Illustration of an active set method

Note the different notation used in these slides compared to the Lecture notes:

	Lecture notes	**These slides**
Optimization variables	x	w
Objective function	$f(x)$	$\Phi(w)$
Inequality constraint function	$g(x)$	$h(w)$
Linear constraint function	$h(x)$	$g(w)$

Active Set Method - Algorithm

Catch the right Active Set $\mathbb{A}$ as fast as possible !!

Active Set Method - Algorithm

Catch the right Active Set $\mathbb{A}$ as fast as possible !!

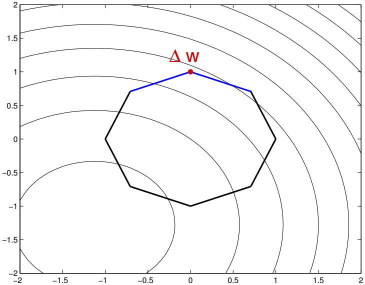
Pseudo-algorithm

Guess: $\mathbb{A} = \{7, 8\}$, Δw feasible ("phase-I")

1. Form $\nabla \mathbf{h}_{\mathbb{A}}^{\mathsf{T}}$ and $\mathbf{h}_{\mathbb{A}}$ (active constraints)
2. Solve :

$$\begin{bmatrix} B & \nabla \mathbf{g} & \nabla \mathbf{h}_{\mathbb{A}} \\ \nabla \mathbf{g}^{\mathsf{T}} & 0 & 0 \\ \nabla \mathbf{h}_{\mathbb{A}}^{\mathsf{T}} & 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta \mathbf{w}^{+} \\ \boldsymbol{\lambda} \\ \boldsymbol{\mu}^{+} \end{bmatrix} = - \begin{bmatrix} \nabla \Phi \\ \mathbf{g} \\ \mathbf{h}_{\mathbb{A}} \end{bmatrix}$$

3. Move $\Delta w \rightarrow \Delta w^{+}$ until hitting a constraint
4. Add the new constraint to $\mathbb{A}$
5. If $\boldsymbol{\mu}_k^{+} \leq 0$, then **remove** k from $\mathbb{A}$
6. If $\boldsymbol{\mu}^{+} > 0$ and $\Delta w^{+} = \Delta w$ reached then **exit**
7. Else **goto 1**



Active Set Method - Algorithm

Catch the right Active Set $\mathbb{A}$ as fast as possible !!

Pseudo-algorithm

Guess: $\mathbb{A} = \{7, 8\}$, Δw feasible ("phase-I")

1. Form $\nabla \mathbf{h}_{\mathbb{A}}^{\mathsf{T}}$ and $\mathbf{h}_{\mathbb{A}}$ (active constraints)
2. Solve :

$$\begin{bmatrix} B & \nabla \mathbf{g} & \nabla \mathbf{h}_{\mathbb{A}} \\ \nabla \mathbf{g}^{\mathsf{T}} & 0 & 0 \\ \nabla \mathbf{h}_{\mathbb{A}}^{\mathsf{T}} & 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta \mathbf{w}^{+} \\ \boldsymbol{\lambda} \\ \boldsymbol{\mu}^{+} \end{bmatrix} = - \begin{bmatrix} \nabla \Phi \\ \mathbf{g} \\ \mathbf{h}_{\mathbb{A}} \end{bmatrix}$$

3. Move $\Delta w \rightarrow \Delta w^{+}$ until hitting a constraint
4. Add the new constraint to $\mathbb{A}$
5. If $\mu_k^{+} \leq 0$, then **remove** k from $\mathbb{A}$
6. If $\mu^{+} > 0$ and $\Delta w^{+} = \Delta w$ reached then **exit**
7. Else **goto 1**

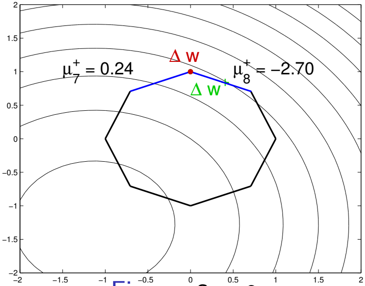


Figure: Step 2

Active Set Method - Algorithm

Catch the right Active Set $\mathbb{A}$ as fast as possible !!

Pseudo-algorithm

Guess: $\mathbb{A} = \{7, 8\}$, Δw feasible ("phase-I")

1. Form $\nabla \mathbf{h}_{\mathbb{A}}^{\mathsf{T}}$ and $\mathbf{h}_{\mathbb{A}}$ (active constraints)
2. Solve :

$$\begin{bmatrix} B & \nabla \mathbf{g} & \nabla \mathbf{h}_{\mathbb{A}} \\ \nabla \mathbf{g}^{\mathsf{T}} & 0 & 0 \\ \nabla \mathbf{h}_{\mathbb{A}}^{\mathsf{T}} & 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta \mathbf{w}^{+} \\ \boldsymbol{\lambda} \\ \boldsymbol{\mu}^{+} \end{bmatrix} = - \begin{bmatrix} \nabla \Phi \\ \mathbf{g} \\ \mathbf{h}_{\mathbb{A}} \end{bmatrix}$$

3. Move $\Delta w \rightarrow \Delta w^{+}$ until hitting a constraint
4. Add the new constraint to $\mathbb{A}$
5. If $\boldsymbol{\mu}_k^{+} \leq 0$, then **remove** k from $\mathbb{A}$
6. If $\boldsymbol{\mu}^{+} > 0$ and $\Delta w^{+} = \Delta w$ reached then **exit**
7. Else **goto 1**

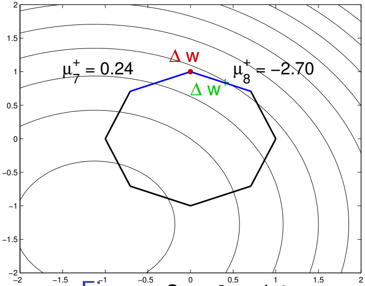


Figure: Step 3 and 4

Active Set Method - Algorithm

Catch the right Active Set $\mathbb{A}$ as fast as possible !!

Pseudo-algorithm

Guess: $\mathbb{A} = \{7, 8\}$, Δw feasible ("phase-I")

1. Form $\nabla \mathbf{h}_{\mathbb{A}}^{\top}$ and $\mathbf{h}_{\mathbb{A}}$ (active constraints)
2. Solve :

$$\begin{bmatrix} B & \nabla \mathbf{g} & \nabla \mathbf{h}_{\mathbb{A}} \\ \nabla \mathbf{g}^{\top} & 0 & 0 \\ \nabla \mathbf{h}_{\mathbb{A}}^{\top} & 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta \mathbf{w}^{+} \\ \boldsymbol{\lambda} \\ \boldsymbol{\mu}^{+} \end{bmatrix} = - \begin{bmatrix} \nabla \Phi \\ \mathbf{g} \\ \mathbf{h}_{\mathbb{A}} \end{bmatrix}$$

3. Move $\Delta w \rightarrow \Delta w^{+}$ until hitting a constraint
4. Add the new constraint to $\mathbb{A}$
5. If $\boldsymbol{\mu}_k^{+} \leq 0$, then **remove** k from $\mathbb{A}$
6. If $\boldsymbol{\mu}^{+} > 0$ and $\Delta w^{+} = \Delta w$ reached then **exit**
7. Else **goto 1**

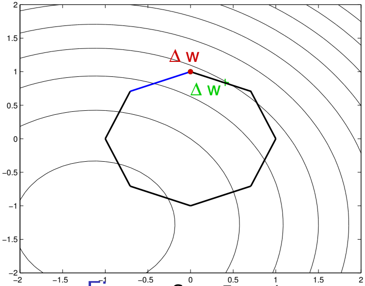


Figure: Step 5 $\rightarrow$ 1

Active Set Method - Algorithm

Catch the right Active Set $\mathbb{A}$ as fast as possible !!

Pseudo-algorithm

Guess: $\mathbb{A} = \{7, 8\}$, Δw feasible ("phase-I")

1. Form $\nabla \mathbf{h}_{\mathbb{A}}^{\mathsf{T}}$ and $\mathbf{h}_{\mathbb{A}}$ (active constraints)
2. Solve :

$$\begin{bmatrix} B & \nabla \mathbf{g} & \nabla \mathbf{h}_{\mathbb{A}} \\ \nabla \mathbf{g}^{\mathsf{T}} & 0 & 0 \\ \nabla \mathbf{h}_{\mathbb{A}}^{\mathsf{T}} & 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta \mathbf{w}^{+} \\ \boldsymbol{\lambda} \\ \boldsymbol{\mu}^{+} \end{bmatrix} = - \begin{bmatrix} \nabla \Phi \\ \mathbf{g} \\ \mathbf{h}_{\mathbb{A}} \end{bmatrix}$$

3. Move $\Delta w \to \Delta w^{+}$ until hitting a constraint
4. Add the new constraint to $\mathbb{A}$
5. If $\boldsymbol{\mu}_k^{+} \leq 0$, then **remove** k from $\mathbb{A}$
6. If $\boldsymbol{\mu}^{+} > 0$ and $\Delta w^{+} = \Delta w$ reached then **exit**
7. Else **goto 1**

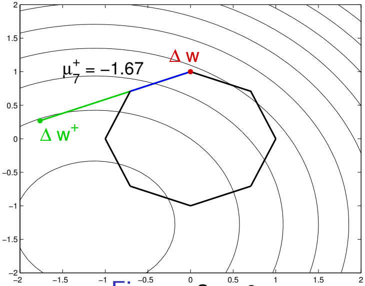


Figure: Step 2

Active Set Method - Algorithm

Catch the right Active Set $\mathbb{A}$ as fast as possible !!

Pseudo-algorithm

Guess: $\mathbb{A} = \{7, 8\}$, Δw feasible (''phase-I'')

1. Form $\nabla \mathbf{h}_{\mathbb{A}}^{\mathsf{T}}$ and $\mathbf{h}_{\mathbb{A}}$ (active constraints)
2. Solve :

$$\begin{bmatrix} B & \nabla \mathbf{g} & \nabla \mathbf{h}_{\mathbb{A}} \\ \nabla \mathbf{g}^{\mathsf{T}} & 0 & 0 \\ \nabla \mathbf{h}_{\mathbb{A}}^{\mathsf{T}} & 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta \mathbf{w}^{+} \\ \boldsymbol{\lambda} \\ \boldsymbol{\mu}^{+} \end{bmatrix} = - \begin{bmatrix} \nabla \Phi \\ \mathbf{g} \\ \mathbf{h}_{\mathbb{A}} \end{bmatrix}$$

3. Move $\Delta w \rightarrow \Delta w^{+}$ until hitting a constraint
4. Add the new constraint to $\mathbb{A}$
5. If $\boldsymbol{\mu}_k^{+} \leq 0$, then **remove** k from $\mathbb{A}$
6. If $\boldsymbol{\mu}^{+} > 0$ and $\Delta w^{+} = \Delta w$ reached then **exit**
7. Else **goto 1**

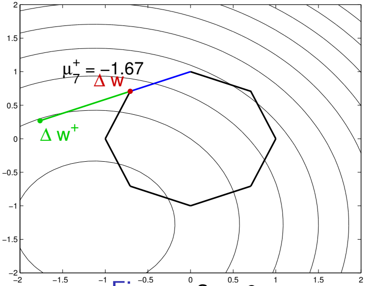


Figure: Step 3

Active Set Method - Algorithm

Catch the right Active Set $\mathbb{A}$ as fast as possible !!

Pseudo-algorithm

Guess: $\mathbb{A} = \{7, 8\}$, Δw feasible ("phase-I")

1. Form $\nabla \mathbf{h}_{\mathbb{A}}^{\top}$ and $\mathbf{h}_{\mathbb{A}}$ (active constraints)
2. Solve :

$$\begin{bmatrix} B & \nabla \mathbf{g} & \nabla \mathbf{h}_{\mathbb{A}} \\ \nabla \mathbf{g}^{\top} & 0 & 0 \\ \nabla \mathbf{h}_{\mathbb{A}}^{\top} & 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta \mathbf{w}^{+} \\ \boldsymbol{\lambda} \\ \boldsymbol{\mu}^{+} \end{bmatrix} = - \begin{bmatrix} \nabla \Phi \\ \mathbf{g} \\ \mathbf{h}_{\mathbb{A}} \end{bmatrix}$$

3. Move $\Delta w \rightarrow \Delta w^{+}$ until hitting a constraint
4. Add the new constraint to $\mathbb{A}$
5. If $\boldsymbol{\mu}_k^{+} \leq 0$, then **remove** k from $\mathbb{A}$
6. If $\boldsymbol{\mu}^{+} > 0$ and $\Delta w^{+} = \Delta w$ reached then **exit**
7. Else **goto 1**

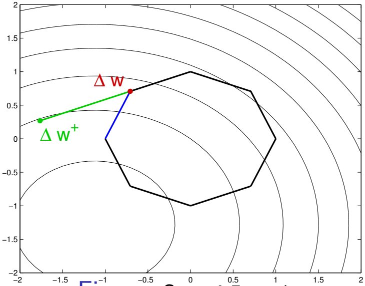


Figure: Step 4,5 $\rightarrow$ 1

Active Set Method - Algorithm

Catch the right Active Set $\mathbb{A}$ as fast as possible !!

Pseudo-algorithm

Guess: $\mathbb{A} = \{7, 8\}$, Δw feasible ("phase-I")

1. Form $\nabla \mathbf{h}_{\mathbb{A}}^{\top}$ and $\mathbf{h}_{\mathbb{A}}$ (active constraints)
2. Solve :

$$\begin{bmatrix} B & \nabla \mathbf{g} & \nabla \mathbf{h}_{\mathbb{A}} \\ \nabla \mathbf{g}^{\top} & 0 & 0 \\ \nabla \mathbf{h}_{\mathbb{A}}^{\top} & 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta \mathbf{w}^{+} \\ \boldsymbol{\lambda} \\ \boldsymbol{\mu}^{+} \end{bmatrix} = - \begin{bmatrix} \nabla \Phi \\ \mathbf{g} \\ \mathbf{h}_{\mathbb{A}} \end{bmatrix}$$

3. Move $\Delta w \rightarrow \Delta w^{+}$ until hitting a constraint
4. Add the new constraint to $\mathbb{A}$
5. If $\boldsymbol{\mu}_k^{+} \leq 0$, then **remove** k from $\mathbb{A}$
6. If $\boldsymbol{\mu}^{+} > 0$ and $\Delta w^{+} = \Delta w$ reached then **exit**
7. Else **goto 1**

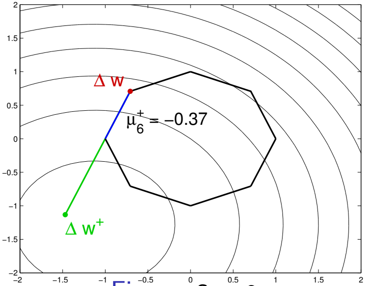


Figure: Step 2

Active Set Method - Algorithm

Catch the right Active Set $\mathbb{A}$ as fast as possible !!

Pseudo-algorithm

Guess: $\mathbb{A} = \{7, 8\}$, Δw feasible ("phase-I")

1. Form $\nabla \mathbf{h}_{\mathbb{A}}^{\mathsf{T}}$ and $\mathbf{h}_{\mathbb{A}}$ (active constraints)
2. Solve :

$$\begin{bmatrix} B & \nabla \mathbf{g} & \nabla \mathbf{h}_{\mathbb{A}} \\ \nabla \mathbf{g}^{\mathsf{T}} & 0 & 0 \\ \nabla \mathbf{h}_{\mathbb{A}}^{\mathsf{T}} & 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta \mathbf{w}^{+} \\ \boldsymbol{\lambda} \\ \boldsymbol{\mu}^{+} \end{bmatrix} = - \begin{bmatrix} \nabla \Phi \\ \mathbf{g} \\ \mathbf{h}_{\mathbb{A}} \end{bmatrix}$$

3. Move $\Delta w \rightarrow \Delta w^{+}$ until hitting a constraint
4. Add the new constraint to $\mathbb{A}$
5. If $\boldsymbol{\mu}_k^{+} \leq 0$, then **remove** k from $\mathbb{A}$
6. If $\boldsymbol{\mu}^{+} > 0$ and $\Delta w^{+} = \Delta w$ reached then **exit**
7. Else **goto 1**

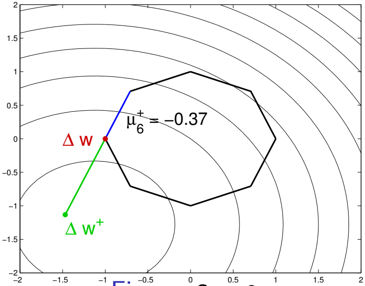


Figure: Step 3

Active Set Method - Algorithm

Catch the right Active Set $\mathbb{A}$ as fast as possible !!

Pseudo-algorithm

Guess: $\mathbb{A} = \{7, 8\}$, Δw feasible ("phase-I")

1. Form $\nabla \mathbf{h}_{\mathbb{A}}^{\mathsf{T}}$ and $\mathbf{h}_{\mathbb{A}}$ (active constraints)
2. Solve :

$$\left[\begin{array}{ccc} B & \nabla \mathbf{g} & \nabla \mathbf{h}_{\mathbb{A}} \\ \nabla \mathbf{g}^{\mathsf{T}} & 0 & 0 \\ \nabla \mathbf{h}_{\mathbb{A}}^{\mathsf{T}} & 0 & 0 \end{array}\right] \left[\begin{array}{c} \Delta \mathbf{w}^{+} \\ \boldsymbol{\lambda} \\ \boldsymbol{\mu}^{+} \end{array}\right] = - \left[\begin{array}{c} \nabla \Phi \\ \mathbf{g} \\ \mathbf{h}_{\mathbb{A}} \end{array}\right]$$

3. Move $\Delta w \rightarrow \Delta w^{+}$ until hitting a constraint
4. Add the new constraint to $\mathbb{A}$
5. If $\boldsymbol{\mu}_k^{+} \leq 0$, then **remove** k from $\mathbb{A}$
6. If $\boldsymbol{\mu}^{+} > 0$ and $\Delta w^{+} = \Delta w$ reached then **exit**
7. Else **goto 1**

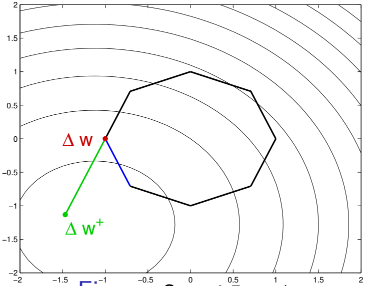


Figure: Step 4,5 $\rightarrow$ 1

Active Set Method - Algorithm

Catch the right Active Set $\mathbb{A}$ as fast as possible !!

Pseudo-algorithm

Guess: $\mathbb{A} = \{7, 8\}$, Δw feasible ("phase-I")

1. Form $\nabla \mathbf{h}_{\mathbb{A}}^{\mathsf{T}}$ and $\mathbf{h}_{\mathbb{A}}$ (active constraints)
2. Solve :

$$\begin{bmatrix} B & \nabla \mathbf{g} & \nabla \mathbf{h}_{\mathbb{A}} \\ \nabla \mathbf{g}^{\mathsf{T}} & 0 & 0 \\ \nabla \mathbf{h}_{\mathbb{A}}^{\mathsf{T}} & 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta \mathbf{w}^+ \\ \boldsymbol{\lambda} \\ \boldsymbol{\mu}^+ \end{bmatrix} = - \begin{bmatrix} \nabla \Phi \\ \mathbf{g} \\ \mathbf{h}_{\mathbb{A}} \end{bmatrix}$$

3. Move $\Delta w \rightarrow \Delta w^+$ until hitting a constraint
4. Add the new constraint to $\mathbb{A}$
5. If $\boldsymbol{\mu}_k^+ \leq 0$, then **remove** k from $\mathbb{A}$
6. If $\boldsymbol{\mu}^+ > 0$ and $\Delta w^+ = \Delta w$ reached then **exit**
7. Else **goto 1**

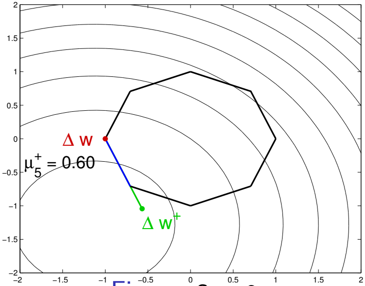


Figure: Step 2

Active Set Method - Algorithm

Catch the right Active Set $\mathbb{A}$ as fast as possible !!

Pseudo-algorithm

Guess: $\mathbb{A} = \{7, 8\}$, Δw feasible ("phase-I")

1. Form $\nabla \mathbf{h}_{\mathbb{A}}^{\mathsf{T}}$ and $\mathbf{h}_{\mathbb{A}}$ (active constraints)
2. Solve :

$$\begin{bmatrix} B & \nabla \mathbf{g} & \nabla \mathbf{h}_{\mathbb{A}} \\ \nabla \mathbf{g}^{\mathsf{T}} & 0 & 0 \\ \nabla \mathbf{h}_{\mathbb{A}}^{\mathsf{T}} & 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta \mathbf{w}^{+} \\ \boldsymbol{\lambda} \\ \boldsymbol{\mu}^{+} \end{bmatrix} = - \begin{bmatrix} \nabla \Phi \\ \mathbf{g} \\ \mathbf{h}_{\mathbb{A}} \end{bmatrix}$$

3. Move $\Delta w \to \Delta w^{+}$ until hitting a constraint
4. Add the new constraint to $\mathbb{A}$
5. If $\boldsymbol{\mu}_k^{+} \leq 0$, then **remove** k from $\mathbb{A}$
6. If $\boldsymbol{\mu}^{+} > 0$ and $\Delta w^{+} = \Delta w$ reached then **exit**
7. Else **goto 1**

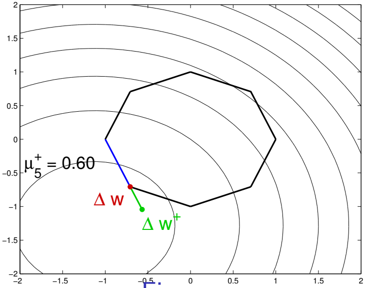


Figure:

Active Set Method - Algorithm

Catch the right Active Set $\mathbb{A}$ as fast as possible !!

Pseudo-algorithm

Guess: $\mathbb{A} = \{7, 8\}$, Δw feasible ("phase-I")

1. Form $\nabla \mathbf{h}_{\mathbb{A}}^\top$ and $\mathbf{h}_{\mathbb{A}}$ (active constraints)
2. Solve :

$$\begin{bmatrix} B & \nabla \mathbf{g} & \nabla \mathbf{h}_{\mathbb{A}} \\ \nabla \mathbf{g}^\top & 0 & 0 \\ \nabla \mathbf{h}_{\mathbb{A}}^\top & 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta \mathbf{w}^+ \\ \boldsymbol{\lambda} \\ \boldsymbol{\mu}^+ \end{bmatrix} = - \begin{bmatrix} \nabla \Phi \\ \mathbf{g} \\ \mathbf{h}_{\mathbb{A}} \end{bmatrix}$$

3. Move $\Delta w \rightarrow \Delta w^+$ until hitting a constraint
4. Add the new constraint to $\mathbb{A}$
5. If $\boldsymbol{\mu}_k^+ \leq 0$, then **remove** k from $\mathbb{A}$
6. If $\boldsymbol{\mu}^+ > 0$ and $\Delta w^+ = \Delta w$ reached then **exit**
7. Else **goto 1**

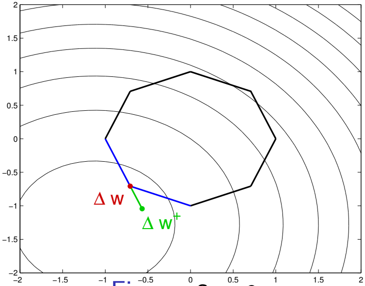


Figure: Step 3

Active Set Method - Algorithm

Catch the right Active Set $\mathbb{A}$ as fast as possible !!

Pseudo-algorithm

Guess: $\mathbb{A} = \{7, 8\}$, Δw feasible ("phase-I")

1. Form $\nabla \mathbf{h}_{\mathbb{A}}^\top$ and $\mathbf{h}_{\mathbb{A}}$ (active constraints)
2. Solve :

$$\begin{bmatrix} B & \nabla \mathbf{g} & \nabla \mathbf{h}_{\mathbb{A}} \\ \nabla \mathbf{g}^\top & 0 & 0 \\ \nabla \mathbf{h}_{\mathbb{A}}^\top & 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta \mathbf{w}^+ \\ \boldsymbol{\lambda} \\ \boldsymbol{\mu}^+ \end{bmatrix} = - \begin{bmatrix} \nabla \Phi \\ \mathbf{g} \\ \mathbf{h}_{\mathbb{A}} \end{bmatrix}$$

3. Move $\Delta w \rightarrow \Delta w^+$ until hitting a constraint
4. Add the new constraint to $\mathbb{A}$
5. If $\boldsymbol{\mu}_k^+ \leq 0$, then **remove** k from $\mathbb{A}$
6. If $\boldsymbol{\mu}^+ > 0$ and $\Delta w^+ = \Delta w$ reached then **exit**
7. Else **goto 1**

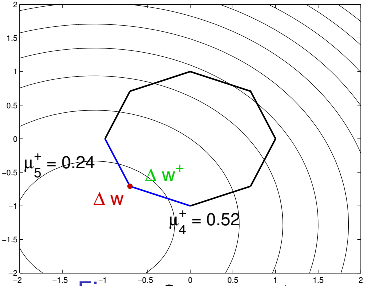


Figure: Step 4,5 $\rightarrow$ 1

Active Set Method - Algorithm

Catch the right Active Set $\mathbb{A}$ as fast as possible !!

Pseudo-algorithm

Guess: $\mathbb{A} = \{7, 8\}$, Δw feasible ("phase-I")

1. Form $\nabla \mathbf{h}_{\mathbb{A}}^{\top}$ and $\mathbf{h}_{\mathbb{A}}$ (active constraints)
2. Solve :

$$\begin{bmatrix} B & \nabla \mathbf{g} & \nabla \mathbf{h}_{\mathbb{A}} \\ \nabla \mathbf{g}^{\top} & 0 & 0 \\ \nabla \mathbf{h}_{\mathbb{A}}^{\top} & 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta \mathbf{w}^{+} \\ \boldsymbol{\lambda} \\ \boldsymbol{\mu}^{+} \end{bmatrix} = - \begin{bmatrix} \nabla \Phi \\ \mathbf{g} \\ \mathbf{h}_{\mathbb{A}} \end{bmatrix}$$

3. Move $\Delta w \rightarrow \Delta w^{+}$ until hitting a constraint
4. Add the new constraint to $\mathbb{A}$
5. If $\boldsymbol{\mu}_k^{+} \leq 0$, then **remove** k from $\mathbb{A}$
6. If $\boldsymbol{\mu}^{+} > 0$ and $\Delta w^{+} = \Delta w$ reached then **exit**
7. Else **goto 1**

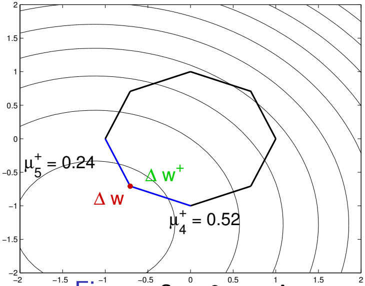


Figure: Step 2 → **exit**

Active Set Method - Algorithm

Catch the right Active Set $\mathbb{A}$ as fast as possible !!

Pseudo-algorithm

Guess: $\mathbb{A} = \{7, 8\}$, Δw feasible ("phase-I")

1. Form $\nabla \mathbf{h}_{\mathbb{A}}^{\top}$ and $\mathbf{h}_{\mathbb{A}}$ (active constraints)
2. Solve :

$$\left[\begin{array}{ccc} B & \nabla \mathbf{g} & \nabla \mathbf{h}_{\mathbb{A}} \\ \nabla \mathbf{g}^{\top} & 0 & 0 \\ \nabla \mathbf{h}_{\mathbb{A}}^{\top} & 0 & 0 \end{array}\right] \left[\begin{array}{c} \Delta \mathbf{w}^{+} \\ \boldsymbol{\lambda} \\ \boldsymbol{\mu}^{+} \end{array}\right] = - \left[\begin{array}{c} \nabla \Phi \\ \mathbf{g} \\ \mathbf{h}_{\mathbb{A}} \end{array}\right]$$

3. Move $\Delta w \rightarrow \Delta w^{+}$ until hitting a constraint
4. Add the new constraint to $\mathbb{A}$
5. If $\boldsymbol{\mu}_k^{+} \leq 0$, then **remove** k from $\mathbb{A}$
6. If $\boldsymbol{\mu}^{+} > 0$ and $\Delta w^{+} = \Delta w$ reached then **exit**
7. Else **goto 1**

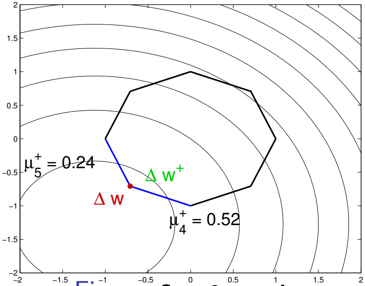


Figure: Step 2 $\rightarrow$ **exit**

1. Very fast for a few changes of the active set
2. No tight complexity bound

Structure exploitation - Condensing

Iterate QP

$$
\begin{aligned}
\min_{\Delta \mathbf{w}} \quad & \frac{1}{2}\Delta \mathbf{w}^T B \Delta \mathbf{w} + \nabla \Phi^T \Delta \mathbf{w} \\
\text{s.t.} \quad & \nabla \mathbf{g}^\mathsf{T} \Delta \mathbf{w} + \mathbf{g} = 0 \\
& \nabla \mathbf{h}^\mathsf{T} \Delta \mathbf{w} + \mathbf{h} \leq 0
\end{aligned}
$$

Factorization of the linear system

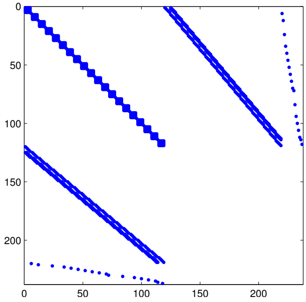
$$
\begin{bmatrix} B & \nabla \mathbf{g} & \nabla \mathbf{h}_{\mathbb{A}} \\ \nabla \mathbf{g}^\mathsf{T} & 0 & 0 \\ \nabla \mathbf{h}_{\mathbb{A}}^\mathsf{T} & 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta \mathbf{w}^+ \\ \boldsymbol{\lambda} \\ \boldsymbol{\mu}^+ \end{bmatrix} = - \begin{bmatrix} \nabla \Phi \\ \mathbf{g} \\ \mathbf{h}_{\mathbb{A}} \end{bmatrix}
$$

is expensive !!

Structure exploitation - Condensing

Iterate QP

$$\begin{aligned} \min_{\Delta \mathbf{w}} \quad & \frac{1}{2}\Delta \mathbf{w}^T B \Delta \mathbf{w} + \nabla \Phi^T \Delta \mathbf{w} \\ \text{s.t.} \quad & \nabla \mathbf{g}^\mathsf{T} \Delta \mathbf{w} + \mathbf{g} = 0 \\ & \nabla \mathbf{h}^\mathsf{T} \Delta \mathbf{w} + \mathbf{h} \leq 0 \end{aligned}$$



Example: 5 integrators, 1 input, $N = 20$ with input & state bounds.

Factorization of the linear system

$$\begin{bmatrix} B & \nabla \mathbf{g} & \nabla \mathbf{h}_{\mathbb{A}} \\ \nabla \mathbf{g}^\mathsf{T} & 0 & 0 \\ \nabla \mathbf{h}_{\mathbb{A}}^\mathsf{T} & 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta \mathbf{w}^+ \\ \boldsymbol{\lambda} \\ \boldsymbol{\mu}^+ \end{bmatrix} = - \begin{bmatrix} \nabla \Phi \\ \mathbf{g} \\ \mathbf{h}_{\mathbb{A}} \end{bmatrix}$$

is expensive !!

Structure exploitation - Condensing

Iterate QP

$$\min_{\Delta \mathbf{w}} \quad \frac{1}{2}\Delta \mathbf{w}^T B \Delta \mathbf{w} + \nabla \Phi^T \Delta \mathbf{w}$$

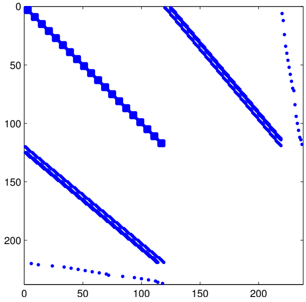
$$\text{s.t.} \quad \nabla \mathbf{g}^{\mathsf{T}} \Delta \mathbf{w} + \mathbf{g} = 0$$

$$\nabla \mathbf{h}^{\mathsf{T}} \Delta \mathbf{w} + \mathbf{h} \leq 0$$

Factorization of the linear system

$$\begin{bmatrix} B & \nabla \mathbf{g} & \nabla \mathbf{h}_{\mathbb{A}} \\ \nabla \mathbf{g}^{\mathsf{T}} & 0 & 0 \\ \nabla \mathbf{h}_{\mathbb{A}}^{\mathsf{T}} & 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta \mathbf{w}^+ \\ \boldsymbol{\lambda} \\ \boldsymbol{\mu}^+ \end{bmatrix} = - \begin{bmatrix} \nabla \Phi \\ \mathbf{g} \\ \mathbf{h}_{\mathbb{A}} \end{bmatrix}$$

is expensive !!

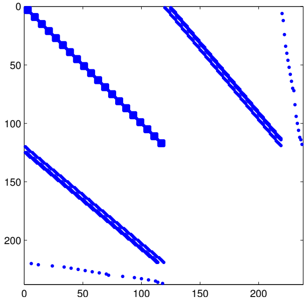


Example: 5 integrators, 1 input, $N = 20$ with input & state bounds.

Structure exploitation - Condensing

Iterate QP

$$\begin{aligned} \min_{\Delta \mathbf{w}} \quad & \frac{1}{2}\Delta \mathbf{w}^T B \Delta \mathbf{w} + \nabla \Phi^T \Delta \mathbf{w} \\ \text{s.t.} \quad & \nabla \mathbf{g}^\mathsf{T} \Delta \mathbf{w} + \mathbf{g} = 0 \\ & \nabla \mathbf{h}^\mathsf{T} \Delta \mathbf{w} + \mathbf{h} \leq 0 \end{aligned}$$



Example: 5 integrators, 1 input, $N = 20$ with input & state bounds.

Condensed QP

Eliminate the states $\Delta \mathbf{x}_k$ using $\nabla \mathbf{g}^\mathsf{T} \Delta \mathbf{w} + \mathbf{g} = 0$, i.e.

$$\Delta \mathbf{x}_{k+1} = \nabla_{\mathbf{x}} \mathbf{f}_k{}^\mathsf{T} \Delta \mathbf{x}_k + \nabla_{\mathbf{u}} \mathbf{f}_k{}^\mathsf{T} \Delta \mathbf{u}_k$$

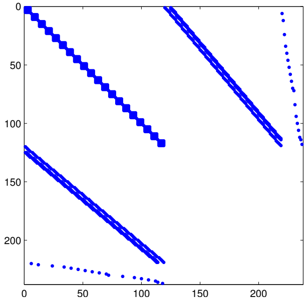
Structure exploitation - Condensing

Iterate QP

$$\min_{\Delta \mathbf{w}} \quad \frac{1}{2}\Delta \mathbf{w}^T B \Delta \mathbf{w} + \nabla \Phi^T \Delta \mathbf{w}$$

$$\text{s.t.} \quad \nabla \mathbf{g}^\top \Delta \mathbf{w} + \mathbf{g} = 0$$

$$\nabla \mathbf{h}^\top \Delta \mathbf{w} + \mathbf{h} \leq 0$$



Example: 5 integrators, 1 input, $N = 20$ with input & state bounds.

Condensed QP

Eliminate the states $\Delta \mathbf{x}_k$ using $\nabla \mathbf{g}^\top \Delta \mathbf{w} + \mathbf{g} = 0$, i.e.

$$\Delta \mathbf{x}_{k+1} = \nabla_{\mathbf{x}} \mathbf{f}_k{}^\top \Delta \mathbf{x}_k + \nabla_{\mathbf{u}} \mathbf{f}_k{}^\top \Delta \mathbf{u}_k$$

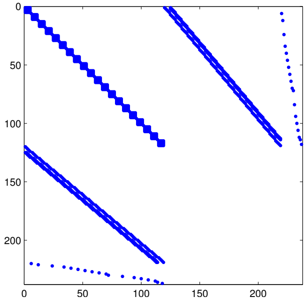
yields by "simulation":

$$\Delta \mathbf{x}_k = \Pi_{i=0}^{k-1} \nabla_{\mathbf{x}} \mathbf{f}_i{}^\top \Delta \mathbf{x}_0 + \sum_{j=0}^{k-1} \Pi_{i=j+1}^{k-1} \nabla_{\mathbf{x}} \mathbf{f}_i{}^\top \nabla_{\mathbf{u}} \mathbf{f}_j{}^\top \Delta \mathbf{u}_j$$

Structure exploitation - Condensing

Iterate QP

$$\min_{\Delta \mathbf{w}} \quad \frac{1}{2}\Delta \mathbf{w}^T B \Delta \mathbf{w} + \nabla \Phi^T \Delta \mathbf{w}$$
$$\text{s.t.} \quad \nabla \mathbf{g}^\top \Delta \mathbf{w} + \mathbf{g} = 0$$
$$\nabla \mathbf{h}^\top \Delta \mathbf{w} + \mathbf{h} \leq 0$$



Example: 5 integrators, 1 input, $N = 20$ with input & state bounds.

Condensed QP

Eliminate the states $\Delta \mathbf{x}_k$ using $\nabla \mathbf{g}^\top \Delta \mathbf{w} + \mathbf{g} = 0$, i.e.

$$\Delta \mathbf{x}_{k+1} = \nabla_{\mathbf{x}} \mathbf{f}_k{}^\top \Delta \mathbf{x}_k + \nabla_{\mathbf{u}} \mathbf{f}_k{}^\top \Delta \mathbf{u}_k$$

yields by "simulation":

$$\Delta \mathbf{x}_k = \Pi_{i=0}^{k-1} \nabla_{\mathbf{x}} \mathbf{f}_i{}^\top \Delta \mathbf{x}_0 + \sum_{j=0}^{k-1} \Pi_{i=j+1}^{k-1} \nabla_{\mathbf{x}} \mathbf{f}_i{}^\top \nabla_{\mathbf{u}} \mathbf{f}_j{}^\top \Delta \mathbf{u}_j$$

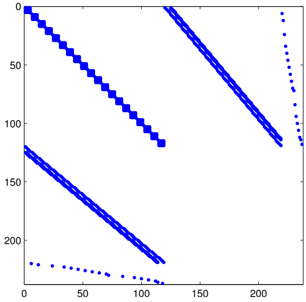
an we can write:

$$\Delta \mathbf{w} = A + M \begin{bmatrix} \Delta \mathbf{u}_0 \\ \dots \\ \Delta \mathbf{u}_{N-1} \end{bmatrix} \equiv A + M \Delta \mathbf{u}$$

Structure exploitation - Condensing

Iterate QP

$$\begin{aligned} \min_{\Delta \mathbf{w}} \quad & \frac{1}{2}\Delta \mathbf{w}^T B \Delta \mathbf{w} + \nabla \Phi^T \Delta \mathbf{w} \\ \text{s.t.} \quad & \nabla \mathbf{g}^\top \Delta \mathbf{w} + \mathbf{g} = 0 \\ & \nabla \mathbf{h}^\top \Delta \mathbf{w} + \mathbf{h} \leq 0 \end{aligned}$$



Example: 5 integrators, 1 input, $N = 20$ with input & state bounds.

Condensed QP

Eliminate the states $\Delta \mathbf{x}_k$ using $\nabla \mathbf{g}^\top \Delta \mathbf{w} + \mathbf{g} = 0$, i.e.

$$\Delta \mathbf{x}_{k+1} = \nabla_{\mathbf{x}} \mathbf{f}_k{}^\top \Delta \mathbf{x}_k + \nabla_{\mathbf{u}} \mathbf{f}_k{}^\top \Delta \mathbf{u}_k$$

yields by "simulation":

$$\Delta \mathbf{x}_k = \Pi_{i=0}^{k-1} \nabla_{\mathbf{x}} \mathbf{f}_i{}^\top \Delta \mathbf{x}_0 + \sum_{j=0}^{k-1} \Pi_{i=j+1}^{k-1} \nabla_{\mathbf{x}} \mathbf{f}_i{}^\top \nabla_{\mathbf{u}} \mathbf{f}_j{}^\top \Delta \mathbf{u}_j$$

an we can write:

$$\Delta \mathbf{w} = A + M \begin{bmatrix} \Delta \mathbf{u}_0 \\ \dots \\ \Delta \mathbf{u}_{N-1} \end{bmatrix} \equiv A + M \Delta \mathbf{u}$$

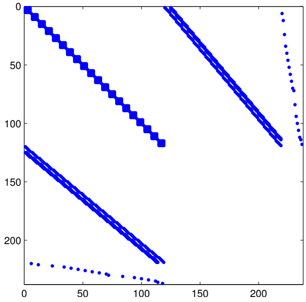
The condensed QP then reads as:

$$\begin{aligned} \min_{\Delta \mathbf{u}} \quad & \frac{1}{2}\Delta \mathbf{u}^\top M^\top B M \Delta \mathbf{u} + \left(\frac{1}{2} A^\top B M + \nabla \Phi^\top M \right) \Delta \mathbf{u} \\ \text{s.t.} \quad & \nabla \mathbf{h}^\top M \Delta \mathbf{u} + \nabla \mathbf{h}^\top A + \mathbf{g} \leq 0 \end{aligned}$$

Structure exploitation - Condensing

Iterate QP

$$
\begin{aligned}
\min_{\Delta \mathbf{w}} \quad & \frac{1}{2}\Delta \mathbf{w}^T B \Delta \mathbf{w} + \nabla \Phi^T \Delta \mathbf{w} \\
\text{s.t.} \quad & \nabla \mathbf{g}^\mathsf{T} \Delta \mathbf{w} + \mathbf{g} = 0 \\
& \nabla \mathbf{h}^\mathsf{T} \Delta \mathbf{w} + \mathbf{h} \leq 0
\end{aligned}
$$



Example: 5 integrators, 1 input, $N = 20$ with input & state bounds.

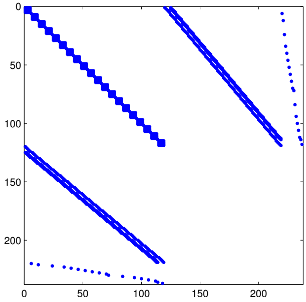
Condensed QP

$$
\begin{aligned}
\min_{\Delta \mathbf{u}} \quad & \frac{1}{2}\Delta \mathbf{u}^\mathsf{T} M^\mathsf{T} B M \Delta \mathbf{u} + \left(\frac{1}{2}A^\mathsf{T} B M + \nabla \Phi^\mathsf{T} M\right)\Delta \mathbf{u} \\
\text{s.t.} \quad & \nabla \mathbf{h}^\mathsf{T} M \Delta \mathbf{u} + \nabla \mathbf{h}^\mathsf{T} A + \mathbf{g} \leq 0
\end{aligned}
$$

Structure exploitation - Condensing

Iterate QP

$$
\begin{aligned}
\min_{\Delta \mathbf{w}} \quad & \frac{1}{2}\Delta \mathbf{w}^T {\color{cyan}B}\Delta \mathbf{w} + \nabla \Phi^T \Delta \mathbf{w} \\
\text{s.t.} \quad & {\color{red}\nabla \mathbf{g}^\mathsf{T} \Delta \mathbf{w} + \mathbf{g} = 0} \\
& {\color{green}\nabla \mathbf{h}^\mathsf{T} \Delta \mathbf{w} + \mathbf{h} \leq 0}
\end{aligned}
$$



Example: 5 integrators, 1 input, $N = 20$ with input & state bounds.

Condensed QP

$$
\begin{aligned}
\min_{\Delta \mathbf{u}} \quad & \frac{1}{2}\Delta \mathbf{u}^\mathsf{T} M^\mathsf{T} B M \Delta \mathbf{u} + \left(\frac{1}{2}A^\mathsf{T} B M + \nabla \Phi^\mathsf{T} M\right)\Delta \mathbf{u} \\
\text{s.t.} \quad & {\color{green}\nabla \mathbf{h}^\mathsf{T} M \Delta \mathbf{u} + \nabla \mathbf{h}^\mathsf{T} A + \mathbf{g} \leq 0}
\end{aligned}
$$

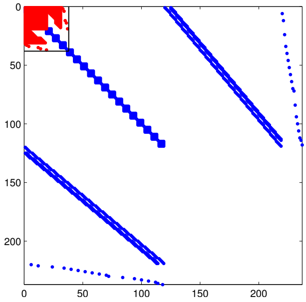
requires the factorisation of the matrix:

$$
\begin{bmatrix} M^\mathsf{T} B M & \left(M^\mathsf{T} \nabla \mathbf{h}\right)_{\color{cyan}\mathbb{A}} \\ \left(\nabla \mathbf{h}^\mathsf{T} M\right)_{\color{cyan}\mathbb{A}} & 0 \end{bmatrix} \begin{bmatrix} \Delta \mathbf{u} \\ \tilde{\boldsymbol{\mu}} \end{bmatrix} = \ldots
$$

Structure exploitation - Condensing

Iterate QP

$$\begin{aligned} \min_{\Delta \mathbf{w}} \quad & \frac{1}{2}\Delta \mathbf{w}^T B \Delta \mathbf{w} + \nabla \Phi^T \Delta \mathbf{w} \\ \text{s.t.} \quad & \nabla \mathbf{g}^\top \Delta \mathbf{w} + \mathbf{g} = 0 \\ & \nabla \mathbf{h}^\top \Delta \mathbf{w} + \mathbf{h} \leq 0 \end{aligned}$$



Example: 5 integrators, 1 input, $N = 20$ with input & state bounds, condensed

Condensed QP

$$\begin{aligned} \min_{\Delta \mathbf{u}} \quad & \frac{1}{2}\Delta \mathbf{u}^\top M^\top B M \Delta \mathbf{u} + \left(\frac{1}{2}A^\top B M + \nabla \Phi^\top M\right)\Delta \mathbf{u} \\ \text{s.t.} \quad & \nabla \mathbf{h}^\top M \Delta \mathbf{u} + \nabla \mathbf{h}^\top A + \mathbf{g} \leq 0 \end{aligned}$$

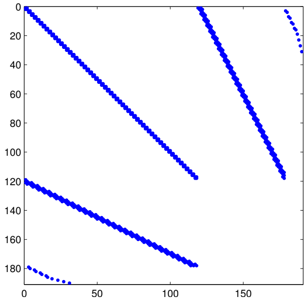
requires the factorisation of the matrix:

$$\begin{bmatrix} M^\top B M & \left(M^\top \nabla \mathbf{h}\right)_{\mathbb{A}} \\ \left(\nabla \mathbf{h}^\top M\right)_{\mathbb{A}} & 0 \end{bmatrix} \begin{bmatrix} \Delta \mathbf{u} \\ \tilde{\boldsymbol{\mu}} \end{bmatrix} = \ldots$$

Structure exploitation - Condensing

Iterate QP

$$\begin{aligned} \min_{\Delta \mathbf{w}} \quad & \frac{1}{2}\Delta \mathbf{w}^T {\color{cyan}B} \Delta \mathbf{w} + \nabla \Phi^T \Delta \mathbf{w} \\ \text{s.t.} \quad & {\color{red}\nabla \mathbf{g}^\top \Delta \mathbf{w} + \mathbf{g} = 0} \\ & {\color{green}\nabla \mathbf{h}^\top \Delta \mathbf{w} + \mathbf{h} \leq 0} \end{aligned}$$



Example: 2 integrators, 2 inputs, $N = 60$ with input & state bounds.

Condensed QP

$$\begin{aligned} \min_{\Delta \mathbf{u}} \quad & \frac{1}{2}\Delta \mathbf{u}^\top M^\top B M \Delta \mathbf{u} + \left(\frac{1}{2}A^\top B M + \nabla \Phi^\top M\right)\Delta \mathbf{u} \\ \text{s.t.} \quad & {\color{green}\nabla \mathbf{h}^\top M \Delta \mathbf{u} + \nabla \mathbf{h}^\top A + \mathbf{g} \leq 0} \end{aligned}$$

requires the factorisation of the matrix:

$$\begin{bmatrix} M^\top B M & \left(M^\top \nabla \mathbf{h}\right)_{\mathbb{A}} \\ \left(\nabla \mathbf{h}^\top M\right)_{\mathbb{A}} & 0 \end{bmatrix} \begin{bmatrix} \Delta \mathbf{u} \\ \tilde{\boldsymbol{\mu}} \end{bmatrix} = \dots$$

Structure exploitation - Condensing

Iterate QP

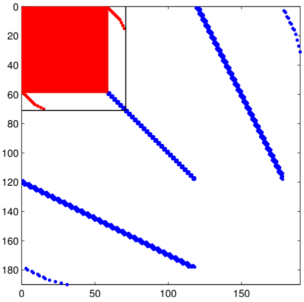
$$\begin{aligned} \min_{\Delta\mathbf{w}} \quad & \frac{1}{2}\Delta\mathbf{w}^T B \Delta\mathbf{w} + \nabla\Phi^T\Delta\mathbf{w} \\ \text{s.t.} \quad & \nabla\mathbf{g}^\top\Delta\mathbf{w} + \mathbf{g} = 0 \\ & \nabla\mathbf{h}^\top\Delta\mathbf{w} + \mathbf{h} \leq 0 \end{aligned}$$

Condensed QP

$$\begin{aligned} \min_{\Delta\mathbf{u}} \quad & \frac{1}{2}\Delta\mathbf{u}^\top M^\top B M \Delta\mathbf{u} + \left(\frac{1}{2}A^\top B M + \nabla\Phi^\top M\right)\Delta\mathbf{u} \\ \text{s.t.} \quad & \nabla\mathbf{h}^\top M\Delta\mathbf{u} + \nabla\mathbf{h}^\top A + \mathbf{g} \leq 0 \end{aligned}$$

requires the factorisation of the matrix:

$$\begin{bmatrix} M^\top B M & \left(M^\top\nabla\mathbf{h}\right)_{\mathbb{A}} \\ \left(\nabla\mathbf{h}^\top M\right)_{\mathbb{A}} & 0 \end{bmatrix} \begin{bmatrix} \Delta\mathbf{u} \\ \tilde{\boldsymbol{\mu}} \end{bmatrix} = \ldots$$

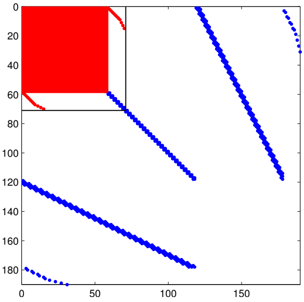


Example: 2 integrators, 2 inputs, $N = 60$ with input & state bounds, condensed

Structure exploitation - Condensing

Iterate QP

$$\begin{aligned} \min_{\Delta \mathbf{w}} \quad & \frac{1}{2}\Delta \mathbf{w}^T B \Delta \mathbf{w} + \nabla \Phi^T \Delta \mathbf{w} \\ \text{s.t.} \quad & \nabla \mathbf{g}^\top \Delta \mathbf{w} + \mathbf{g} = 0 \\ & \nabla \mathbf{h}^\top \Delta \mathbf{w} + \mathbf{h} \leq 0 \end{aligned}$$



Example: 2 integrators, 2 inputs, $N = 60$ with input & state bounds, condensed

Condensed QP

$$\begin{aligned} \min_{\Delta \mathbf{u}} \quad & \frac{1}{2}\Delta \mathbf{u}^\top M^\top B M \Delta \mathbf{u} + \left(\frac{1}{2}A^\top B M + \nabla \Phi^\top M\right)\Delta \mathbf{u} \\ \text{s.t.} \quad & \nabla \mathbf{h}^\top M \Delta \mathbf{u} + \nabla \mathbf{h}^\top A + \mathbf{g} \leq 0 \end{aligned}$$

requires the factorisation of the matrix:

$$\begin{bmatrix} M^\top B M & \left(M^\top \nabla \mathbf{h}\right)_{\mathbb{A}} \\ \left(\nabla \mathbf{h}^\top M\right)_{\mathbb{A}} & 0 \end{bmatrix} \begin{bmatrix} \Delta \mathbf{u} \\ \tilde{\boldsymbol{\mu}} \end{bmatrix} = \dots$$

Large/sparse QP $\rightarrow$ **small/dense** QP, but...

- Condensing is unstable for locally unstable systems
- Dense factorization has cubic complexity

Unfavorable for

- unstable systems
- many inputs
- long horizon