arXiv:2303.03903v1 [cs.RO] 7 Mar 2023

基于本体感觉传感器的多接触点同步定位与力识别：应用于机械臂

Seo Wook Han[1] 和 Min Jun Kim[1]

摘要—本文提出一种可同时估计接触点和力的算法。我们研究的对象是配备本体感知传感器的协作机器人，具体包括关节扭矩传感器（JTS）和基座力/扭矩（F/T）传感器。所提出的方法具有以下优点。首先，通过适当的机器人网格预处理实现快速计算。其次，借助基座 F/T 传感器可以识别多点接触，而仅配备 JTS 时则难以做到。所提出的方法是在标准粒子滤波器的基础上进行修改，以适应网格预处理和可用的传感器数据。在仿真验证中，对于一台七自由度机器人，该算法以 99.96% 的成功率运行，单点接触情况下运行频率达到 2200Hz 。在运行时间方面，与现有方法相比，所提出的方法快了 ≥3.5 倍。文中还报告了双点和三点接触的结果。

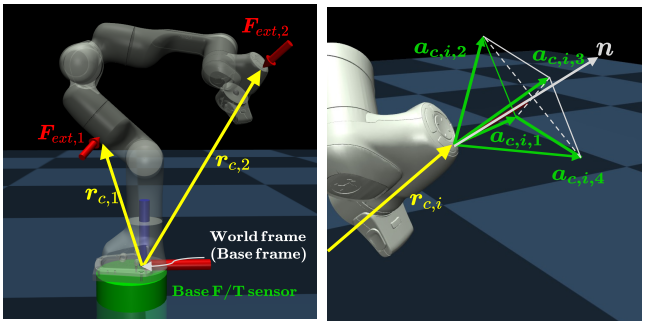
I. 引言

如今，协作机器人日益普及，其设计初衷是与人类共享工作空间。因此，如何在碰撞或接触情况下实现安全的物理交互，已成为一项重要的研究课题。在一些研究中，例如采用无碰撞路径规划 [1]–[3]和基于无源性的控制器 [4], [5] 来确保安全。另一些研究则致力于尽可能快速地识别碰撞是否发生，以保障安全 [6]–[11]。

为进一步理解并妥善处理物理交互，研究人员尝试同时估计接触位置和力（例如，估计图 1(a) 中的 $\boldsymbol{r}_{c,\,i}$ 和 $\boldsymbol{F}_{ext,\,i}$）。针对这一问题，一项主要研究方向是开发触觉传感器，尝试模拟人类皮肤的感觉受体 [12], [13]。尽管触觉传感器是一项极具前景的技术，但迄今为止，更实用的方案是使用本体感知传感器，例如关节扭矩传感器（JTS）或六轴力/扭矩（F/T）传感器，这些传感器已经相当成熟，且已在市场上销售。

事实上，已有不少基于本体感知传感器的解决方案被提出 [14]–[19]。例如， [14] 提出了一种基于机器学习技术的方法，而[15] 提出了一种所谓的力作用线方法，通过几何方式计算接触位置。然而，已知这些方法能够识别的接触位置有限。 [14] 仅考虑机器人连杆上的少数离散点，而 [15]–[17]中的方法可能会因连杆几何形状而无法得到可行解。

本研究得到韩国政府（MSIT）资助的韩国国家研究基金会（NRF）项目支持，项目编号为 2021R1C1C1005232 和 2021R1A4A3032834。
[1]作者来自韩国大田的韩国科学技术院智能机器人系统实验室。电子邮箱：tjdnr7117, minjun.kim@kaist.ac.kr



(a) (b)

图 1 $\mathbb{R}^3$ $\boldsymbol{F}_{c,i} \in \mathbb{R}^3$ $\boldsymbol{r}_{c\,i} \in$ 。（a）本文旨在利用本体感知传感器（关节扭矩传感器、基座力/扭矩传感器）估计接触位置（. ）和力（. ）。（b）通过摩擦锥公式实现点接触假设。

为在更一般的设置下估计接触位置， [20]–[22]提出了基于粒子滤波器（PF）的方法。在这种方法中，粒子会被更新并重新采样，以找到一组能够解释所有传感器测量值的位置和力，这一过程采用二次规划（QP）的思路。此外，在粒子滤波器框架内，只要感知信息足够丰富，就可以估计多点接触。

然而，基于粒子滤波器的方法仍面临一些挑战。首先，运行时间通常较长，主要是由于粒子滤波器运动模型的计算耗时；例如， [20] 提出的运动模型会在粒子更新至三维空间后，将其投影到机器人表面。 [20]报告的运行时间约为 10 Hz，适用于一台 30 自由度人形机器人。 [21] 也提出了一种基于粒子滤波器的方法，其在一台 7 自由度机械臂上的运行频率约为 200 Hz。 [19],提出的方法严格来说并非粒子滤波器的变体；该方法表明，通过适当地对机器人网格进行预处理，可以缩短运行时间，并在一台 7 自由度机器人上实现了≥600Hz。其次，以往工作中提出的算法往往受到奇异性的限制：在这些情况下，存在多个能够解释所有传感器测量值的解。事实上，如 [18], [19]所分析的那样，仅考虑关节扭矩传感器时，此类奇异性可能相当频繁地出现。

本文提出一种解决方案，应对上述两项挑战。首先，受 [19], 启发，我们通过网格预处理来提升 PF 的运行时间，该预处理会离线计算网格数据，例如所有面片对之间的距离。然而，在我们的设置中，这只能针对每个连杆分别进行，因此粒子无法跨连杆更新。为此，我们修改了 PF，引入探索粒子，强制它们探查相邻连杆。其次，如图 1(a) 所示，我们在机器人的基座处安装一个力/扭矩传感器，以缓解奇异性。通过

这种设置，我们可以估计任意连杆上的接触，包括连杆 1 和 2；即使对于自由度较少的协作机器人，也能估计多个接触点。我们指出，众所周知，当机器人仅配备关节扭矩传感器时，这些任务极具挑战性 [18], [19]。

总而言之，本文针对协作机械臂在多点接触场景下的接触力和位置估计问题展开研究。我们提出了一种称为多点接触粒子滤波器与探索粒子（MCP-EP）的新算法，该算法以接触粒子滤波器（CPF） [20]为基础。与以往工作相比，(i) 通过引入网格预处理步骤和探索粒子，改善了运行时间；(ii) 使用基座力/扭矩传感器处理多点接触场景，同时减轻奇异性问题。采用所提出的方法，对于一台具有 7 个自由度的协作机械臂，单点接触的计算时间为 0.45ms（2200Hz），双接触的计算时间为 1.66ms（600Hz）。通过使用基座力/扭矩传感器缓解奇异性问题，单点接触的成功率可达到 99.96%。此外，本文还报告了最多包含 3 个接触点的多点接触场景。

本文的组织结构如下。第二节介绍了结合 JTS 和基座力/扭矩传感器的接触力识别二次规划公式。第三节介绍了用于多点接触位置定位和力识别的 MCP-EP。第四节通过仿真研究对所提出的算法进行了验证。

II. 基于二次规划的无定位接触力识别

本节提出一种二次规划公式，用于估计接触力，而不进行接触点定位；接触点定位将在第三节介绍。二次规划将接触力作为决策变量输出，以解释给定候选接触点处的传感器测量值（关节扭矩传感器和基座力/扭矩传感器）。在建立二次规划问题之前，我们首先介绍如何利用本体感知传感器（即关节扭矩传感器和基座六轴力/扭矩传感器）估计外部关节力矩和接触诱发的基座力旋量。全文假设为不包含力矩的点接触 [14], [15], [20]–[24]。

A. 使用关节扭矩传感器估计外部关节力矩

考虑以下铰接多体动力学（也称为柔性关节机器人模型的连杆动力学）：

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}}+\boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}}+\boldsymbol{g}(\boldsymbol{q})=\boldsymbol{\tau}_j+\boldsymbol{\tau}_{ext}, \tag{1}$$

其中，$\boldsymbol{q}\in\mathbb{R}^n$ 为关节变量，$\boldsymbol{M}(\boldsymbol{q})$ 为连杆侧惯性矩阵，$\boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}})$ 为科里奥利/离心矩阵，$\boldsymbol{g}(\boldsymbol{q})$ 为重力向量，$\boldsymbol{\tau}_j$ 为可通过关节扭矩传感器测得的关节扭矩，$\boldsymbol{\tau}_{ext}$ 为接触力所产生的外部扭矩。

假设机器人中存在 k 个接触点（每个连杆最多一个），如图 1(a) 所示。外部关节力矩 τ_{ext} 为

$$\boldsymbol{\tau}_{ext}=\sum_{i=1}^{k}\boldsymbol{J}_i^T(\boldsymbol{q},\boldsymbol{r}_{c,i})\boldsymbol{F}_{ext,i}, \tag{2}$$

其中 $\boldsymbol{r}_{c,\,i}\in\mathbb{R}^3$ 是 i^{th} 接触点的位置向量，而 $\boldsymbol{J}_i(\boldsymbol{q},\ \boldsymbol{r}_{c,\,i})\in\mathbb{R}^{3\times n}$ 是该接触点对应的位置雅可比矩阵。对于配备关节扭矩传感器的机器人，$\boldsymbol{\tau}_{ext}$ 可以使用以下基于动量的观测器 [6], [7] 进行估计：

$$\hat{\boldsymbol{\tau}}_{ext}(t)=\mathbf{K}_{\mathrm{o}}\left\{\boldsymbol{p}-\int_0^t\left(\boldsymbol{\tau}_j+\boldsymbol{n}(\boldsymbol{q},\dot{\boldsymbol{q}})+\hat{\boldsymbol{\tau}}_{ext}\right)\mathrm{d}s\right\}, \tag{3}$$

其中，$\boldsymbol{p}=\boldsymbol{M}(\boldsymbol{q})\,\dot{\boldsymbol{q}}$ 为机器人广义动量，$\boldsymbol{n}(\boldsymbol{q},\dot{\boldsymbol{q}})=\boldsymbol{C}^T(\boldsymbol{q},\dot{\boldsymbol{q}})\,\dot{\boldsymbol{q}}-\boldsymbol{g}(\boldsymbol{q})$，而 $\mathbf{K}_{\mathrm{o}}$ 为对角增益矩阵。这可以视为实际外部关节力矩 $\boldsymbol{\tau}_{ext}$的一阶低通滤波器。当$\mathbf{K}_{\mathrm{o}}$足够大时，我们可以简单地假设 $\hat{}\boldsymbol{\tau}_{ext}\simeq\boldsymbol{\tau}_{ext}$ [6], [25]。

B. 使用基座力/扭矩传感器估计接触力产生的力旋量

令 $\boldsymbol{W}_{ext,b}\in\mathbb{R}^6$ 表示外部接触力在基座处产生的力旋量。那么，

$$\boldsymbol{W}_{ext,b}=\sum_{i=1}^{k}\underbrace{\begin{bmatrix}\mathbf{I}_{3\times3}\\ skew(\boldsymbol{r}_{c,i})\end{bmatrix}}_{\triangleq\boldsymbol{X}_{c,i}(\boldsymbol{r}_{c,i})}\boldsymbol{F}_{ext,i}, \tag{4}$$

$skew(\cdot)$使 $\mathbb{R}^3$ 向量成为一个 $\mathbb{R}^{3\times3}$ 反对称矩阵。事实上，$skew(\cdot)$定义了一个叉积，即 $\forall a,b\in\mathbb{R}^3$

$$skew(a)b=a\times b. \tag{5}$$

需要注意的是，基座力/扭矩传感器的测量值并不直接提供由接触力产生的力旋量，因为测量值中包含了机器人的动力学。因此，我们采用[16]提出的一种方法，使用基座力/扭矩传感器估计 $\boldsymbol{W}_{ext,b}$ 。首先，通过用（1）式减去（3）式，并对惯性矩阵求逆，计算出一个标称 $\ddot{\boldsymbol{q}}_n$ 。该标称值代表不存在接触力的情形。其次，给定机器人状态（$\boldsymbol{q},\dot{\boldsymbol{q}}$）和 $\ddot{\boldsymbol{q}}_n$，使用递归牛顿-欧拉算法（RNEA）计算基座力/扭矩传感器施加于机器人基座连杆的标称基座力旋量。然后，我们可以通过从基座力/扭矩传感器的测量值中减去标称基座力旋量，得到 $\boldsymbol{W}_{ext,b}$，它是 $\boldsymbol{W}_{ext,b}$的估计值。

C. 接触力识别的二次规划公式

^定义向量 $\boldsymbol{W}$ 为

$$\hat{\boldsymbol{W}}=\begin{bmatrix}\hat{\boldsymbol{\tau}}_{ext}\\ \hat{\boldsymbol{W}}_{ext,b}\end{bmatrix}\in\mathbb{R}^{n+6}. \tag{6}$$

现在，需要找出最能解释 $\boldsymbol{W}$的接触位置和力^。由于我们假设为点接触，因此不能在机器人表面施加拉力。为对此进行数学表达，我们将 i^{th} 接触力约束在摩擦锥 $\mathcal{F}_c(\boldsymbol{r}_{c,\,i})$内，即$\boldsymbol{F}_{c,\,i}\in\mathcal{F}(\boldsymbol{r}_{c,\,i})$。

根据公式（2）、（4）和摩擦锥约束，我们可以构建一个优化问题，其中接触

点和力是决策变量：

$$\min_{\boldsymbol{r}_c,\boldsymbol{F}_c} \left\| \hat{\boldsymbol{W}} - \sum_{i=1}^{k} \begin{bmatrix} \boldsymbol{J}_i & \boldsymbol{X}_{c,i}^T \end{bmatrix}^T \boldsymbol{F}_{c,i} \right\|^2 \tag{7}$$

满足约束条件 $\boldsymbol{r}_{c,\,i} \in \mathcal{S}$，$\boldsymbol{F}_{c,\,i} \in \mathcal{F}(\boldsymbol{r}_{c,\,i})\quad \forall i \in \{1,\ldots,k\}$，

其中，$\boldsymbol{r}_c = (\boldsymbol{r}_{c,\,1},\ \ldots,\ \boldsymbol{r}_{c,\,k})^T$、$\boldsymbol{F}_c = (\boldsymbol{F}_{c,\,1},\ \ldots,\ \boldsymbol{F}_{c,\,k})^T$，且 $\mathcal{S}$ 表示机器人表面。这是一个非线性优化问题，因为例如，$\boldsymbol{r}_c$ 和 $\boldsymbol{F}_c$ 在 $\begin{bmatrix} \boldsymbol{J}_i & \boldsymbol{X}_{c,i}^T \end{bmatrix}^T \boldsymbol{F}_{c,\,i}$ 中以叉积的形式出现，如式 (4) 和 (5) 所示。

为简化该问题，如图 1(b) 所示，将摩擦锥近似为多面锥。在此近似下，i^{th} 接触力可表示为

$$\boldsymbol{F}_{c,i} = \begin{bmatrix} \boldsymbol{a}_{c,i,1} & \ldots & \boldsymbol{a}_{c,i,4} \end{bmatrix} \begin{bmatrix} f_{c,i,1} \\ \vdots \\ f_{c,i,4} \end{bmatrix} = \boldsymbol{A}_{c,i} \boldsymbol{f}_{c,i}, \tag{8}$$

其中，$\boldsymbol{a}_{c,\,i,\,j} \in \mathbb{R}^3$ 是多面摩擦锥的一个支撑向量，而 $f_{c,\,i,\,j}$ 是各支撑向量的权重。

为简便起见，在式 (7) 中，令 $\sum_{i=1}^{k} \begin{bmatrix} \boldsymbol{J}_i & \boldsymbol{X}_{c,i}^T \end{bmatrix}^T \boldsymbol{F}_{c,\,i}$ 表示为 $\boldsymbol{Q}^T \boldsymbol{f}_c$，其中 $\boldsymbol{f}_c = (\boldsymbol{f}_{c,\,1},\ \ldots,\ \boldsymbol{f}_{c,\,k})^T$。这里，$\boldsymbol{Q} \in \mathbb{R}^{4k \times (n+6)}$ 是对 $\boldsymbol{A}_{c,\,i}^T \boldsymbol{J}_i$ 和 $\boldsymbol{X}_{c,\,i} \boldsymbol{A}_{c,\,i}$ 矩阵的适当增广。此外，若 $\boldsymbol{r}_c$ 固定，则式 (7) 可写为

$$\min_{\boldsymbol{f}_c} \left\| \hat{\boldsymbol{W}} - \boldsymbol{Q}^T \boldsymbol{f}_c \right\|^2 \tag{9}$$

$$\text{subject to}\ \ f_{c,i,1}, \ldots, f_{c,i,4} \le 0,\ \ \forall i \in \{1,\ldots,k\},$$

这是一个凸二次规划问题，因为对于固定的 $\boldsymbol{r}_c$，$\boldsymbol{Q}$ 是常数，且摩擦锥是一个凸集。注意，接触力 $\boldsymbol{f}_c$ 是此问题中唯一的决策变量。

值得一提的是，借助基座力/扭矩传感器，无论接触位置如何，式（9）中的感知自由度都会增加 6 个。例如，如果接触发生在连杆 1 上，可用的传感器自由度为 7（其中 6 个来自基座力/扭矩传感器，1 个来自关节 1 处的关节扭矩传感器）。相比之下，如果机器人仅配备关节扭矩传感器，则此时感知自由度为 1，因此无法估计接触力和接触位置。

III. 用于接触点定位和力识别的粒子滤波算法

本节介绍用于多点接触点定位和力识别的 MCP-EP算法。在本节中，$\mathcal{X}_i$ 表示由粒子 $\{x_i^{[1]}, \cdots, x_i^{[m]}\}$ 组成的 i^{th} 集，$\mathfrak{X} = \{\mathcal{X}_1, \ldots, \mathcal{X}_k\}$ 表示粒子集的集合。粒子 $x_i^{[m]}$ 的权重记为 $w_i^{[m]}$。我们假设多点接触按顺序发生，从实际角度来看这是合理的。i^{th} 粒子集 $\mathcal{X}_i \in \mathfrak{X}$ 将估计 i^{th} 接触点。

A. 运动模型

根据粒子滤波器的运动模型更新粒子时，由于运动模型没有输入，一种常见做法是采用随机游走方法更新粒子 [20][21]。为缩短算法运行时间，本文对网格进行预处理，并相应地修改运动模型。

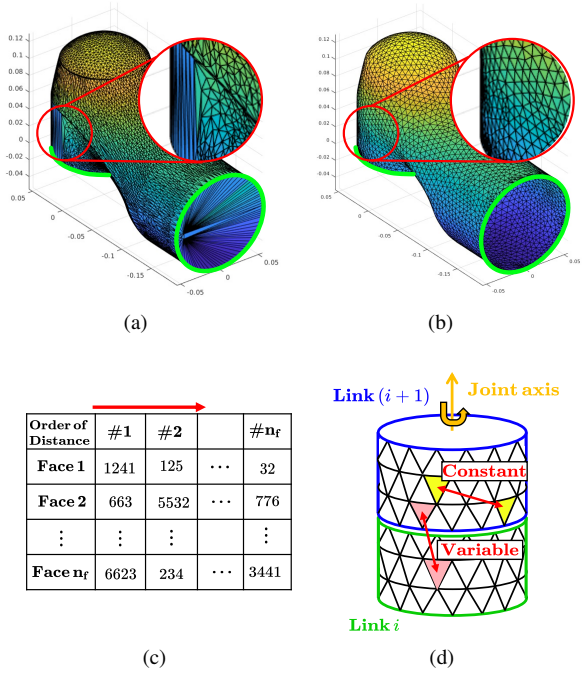


图 2. (a) 原始网格的三角面不规则（见红圈），且包含非接触部分（见绿圈）。(b) 去除非接触部分后，各面的尺寸几乎一致。(c) 数据结构按测地距离升序存储相邻面的索引。(d) 黄色面之间的距离保持不变，而红色面之间的距离会随关节旋转而变化。也就是说，不同连杆之间的距离信息无法预先计算。

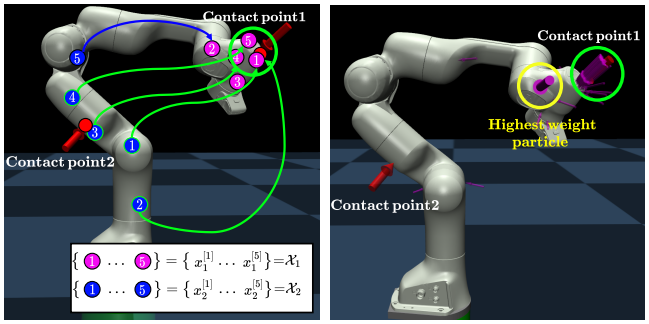
1) 网格预处理：如图 2(a) 绿色圆圈所示，存在一个无法发生接触的区域。我们使用 Blender 手动将其移除 [26]。此外，由于原始网格不规则（图 2(a) 中的红色圆圈），我们还应用了各向同性重网格算法[27]。图 2(b) 显示了生成的面。因此，我们可以使用连杆和面索引编号来表示粒子的位置，如下所示：

$$x_i^{[m]} = (link\#,\ face\#). \tag{10}$$

此外，我们使用 GP-toolbox[28] 计算所有面片对之间的测地距离（表面上的最短距离），并按升序存储，如图 2(c) 所示。借助这种数据结构，我们可以快速找到任意粒子附近的面。请注意，整个过程均离线完成，不会增加 MCP-EP 算法的计算负担。

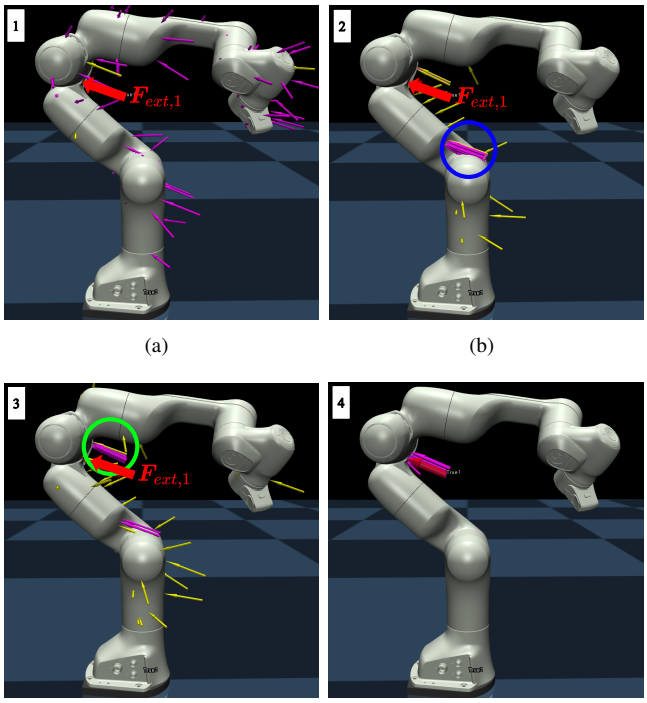
2) 改进运动模型：借助图 2(c) 中的数据结构，只需更改粒子的面索引（即 $face\#$ of $x_i^{[m]}$），即可简单地实现随机游走。除更改索引外，无需进行任何算法计算，因此计算成本极低。在我们的实现中，更新一个粒子约需 0.06μs。

然而，如图 2(d) 所示，不同连杆上各面的间距会随关节旋转而变化。因此，图 2(c) 中预先计算的表格仅对每个连杆有效，所以运动模型无法跨连杆更新粒子；也就是说，运动模型会改变 $face\#$，但不会改变 $link\#$。为解决这一问题，我们将在第三节-C 中引入探索粒子 。



(a) (b)

图 3. (a) 由于 $\mathcal{X}_1$ 已收敛到接触点 1，因此可以利用采样粒子（$x^*(\mathcal{X}_1)$）更新 $\mathcal{X}_2$ 中粒子的权重。(b) 假设施加了第二个力，但尚未添加新的粒子集。黄色和绿色圆圈分别表示权重最高的粒子和真实接触点 1。在这种情况下，权重最高的粒子并不代表真实接触点 1。



(a) (b) (c) (d)

图4。(a) 将接触力施加于连杆3，并初始化一个粒子集，使其散布在整个机器人上。(b) 粉色和黄色箭头分别表示基本粒子和探索粒子。所有基本粒子都收敛到了连杆2（见蓝色圆圈）。由于基本粒子无法从二次规划误差的角度充分解释传感器测量值，因此探索粒子被散布到相邻连杆上。(c) 一些基本粒子在更能解释传感器测量值的探索粒子周围重新采样（见绿色圆圈）。(d) 最终，所有基本粒子都收敛到了连杆3上的真实接触点。

B. 测量模型

由于较小的 QP 误差意味着粒子能够较好地解释传感器测量值，因此我们可以使用 QP 误差来定义粒子 $w_i^{[m]}$ 的权重。也就是说，QP 误差较小的粒子更有可能是真实接触点。因此，对于单接触情况，我们可以通过评估所有粒子的 QP 误差来计算相应的权重。

然而，多点接触情况会变得更加复杂，因为存在多个粒子集。计算粒子的权重时，应联合考虑其他粒子集。这也可以从 QP 问题（9）中的 $\boldsymbol{Q}$ 矩阵看出，其大小为 $4k \times (n+6)$，其中 k 为接触点数量。由于评估所有可能的粒子对需要大量计算，我们提出如下基于采样的方法。

假设我们关注粒子集 $k > 1$ 中的粒子 $x_1^{[m]} \in \mathcal{X}_1$ 的权重 $w_1^{[m]}$。那么，让我们从其他粒子集中采样粒子：$\boldsymbol{x}^* = \{x^*(\mathcal{X}_2)\ x^*(\mathcal{X}_3) \ldots, x^*(\mathcal{X}_k)\}$，其中 $x^*(\mathcal{X}_j)$表示根据相应权重从 $\mathcal{X}_j$ 中采样得到的粒子。通过此设置，可以通过求解以下二次规划问题来计算权重 $w_i^{[m]}$：

$$w_i^{[m]} = p(\hat{\boldsymbol{W}} \mid x_i^{[m]}) \propto \mathbf{exp}\left(-\alpha QP(\hat{\boldsymbol{W}} \mid x_i^{[m]}, \boldsymbol{x}^*)\right), \tag{11}$$

$$QP(\hat{\boldsymbol{W}} \mid x_i^{[m]}, \boldsymbol{x}^*) = \min_{\boldsymbol{f}_c} \left\| \hat{\boldsymbol{W}} - \boldsymbol{Q}^T \boldsymbol{f}_c \right\|^2, \tag{12}$$

其中 $\alpha > 0$ 用于确保权重与二次规划误差成反比。

以这种方式更新粒子的权重，假设其他粒子集已收敛到真实接触点。这个假设是合理的，因为力会依次施加到机器人上，而所提出的算法速度足够快，能够在下一个接触发生前完成收敛。例如，在图 3(a) 中，可以使用采样得到的粒子 $x^*(\mathcal{X}_1)$ 来更新 $\mathcal{X}_2$ 中粒子的权重，该粒子能够相当准确地估计真实接触点。

备注*1*：发生新的接触时，算法会添加一个新的粒子集，具体内容将在后文介绍。然而，算法无法立即识别这一情况；与此同时，如图3(b)所示，粒子会尝试移动以解释新的接触，而不添加额外的粒子集。根据我们的经验，采样 $\boldsymbol{x}^*$ 比按照某项标准手动选择 $\boldsymbol{x}^*$（例如，选择权重最高的粒子）表现出更强的鲁棒性，因为采样为测量模型引入了随机性。例如，在图3(b)中，黄色圆圈中的粒子并不代表真实接触点（位于绿色圆圈中），尽管它的权重最高。

C. 探索粒子

请注意，运动模型无法跨连杆更新粒子，因为它只改变 $face\#$。因此，一旦所有粒子都收敛到错误的连杆上，就没有机会在正确的连杆上重新采样粒子。为了解决这个问题，我们引入探索粒子 来考察相邻连杆（见图4(a)和(b)）。

1）更新探索粒子：算法1介绍了一种更新探索粒子的方法。其主要思路是将粒子集 $\mathcal{X}_i$ 划分为两个子集：基础子集（$\mathcal{X}_{i,\ basic}$）和探索子集（$\mathcal{X}_{i,\ explore}$）。如果粒子集的最大权重低于某个阈值 ϵ（即粒子集无法很好地解释接触点），则随机丢弃一个粒子

算法1更新探索粒子($\mathfrak{X}$)

对 $\mathcal{X}_i$ 中的 $\mathfrak{X}$ **执行**
  如果获取最大权重($\mathcal{X}_i$) $< \bar{\epsilon}$ **则**
    采样 $x_i^{[m]} \in \mathcal{X}_{i,\ basic}$
    从$\mathcal{X}_i$中丢弃 $x_i^{[m]}$, $basic$
    $x_i^{[m]} \leftarrow$ 运动模型近邻连杆($x_i^{[m]}$)
    将 $x_i^{[m]}$ 添加到 $\mathcal{X}_{i,\ explore}$
  else
    $\mathcal{X}_{i,\ basic} = \mathcal{X}_{i,\ basic} \cup \mathcal{X}_{i,\ explore}$
    $\mathcal{X}_{i,\ explore} = \emptyset$
  结束 if
结束 for
返回 $\mathfrak{X}$

从 $\mathcal{X}_{i,\ basic}$中。然后使用 Motion-Model-near-Link（第5行），向 $\mathcal{X}_{i,\ explore}$中添加一个与 $link\#$ 相邻的新粒子。否则，将 $\mathcal{X}_{i,\ explore}$ 合并到$\mathcal{X}_{i,\ basic}$ 中（第8–9行）。

2) 重要性重采样 *EP*（探索粒子）：探索粒子经历的重新采样过程与$\mathcal{X}_{i,\ basic}$中的粒子不同。基本粒子按照标准重要性重采样过程进行更新。不过，探索粒子不会进行重新采样。也就是说，基本粒子可以在探索粒子周围重新采样，但探索粒子不会在重新采样过程中消失。即使所有基本粒子都收敛到了错误的连杆上，也可以由于探索粒子的存在而在正确的连杆上重新采样（见图4(c)和(d)）。

算法 2管理粒子集($\mathfrak{X}$)

如果 $\epsilon(\hat{\boldsymbol{W}},\ \mathfrak{X}) > \bar{\epsilon}$ **然后**
  将 $\mathcal{X}_{init}$ 添加到 $\mathfrak{X}$
  返回 $\mathfrak{X}$
结束如果
遍历 $\mathcal{X}_i$ 于 $\mathfrak{X}$ **执行**
  如果 $\epsilon(\hat{\boldsymbol{W}},\ \mathfrak{X}\backslash\mathcal{X}_i) < \bar{\epsilon}$ **然后**
    从 $\mathfrak{X}$中丢弃 $\mathcal{X}_i$
    返回 $\mathfrak{X}$
  结束如果
结束循环
返回 $\mathfrak{X}$

D. 粒子集管理

由于多个接触是依次发生的，因此应适当调整粒子集的数量。为此，我们采用了 [20] 提出的方法，未作任何修改，并在算法2中给出该方法，使本文内容完整自洽。

在算法 2 中，$\epsilon(\hat{\boldsymbol{W}},\ \mathfrak{X}) = QP(\hat{\boldsymbol{W}} \mid \boldsymbol{x}^{\#}(\mathfrak{X}))$，其中$\boldsymbol{x}^{\#}(\mathfrak{X}) = \{x^{\#}(\mathcal{X}_1) \ldots,\ x^{\#}(\mathcal{X}_k)\}$，而 $x^{\#}(\mathcal{X}_i)$是 $\mathcal{X}_i$中权重最高的粒子。如果 $\mathfrak{X}$ 为空，则定义$QP(\hat{\boldsymbol{W}} \mid \boldsymbol{x}^{\#}(\mathfrak{X})) = \hat{\boldsymbol{W}}^T\hat{\boldsymbol{W}}$以初始化粒子滤波器。在第 1–4 行中，如果当前粒子集无法解释传感器测量值，则将新的粒子集 $\mathcal{X}_{init}$ 添加到 $\mathfrak{X}$中。在第 5–10 行中，如果无需某个粒子集 $\mathcal{X}_i$也能很好地解释传感器测量值，则将其从 $\mathfrak{X}$中丢弃。

算法3MCP-EP($\mathfrak{X}$)

1: $\mathfrak{X} \leftarrow$ 运动模型($\mathfrak{X}$) ▷在第 III-A 节中
2: $\mathfrak{X} \leftarrow$ 测量模型($\mathfrak{X}$) ▷在第 III-B 节中
3: $\mathfrak{X} \leftarrow$ 更新探索粒子($\mathfrak{X}$) ▷在第 III-C.1 节中
4: $\mathfrak{X} \leftarrow$ 重要性重采样 EP($\mathfrak{X}$) ▷在第 III-C.2 节
5: $\mathfrak{X} \leftarrow$ 管理粒子集 ($\mathfrak{X}$) ▷在第 III-D 节
6: **返回**($\mathfrak{X}$)

# of 接触	# of 粒子	成功率 (%)	收敛 step	RMSE 位置（厘米）	RMSE 力（牛顿）	运行时间 (ms)
1	100	99.96	18.25	0.16	0.01	0.45
2	200	96.70	11.00	1.08	2.00	1.66
3	300	81.63	14.37	3.50	4.63	4.50

表 I 仿真结果：概述

E. 从粒子集获取估计接触点和力

MCP-EP 的整体流程如算法3所示，其中上述方法按顺序执行。经过多次迭代算法3后，每个粒子集都会收敛到各自的接触点。使用 $x^{\#}(\mathcal{X}_i)$可以得到估计接触点（$\boldsymbol{r}_{c,\ i}$），其中 $x^{\#}(\mathcal{X}_i)$是 $\mathcal{X}_i$中权重最高的粒子。通过求解给定估计接触点（$QP(\hat{\boldsymbol{W}} \mid \boldsymbol{x}^{\#}(\mathfrak{X}))$）时的二次规划，可以得到估计接触力（$\boldsymbol{F}_{c,\ i}$）。

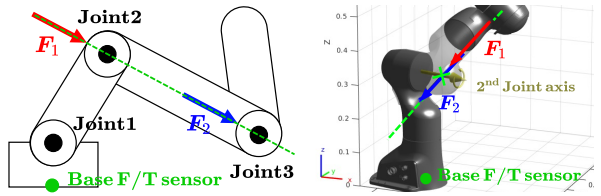
IV. 仿真验证

所提出的 MCP-EP算法在 Mujoco 仿真器中实现 [29]。实验采用配备关节扭矩传感器的 7 自由度 Franka E mika Panda 机械臂进行。使用 qpSWIFT [30]求解二次规划 (9)。该算法在搭载 Intel Core i7-12700 CPU、32 GB 内存的 PC 上，以 C++单线程程序运行并进行测试。

对单接触、双接触和三接触场景分别进行了 10000 次仿真。每次运行时，机器人构型均随机初始化，并在随机位置施加摩擦锥内的 20N 力（但每个连杆最多 1 个接触点）。为对算法进行定量评估，我们将定位误差为 $\leq$ 2.25cm（大致对应于 5 个面）的情况判定为成功。随后，成功率可计算为“成功次数/10000”。我们还报告了收敛步数，即成功前所需的迭代次数，以及位置和力的均方根误差（RMSE）和算法运行时间（每次迭代）。表 I 汇总了仿真结果。请注意，RMSE 的评估包含失败情形。下文将更详细地讨论单接触和双接触情形。

接触连杆	连杆1	连杆2	连杆3	连杆4	连杆5	连杆6	连杆7
成功率 (%)	100.00	99.93	100.00	100.00	100.00	99.86	99.93
RMSE 位置（厘米）	0.17	0.12	0.18	0.13	0.09	0.29	0.14
RMSE 力（牛顿）	0.01	0.01	0.01	0.01	0.01	0.02	0.02

表 II 仿真结果：单点接触情况



(a) (b)

图 5. (a) 单点接触下典型奇异情形的示意图。 F_1 和 F_2 无法区分，因为它们产生相同的传感器测量值。(b) 7 自由度机器人的一种奇异情形。

A. 单点接触情形

如表 I 所示，平均而言，成功率为 99.96%，位置和力的均方根误差分别为 0.16cm 和 0.01N。算法运行时间为 0.45ms，收敛需要 18.25 步。因此，在施加力后，该算法平均可在 8.21ms 内估计出接触情况。我们想强调，与现有方法 [19]–[21], [23] 相比，MCP-EP 的运行时间至少快 ≥ 3.5 倍。更详细的数据见表 II。

在 10000 次试验中，接触点定位仅失败 4 次，且均由奇异性导致。尽管使用基座力/扭矩传感器大幅缓解了奇异性问题，但仍存在某些情形：多个点力对都能完美解释给定的传感器测量值。例如，图 5 中的 F_1 和 F_2 会产生相同的传感器测量值，因此 MCP-EP算法可能会失败。尽管如此，除奇异性情况外（我们认为这类情况很少见），单点接触都能被准确识别。我们指出，表 II 中所有以厘米为尺度的均方根误差位置误差均小于 1 个面误差（粗略来说，1 个面误差对应 0.45 cm 的误差）。

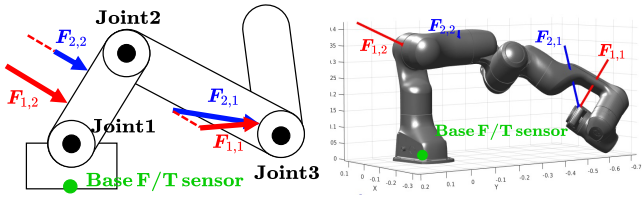
第二接触连杆

1st contact link		连杆1	连杆2	连杆3	连杆4	连杆5	连杆6	连杆7
连杆1	成功率（%）	X	97.84	95.2	98.91	100.00	97.96	98.71
	均方根误差（厘米）	X	0.99	1.46	0.62	0.44	0.92	0.38
	均方根误差（牛顿）	X	3.09	2.22	1.33	0.54	0.68	0.27
连杆2	成功率（%）	95.91	X	95.8	98.73	96.92	97.08	97.73
	均方根误差（厘米）	1.50	X	1.70	1.14	1.18	0.88	0.88
	均方根误差（牛顿）	3.62	X	4.00	2.52	1.36	0.88	0.68
连杆3	成功率（%）	97.03	95.42	X	94.07	98.69	98.04	98.21
	均方根误差（厘米）	1.37	1.33	X	1.91	1.03	0.97	0.80
	均方根误差（牛顿）	2.68	3.59	X	4.18	1.63	1.37	0.91
连杆4	成功率（%）	97.17	96.75	96.04	X	93.39	95.02	95.71
	均方根误差（厘米）	0.89	0.85	1.41	X	1.82	1.71	1.32
	均方根误差（牛顿）	1.33	1.69	3.85	X	3.83	2.82	1.79
连杆5	成功率（%）	97.49	97.7	96.81	91.91	X	96.36	97.17
	均方根误差（厘米）	1.01	1.06	1.18	1.49	X	1.42	0.89
	均方根误差（牛顿）	0.76	1.05	2.32	3.77	X	3.44	1.91
连杆6	成功率（%）	96.85	94.74	96.54	92.00	91.53	X	94.05
	均方根误差（厘米）	0.77	1.01	0.84	1.92	1.67	X	1.20
	均方根误差（牛顿）	0.64	0.97	1.07	2.40	3.55	X	4.05
连杆7	成功率（%）	99.55	98.82	99.61	98.71	97.61	97.91	X
	均方根误差（厘米）	0.35	0.36	0.42	0.64	1.00	0.88	X
	均方根误差（牛顿）	0.27	0.35	0.54	0.90	1.66	3.54	X

表III S仿真 结果：双接触-情形

B. 双接触情形

为模拟双接触场景，每隔 0.1 秒依次对机器人施加力。



(a) (b)

图6。(a) 双接触情况下奇异情形的示意图。红色和蓝色箭头会产生非常相似的传感器测量值。(b) 7 自由度机器人发生奇异的情形。

算法运行时间平均为 1.66 毫秒，且需要 11 步才能收敛（该数值表示添加第二个粒子集后的步数）。因此，施加第二个力后，MCP-EP 平均需要 18.26 毫秒来估计第二个接触点。

如表 I 所示，总体成功率为 96.70%。根据我们的分析，大多数失败案例属于以下两类之一。第一类是成功估计出一个接触点，但另一个接触点落入图 5 所示的奇异情形。第二类是 MCP-EP算法收敛到错误的估计结果，但该结果仍能较好地解释传感器测量值（因此陷入局部极小值）。事实上，如前所述，第一类情况很少见，因此大多数失败都属于第二类。图6展示了第二类情况的一个示例。由于红色和蓝色箭头可能产生非常相似的传感器测量值（尽管二者产生的数值并不完全相同），MCP-EP算法可能会收敛到错误的结果。

根据上述观察，96.70% 的成功率似乎相当合理，因为双接触情况下出现奇异性的可能性更高。此外，从表III中可以看出，第一个和第二个接触连杆距离越远，成功率越高。这一直观上是合理的，因为两个接触点之间的感知信息越丰富，算法越不容易陷入奇异性。我们还注意到，当第一次接触发生在连杆7上时（这在实际应用中是最常见的情况之一），平均成功率为 98.70%。

V. 结论

在本研究中，我们提出了一种名为 MCP-EP（带探索粒子的多接触粒子滤波器）的算法，用于解决多接触点定位与力识别问题。在所提出的算法中，基座力/扭矩传感器的使用使得能够以较小的奇异性估计多个接触点。在验证中，我们展示了该算法对一台 7 自由度机器人上三个接触点的估计结果（每个连杆最多一个接触点）。此外，所提出的算法通过适当的网格预处理提升了运行时间效率，其中所有面片对的距离信息均预先计算。然而，预处理后的数据仅在单个连杆范围内有效，因此粒子不会通过粒子滤波器的运动模型在不同连杆之间更新。为解决这一问题，我们在粒子滤波器算法中引入了探索粒子 这一概念，强制粒子探索相邻连杆。仿真研究展示了所提出算法的有效性。

参考文献

[1] D. M. Ebert 和 D. D. Henrich，“安全的人机协作：面向工业机器人的基于图像的碰撞检测”，载于 IEEE/RSJ智能机器人与系统国际会议，第 2 卷，2002 年，第 1826–1831 页。[2] S. Kuhn 和 D. Henrich，“已知物体与未知物体之间最小距离的快速视觉测定”，载于2007年IEEE/RSJ智能机器人与系统国际会议，第 2186–2191 页。[3] K. Kim 和 M. J. Kim，“一种用于机器人任务与运动规划的基于可达树的算法”，载于 2023年IEEE机器人与自动化国际会议（ICRA）。[4] M. J. Kim、W. Lee、J. Y. Choi、G. Chung、K.-L. Han、I. S. Choi、C. Ott 和 W. K. Chung，“一种基于无源性的非线性导纳控制及其在未知环境交互下动力辅助上肢控制中的应用”，IEEE/ASME机电一体化汇刊，第 24 卷，第 4 期，第 1473–1484 页，2019 年。[5] M. J. Kim、W. Lee、C. Ott 和 W. K. Chung，“一种利用反馈互联的基于无源性的导纳控制设计”，载于 2016年IEEE/RSJ智能机器人与系统国际会议（IROS），第 801–807 页。[6] A. De Luca、A. Albu-Schaffer、S. Haddadin 和 G. Hirzinger，“DLR-III轻型机械臂的碰撞检测与安全反应”，载于 2006年IEEE/RSJ智能机器人与系统国际会议，第 1623–1630 页。[7] M. J. Kim、Y. J. Park 和 W. K. Chung，“刚性与柔性关节机器人的基于动量的扰动观测器设计”，《智能服务机器人》，第 8 卷，第 57–65 页，2015 年。[8] Y. J. Heo、D. Kim、W. Lee、H. Kim、J. Park 和 W. K. Chung，“工业协作机器人的碰撞检测：一种深度学习方法”，IEEE机器人与自动化快报，第 4 卷，第 2 期，第 740–746 页，2019 年。[9] S. A. B. Birjandi、J. K¨uhn 和 S. Haddadin，“利用本体感知与惯性测量单元传感，在机器人操作臂中进行碰撞监测的观测器扩展直接法”，IEEE机器人与自动化快报，第 5 卷，第 2 期，第 954–961 页，2020 年。[10] D. Kim、D. Lim 和 J. Park，“利用通用模块化神经网络实现协作机械臂可迁移的碰撞检测学习”，IEEE机器人学汇刊，2021 年。[11] K. M. Park、Y. Park、S. Yoon 和 F. C. Park，“使用无监督异常检测算法进行机器人操作臂碰撞检测”，IEEE/ASME机电一体化汇刊，2021 年。[12] J. M. Gandarias、A. J. Garc´ıa-Cerezo 和 J. M. G´omez-de Gabriel，“利用高分辨率触觉传感器进行物体识别的基于卷积神经网络的方法”，IEEE传感器期刊，第 19 卷，第 16 期，第 6872–6882 页，2019 年。[13] J. Liang、J. Wu、H. Huang、W. Xu、B. Li 和 F. Xi，“用于护理机器人的安全控制的柔性敏感皮肤：结合邻近传感器与触觉传感器”，IEEE传感器期刊，第 20 卷，第 7 期，第 3822–3830 页，2020 年。[14] A. Zwiener、C. Geckeler 和 A. Zell，“结合本体感知传感器与机器学习的关节式机械臂接触点定位”，载于 2018年IEEE机器人与自动化国际会议（ICRA），第 323–329 页。[15] S. Haddadin、A. De Luca 和 A. Albu-Sch¨affer，“机器人碰撞：检测、隔离与识别综述”，IEEE机器人学汇刊，第 33 卷，第 6 期，第 1292–1312 页，2017 年。[16] G. Buondonno 和 A. De Luca，“结合真实与虚拟传感器测量作用于机器人的交互力与力矩”，载于 2016年IEEE/RSJ智能机器人与系统国际会议（IROS），第 794–800 页。[17] M. Iskandar、O. Eiberger、A. Albu-Sch¨affer、A. De Luca 和 A. Diet-rich，“利用传感冗余在 DLR SARA机器人上进行碰撞检测、识别与定位”，载于 2021年IEEE机器人与自动化国际会议（ICRA），第 3111–3117 页。[18] T. Pang、J. Umenberger 和 R. Tedrake，“识别串联机械臂关节力矩测量中的外部接触及其局限性”，载于 2021年IEEE机器人与自动化国际会议（ICRA），第 6476–6482 页。[19] D. Popov、A. Klimchik 和 A. Pashkevich，“多个潜在接触位置与接触力的实时估计”，IEEE机器人与自动化快报，第 6 卷，第 4 期，第 7025–7032 页，2021 年。[20] L. Manuelli 和 R. Tedrake，“利用本体感知传感器定位外部接触：接触粒子滤波器”，载于 2016年IEEE/RSJ智能机器人与系统国际会议（IROS），第 5062–5069 页。

[21] A. Zwiener、R. Hanten、C. Schulz 和 A. Zell，“Armcl：通过蒙特卡洛定位进行机械臂接触点定位”，载于*2019*年*IEEE/RSJ*智能机器人与系统国际会议（*IROS*），第7105–7111页。[22] J. Bimbo、C. Pacchierotti、N. G. Tsagarakis 和 D. Prattichizzo，“利用关节力矩传感进行机器人碰撞检测与隔离”，载于*2019*年*IEEE/RSJ*智能机器人与系统国际会议（*IROS*），第7604–7609页。[23] D. Popov 和 A. Klimchik，“协作机器人实时外部接触力估计与定位”，载于*2019*年*IEEE*机电一体化国际会议（*ICM*），第1卷，第646–651页。[24] N. Likar 和 L. ˇZlajpah，“基于外部关节力矩的接触信息估计”，《国际先进机器人系统杂志》，第11卷，第7期，第107页，2014年。[25] S. Haddadin、A. Albu-Schaffer、A. De Luca 和 G. Hirzinger，“碰撞检测与响应：对安全物理人机交互的贡献”，载于*2008*年*IEEE/RSJ*智能机器人与系统国际会议，第3356–3363页。[26] R. Hess，《Blender基础：学习Blender 2.6必备指南》。Focal Press，2010年。[27] D.-M. Yan、B. L´evy、Y. Liu、F. Sun 和 W. Wang，“通过快速精确计算受限Voronoi图进行各向同性重网格化”，载于《计算机图形论坛》，第28卷，第5期。Wiley在线图书馆，2009年，第1445–1454页。[28] A. Jacobson 等，“gptoolbox：几何处理工具箱”，2021年，http://github.com/alecjacobson/gptoolbox。[29] E. Todorov、T. Erez 和 Y. Tassa，“MuJoCo：用于基于模型控制的物理引擎”，载于*2012*年*IEEE/RSJ*智能机器人与系统国际会议，第5026–5033页。[30] A. G. Pandala、Y. Ding 和 H.-W. Park，“qpSWIFT：面向机器人应用的实时稀疏二次规划求解器”，*IEEE*机器人与自动化快报，第4卷，第4期，第3355–3362页，2019年。