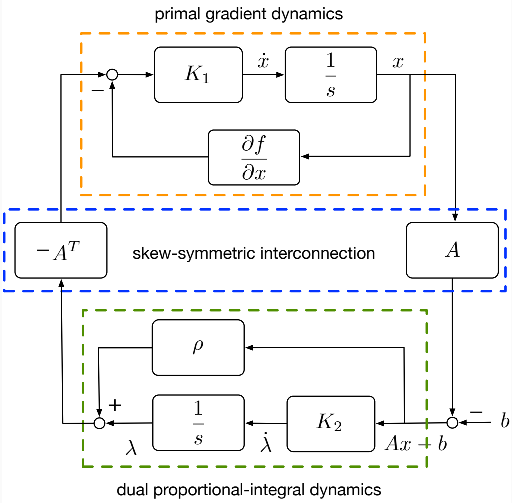
Block-diagram of augmented primal-dual saddle-point flow



$$\dot{x} = -K_1 \frac{\partial f(x)}{\partial x} - K_1 A^\top \lambda$$
$$-\rho\, K_1 A^T (Ax - b)$$
$$\dot{\lambda} = K_2\,(Ax - b)$$

1. primal gradient descent

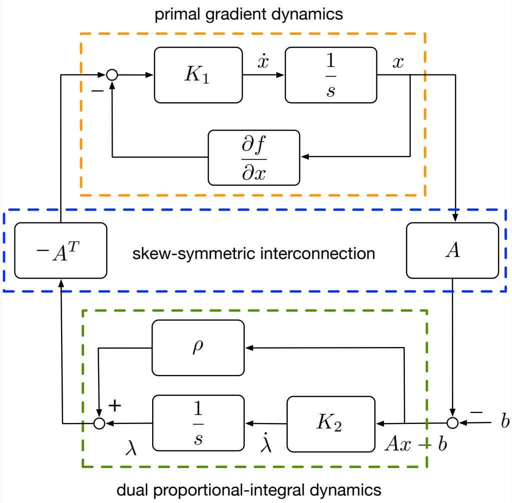
Block-diagram of augmented primal-dual saddle-point flow



$$\dot{x} = -K_1 \frac{\partial f(x)}{\partial x} - K_1 A^\top \lambda$$
$$-\rho K_1 A^T (Ax - b)$$
$$\dot{\lambda} = K_2 (Ax - b)$$

1. primal gradient descent
2. dual proportional-integral control penalizing constraint violation

Block-diagram of augmented primal-dual saddle-point flow

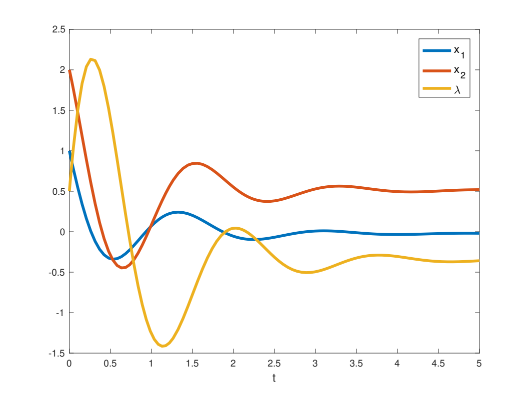


$$\dot{x} = -K_1 \frac{\partial f(x)}{\partial x} - K_1 A^\top \lambda$$
$$-\rho K_1 A^T (Ax - b)$$
$$\dot{\lambda} = K_2 (Ax - b)$$

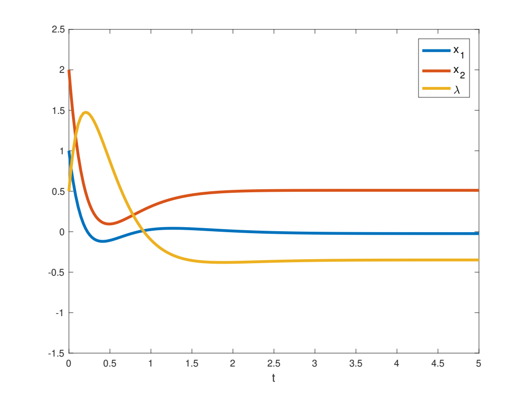
1. primal gradient descent
2. dual proportional-integral control penalizing constraint violation
3. skew-symmetric interconnection

Linear-Quadratic Case

Standard and augmented saddle-point flow for LQ program



standard saddle-point flow



augmented saddle-point flow ($\rho = 1$)

augmentation induces (stricter) convexity & better performance (damping) more algorithms in the next lecture

distributed optimization

Setup in distributed optimization

1. **basic problem:** n distributed agents want to solve

$$\begin{array}{ll} \text{minimize}_{y \in \mathbb{R}} & \sum_{i=1}^{n} f_i(x) \\ \text{subject to} & \mathcal{C}(x) \end{array}$$

1. **knowledge:** $f_i : \mathbb{R} \to \mathbb{R}$ is private cost function known only to agent i

Setup in distributed optimization

1. **basic problem:** n distributed agents want to solve

$$\begin{array}{ll} \text{minimize}_{y \in \mathbb{R}} & \sum_{i=1}^{n} f_i(x) \\ \text{subject to} & \mathcal{C}(x) \end{array}$$

1. **knowledge:** $f_i : \mathbb{R} \to \mathbb{R}$ is private cost function known only to agent i
2. **assume:** $f_i(y)$ is continuously differentiable and strictly convex

Setup in distributed optimization

1. **basic problem:** n distributed agents want to solve

$$\begin{array}{ll} \text{minimize}_{y\in\mathbb{R}} & \sum_{i=1}^{n} f_i(x) \\ \text{subject to} & \mathcal{C}(x) \end{array}$$

1. **knowledge:** $f_i : \mathbb{R} \to \mathbb{R}$ is private cost function known only to agent i
2. **assume:** $f_i(y)$ is continuously differentiable and strictly convex

$\Rightarrow$ since y is a *global* variable, the agents need to **coordinate**

Setup in distributed optimization

1. **basic problem:** n distributed agents want to solve

$$\begin{array}{ll} \text{minimize}_{y \in \mathbb{R}} & \sum_{i=1}^{n} f_i(x) \\ \text{subject to} & \mathcal{C}(x) \end{array}$$

1. **knowledge:** $f_i : \mathbb{R} \to \mathbb{R}$ is private cost function known only to agent i
2. **assume:** $f_i(y)$ is continuously differentiable and strictly convex

$\Rightarrow$ since y is a *global* variable, the agents need to **coordinate**

3. **communication:** undirected and connected communication graph with n nodes (processors) and m edges (communication links)

Setup in distributed optimization

1. **basic problem:** n distributed agents want to solve

$$\begin{array}{ll} \text{minimize}_{y \in \mathbb{R}} & \sum_{i=1}^{n} f_i(x) \\ \text{subject to} & \mathcal{C}(x) \end{array}$$

1. **knowledge:** $f_i : \mathbb{R} \to \mathbb{R}$ is private cost function known only to agent i
2. **assume:** $f_i(y)$ is continuously differentiable and strictly convex

$\Rightarrow$ since y is a *global* variable, the agents need to **coordinate**

3. **communication:** undirected and connected communication graph with n nodes (processors) and m edges (communication links)

$\Rightarrow$ **key idea:** local copies of global variable x and consensus constraint

Distributed Optimization

Saddle points of distributed optimization problem

Consider the **augmented Lagrangian** $\mathcal{L}(y,\lambda) = \tilde{f}(y) + \lambda^T L y + \frac{1}{2} y^T L y$

and its set of **saddle points** $(y^\star, \lambda^\star) \in \mathbb{R}^n \times \mathbb{R}^n$

$$\mathbf{0} = \frac{\partial \mathcal{L}(y,\lambda)}{\partial y} = \frac{\partial \tilde{f}(y)}{\partial y} + Ly + L\lambda \qquad \mathbf{0} = \frac{\partial \mathcal{L}(y,\lambda)}{\partial \lambda} = Ly$$

Saddle points of distributed optimization problem

Consider the **augmented Lagrangian** $\mathcal{L}(y,\lambda) = \tilde{f}(y) + \lambda^T L y + \frac{1}{2} y^T L y$

and its set of **saddle points** $(y^\star, \lambda^\star) \in \mathbb{R}^n \times \mathbb{R}^n$

$$\mathbf{0} = \frac{\partial \mathcal{L}(y,\lambda)}{\partial y} = \frac{\partial \tilde{f}(y)}{\partial y} + Ly + L\lambda \qquad \mathbf{0} = \frac{\partial \mathcal{L}(y,\lambda)}{\partial \lambda} = Ly$$

Lemma: Properties of saddle points

1. **symmetry:** if $(y^\star, \lambda^\star) \in \mathbb{R}^n \times \mathbb{R}^n$ is a saddle point of $\mathcal{L}$, then so is $(y^\star, \lambda^\star + \alpha \mathbf{1})$ for any $\alpha \in \mathbb{R}$.

Saddle points of distributed optimization problem

Consider the **augmented Lagrangian** $\mathcal{L}(y,\lambda) = \tilde{f}(y) + \lambda^T L y + \frac{1}{2} y^T L y$

and its set of **saddle points** $(y^\star, \lambda^\star) \in \mathbb{R}^n \times \mathbb{R}^n$

$$\mathbf{0} = \frac{\partial \mathcal{L}(y,\lambda)}{\partial y} = \frac{\partial \tilde{f}(y)}{\partial y} + Ly + L\lambda \qquad \mathbf{0} = \frac{\partial \mathcal{L}(y,\lambda)}{\partial \lambda} = Ly$$

Lemma: Properties of saddle points

1. **symmetry:** if $(y^\star, \lambda^\star) \in \mathbb{R}^n \times \mathbb{R}^n$ is a saddle point of $\mathcal{L}$, then so is $(y^\star, \lambda^\star + \alpha\mathbf{1})$ for any $\alpha \in \mathbb{R}$.
2. **optimality:** if $(y^\star, \lambda^\star) \in \mathbb{R}^n \times \mathbb{R}^n$ is a saddle point, then $y^\star = x^\star\mathbf{1}$ where $x^\star \in \mathbb{R}$ is a solution of the original optimization problem.

Saddle points of distributed optimization problem

Consider the **augmented Lagrangian** $\mathcal{L}(y, \lambda) = \tilde{f}(y) + \lambda^T L y + \frac{1}{2} y^T L y$

and its set of **saddle points** $(y^\star, \lambda^\star) \in \mathbb{R}^n \times \mathbb{R}^n$

$$\mathbf{0} = \frac{\partial \mathcal{L}(y,\lambda)}{\partial y} = \frac{\partial \tilde{f}(y)}{\partial y} + Ly + L\lambda \qquad\qquad \mathbf{0} = \frac{\partial \mathcal{L}(y,\lambda)}{\partial \lambda} = Ly$$

Lemma: Properties of saddle points

1. **symmetry:** if $(y^\star, \lambda^\star) \in \mathbb{R}^n \times \mathbb{R}^n$ is a saddle point of $\mathcal{L}$, then so is $(y^\star, \lambda^\star + \alpha \mathbf{1})$ for any $\alpha \in \mathbb{R}$.
2. **optimality:** if $(y^\star, \lambda^\star) \in \mathbb{R}^n \times \mathbb{R}^n$ is a saddle point, then $y^\star = x^\star \mathbf{1}$ where $x^\star \in \mathbb{R}$ is a solution of the original optimization problem.
3. **inverse optimality:** if $x^\star \in \mathbb{R}$ is a solution of the original optimization problem, then there are $\lambda^\star \in \mathbb{R}^n$ and $y^\star = x^\star \mathbf{1}$ satisfying $L\lambda^\star + \frac{\partial \tilde{f}(y^\star)}{\partial y} = \mathbf{0}$ so that $(y^\star, \lambda^\star)$ is a saddle point of $\mathcal{L}$.

Distributed saddle-point flow

The **saddle-point flow** associated with $\mathcal{L}(y,\lambda) = \tilde{f}(y) + \frac{1}{2}y^T L y + \lambda^T L y$ is

$$\dot{y} = -\frac{\partial \mathcal{L}(y,\lambda)}{\partial y} = -\frac{\partial \tilde{f}(y)}{\partial y} - Ly - L\lambda \qquad \dot{\lambda} = +\frac{\partial \mathcal{L}(y,\lambda)}{\partial \lambda} = +Ly$$

Distributed saddle-point flow

The **saddle-point flow** associated with $\mathcal{L}(y,\lambda) = \tilde{f}(y) + \frac{1}{2}y^T L y + \lambda^T L y$ is

$$\dot{y} = -\frac{\partial \mathcal{L}(y,\lambda)}{\partial y} = -\frac{\partial \tilde{f}(y)}{\partial y} - Ly - L\lambda \qquad \dot{\lambda} = +\frac{\partial \mathcal{L}(y,\lambda)}{\partial \lambda} = +Ly$$

For processor i the saddle-point dynamics **read component-wise** as

$$\begin{aligned}\dot{y}_i &= -\frac{\partial \tilde{f}_i(y_i)}{\partial y_i} - \sum\nolimits_{j=1}^{n} a_{ij}(y_i - y_j) - \sum\nolimits_{j=1}^{n} a_{ij}(\lambda_i - \lambda_j)\\ \dot{\lambda}_i &= \sum\nolimits_{j=1}^{n} a_{ij}(y_i - y_j)\,.\end{aligned}$$

Distributed saddle-point flow

For processor i:

$$\dot{y}_i = -\frac{\partial \tilde{f}_i(y_i)}{\partial y_i} - \sum_{j=1}^{n} a_{ij}(y_i - y_j) - \sum_{j=1}^{n} a_{ij}(\lambda_i - \lambda_j)$$

$$\dot{\lambda}_i = \sum_{j=1}^{n} a_{ij}(y_i - y_j)\,.$$

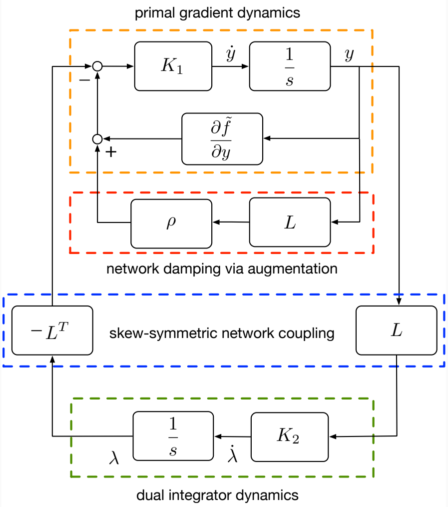
Distributed saddle-point flow

For processor i:

$$\dot{y}_i = -\frac{\partial \tilde{f}_i(y_i)}{\partial y_i} - \sum_{j=1}^{n} a_{ij}(y_i - y_j) - \sum_{j=1}^{n} a_{ij}(\lambda_i - \lambda_j)$$
$$\dot{\lambda}_i = \sum_{j=1}^{n} a_{ij}(y_i - y_j)\,.$$

- **distributed:** local computation & nearest neighbor communication;
- **converges** to the set of saddle points $(y^\star, \lambda^\star)$; and
- **recovers** the globally optimal solution $y^\star = x^\star \mathbf{1}$.

Block-diagram of distributed primal-dual saddle-point flow

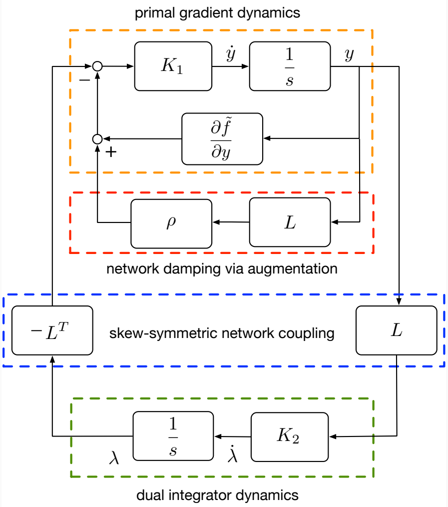


$$\dot{y} = -\frac{\partial \tilde{f}(y)}{\partial y} - Ly - L\lambda$$

$$\dot{\lambda} = Ly$$

1. primal gradient descent

Block-diagram of distributed primal-dual saddle-point flow

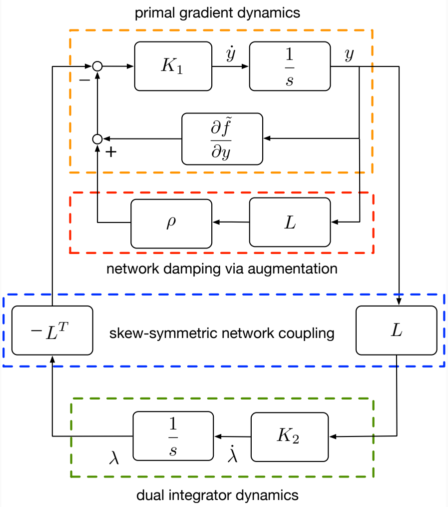


$$\dot{y} = -\frac{\partial \tilde{f}(y)}{\partial y} - Ly - L\lambda$$

$$\dot{\lambda} = Ly$$

1. primal gradient descent
2. dual integral control penalizing violation of consensus constraint

Block-diagram of distributed primal-dual saddle-point flow

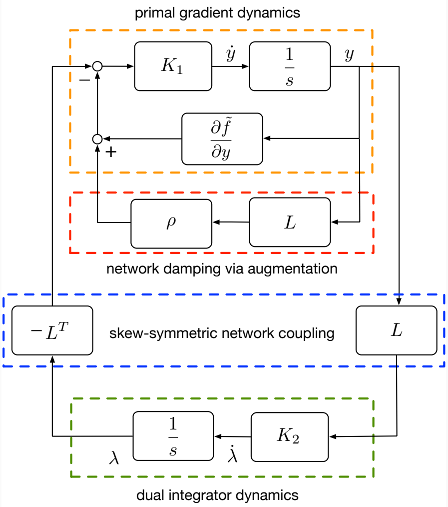


$$\dot{y} = -\frac{\partial \tilde{f}(y)}{\partial y} - Ly - L\lambda$$

$$\dot{\lambda} = Ly$$

1. primal gradient descent
2. dual integral control penalizing violation of consensus constraint
3. skew-symmetric network coupling

Block-diagram of distributed primal-dual saddle-point flow



$$\dot{y} = -\frac{\partial \tilde{f}(y)}{\partial y} - Ly - L\lambda$$

$$\dot{\lambda} = Ly$$

1. primal gradient descent
2. dual integral control penalizing violation of consensus constraint
3. skew-symmetric network coupling
4. Laplacian damping from augmentation

Next week:

1. More distributed optimization

Next week:

1. More distributed optimization
2. Algorithms

Next week:

1. More distributed optimization
2. Algorithms
3. Exercise sessions: discrete-time counterparts