

# MPC Reference Governor FTC Learning Log

## 一、研究目的

2024年5月27日

探讨在执行器卡死情况下，AUV 的控制算法。

由于 MPC 可以显式地处理约束问题，所以可以通过将故障设置为约束的形式来处理卡斯故障。

然而由于故障导致的控制性能下降，会引起参考信号无法跟踪（或者勉强跟踪）的情况。出现这种情况的直接原因是控制性能下降，本质原因是参考指令过于激进，导致执行器饱和以后仍然无法达到参考目标，在 MPC 中的表现就是求解器无解（recursive feasibility 丧失）。

在这种情况下，会导致系统失稳，所以需要使用 Reference Governor 求出最优的满足约束的参考指令，这能够保证 recursive feasibility。递归可行性得到保证后，就可以保证系统的稳定性，并且跟踪性能最优。

Limón D, Alvarado I, Alamo T, et al. MPC for tracking piecewise constant references for constrained linear systems[J]. Automatica, 2008, 44(9): 2382-2387.

Limon D, Ferramosca A, Alvarado I, et al. Nonlinear MPC for tracking piece-wise constant reference signals[J]. IEEE Transactions on Automatic Control, 2018, 63(11): 3735-3750.

## 二、研究对象

2024年5月27日

选用 x-rudder AUV，因为执行器够多，能够补偿容错。

仿真环境中使用 Fossen 的 remus100，并把十字舵替换为 X 舵。

Fossen T I. Handbook of marine craft hydrodynamics and motion control[M]. John Wiley & Sons, 2011.

Zhang Y, Li Y, Sun Y, et al. Design and simulation of X-rudder AUV's motion control[J]. Ocean Engineering, 2017, 137: 204-214.

## 三、仿真场景

2024年5月27日

使用 Fossen 的 `Python Vehicle Simulator` :

<https://github.com/cybergalactic/PythonVehicleSimulator>

对 AUV 进行深度控制，较大尺度的深度变化，能体现出 RG 的重要性。

同时对姿态约束：纵倾角不能太大。这样符合实际应用需求，并且约束更强，更可以体现 RG 的重要性。

目前可以确定的是：使用 `Reference Governor + MPC` 能够保证 MPC 的 `迭代可行性`，加上文中提出的 `不变集、终端代价` 的计算，从而保证系统 `稳定性`。（RG 更多的是体现对 MPC 算法的改进，没有应用层面的体现）

如果能够在外环就对参考指令进行调节（这里的功能不是为了保证迭代可行性，而是为了保证后续的参考指令能够到达），更能够体现 AUV 在海底运行的实际情况。

Yan Z, Gong P, Zhang W, et al. Model predictive control of autonomous underwater vehicles for trajectory tracking with external disturbances[J]. Ocean Engineering, 2020, 217: 107884.