

Reference Management for Fault-Tolerant Model Predictive Control

Questions

1. 从路径跟踪到轨迹跟踪，这篇文章涵盖了哪方面？
2. 如何进行轨迹跟踪？
3. Reference Management 体现在哪里？

Abstract & Introduction

该篇文章提出了一种用于容错模型预测控制的参考管理技术。要跟踪的参考指令首先要通过一组**静态容许指令**进行过滤。这组指令表达了机身达到新平衡点的物理能力，它是通过考虑**输入和受控系统状态的约束**而构建的。指令集采用多面体形式，由线性不等式定义。在稳态时获得状态和控制向量，以保证模型预测控制调节器所面临的约束数值问题的可行性。

考虑到目前 MPC 和容错控制器的局限性，该文提出了一种新的 MPC 参考管理方法。这种方法将计算受约束控制系统的静态容许集的方法 [15] 与称为可行目标跟踪 MPC 的技术 [19] 相结合。在输入和/或输出约束条件下，完整的控制系统能够**以静态和动态两种方式过滤需要遵循的参考**。首先检查参考指令是否与新的平衡点对应（静态），然后再次过滤参考值，以保证约束优化问题的可行性（动态）。

Set of Statically Admissible Commands

离散时域模型：

$$\begin{aligned} x_{k+1} &= \Phi x_k + \Gamma u_k + \Gamma_d d_k & d_{k+1} &= d_k \\ y_k &= E x_k & z_k &= H y_k \end{aligned} \quad (1)$$

where $x_k \in \mathbb{R}^n$ is the state vector, $u_k \in \mathbb{R}^m$ is the control vector, $y_k \in \mathbb{R}^p$ is the vector of observed variables, $d_k \in \mathbb{R}^q$ is the vector of disturbances and $z_k \in \mathbb{R}^q$, $q \leq m$, is the vector of controlled variables. The disturbance vector is included to model mismatches between the plant and the nominal model as well as external disturbances acting on the plant. The pair (E, Φ) is assumed to be detectable with E full row rank. Additionally, the disturbance vector is assumed to be estimated by a proper observer. In this work, the disturbance model is simply a constant value.

控制目标：

$$z_k \xrightarrow[k \rightarrow \infty]{} r_{ss} \quad (2)$$

考虑系统约束以及扰动，对稳态值的要求如下：

$$\begin{aligned}
(\Phi - I)x_{ss} + \Gamma u_{ss} + \Gamma_d d_k &= 0 \\
HEx_{ss} &= r_{ss} \quad u_{ss} \in \mathbf{U} \quad x_{ss} \in \mathbf{X}
\end{aligned} \tag{3}$$

故障情况下，或者扰动 d_k 较大时，很可能出现 (3) 式中约束和等式无法同时满足的情况，此时可以通过求解优化问题得到最优解：

$$\begin{aligned}
\min_{x_{ss}, u_{ss}} J(r_{ss}, d_k) &= (HEx_{ss} - r_{ss})^T Q_{ss} (HEx_{ss} - r_{ss}) + u_{ss}^T R_{ss} u_{ss} \\
\text{subject to: } (\Phi - I)x_{ss} + \Gamma u_{ss} + \Gamma_d d_k &= 0 \\
u_{ss} &\in \mathbf{U} \quad x_{ss} \in \mathbf{X}
\end{aligned} \tag{4}$$

where Q_{ss} and R_{ss} are weighting matrices.

通过求解公式 (4)，在每个采样瞬间解决这个问题，就能得到与预期输出目标相对应的稳态变量。通过求解每对 (r_{ss}, d_k) ，可以确定可实现的稳态和相关输出的完整集合。然而，该集合的构建在数值上代价高昂，而且实际使用时只需要获得该集合边界的近似值，根本不需要计算出完整的集合（这也证明了数值计算代价太高）。为了提供一个更好、更实用的可实现集合，该文建议使用**投影算法**。

多面体是有限个封闭半空间的交集：

$$\mathbf{P} = \{v \in \mathbb{R}^n | L_c v \leq k_c\} \tag{5}$$

给定一个多面体 $P \subset V \times Y$ ，其中 V 和 Y 都是子空间， P 在 V 上的投影为：

$$\pi_V \mathbf{P} = \{v \in V | \exists y \in Y, (v, y) \in \mathbf{P}\} \tag{6}$$

而 P 在向量 $r \in R_p$ 处的切片 (slice) 定义为：

$$\sigma_r \mathbf{P} = \{v \in \mathbb{R}^n | (v, r) \in \mathbf{P}\} \tag{7}$$

这两个定义的图解：

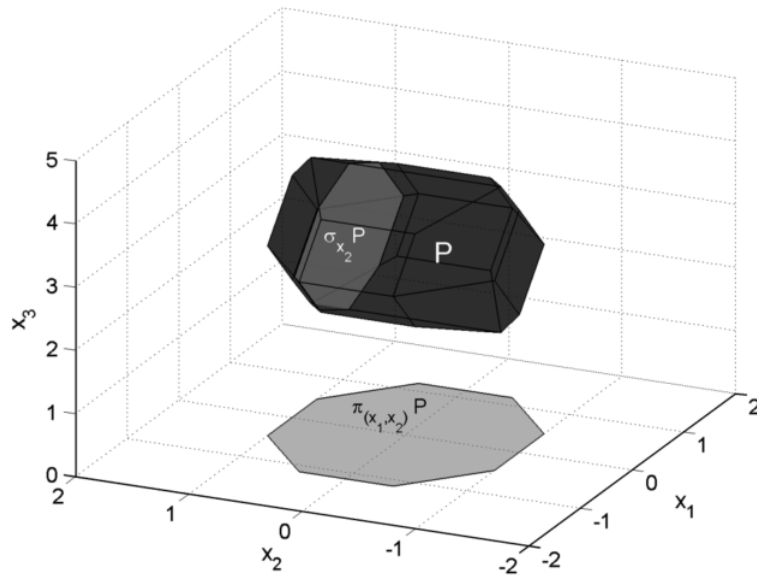


Fig. 1 Projection and slice of a polyhedron.

简单来说，**投影会降维，而切片不会降维**（将某个变量定位固定值）。

所提技术的关键概念是构建与稳态约束条件相对应的多面体 P_{ss} ，并将 P_{ss} 投影到所需的子空间上。如果关注的是稳态下的性能特征，则可以构建一个包含状态矢量的子空间。如果考虑的是操纵性问题，则空速、飞行路径角和转弯率等输出可定义另一个子空间。

通过引入任意小的公差向量 ϵ_x 和 ϵ_z ，状态空间方程和目标方程所代表的等式约束可以转换为不等式约束：

$$\begin{aligned} (\Phi - I)x_{ss} + \Gamma u_{ss} + \Gamma_d d_k &\leq \epsilon_x I \\ (\Phi - I)x_{ss} + \Gamma u_{ss} + \Gamma_d d_k &\geq -\epsilon_x I \\ HEx_{ss} - r_{ss} &\leq \epsilon_z I \\ HEx_{ss} - r_{ss} &\geq -\epsilon_z I \\ u_{ss} &\in \mathbf{U} \quad x_{ss} \in \mathbf{X} \end{aligned} \quad (8)$$

那么就可以用一个集合来表示：

$$\mathbf{P}_{ss} = \{v \in \mathbb{R}^{n+m+2q} | L_{ss} v \leq k_{ss}\} \quad (9)$$

where $v = [x_{ss}^T \quad u_{ss}^T \quad r_{ss}^T \quad d_k^T]^T$ and

$$L_{ss} = \begin{bmatrix} C_c & 0 & 0 & 0 \\ -C_c & 0 & 0 & 0 \\ 0 & I & 0 & 0 \\ 0 & -I & 0 & 0 \\ 0 & 0 & 0 & I \\ 0 & 0 & 0 & -I \\ \Phi - I & \Gamma & 0 & \Gamma_d \\ -\Phi + I & -\Gamma & 0 & -\Gamma_d \\ HE & 0 & -I & 0 \\ -HE & 0 & I & 0 \end{bmatrix}, \quad k_{ss} = \begin{bmatrix} x_{\max} \\ x_{\min} \\ u_{\max} \\ u_{\min} \\ d_{\max} \\ d_{\min} \\ \epsilon_x \\ -\epsilon_x \\ \epsilon_z \\ -\epsilon_z \end{bmatrix} \quad (10)$$

where x_{\max} , x_{\min} , u_{\max} , u_{\min} , d_{\max} and d_{\min} are the bounds of the state, control and disturbance vectors, respectively. The projection of \mathbf{P}_{ss}

假定扰动可通过某个观测器获得。因此，通过对估计的干扰值进行**切片**运算，就可以轻松计算出 r_{ss} 的极限值（因为有降维）。切片运算**简化了线性不等式**，当某些变量先验已知时，线性不等式定义了一个多面体。因此，切片操作会导致多面体的维度降低。投影算法在实时应用中限制于低阶多面体。不过，通过假设不同的故障组合情况会产生不同的矢量 k_{ss} ，而每个矢量的 u_{\max} 和 u_{\min} 值都不同，离线确定可操控性多面体似乎很有用。对估算出的干扰矢量进行切片是一个非常简单的操作，可以实时执行。

因此，在给定一个执行器故障情况下，可通过公式（10）获得相关的可变多面体 P_m 。每个故障组合都与一个多面体 P_m 相关联，由于只用一个矩阵和一个矢量表示，因此可以很容易地存储起来（ L_{ss} 和 k_{ss} ），以便在实时应用中使用。因此，只需进行简单的操作，就能验证给定参考 r_{ss} 的可行性。最后，对所选公差做一些说明。为了保持状态和输出约束的完整性，公差值越小越好。然而，过小的公差可能会导致投影算法不可行，从而产生空多面体。因此，必须谨慎选择这些参数，以确保构建出非空的可操控性多面体。

总结

通过公式（4）实时计算参考指令是可行的，但是对算力要求太高，所以文章提出了一个可以先离线计算，从而降低算力要求的方法：

1. 离线：给出多个可能出现的故障下的矩阵 L_{ss} 和 k_{ss} ，计算出多面体 P_{ss} ，根据所需要使用到的变量（比如控制变量），计算出投影矩阵 P_w 。
2. 在线：实时根据估算的扰动 \hat{d}_k 计算出 P_w 的切面 P_m ，这个多面体包含了所需要的容许参考 r_{ss} 。

Feasible Target-Tracking Model Predictive Control

图 2 显示了解决方案的整体结构。离散控制器（由虚线围成）有三个主要功能：可行目标计算、MPC 优化器和线性观测器。假定故障检测和隔离系统（FDI）可为目标计算和 MPC 方案提供有关执行器状态的正确信息。

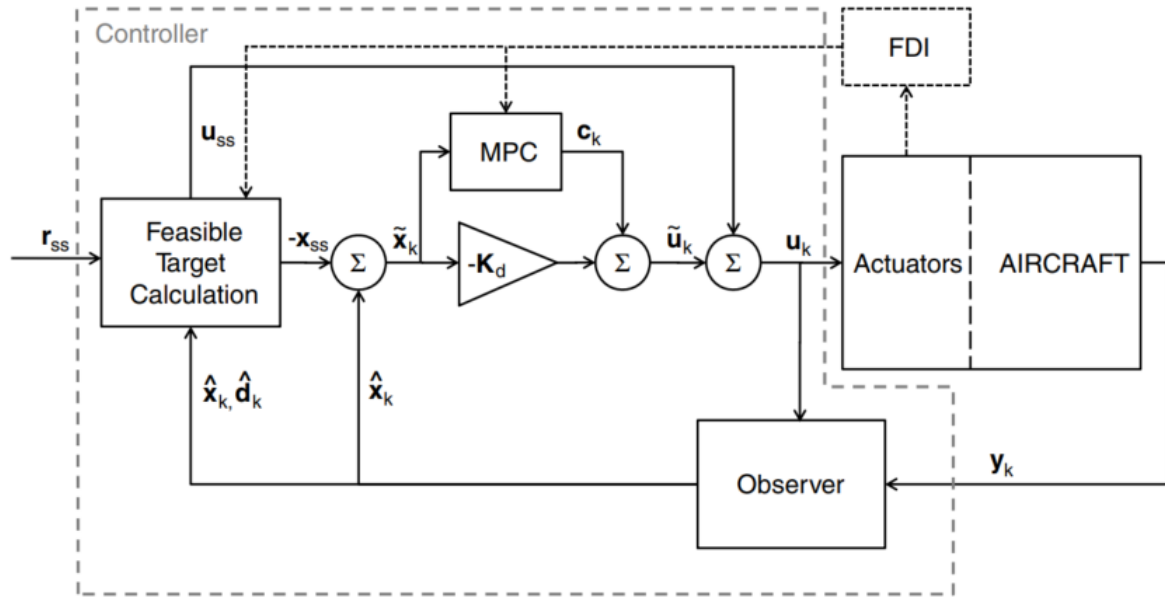


Fig. 2 Overall structure of the feasible target-tracking MPC.

可行目标计算子系统计算稳态下的状态和控制矢量（ x_{ss} 和 u_{ss} ），这是对静态可接受参考信号 r_{ss} 进行无偏移跟踪所必需的。该计算考虑了**执行器的限制和约束优化问题的可行性**。

利用计算出的稳态值，从 x_{ss} 中减去估计状态向量 \hat{x}_k ，从而将跟踪问题转换为对所需稳态条件的调节。反过来，调节器控制法则两个部分的总和：一个是具有静态线性反馈增益 K_d 的线性部分，另一个是由 MPC 系统计算得出的**非线性修正** c_k 。在执行器出现故障时，非线性修正 c_k 会在可用的执行器之间重新分配控制力。

Feasible Target Calculation

控制律：

$$\begin{aligned} u_j &= -K_d(x_j - x_{ss}) + u_{ss} + c_j & j = 0, \dots, N-1 \\ u_j &= -K_d(x_j - x_{ss}) + u_{ss} & j \geq N \end{aligned} \quad (11)$$

MPC 计算的是 c_j ， K_d 是固定的。

假设稳态值（ x_{ss}, u_{ss} ）固定，那么将状态向量增广，并将控制律代入系统模型：

$$\begin{bmatrix} x_{j+1} \\ x_{ss} \\ u_{ss} \\ d_{j+1} \end{bmatrix} = \begin{bmatrix} \Phi - \Gamma K_d & \Gamma K_d & \Gamma & \Gamma_d \\ 0 & I & 0 & 0 \\ 0 & 0 & I & 0 \\ 0 & 0 & 0 & I \end{bmatrix} \begin{bmatrix} x_j \\ x_{ss} \\ u_{ss} \\ d_j \end{bmatrix} + \begin{bmatrix} \Gamma \\ 0 \\ 0 \\ 0 \end{bmatrix} c_j \quad (12)$$

对应的约束：

$$\begin{aligned} C_c x_j &\leq x_{\max} & -C_c x_j &\leq -x_{\min} & C_c x_{ss} &\leq x_{\max} \\ -C_c x_{ss} &\leq -x_{\min} & u_{ss} &\leq u_{\max} & -u_{ss} &\leq -u_{\min} \\ d_j &\leq d_{\max} & -d_j &\leq -d_{\min} & -K_d x_j + K_d x_{ss} + u_{ss} &\leq u_{\max} \\ & & & & K_d x_j - K_d x_{ss} - u_{ss} &\leq -u_{\min} \end{aligned} \quad (13)$$

根据参考文献[4]可以计算出不变集，根据不变集可以计算出终端约束 X_N 。

$$\begin{aligned} \mathbf{X}_N = \{ &(x_k, x_{ss}, u_{ss}, d_k) | M_x x_k + M_{x_{ss}} x_{ss} \\ &+ M_{u_{ss}} u_{ss} + M_d d_k \leq k_N \} \end{aligned} \quad (14)$$

终端约束是为了保证 MPC 有可行解，那么只要让给定的稳态目标 x_{ss} 和 u_{ss} 在终端约束中即可：

Finally, the set \mathbf{X}_{ssN} of admissible steady-state values, which assures the feasibility of the MPC regulator, is computed at each sampling time with the slice operation $\sigma_{(\hat{x}_k, \hat{d}_k)} \mathbf{X}_N$. The computation of \mathbf{X}_{ssN} , for each actuator fault condition, is summarized by Algorithm 2.

The feasible target values of x_{ss} and u_{ss} are obtained through the solution of the QP problem:

$$\begin{aligned} \min_{x_{ss}, u_{ss}} J(r_{ss}, \hat{x}_k, \hat{d}_k) &= (HEx_{ss} - r_{ss})^T Q_{ss} (HEx_{ss} - r_{ss}) + u_{ss}^T R_{ss} u_{ss} \\ \text{subject to: } &(\Phi - I)x_{ss} + \Gamma u_{ss} + \Gamma_d d_k = 0 \\ &(x_{ss}, u_{ss}) \in \mathbf{X}_{ssN} \end{aligned} \quad (15)$$

总结

该章节和第二节提供了两个方法，第二节面向 r_{ss} ，这一节面向 x_{ss} 和 u_{ss} 。

第二节通过切片和投影，可以获取 r_{ss} 的容许范围，这相当于第一次过滤 (filter)，从控制对象的物理能力（状态模型和约束）出发，筛选出能够完成的 r_{ss} 。

该节通过终端约束的切面，来确定 x_{ss} 和 u_{ss} ，保证了 MPC 有解。

这篇文章的参考管理体现在第二节，保证了在故障情况下，系统的参考输入有对应的稳态值。这解决了开头提出的 **三个问题**：文章涵盖了轨迹跟踪，并且能够对上层给出的轨迹进行适当调整，调整方法是投影和切片，保证在故障情况下依然能够完成参考指令。

结合我们的场景，我们需要在前方出现障碍物时，**根据参考指令的变化快速调动执行器，以避免障碍物**。这篇文章能够在保证可解的同时，最大程度尽量完成上层给出的参考指令，解决了 MPC 有解的问题，但本质上还是在控制层，需要在“**规划层**”调整参考指令。