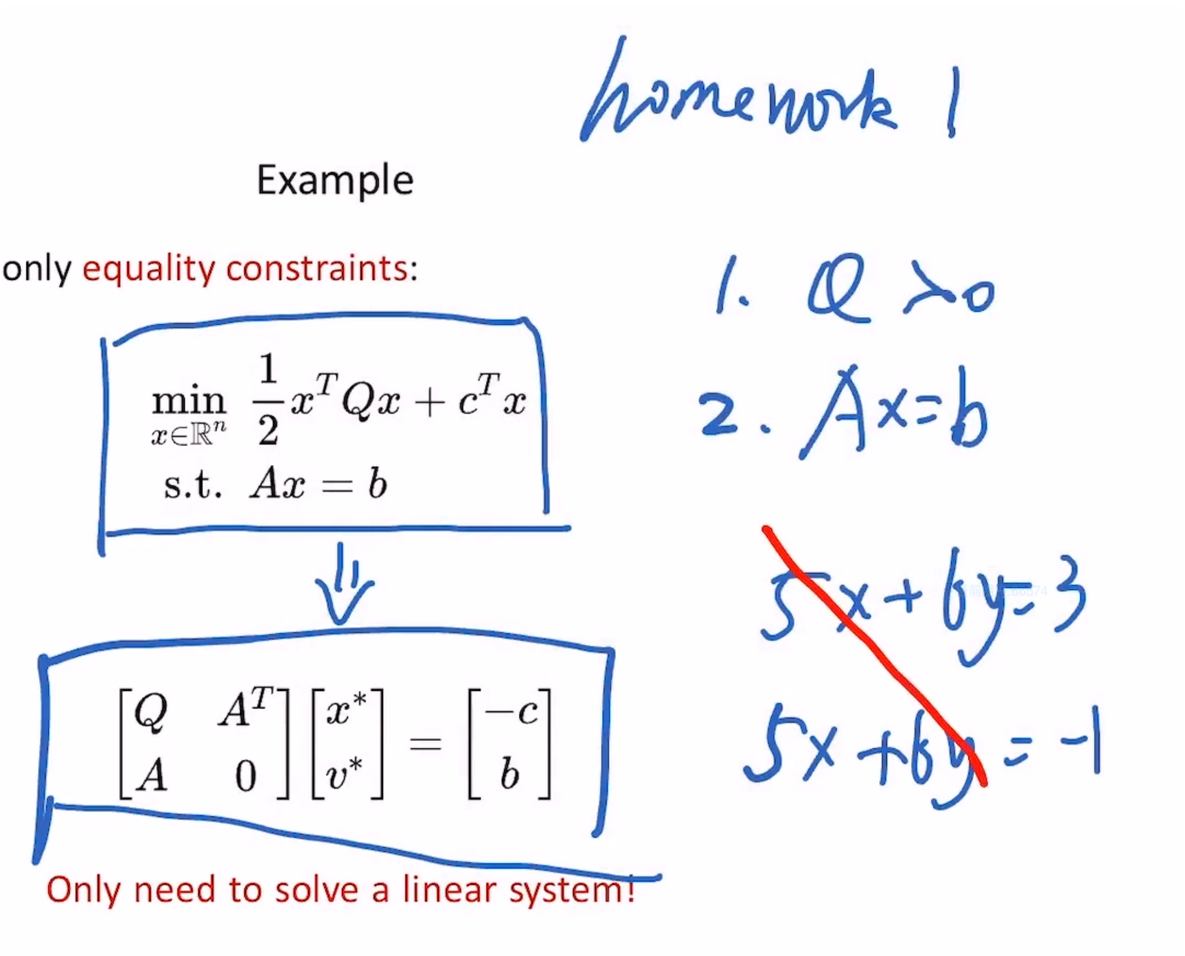
hw-l3-numerical optiamization of robot

Task 1



Q 矩阵严格正定情况，等式约束 KKT 条件推导

首先获得 KKT 相关条件，由于仅仅存在线性等式约束

故存在

- 原始可行性条件

$$Ax^* = b$$

- 稳定性条件

设拉格朗日乘子为 v 目标函数为

$$\frac{1}{2}x^TQx + c^Tx + v(Ax - b)$$

对 x 求偏导, 设 KKT 点为 x^* ， 那么满足 kkt 条件为

$$\frac{\partial}{\partial x}\left(\frac{1}{2}x^TQx + c^Tx + v^T(Ax - b)\right)\Bigg|_{x=x^*} = 0$$

整理得

$$Qx^* + A^Tv^* = -c$$

满足上述条件的线性等式方程为

$$\begin{bmatrix} Q & A^T \\ A & 0 \end{bmatrix}\begin{bmatrix} x^* \\ v^* \end{bmatrix} = \begin{bmatrix} -c \\ b \end{bmatrix}$$

task2 low-Dimensional QP

task2 low-Dimensional QP

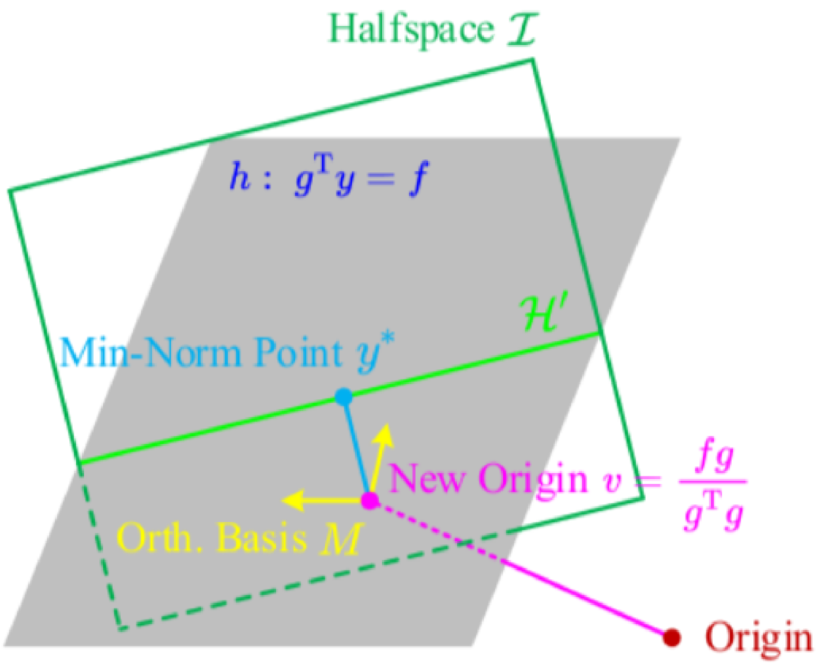
第 1 列

$\text{LowDimMinNorm}(\mathcal{H})$
    if $\dim(\mathcal{H}) = 1$
        $y \leftarrow \text{OneDimMinNorm}(\mathcal{H})$
    end if
    $\mathcal{I} \leftarrow \{\}$
    for $h \in \mathcal{H}$ in a random order
        if $y \notin h$
            $\boxed{\{M, v, \mathcal{H}'\} \leftarrow \text{HouseholderProj}(\mathcal{I}, h)}$
            $y' \leftarrow \text{LowDimMinNorm}(\mathcal{H}')$
            $y \leftarrow My' + v$
        end if
        $\mathcal{I} \leftarrow \mathcal{I} \cup \{h\}$
    end for
return y

第 2 列

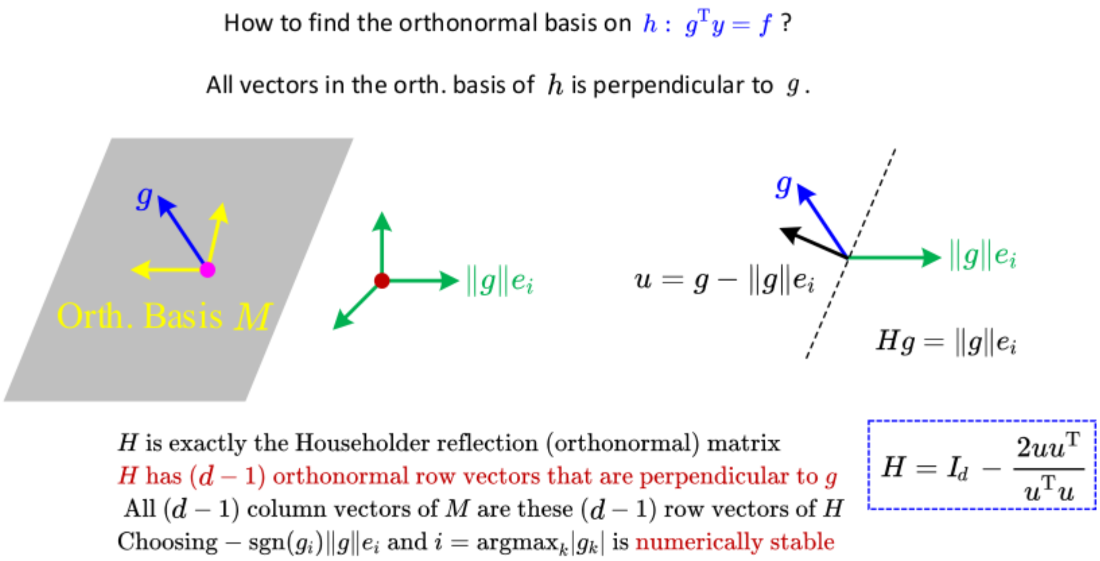
处理的核心逻辑是通过投影降低维数。

首先判断约束是否为一维，为一维直接去在满足约束下找到即可，否则需要对于违反当前 $x \notin h$ 最优解的 d 维新约束 $g^T y \le f$,要将之前的约束进行投影，在 h 平面上：$g^T y = f$， 也就是用 h 上 的一组 d-1 维的标准正交基。



$$\|My'\|^2 + \|v - o\|^2 = \|y'\|^2 + \|v - o\|^2 = \|y^*\|^2$$

How to find the orthonormal basis on $h: \ g^{\mathrm{T}} y = f$?



Householder 投影具体操作：

- 首先找到新的原点 v，投影在 h 平面上
- 寻找一组标准正交基

进行 Household 反射变换

为了数值稳定性， 镜面的法向向量 $u = g + sgn(g_i)||g||e_i$

其中 $i = argmax_k|g_k|$

则根据向量相加几何特性可知

$$g - 2g^{T}\frac{u}{\|u\|}\frac{n}{\|n\|} = -\operatorname{sgn}(g_i)\|g\|e_i$$

其中 n=u^T

$$H = I_d - \frac{2uu^{T}}{u^{T}u}$$

H 便是反射矩阵。

降维的约束 H'=Hh

源码:

C++

```
  ////////////////////////////////// HOMEWORK START //////////////////////////
  // TODO
//                 id=argmax(gi)
  cpy<d>(plane_i, reflx);
  double g_norm = std::sqrt(sqr_norm<d>(plane_i));
  reflx[id]+=plane_i[id]<0.0 ? -g_norm:g_norm;//u=g+sigan(g_i)|g|
  double h=-2.0/ sqr_norm<d>(reflx);
  for(int j=0;j!=i;j=next[j]){
      const double *halfspace = halves + (d + 1) * j;
      double *proj_halfspace = new_halves + d * j;
      double term=h*dot<d>(halfspace,reflx);
      int temp=0;
      for(int k=0;k<d+1;k++){
          if(k==id) continue;
          proj_halfspace[temp++]=halfspace[k]+term*reflx[k];
      }
      proj_halfspace[d-1]=dot<d>(new_origin,halfspace)+halfspace[d];
  }
```

Result

```Python
optimal sol: 4.11111 9.15556 4.50022
optimal obj: 201.14
cons precision: 8.88178e-16
```

task3 PHR ALM NMPC

$$\begin{array}{cl}\min\limits_{s_1,\ldots,s_N,u_0,\ldots,u_N} & J\left(s_1,\ldots,s_N,u_0,\ldots,u_N\right)\\ \text{s.t.} & F\left(s_k,u_k\right)=s_{k+1},\forall i\in\{0,\ldots,N\}\\ & G\left(s_k,u_k\right)\leq 0,\quad \forall i\in\{0,\ldots,N\}\end{array}$$

其中， $s_k=[x_k,y_k,\phi_k,v_k]^T$ 为状态变量， $u_k=[a_k,\delta_k]^T$ 为输入变量，
$F(s_k,u_k)=f(s_k,u_k)\tau+s_k$ 为离散运动学模型:如下所示

RK45(龙格库塔)也可以用

$$\begin{cases} \dot{x}=v\cos(\phi) \\ \dot{y}=v\sin(\phi) \\ \dot{\phi}=v\tan(\delta)/L \\ \dot{v}=a \end{cases}$$

$G(s_k,u_k)\leq 0$ 为状态与输入约束的集成, 表示为:

$$\begin{aligned} & a_{\min}\leq a_k\leq a_{\max} \\ & \delta_{\min}\leq\delta_k\leq\delta_{\max} \\ & v_{\min}\leq v_k\leq v_{\max} \\ & k\in\{0,1,\ldots,N\} \end{aligned}$$

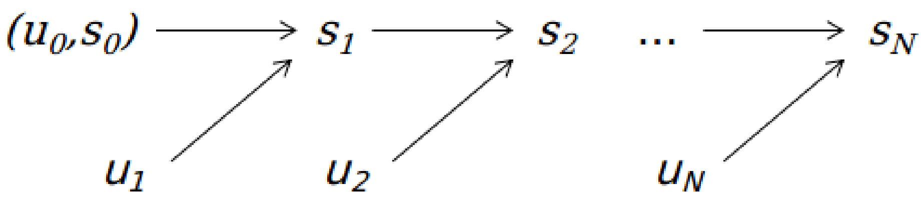
目标函数为

$$J(s_1,\ldots,s_N,u_0,\ldots,u_N):=\sum_{k=1}^{N}\left[\left(x_k-x_k^{\text{ref}}\right)^2+\left(y_k-y_k^{\text{ref}}\right)^2+w_v\left(a_k-a\right.\right.$$

优化量分别为横向误差与纵向误差,控制的相邻时刻变化量的,

todo: 横向误差与纵向误差的权重

为了快速求解,通过运动学模型将状态量转化为与输入相关的量,从而消除这个等式约束,
如下图所示



这样 u 可以看作唯一的优化变量,

$$F(s_k,u_k)=s_{k+1},\ \forall\, i\in\{0,\dots,N\} \xleftrightarrow{u_{0:N}=(u_0,\dots,u_N)} s_k(u_{0:N}),\ \forall\, i\in\{1,\dots,N\}$$

于是目标函数转化为

$$\begin{aligned}&\min_{u_{0:N}} J\left(s_1\left(u_{0:N}\right),\dots,s_N\left(u_{0:N}\right),u_{0:N}\right)\\ &\text{s.t. } G\left(s_k\left(u_{0:N}\right),u_k\right)\le 0,\forall i\in\{0,\dots,N\}\end{aligned}$$

在不等式约束引入辅助变量 s ，使其转化为等式约束,如下所示

$$G\left(s_k\left(u_{0:N}\right),u_k\right)+[s]^2=0$$

可由 PHR-ALM 得到相应的增广拉格朗日函数

$$\mathcal{L}_\rho(u_{0:N},\mu):=J\left(s_1\left(u_{0:N}\right),\dots,s_N\left(u_{0:N}\right),u_{0:N}\right)+\frac{\rho}{2}\left\|\max\left[G\left(s_k\left(u_{0:N}\right),\right.\right.\right.$$

第二项对 u 的导为:

$$\frac{d}{du}\left(\frac{\rho}{2}\left\|\max\left[G\left(s_k\left(u_{0:N}\right),u_k\right)+\frac{\mu}{\rho},0\right]\right\|^2\right)=\begin{cases}\rho(G+\frac{\mu}{\rho})\frac{dG}{du} & \text{, if } G\left(s_k\left(u_{0:\right.\right. \\ 0 & \text{, otherwise}\end{cases}$$

外层采用 lbfgs 算法更新 ALM 的优化变量 $u_{0:N} \leftarrow \underset{u_{0:N}}{argmin}\, \mathcal{L}_\rho(u_{0:N}, \mu)$ 。

对于内层的乘子更新按以下方式:

$$\begin{cases} \mu \leftarrow \max[\mu + \rho G\left(s_k\left(u_{0:N}\right), u_k\right), 0] \\ \rho \leftarrow \min[(1+\gamma)\rho, \beta] \end{cases}$$

对于存在不等式约束的 PHR-ALM 问题,需要满足稳定性条件和对偶可行性条件

迭代停止条件为:

$$\left\| \max\left[G(s_k(u_{0:N}, u_k), -\frac{\mu}{\rho} \right] \right\|_\infty \Bigg] < \epsilon_{\text{cons}} \ ,$$
$$\left\| \nabla_x \mathcal{L}_\rho(u_{0:N}, \mu) \right\|_\infty < \epsilon_{\text{prec}}$$

结果

见 mpc1.mp4