WBC学习笔记

WBC概述

WBC的问题背景

在目前的了解下，WBC产生于以下问题：
多任务高自由度机器人，存在以下两个特点：

1. 通常同时存在多个控制任务，如末端轨迹跟随和连杆避障，如下肢任务与上肢任务。
2. 在一些关节奇异位置，可能无法同时满足所有控制任务，必须舍弃一些。因此，引入任务优先级的概念。

WBC的优势

在高自由度机器人中，Whole-Body Control 相较于传统的独立关节级控制或基于解耦假设的运动规划方法具有显著优势：

1. 任务一致性与最优性（Task Consistency & Optimality）

WBC 能够在满足机器人动力学约束的前提下，通过优化框架同时处理多个任务。它保证**高优先级任务严格执行**，而在不违反上层任务的条件下尽可能地优化低优先级任务，实现全局一致性。

---

2. 可处理冗余自由度（Exploiting Redundancy）

冗余自由度常常被用于实现附加目标，例如避障、姿态优化、力控制。WBC 通过投影或层级优化的方法（如 Hierarchical QP）自然地利用冗余自由度，使机器人在执行主任务时，还能实现诸如：

- 保持平衡
- 避免自碰撞
- 关节空间优化（joint limit avoidance）
- 改善能耗或动态稳定性

WBC经典应用场景

1. 仿人机器人（Humanoid Robots）

仿人机器人通常具有 20–40 个自由度，且存在多接触、平衡保持和运动协调等复杂问题，因此是 WBC 的最经典应用场景：

- 行走、奔跑中的动态平衡控制
- 双手搬运物体时，手臂任务与足部任务冲突的处理
- 奇异位形（如伸手过头）下的任务投影和冗余处理
  代表系统：HRP 系列、Atlas、Talos、iCub

---

2. 多足机器人（Quadruped / Hexapod Robots）

WBC 能够在优化中分配地面反作用力，因此特别适用于具有多脚接触的机器人：

- 不平整地形下的动态步态调节
- 质心/角动量控制
- 在跳跃、落地过程中的接触力优化
  代表：MIT Mini Cheetah、ANYmal

数学基础

为了更好地理解WBC背后的数学原理，读者最好具有以下数学基础。

雅可比矩阵

设我们有一个向量函数：

$$f(x) = \begin{pmatrix} f_1(x_1, x_2, \cdots, x_n) \\ f_2(x_1, x_2, \cdots, x_n) \\ \vdots \\ f_m(x_1, x_2, \cdots, x_n) \end{pmatrix}$$

其中：$f : \mathbb{R}^n \to \mathbb{R}^m$
则雅可比矩阵定义为所有一阶偏导数组成的矩阵：

$$J(x) = \frac{\partial f(x)}{\partial x} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \cdots & \frac{\partial f_2}{\partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_m}{\partial x_1} & \frac{\partial f_m}{\partial x_2} & \cdots & \frac{\partial f_m}{\partial x_n} \end{bmatrix}$$

不难看出，雅可比矩阵**描述了多维函数在当前点的线性化近似**，也就是说，$J(x)$是函数矩阵在该点上的线性转移矩阵。

在机器人正运动学中，我们往往能够得到机器人关节空间和任务空间之间的函数矩阵：

$$x(q)=\begin{pmatrix} x_1(q_1,q_2,\cdots,q_n) \\ x_2(q_1,q_2,\cdots,q_n) \\ \vdots \\ x_m(q_1,q_2,\cdots,q_n) \end{pmatrix}$$

通过计算雅可比矩阵，可以在给定构型下建立关节空间速度 $\dot{q}$与任务空间速度$\dot{x}$ 之间的线性映射关系，即:

$$\dot{x}=J(q)\dot{q}$$

该矩阵描述了关节变量的微小变化如何在任务空间中产生相应的瞬时运动。

伪逆矩阵

伪逆矩阵是普通逆矩阵的推广 —— 普通逆仅适用于可逆方阵，而伪逆可用于**任意矩阵**（包括非方阵、奇异方阵），解决 "逆运算" 的需求。

对于任意复矩阵 $A\in\mathbb{C}^{m\times n}$，其 Moore-Penrose 伪逆记为 $A^\dagger\in\mathbb{C}^{n\times m}$，是唯一满足以下 4 个 "Moore-Penrose 条件" 的矩阵：

$$\begin{cases} 1.\ AA^\dagger A=A \quad （自反性：伪逆作用后不改变原矩阵的核心信息） \\ 2.\ A^\dagger AA^\dagger=A^\dagger \quad （自反性：伪逆自身也满足类似性质） \\ 3.\ (AA^\dagger)^H=AA^\dagger \quad （\text{Hermite}对称性：AA^\dagger\ 是正交投影矩阵） \\ 4.\ (A^\dagger A)^H=A^\dagger A \quad （\text{Hermite}对称性：A^\dagger A\ 是正交投影矩阵） \end{cases}$$

其中 $(\cdot)^H$ 表示**共轭转置**（若 A 是实矩阵，共轭转置退化为普通转置 $(\cdot)^T$)。

伪逆矩阵的求解方法

伪逆的求解方法中，**奇异值分解（SVD）法**是最常用、数值最稳定的方式：

对任意矩阵 $A\in\mathbb{C}^{m\times n}$，其 SVD 分解为：$A=U\Sigma V^H$其中：

- $U\in\mathbb{C}^{m\times m}$、$V\in\mathbb{C}^{n\times n}$ 是**酉矩阵**（实矩阵下为正交矩阵，列向量两两垂直且模长为 1）；
- $\Sigma\in\mathbb{R}^{m\times n}$是对角矩阵，对角线上的非负元素 $\sigma_1\geq\sigma_2\geq\cdots\geq\sigma_r>0,(r=\mathrm{rank}(A)$ 是 A 的秩）称为**奇异值**，其余位置为 0。

此时，A 的伪逆可通过 "调整 SVD 分量" 得到：$A^\dagger=V\Sigma^\dagger U^H$其中 $\Sigma^\dagger\in\mathbb{R}^{n\times m}$ 是 Σ 的 "伪逆对角矩阵"—— 将 Σ 的非零奇异值取倒数($\sigma_i^\dagger=1/\sigma_i$)，再交换 Σ的行列（即转置），其余元素仍为 0。

伪逆矩阵的核心性质

伪逆的性质是其应用的基础，关键性质包括：

1. **基本自反性**：
   $AA^\dagger A=A,\quad A^\dagger AA^\dagger=A^\dagger$

2. **逆矩阵的推广**：若 A 是可逆方阵，伪逆等价于普通逆：
   $A^\dagger = A^{-1}$
3. **投影性质**：
   - $AA^\dagger$ 是从 $\mathbb{C}^m$ 到 A 的**列空间**的正交投影矩阵；
   - $A^\dagger A$是从 $\mathbb{C}^n$到 A 的**行空间**的正交投影矩阵。
4. **共轭转置的兼容性**：伪逆与共轭转置可交换顺序：
   $(A^\dagger)^H = (A^H)^\dagger$

零空间

目前WBC的控制器类型也比较多，下面主要介绍WBC最核心的思想，零空间方法。
首先我们需要搞懂什么是零空间：
对于一个矩阵，比如：

$A = \begin{pmatrix} 1 & 2 & 3 & 4 \\ 5 & 4 & 7 & 1 \\ 4 & 2 & 4 & -3 \end{pmatrix}$我们可以计算得到关于行空间的维度，也就是行空间基向量的个数，由所有基向

那么我们就称所有的向量x所张成的空间为矩阵的零空间。

我们也很容易看出来，行空间和零空间是两个**相互正交**的空间。
那这与WBC有什么关系呢，假设说现在有两个优先级不同的任务需要去完成，系统希望任务二（低优先级）的完成不影响任务一（高优先级）的完成。
那么其实只需要满足实现任务二的某个关节空间速度$\dot{q}_2$乘以任务一的雅可比矩阵J_1为零即可。

基于零空间投影的WBC

从上述数学基础，我们便可以得到零空间矩阵，应为：

$$N_{i-1}^A = I - (J_{i-1}^A)^\dagger J_{i-1}^A$$

它的性质为：

$$\begin{aligned} J_{i-1}^A \cdot N_{i-1}^A &= J_{i-1}^A \cdot \left(I - \left(J_{i-1}^A \right)^\dagger J_{i-1}^A \right) \\ &= J_{i-1}^A - \underbrace{J_{i-1}^A \left(J_{i-1}^A \right)^\dagger J_{i-1}^A}_{\text{根据伪逆性质 } AA^\dagger A = A} \\ &= J_{i-1}^A - J_{i-1}^A \\ &= \mathbf{0} \end{aligned}$$

那么任意一个关节速度向量$\dot{q}_i$乘以某一任务的雅可比矩阵的零空间矩阵N后，得到的新的关节速度将不会对该任务有任何贡献，因为有上述可知，无论$\dot{q}$为多少，总是满足：

$$J \cdot N \cdot \dot{q} = \mathbf{0}$$

这个性质就可以用来计算完成低优先级任务的关节速度是如何跟完成高优先级的关节速度耦合在一起的。
假设现在有：

$$J_{i-1} \cdot \dot{q}_{i-1} = \dot{x}_{i-1}$$

现在计算实现第i个任务的关节速度，即满足：

$$J_i \cdot \dot{q}_i = \dot{x}_i$$

第i个任务的关节速度是在第i-1个任务上叠加一部分关节速度，但是这部分关节速度不能对第i-1个任务有贡献，所以即有以下关系式：

$$\dot{q}_i = \dot{q}_{i-1} + N_{i-1}\dot{q}_{\forall}$$

代入上式后有：

$$J_i \cdot (\dot{q}_{i-1} + N_{i-1}\dot{q}_{\forall}) = \dot{x}_i$$

可以解得

$$\dot{q}_{\forall} = (J_i N_{i-1})^{\dagger}(\dot{x}_i - J_i\dot{q}_{i-1})$$

将$\dot{q}_{\forall}$带入后有：

由此便得到了迭代式，可以从第一个任务迭代计算到第i个任务。

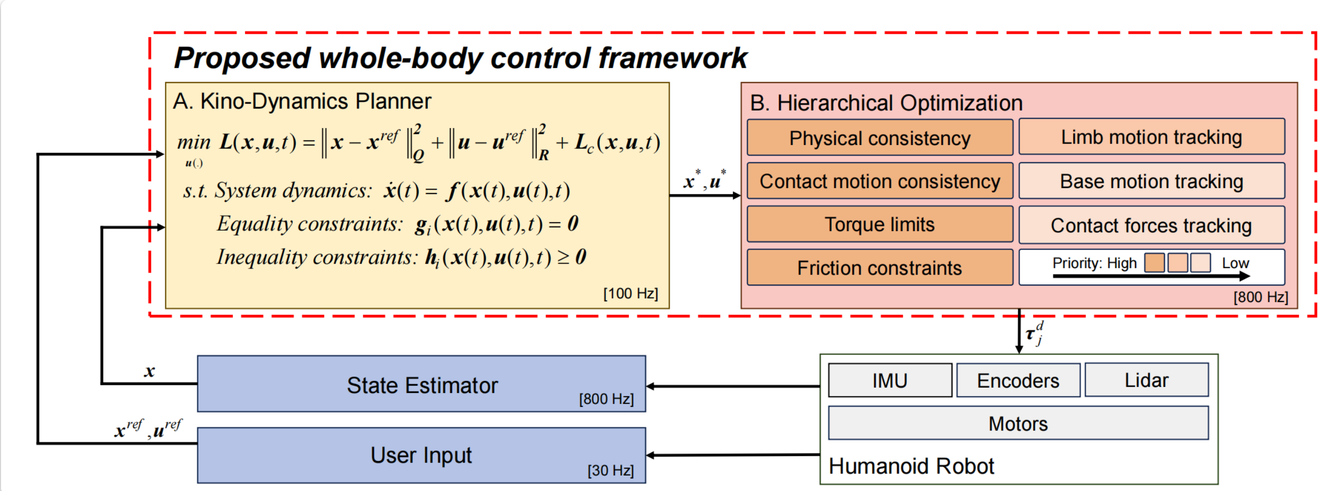
$$\dot{q}_i = \dot{q}_{i-1} + (J_i N_{i-1})^{\dagger}(\dot{x}_i - J_i\dot{q}_{i-1})$$

在Matlab中的案例

在人形机器人上的应用

人形机器人

在基于模型的方法用于人形机器人控制时，通常将系统划分为**规划模块**（planning module）和**跟踪模块**（tracking module）两个部分。在文献 *“Whole-Body Control Framework for Humanoid Robots with Heavy Limbs: A Model-Based Approach”* 中，作者采用模型预测控制（MPC）作为规划模块，用于在线生成机器人未来时域内的参考轨迹，包括四肢末端轨迹、基体（floating base）轨迹以及期望的接触力轨迹等。
系统的baseline是这样的：



系统定义的任务及优先级如下图所示：

Priority	Task
1	Physical consistency
	Contact motion consistency
	Torque limits
	Friction constraints
2	Limb motion tracking
3	Base motion tracking
4	Contact forces tracking

首先，动力学规划模块基于机器人动力学模型与预测地平线，通过 MPC 生成未来一段时间内的期望接触力、关节位置与速度、以及基座的线速度和角速度等参考量。这些参考轨迹随后被输入到跟踪控制模块。

在跟踪部分，系统采用基于层级优化（Hierarchical Optimization）与分层二次规划（HQP）的方法，构建一个多优先级的最优化问题，从而求解满足动力学约束、接触约束及任务约束的控制力矩指令，实现整机的 Whole-Body Control。

基于层级优化的方法是二次规划和零空间投影的结合，具体原理如下。

HQP

一般的二次规划问题

对于标准的二次规划问题有以下形式：

$$\begin{aligned} \min_{x} \quad & \frac{1}{2}x^T x + {}^T x \\ .\text{t.} \quad & Ax = \\ & x \end{aligned}$$

假设说我们的代价项为以下的最小二乘的形式：

$$\frac{1}{2} Ax - {}^2$$

那么最小化上述代价函数就可以被转化为一个标准的二次规划问题：

$$\begin{aligned} \frac{1}{2} Ax - {}^2 &= \frac{1}{2}(Ax -)^T(Ax -) \\ &= \frac{1}{2}\left(x^T A^T Ax - x^T A^T - {}^T Ax + {}^T\right) \\ &= \frac{1}{2}\left(x^T A^T Ax - 2^T Ax + {}^T\right) \end{aligned}$$

因为^{T}是一个固定项，可以将其丢弃，那么上述问题中的与为：

$$\begin{aligned} &= A^T A \\ &= \left(-{}^T A\right)^T = -A^T \end{aligned}$$

分层二次规划

$$\begin{aligned} & Ax = \\ & x \end{aligned}$$

我们将上述的任务定义为：

$$: \quad \begin{aligned} & A - \ = \\ & - \end{aligned}$$

在这里v和被称为松弛变量，将其作为二次规划的目标：

$$\begin{aligned} \min \quad & \frac{1}{2}{}^2 + \frac{1}{2}{}^2 \\ .\text{t.} \quad & A - \ = \\ & - \end{aligned}$$

或者：

$$\begin{aligned} \min \quad & \frac{1}{2} A - {}^2 + \frac{1}{2}{}^2 \\ .\text{t.} \quad & - \end{aligned}$$

设需要优化的变量为 $= x^T \omega^{TT}$,那么与上述问题所等价的表达形式为：

$$\begin{aligned} \min \quad & \frac{1}{2}{}^T + {}^T \\ .\text{t.} \quad & \end{aligned}$$

其中：

$$= A^T A, \quad = -A^T$$

且：

$$A = \begin{matrix} A & \\ & I \end{matrix}, \qquad = \quad - \quad ,$$
$$= \quad -I, \qquad =$$

现在我们有带有**优先级**的多个任务：$T_1, T_2, \cdots, T_n$,对于具有优先级的任务，可以定义为：

$$: \quad \begin{matrix} Ax - \quad = v \\ x - \end{matrix}$$

对于矩阵A的零空间矩阵我们由上述笔记内容可知：

$$N = Null(A) = I - A\dagger A$$

现在我们将上述任务雅可比矩阵推广一下：

$$A = \begin{bmatrix} A_1 \\ A_2 \\ \vdots \\ A \end{bmatrix}$$

那么上述增广任务雅可比矩阵的零空间投影矩阵为：

$$N = Null(A) = I - A\dagger A$$

但是这在计算上太复杂了，我们可以用一种递归的表达式进行求解，即：

$$N = N_{-1} Null(A N_{-1})$$

在有了零空间投影矩阵后，我们就可以定义给定任务的解的集合。我们假设x_1是任务T_1优化得到的最优解。

$$x_1, \ _1 = \min_{x,} \frac{1}{2} A_1 x - \ _1{}^2 + \frac{1}{2} \ ^2$$
$$.\text{t.} \quad _1 x - \ _1$$

这时我们考虑任务T_2,此时对于任务T_1和T_2整体优化后的最优解应该为：

$$x_2 = N_1 x_2 + x_1$$

在这里N_1是增广的零空间投影矩阵，而且x_1是增广后的最优解，只不过由于是第一个任务，所以我们有$N_1 = N_1$以及$x_1 = x_1$，N_1当然就是任务T_1的零空间投影矩阵了。所以在N_1的作用下，x_2被投影到了任务T_1的零空间中，所以不会对任务T_1有任何的作用。为了计算得到x_2是多少，我们需要解决以下问题：

$$
\begin{aligned}
x_{2}, {}_{2} = \min_{x,} \frac{1}{2} A_2 \left(N_1 x + x_1\right) - {}_2{}^2 + \frac{1}{2}{}^2 \\
.\text{t.} \quad {}_2 \left(N_1 x + x_1\right) - {}_2 \\
{}_1 \left(N_1 x + x_1\right) - {}_1 \quad {}_1
\end{aligned}
$$

从这里我们也可以看到，我们在求解实现任务T_2的最优解时，并不是解一个和任务T_1一样形式的优化问题，直接将x_2求解出来；因为那没有约束，解出来新的最优解可能会对上一个任务的执行产生影响，所以才需要写成上述的形式，求解出x_2后再通过零空间投影得到x_2.

我们接着考虑任务T_3,此时对于任务T_1,T_2和T_3整体优化后的最优解应该为:

$$x_3 = N_2 x_3 + x_2$$

此时的零空间投影矩阵N_2便是增广任务$\begin{matrix} A_1 \\ A_2 \end{matrix}$的零空间投影矩阵。

为了求解x_3,我们需要解决：

$$
\begin{aligned}
x_{3}, {}_{3} = \min_{x,} \frac{1}{2} A_3 \left(N_2 x + x_2\right) - {}_3{}^2 + \frac{1}{2}{}^2 \\
.\text{t.} \quad {}_3 \left(N_2 x + x_2\right) - {}_3 \\
{}_2 \left(N_2 x + x_2\right) - {}_2 \quad {}_2 \\
{}_1 \left(N_2 x + x_2\right) - {}_1 \quad {}_1
\end{aligned}
$$

到这里我们便能清楚的看到分层二次规划是怎么利用零空间投影和二次规划来进行多层任务的求解了。我们考虑整体上，我们有任务$T_1, T_2, \cdots, T_{+1}$，我们求得对于上述p+1个任务的最优解应该为：

$$x_{+1} = N x_{+1} + x$$

而x_{+1}通过以下问题被求解：

$$
\begin{aligned}
x_{+1}, {}_{+1} = \min_{x,} \frac{1}{2} A_{+1} \left(N x + x\right) - {}_{+1}{}^2 + \frac{1}{2}{}^2 \\
.\text{t.} \quad {}_{+1} \left(N x + x\right) - {}_{+1} \\
\left(N x + x\right) - \\
\\
{}_2 \left(N x + x\right) - {}_2 \quad {}_2 \\
{}_1 \left(N x + x\right) - {}_1 \quad {}_1
\end{aligned}
$$

由此我们便得到了分层二次规划是怎么解决多个具有优先级的任务的控制问题的。
参考链接Hierarchical QP

动力学公式推导

所以回归上述人形机器人运动问题，我们尝试回归到动力学的世界，将上述优化变量，任务矩阵与动力学分析中的符号进行适配。

在论文中规定了优化向量为 $= \dot{v}^T, {}_T$，其中v为广义速度变量，维度为 $+n$，x，y，轴方向上的线速度加角速度与n个驱动关节的速度。而${}_T$便为地面接触力。

这个系统首先需要满足的便为Physical consistency，即符合牛顿第二定律。由于本文在MPC规划环节并未加上该约束条件，所以需要满足以下方程：

$$
\begin{aligned}
\dot{v} + n &= J_T, \\
\dot{v} + n &= \; + J_T,
\end{aligned}
$$

将其换为我们先前定义的优化向量，则有：

$$
, -J_T \quad = -n.
$$

其中为广义质量矩阵，下标代表是以浮动基坐标系为参考系。J_T为接触力的雅可比矩阵，由虚功原理导出，n为惯性力（如向心力，科里奥利力等）。

于是我们便得到了如$Ax =$ 式的等式约束，就能构建如

$$
\frac{1}{2} Ax - {}^2
$$

的二次规划问题。