1-xrudder本身的控制难点

- 控制分配问题

2-xrudder最近的研究成果，分为不考虑容错和考虑容错
3-lqr的xrudder控制有没有
4-滑膜控制xrudder
5-仿真工具箱学习
6-地形跟踪auv（避免碰撞）

X-rudder 的类别

- 固定的 X 舵(Fixed X rudder)：

  这种 X 形舵的方向舵不能操作。由 Tao（2002 年）设计的名为 CR-02 的 AUV 就配备了这种 X 形舵。AUV 的运动控制由垂直和水平推进器以及四个主推进器完成。这种 X 舵不提供控制能力，只提供了额外的稳定性。

- 对角连接 X 舵(Diagonal-linkage X rudder)：

  这种 X 型舵以对角舵为单位驱动。例如，吉田设计的名为 "Yumeiruka "的自动潜航器（2013 年）在船头和船尾分别安装了两个 X 形舵。AUV 可以利用它们产生相应的力矩来控制偏航和俯仰。2013 年 3 月，AUV 在相模湾进行了为期 15 天的海上试验。

- **独立 X 舵(Independent X rudder)**：

  这种 X 型舵的舵面可以独立操作。瑞典的哥特兰级（Gotland-class Submarine）、德国的 212 A 型（Type 212 Submarine）和日本的双龙级（Sōryū-class Submarine）等一些国家的潜艇通常装备有这种 X 型舵。但很少有公开文献介绍其控制方法。一些鱼雷也使用了这种 X 形舵（Tiansen，2007 年），在鱼雷控制中通常同时操作四个舵。在这种 X 形舵中，每个舵的操作都会引起飞行器偏航和俯仰的变化。为 AUV 提供更多的控制方式意味着为 AUV 提供更多的安全性。

X-rudder 和 Corss-rudder 的区别

- 标准化和去标准化

  对于十字舵 AUV ，将控制器的输入标准化(限定到-1到+1之间)，再将其输出去标准化，这两个步骤都是线性的，设计起来比较简单。这样做的好处是通过微小的控制律调整就可以引起较大的输出变化。

  $$\begin{cases} \delta_v = CTR_{yaw.cross} \times \delta_{v.\max} \\ \delta_h = CTR_{pitch.cross} \times \delta_{h.\max} \\ -1 \le CTR_{yaw.cross},\ CTR_{pitch.cross} \le +1 \end{cases} \tag{5}$$

  可以看到，对于十字舵，拥有水平和垂直舵面两个执行器，属于欠驱动，所以只控制两个量，达到控制舰体的目的，控制输出 $CTR_{yaw,cross}, CTR_{pitch,cross}$ (偏航角和俯仰角，对应竖直舵和水平舵)，可以直接对应到两个舵，去标准化非常简单。

  对于 X 舵，其四个舵面都可以独立控制，属于过驱动，所以可以控制完整的三个角度(yaw, pitch, roll)。而四个舵面都可以影响这三个量，所以不能像十字舵那样简单的去标准化。

- 控制分配

注意公式 (5)，由于水平舵和垂直舵刚好与俯仰角、方向角对应，所以去标准化后可以直接得到对应的舵角。然而对于 X 舵，去标准化后只能得到 $\delta_{VR} = [\delta_{VR,roll}, \delta_{VR,pitch}, \delta_{VR,yaw}]^T$，这表示三个虚拟舵角用于控制三个角度。接下来就要进行控制分配，用四个舵面来完成三个角度的控制。

> Zhang, Ying-hao et al. “Design and simulation of X-rudder AUV's motion control.” *Ocean Engineering* 137 (2017): 204-214.

以上文章提出了去标准化和控制分配（伪逆）的方法。

运动学建模和动力学建模

对于飞行器或水下航行器，以垂直面控制为例，上层给出的参考指令 $y_d = [x_d, z_d, \theta_d]$，表示目标坐标和俯仰角。系统状态 $x = [x, z, \theta, u, w, q]$，后三个参数表示浪涌速度，升沉速度，俯仰角速度。

对于线性系统：

$$\begin{aligned} x^+ &= Ax + Bu \\ y &= Cx + Du, \end{aligned} \tag{1}$$

在上层给出参考指令 y_d 后，可以计算出对应的x_d，后续直接跟踪 x_d 即可。这种方法相当于将运动学和动力学方程融合在一起，如果只对状态 x 的后三个参数 u, w, q 进行控制，也可以完成跟踪任务，因为它们就是由前三个参数的目标值计算出来的。当后三个参数达到目标值时，前三个值也会达到。

将运动学和动力学方程融合在一起的好处在于能够抗干扰，防止动力学参数达到目标值时，运动学参数没有达到。

然而对于 AUV，其运动学方程和动力学方程为：

$$\begin{cases} \dot{x} = u\cos\theta + w\sin\theta \\ \dot{z} = -u\sin\theta + w\cos\theta \\ \dot{\theta} = q \end{cases} \tag{1}$$

$$\begin{cases} \dot{u} = f_u(u, w, q) + g_u X_T + d_u \\ \dot{w} = f_w(u, w, q) + g_w \tau_w + d_w \\ \dot{q} = f_q(u, w, q) + g_q \tau_q + d_q \end{cases} \tag{3}$$

对于该系统，给出参考指令 $[x_d, z_d, \theta_d]$ 后，可以计算出 u, w, q 的目标稳态。然而系统是非线性的，将两个模型融合在一起只会加剧非线性的程度，所以分为两层进行控制：

1. 根据给定的 $[x_d, z_d, \theta_d]$ 和运动学方程计算出 $[u_d, w_d, q_d]$
2. 根据 $[u_d, w_d, q_d]$ 计算出控制输出。

这种分两层控制的方法适用于非线性控制，以下论文设计了这种模式的跟踪控制算法：

> Xia, Yingkai et al. “Optimal robust trajectory tracking control of a X-rudder AUV with velocity sensor failures and uncertainties.” *Ocean Engineering* 198 (2020): 106949.

> Elmokadem, Taha et al. “Trajectory tracking sliding mode control of underactuated AUVs.” *Nonlinear Dynamics* 84 (2015): 1079-1091.

非线性 MPC

使用非线性 MPC 可以避免分两层计算进行控制，只进行一层优化计算。

> Shen, Chao and Yang Shi. “Distributed implementation of nonlinear model predictive control for AUV trajectory tracking.” *Autom.* 115 (2020): 108863.

本文研究了自主潜水器（AUV）的轨迹跟踪控制。我们研究了非线性模型预测控制（NMPC）方法，寻找可能的方法来减轻沉重的计算负担。利用 AUV 运动的动态特性，我们开发了新型分布式 NMPC 算法。通过将原始优化问题适当分解为更小的子问题，然后以分布式方式求解，可以显著减少预期浮点运算（flops）。我们的研究表明，在分解后的子问题中采用拟议的收缩约束可以保证 AUV 轨迹的收敛性。我们还证明了递归可行性和闭环稳定性。利用所保证的稳定性，进一步开发了一种实时分布式实现算法，以自动权衡控制性能和计算复杂性。对猎鹰 AUV 模型进行的大量仿真研究证明了所提方法的有效性和鲁棒性。

控制分配问题

> Wang, Wenjin et al. “A Fault-tolerant Steering Prototype for X-rudder Underwater Vehicles.” *Sensors (Basel, Switzerland)* 20 (2020): n. pag.

在上面文献的 Introduction 中写道：上述 xAUV **仍采用虚拟水平舵和垂直舵指令，需要将其转换为 X 舵指令**。在文献[6,7]中，首先利用 PID 控制器产生水平舵和垂直舵指令，然后利用指令变换来计算 X-舵指令。然而，这种指令变换可能会产生不期望的滚转扭矩，影响平移和旋转稳定性。为了解决这一问题，Zhang 等人[8]通过引入最小能量作为准则，将变换表述为一个约束最小化问题，而该优化问题的伪逆解正是[6,7]中描述的变换公式。虽然现有文献对控制分配问题进行了研究，但大多数研究都是通过模拟进行的，现有文献都没有进行现场测试。

在 `X-rudder 和 Corss-rudder 的区别` 一节中，引用的 `Design and simulation of X-rudder AUV's motion control.` 文献中，虚拟舵不再是水平舵和垂直舵，而是控制三个角度（roll，pitch，yaw）的虚拟舵。

不管采用哪种虚拟舵的设计模式，到最后都避免不了控制分配问题，所以 X-rudder 的核心问题是如何进行控制分配问题。当前文献对于执行器故障的容错控制，也是如何针对故障进行对应的控制分配。

无故障控制分配

伪逆法

（2）伪逆法。控制分配技术中应用最广，最具代表性的当属伪逆控制分配，伪逆法最早出现于飞行器控制分配问题中[31]，2006 年，T.I. Fossen 等[32]将伪逆法用于水下机器人的控制分配。伪逆法具有计算量小和易于实现等优点,但未考虑执行机构的饱和（位置和偏转速率）约束限制，也不能实现可达集的全覆盖。1995 年，Bordignon 等[33]通过 F-18HARV 实验得出伪逆法只可实现可达集中的部分虚拟指令，最好的情况也只能达到其中的 42.7%。另外伪逆法为单一目标优化解，不能用于多目标优化求解，也不能用于非线性的控制分配问题求解。2020 年，Jagadeesh 等[34]将加权伪逆法用于水下机器人的控制分配，实现了执行机构的使用权重，但仍未解决伪逆法的缺陷。

如果矩阵B行满秩，行向量线性无关，$\mathbf{B}\boldsymbol{u}=\boldsymbol{\tau}$为欠定方程组，此时存在零个或者无穷多个解，那么可以利用右逆直接求得一个解：

$$\boldsymbol{u}=\mathbf{B}^{\mathrm{T}}(\mathbf{B}\mathbf{B}^{\mathrm{T}})^{-1}\boldsymbol{\tau}=\mathbf{B}^{+}\boldsymbol{\tau} \tag{3-16}$$

在上式中，$\mathbf{B}^{+}$为B的右逆矩阵。

在X舵AUV的正常控制分配中，伪逆控制分配，即是求得其右逆解，可以通过证明[66]，矩阵的伪逆是唯一的，欠定方程组的伪逆解即是最小范数解，是方程组的一个可行解，也是一个最优解。对于如下的优化问题：

$$\begin{cases}\min & f=\boldsymbol{u}^{\mathrm{T}}\mathbf{W}\boldsymbol{u}\\ s.t. & \mathbf{B}\boldsymbol{u}-\boldsymbol{\tau}=0\end{cases} \tag{3-17}$$

伪逆法求得的解就是上式的最优解，因此，伪逆法是一种单一目标优化的控制分配方法，对于多目标优化问题，伪逆法不再适用，但可以采用非线性规划法。

顺序二次规划SQP & 改进二次规划 IQP

AUV在水下工作时，其工作任务时长受携带的能源限制，不能无限期工作，而AUV携带的能源受其自身体积的约束，能源就显得十分珍贵，因此，将能源的消耗量作为控制分配的一个优化目标。AUV控制分配的结果体现在舵的执行转角上，舵角分配的越小，能源的消耗越小。在考虑能源消耗的同时，还必须考虑控制分配的精确度，控制分配的结果越精确，AUV的运动控制响应越好，就越容易控制，因此将能源消耗和控制分配精度作为控制分配的优化目标，优化目标函数定义如下：

$$\min\ f=(1-\vartheta)\ \boldsymbol{u}^{\mathrm{T}}\mathbf{w}_1\boldsymbol{u}+\vartheta(\tau-\mathbf{B}\boldsymbol{u})^{\mathrm{T}}\mathbf{w}_2(\tau-\mathbf{B}\boldsymbol{u}) \tag{3-38}$$

等式约束：$\boldsymbol{\tau}=\mathbf{B}\boldsymbol{u}$

不等式约束：$\bar{\boldsymbol{u}}^2-\boldsymbol{u}^2\geq 0$

上式中，$\mathbf{w}_1$，$\mathbf{w}_2$为对角权值矩阵，调节其大小，可以改变对应舵的使用率及优先使用权。舵偏转位置约束$\boldsymbol{u}_{\min}=[-25^{\circ},-25^{\circ},-25^{\circ},-25^{\circ}]$和$\boldsymbol{u}_{\max}=[25^{\circ},25^{\circ},25^{\circ},25^{\circ}]$，$\mathbf{w}_1$和$\mathbf{w}_2$取单位对角阵。$\vartheta=0.5$，是考虑能源消耗问题与分配精确问题同等重要。

故障情况控制分配

力矩再分配

（1）单舵缺失

当X舵AUV某一个舵叶全部丢失时，那么对应的舵将不再产生力矩，实际产生的力矩与期望的控制力矩将会存在误差，会影响AUV的正常工作。以4号尾舵缺失为例，期望力矩由剩下的1、2、3号舵产生，此时，控制分配效率矩阵为：

$$\mathbf{B}_f=\begin{bmatrix}-0.0221v^2 & -0.0221v^2 & -0.0221v^2\\ 0.0729v^2 & -0.0729v^2 & -0.0729v^2\\ -0.0729v^2 & -0.0729v^2 & 0.0729v^2\end{bmatrix}$$

期望力矩$\boldsymbol{\tau}$与剩余执行机构输入指令向量$\boldsymbol{u}_f$的关系如下：

$$\boldsymbol{\tau}=\boldsymbol{B}_f\boldsymbol{u}_f \tag{4-25}$$

此时，$\boldsymbol{\tau}$与$\boldsymbol{u}_f$的维度相同，求解式（4-25）即是一个正常的线性方程组求解问题，可以很容易得到可行解。

（1）单舵卡舵

卡舵与舵叶缺失的情况不同，卡舵时，舵叶仍然产生力矩，只不过其力矩是固定大小，无论舵机输入指令为多少，都只输出固定的力矩大小，方向也一定。假设 4 号舵发生卡舵，卡舵舵角为u_{t4}，4 号舵的固定力矩$\boldsymbol{\tau}_f$输出为：

$$\boldsymbol{\tau}_f=\begin{bmatrix} -0.0221v^2 \cdot u_{t4} \\ 0.0729v^2 \cdot u_{t4} \\ 0.0729v^2 \cdot u_{t4} \end{bmatrix} \tag{4-28}$$

单舵卡舵情况下的控制分配效率矩阵为：

$$\mathbf{B}_f=\begin{bmatrix} -0.0221v^2 & -0.0221v^2 & -0.0221v^2 \\ 0.0729v^2 & -0.0729v^2 & -0.0729v^2 \\ -0.0729v^2 & -0.0729v^2 & 0.0729v^2 \end{bmatrix} \tag{4-29}$$

此时，期望力矩$\boldsymbol{\tau}$与剩余执行机构输入指令向量$\boldsymbol{u}_f$的关系为：

$$\boldsymbol{\tau}-\boldsymbol{\tau}_f=\boldsymbol{B}_f\boldsymbol{u}_f \tag{4-30}$$

非线性规划NLP

卡舵故障映射到虚拟舵：

$$\begin{bmatrix} \bar{\delta}_r & \bar{\delta}_s \end{bmatrix}^T = B\begin{bmatrix} \delta_1 & \delta_2 & \delta_3 & \delta_4 \end{bmatrix}^T \tag{32}$$

where the configuration matrix B is given as:

$$B=\frac{1}{4}\begin{bmatrix} 1 & -1 & 1 & -1 \\ -1 & -1 & -1 & -1 \end{bmatrix}$$

(32) 式中 的 $\bar{\delta}$ 表示故障舵映射到虚拟舵面的作用。那么非线性优化问题可以表示为：

$$J=(1-\xi)W\|(\tau_d-\tau)\|_2+\xi\|u^t\|_2 \tag{33}$$

where minimization of allocation errors and control inputs are considered as criterions. In Equation (33), $\tau_d=\begin{bmatrix} \delta_r & \delta_s \end{bmatrix}^T$ is the virtual rudder commands produced by DIFC, $\tau=\begin{bmatrix} \bar{\delta}_r & \bar{\delta}_s \end{bmatrix}^T$ is the actual control efforts derived from Equation (32), $u=\begin{bmatrix} \delta_1 & \delta_2 & \delta_3 & \delta_4 \end{bmatrix}^T$ is the actuator's control inputs to be solved, W is the weight matrix for balancing horizontal and vertical control, and ξ is a parameter for balancing criterion components. When J is optimized to its minimum, control allocation error $(\tau_d-\tau)$ is minimum and rudder deflection u yields:

$$\begin{cases} u\in[-\kappa,\kappa] \\ |\dot{u}|\in[-\gamma,\gamma] \\ \delta_i=D,\ \text{where} i \text{represents the index of faulty rudder} \end{cases} \tag{34}$$

通过约束条件来限制卡死舵面。

地形探测

seafloor mapping

seafloor observation