

# 论文阅读与复现报告

## 一、研究背景

这篇文章的研究背景是机器人群体中的自重构机器人系统。自重构机器人是指能够动态连接和交互的机器人，它可以通过重新排列模块的连接来改变其形状，以执行新任务，并且适应不断变化的环境。

与传统机器人不同的是，传统机器人大多通过自身控制程序的改变来实现不同的工艺操作，但其固定的构形仍然大大限制了所能完成的任务范围；而自重构模块化机器人则能够在运行期间配置其形状。它们可以形成复杂的结构，如桥梁和绳索，并且可以在环境中独立导航。鉴于其灵活度高、环境适应能力强以及成本低等特点，可重构模块机器人在工业装配、医疗、军事、灾难救援、深空探测及核泄露事故处理等领域具有广泛的应用前景。

具体而言，自重构模块化机器人由一组相互连接的模块组成，无需使用额外工具，便可利用改变其配置和结构的优势来执行特定或复杂的任务。Feczko 等将这种特性定义为“自我重构”，这表明自重构模块化机器人可以在没有任何外部干预的情况下改变其结构形成<sup>[1]</sup>。在最新的研究论文和综述，我们总结了部分先进的自重构模块化机器人。

表 1 自重构模块化机器人总结

名字	结构	时间
FreeBot	Free Form	2020
Mosar WM	Chain	2020
ARC-Alpha	Chain	2020
Karakassa	Hybrid	2020
Isoperimetric	Truss	2020
Legged Swarm	Hybrid	2021
Datom	Larrice	2021
SMMRob	Hybrid	2022

比较先进的方案，也是这篇文章关注的方案，是链式类型，它们按照按串行顺序连接模块。最大的优势在于它允许自重构模块化机器人在一个平台上受益于各种优势。例如，链式队形使得在崎岖地形上的运动更加容易，也能够实现爬楼梯和进入狭窄的洞等特殊任务，就像一条蛇一样。

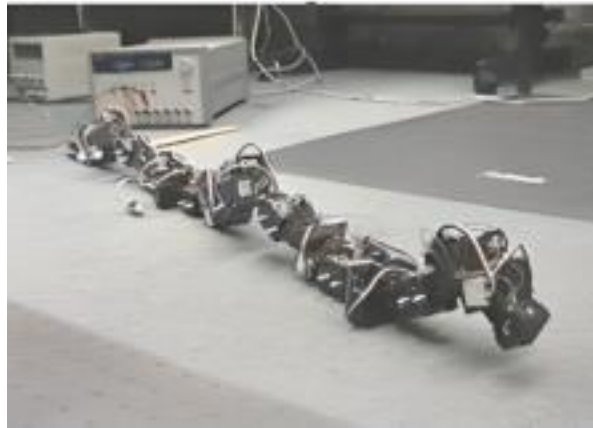


图 1 蛇形-自重构模块化机器人

同时，也可以通过设置链环执行滚动步态，从而减少行进时间，类似多足类动物。例如，建立四足队形还可以提供步行步态的好处，一群小型机器人可以在执行指定任务时合并，完成从一个点移动到另一个点的任务，当融合机器人到达目标时，小模块从主体结构上脱离，以完成其他任务。这样，通过不同模块的协同工作，可以为不同的目的创建不同的阵型。



图 2 链环形-自重构模块化机器人

很自然的，自重构模块化机器人的机械结构应该是模块化的，并且模块应该具有运动能力。在大多数时候，模块应当可以相互连接和分离。但是，这也带来了一个挑战，就是如何设计相应的模块的对接和脱离机制，以提供这种能力。

在过去的研究中，这种动态耦合和解耦的模块化机器人大多使用机械或主动式的磁力连接，例如 ATRON<sup>[2]</sup>，M-TRAN III<sup>[3]</sup> 和 M-blocks<sup>[4]</sup>。但是这种连接机制存在以下两个问题：

一是由于动力和动态结构专门用于耦合机制，每个模块的机动性与常规移动机器人相比有限，从而限制了它们在环境中的导航能力；

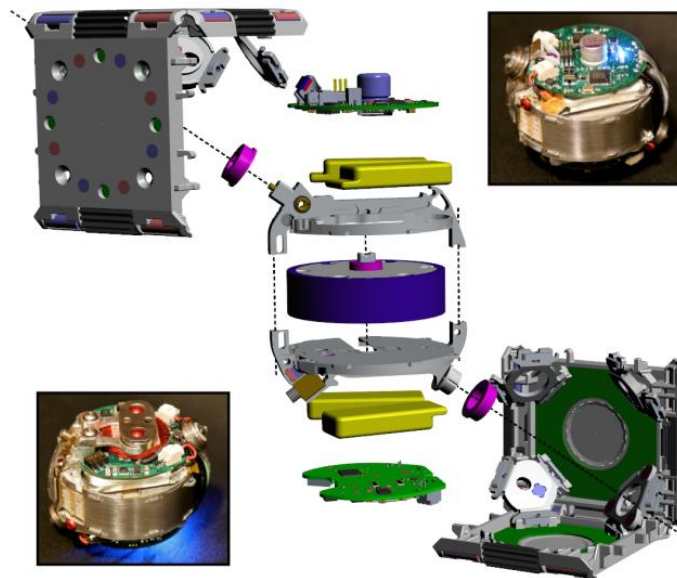


图 3 M-blocks 示意图

二是磁力致动耦合机制在维持耦合状态时会消耗较大功率；且尽管磁性连接的剪切力明显较低，磁铁的极性也限制了连接的方向性，因此连接的强度对错位非常敏感<sup>[5]</sup>。

为了解决第二个问题，很多研究使用了被动耦合机制来形成和维持结构，如 ModQuad<sup>[6]</sup>，Roboats<sup>[7]</sup>，SlimeBot<sup>[8]</sup>和 PuzzleBots<sup>[9]</sup>。然而，使用这些被动机制时，机器人在耦合在一起时形成了刚性的 2D 连接，这极大地限制了它们的功能性和机动性。

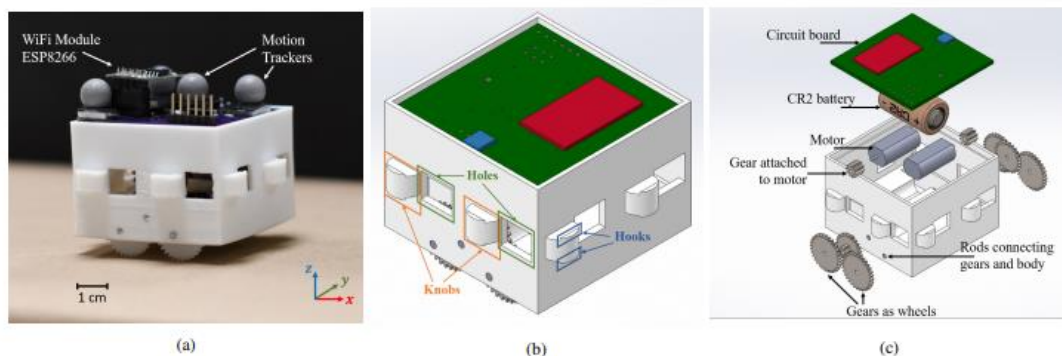


图 4 PuzzleBots 示意图

以 PuzzleBots 为例，Yi 等人<sup>[9]</sup>设计了一种低成本的机器人群，机器人可以在物理上相互耦合，并以最低的能耗形成功能结构。每个机器人的侧面都有旋钮和孔，以便机器人可以通过插入旋钮来耦合。Yi 等利用基于二次规划（QP）的配置控制算法，限制两个机器人之间的一对点来维持耦合状态。这种点约束对定位误差非常敏感，并严重限制了机器人的运动。

进一步的研究把机器人不再视为点质量，而是选择使用多面体来描述耦合行为的几何性质。这里的核心关切是避免碰撞<sup>[10, 11]</sup>。比如，Thirugnanam 等人<sup>[10]</sup>提出使用非光滑的基于对偶的安全关键最优控制函数，通过基于 QP 的优化问题，

并引入对偶优化，来表示多胞体和多胞体之间的最小距离，最终应用对偶形式的拉格朗日函数来构造控制屏障功能。

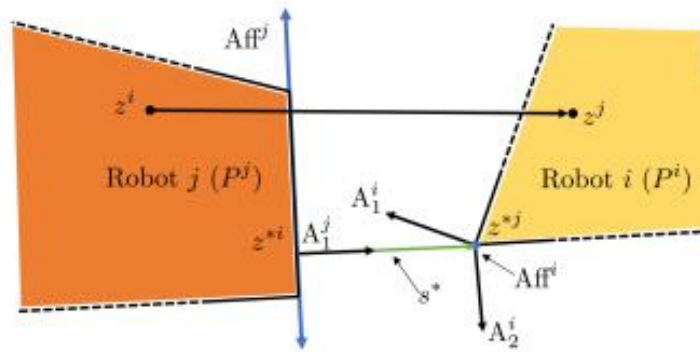


图 5 耦合的几何建模

这篇文章即聚焦于这一方向，文章的目标是设计一种被动耦合机制，使机器人能够形成与环境结构相符的功能配置，并控制组装的结构以执行集体行为。作者注意到，过去方案的方案只能形成刚性结构，而由于柔性物体的变形性和灵活的状态，原有的方案对其进行建模是困难的。通过被动耦合机制，机器人需要对其运动进行精确控制以保持连接。这样的维护约束应具有足够的约束性，以使机器人不容易解耦，并具有足够的放松程度，以使组装结构可以在环境中移动。

所以，作者为了解决现有机机械或磁性连接机制的问题，提出了一种低成本的软锚设计，在具有强连接的同时，让机器人仍然能够动态解耦，这样就实现了机器人之间的柔性耦合和解耦。通过模型预测控制和多边形约束，维持耦合机制的组装结构，使机器人能够耦合、解耦并在模拟环境和硬件平台上执行各种行为。

以垂直结构为例，当锚插入到另一个机器人的空间中时，它充当一个未经激活的关节。特别的是，这个锚不具有对称性，也就是说，将其插入一个空间所需的力很小，而将其拉出所需的力量大。柔性锚形成的未经激活的关节也为组装结构中的机器人提供了在关节限制下旋转或平移的空间，与机器人的运动相符。作者将柔性锚设计限制为仅具有一个自由度（偏航旋转），其状态可以通过相对位置和位移进行估计。通过获取柔性锚的力学特性，最终可以将其建模并仿真为刚性组件链接。

## 二、主要贡献

作者将本文的主要贡献概括为三个方面<sup>[12]</sup>：

第一，提出了一种非对称的软体结合机构，这一机构允许机器人相互结合形成灵活的/刚体的结构；

第二，在仿真中提出了一种三杆联结系统，可以有效地实现对软锚的建模，并且在仿真环境中模拟出相似于真正硬件系统的机械行为；

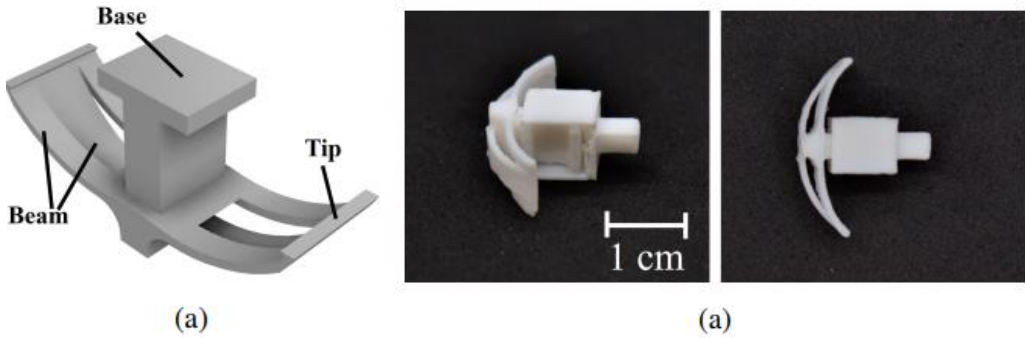
第三，基于以上的机构设计，提出了一个带有多边形约束的基于模型的预测控制框架来为结合形体的结合形态进行精确的建模。

而我们小组在阅读之后认为本文的创新点可以进一步概括为两个方面：

第一，提出了一种非对称的软锚机构，可以让机器人在较低功耗状态下完成

结合，需要额外能量维持结合状态并且面对干扰、外力破坏、计算误差具有鲁棒性，同时该结构具有一定的灵活性，可与机器人侧边的固定结合纽扣搭配构成更加多样化的集合体；

第二，提出了一种专用的控制算法，通过多边形约束确保维持结合状态，通过前向叠加误差函数来拟合目标运动状态，从而在较低的计算量下完成了对本论文中的软锚小车的预测、规划和控制。



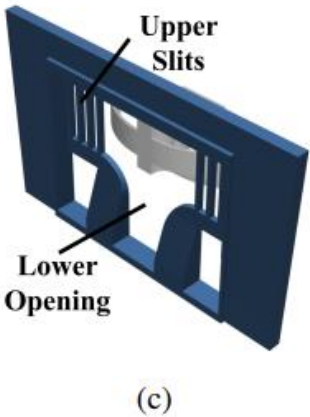
### 三、理论介绍

#### 3.1. 软锚机构

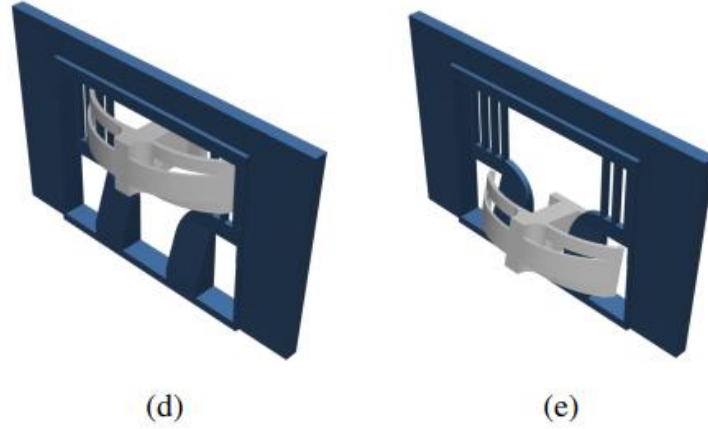
本文的第一个创新点在于提出了这样一种软锚机构。作者团队的设计初衷是实现三个性能的兼顾：激活后有足够的力来维持结合状态，对抗外力；采用全被动的结合形式来节约能源；当其作为一个集成机构被用在机器人集群时，要有充足的灵活性。而软锚的提出确实达成了团队的设计理念。

##### 1、方法

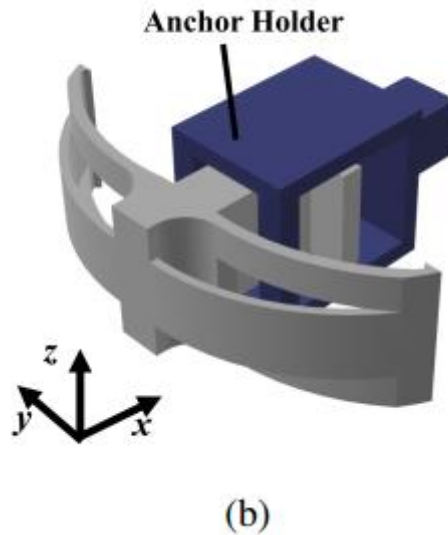
在结合过程中，首先需要锚头对准被结合的机器人，然后挤压锚头使其进入被结合机器人的箱体内部（见图 2c），从而锚头从被挤压的形变状态恢复到自然伸展状态，其宽度可以使得两边的尖端可以卡进滑孔或空槽内（见图 2d，2e）。这两种不同的结合状态 对应于不同的功能，前者用于一般的平面运动，后者可以在有高低落差的情景下形成稳定的链条来运动。这一设计也为机器人集群提供了额外的灵活性。







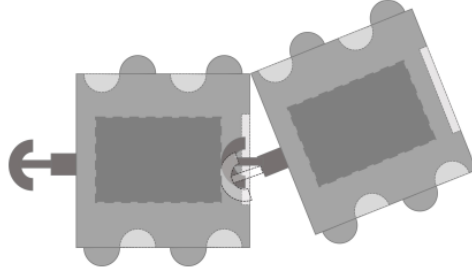
结合后的软锚不仅具有被动的刚体结合的稳定性的，还有一定的灵活活动的空间。以下图 2b 中的坐标系为例，软锚在结合后仍然可以在一定范围内沿 x 轴方向自由移动，或者围绕偏航角方向（图中 z 轴）自由转动，在 y 轴方向和俯仰角方向（图中 y 轴）上也有微小的移动范围（分别为 $<0.5\text{mm}$  和 $<0.\text{rad}$ ）。而这些有限的移动空间的精确表述，也为后续建模预测控制的框架设计奠定了基础。



在解耦过程中，团队使用了一种非常精妙的摆动模式来完成，是由一个固定速度的平移运动和一个角速度周期性变化的转动运动合成来实现的，公式如下。在这个过程中，锚头会相对于被结合机器人产生如下偏转（见图 4b），从而使得锚头可以斜侧着从结合口中平移出来，完成解耦过程。由于这一过程需要转动运送和平移运动之间精确的配合，所以我们只需要在 MPC 控制框架中加入一个约束见下，即可确保，在机器人完成作业的过程中不会发生我们不需要的解耦过程。

以上这些设计不仅使得结合机构具有一定的灵活度，并且在面对外力破坏时，可以提供相当大的阻力来维护，而且对于干扰和计算误差也是鲁棒。相比于基于磁力的结合机构，本文中的软锚不需要任何额外电力就可以维持结合状态，并且可以提供一定程度的切变力。可以说软锚结构在理论上是同时具备磁力结合和钢体结合两方面的优势的。这些特性在下面的实验中也获得了数据支持。

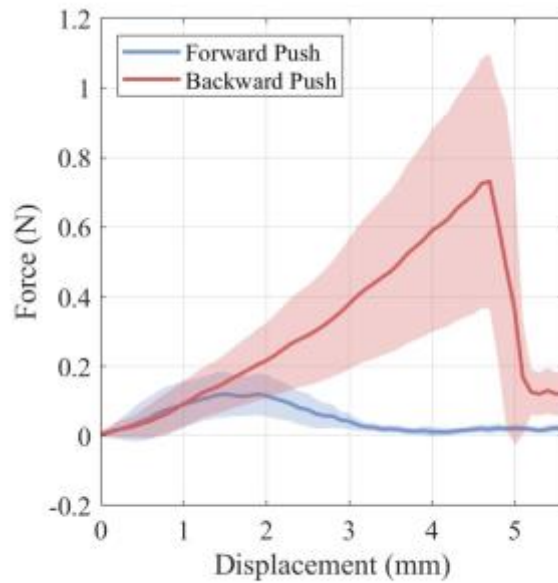
$$\begin{bmatrix} v \\ w \end{bmatrix} = \begin{bmatrix} v_{\text{bias}} \\ w_{\text{max}} \sin(Bt) \end{bmatrix}$$



(b)

## 2、结果

本文关于软锚力学特性的实验验证，主要针对软锚结合时需要的力和解耦时所需力来进行。如下图所示，作者团队直接测量了前向推力和后向拉力与软锚机构不同程度的形变之间的对应关系。可以看出在两毫米到五毫米之间的形变范围内，所需的前向推力和后向拉力具有相当显著的差异。前向推力始终未超过 0.2N 而后向拉力则保持在 0.2N 以上甚至超过了 0.4N。更具体来说，从后向施加拉力使得软锚解耦所需的形变程度是 3mm，对应的力约为 0.4N，数倍于结合过程中所需的前向推力（文中未给出，结合几何特征可以认为所需小于 3mm 所需推力小于 0.15N）。这一特质赋予了软锚机构，结合时省力，而结合后可以承载较大外力的良好特性。而在进一步的实验中，由于在最大速度下运行的机器人集群中，软锚承担推力力约是 0.5N。以上这些数字刚好能保证软锚轻松结合，同时也使得软锚很难在一个直接拉力的作用下解耦。在另一项实验中，作者团队测量了，滑槽结合状态下（见图 2e）的单个软锚可以负载的质量为 500g，约是七个完整机器人的重量，显示出该机构强大的负载能力。



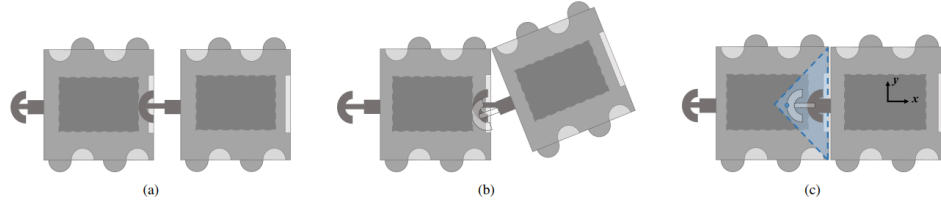
(a)

### 3.2. MPC 框架

本文的第二个创新点在于提出了一套计算经济且行之有效的 MPC 框架，从而将软锚的理论可能转化为可以在实际作业中发挥作用的特性，充分利用易结合，易保持，不易解耦这些性能。

#### 1、方法

第一步，将保持结合状态的约束形式转换为多边形内点问题，从而得到方便计算的形式化约束。如何让将保持结合状态的约束引入控制算法，是所有被动式结合机器人控制都要面临的问题。在此前的工作中，研究人员往往是通过最小化结合处两个机械单元上的某个固定点位之间的距离，来得到保持结合状态所需的约束条件。这种方法在形式上已经限制了结合处的移动空间，降低了灵活性，而在集成后更是大大缩小了机器人集群的移动空间。本文为了克服这些困难，提出了松弛的多边形约束以保持结合状态。首先我们将自然延展的软锚刚进入箱体时的中立位置定义为零位置（见图 4a）。当两侧的机器人发生相对运动时，软锚的位姿也发生变化。在实际运动和作业的过程中需要保证软锚的位姿不会出现易于解耦的情况，所以，研究人员对软锚头部顶点的位置（见图 4c 中的蓝色小点）施加了一个三角形的约束范围（见图 4c 中蓝色虚线围成的三角区域）。



而这个约束就转化为经典的多边形内点问题，已经有很多现成的经典方法可以解决（如光线投射算法和回转数算法等等），但这些方法由于其昂贵的而计算代价难以集成到 MPC 框架中来。研究人员通过观察这个具体的内点问题的特性，发现此处需要计算的多边形边数较少且是凸的，所以可以直接通过向量叉乘来判断内点。依次定义多边形顶点为 $A_1, A_2, \dots, A_K$ ，其中 $A_k \in R^2, k \in 1, 2, \dots, K$ ，待判定的点 $P \in R^2$ ，则待判定点为内点等价于下式。

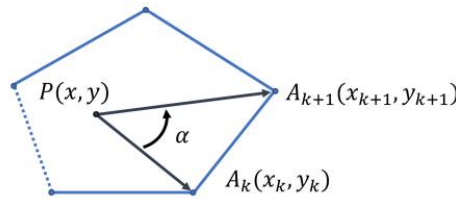


Fig. 5: A point inside a convex polygon.

$$\overrightarrow{PA_k} \times \overrightarrow{PA_{k+1}} \geq 0.$$

经过化简可以得到对待判定点坐标的约束为：

$$\begin{bmatrix} y_{k+1} - y_k & -(x_{k+1} - x_k) \end{bmatrix} \left( \begin{bmatrix} x \\ y \end{bmatrix} - \begin{bmatrix} x_k \\ y_k \end{bmatrix} \right) \leq 0 \quad (3)$$

以上约束简记作：

$$pip(P|A_1, \dots, A_K) \leq 0,$$



第二步，针对不同的任务类型设计不同的最小化函数。本文共提出了三种不同的任务类型。

1)、链接对对齐

$$J_c(\mathcal{C}_{active}) = \sum_{(C_i, C_j) \in \mathcal{C}_{active}} \left( \Delta p^\top W_p \Delta p + W_\theta \tan^2 \frac{1}{2} \Delta \theta \right) \quad (5)$$

其中  $\mathcal{C}_{active} = (C_i, C_j), \dots, i, j \in 1, \dots, N$ ,  $C_i, C_j$  为机器人  $i, j$  各自的连接点由各自的刚体变换阵来定义见下式。(先从世界框架变换到机器人框架, 再考虑相对于机器人框架下的位姿)

$$g_{SB_i} = \begin{bmatrix} R(\theta_i) & p_i \\ 0 & 1 \end{bmatrix}, g_{B_i C_i} = \begin{bmatrix} R(\theta_{C_i}) & p_{C_i} \\ 0 & 1 \end{bmatrix} \in SE(2) \quad (4)$$

$$\Delta p = R(\theta_i) p_{\{C_i\}} + p_i - (R(\theta_j) p_{C_j} + p_j), \Delta \theta = \theta_i + \theta_{C_i} - \theta_j + \theta_{C_j},$$

$W_p \in R^{2 \times 2}, W_\theta \in R$  分别为位置差异的对角权重矩阵和角度差异的权重。为了能

够给出合适的值加权计算, 并且保证可导性, 这里使用了正切函数。

2)、实现给定的目标位姿。

$$J_g = \sum_{g_i \in \mathcal{G}} (p_i - g_i)^\top W_g (p_i - g_i) \quad (6)$$

其中  $G = g_i, \dots, g_i \in R^2, i \in 1, \dots, N$  为给定的目标位姿,  $W_g \in R^{2 \times 2}$  为对角权重矩阵

3)、拟合给定的指令速度

$$J_v = [v_0 - v_0^*, \dots, w_N - w_N^*] W_v [v_0 - v_0^*, \dots, w_N - w_N^*]^\top \quad (7)$$

其中  $[v_0^*, \omega_0^*, \dots, v_N^*, \omega_N^*]^\top$  为给定的速度指令,  $W_v \in R^{2N \times 2N}$  为对角权重矩阵。综合以上推导, 可以得到实际使用的优化问题的形式。

$$\begin{aligned} \min_{x,u} \quad & W_f J(t + H_m|t) + W_c \sum_{k=0}^{H_c} J_c(\mathcal{C}_{conn}, t + k|t) \\ & + W_m \sum_{k=0}^{H_m-1} J(t + k|t) + W_s u^\top u \end{aligned} \quad (8)$$

$$\text{s.t. } x(t|t) = x(t) \quad (9)$$

$$\begin{aligned} x(t + k + 1|t) = x(t + k|t) \\ + f(x(t + k|t), u(t + k|t)) \Delta t \end{aligned} \quad (10)$$

$$\text{for } k = 0, \dots, H_m - 1$$

$$\text{pip}(C_i(t + k|t)|R_j(t + k|t), C_j^r(t + k|t), C_j^l(t + k|t)) \quad (11)$$

$$\text{for } k = 1, \dots, H_c, (C_i, C_j) \in \mathcal{C}_{conn}$$

$$x(t + k|t) \in \mathcal{X}, u(t + k|t) \in \mathcal{U}, k = 0, \dots, H_m \quad (12)$$

其中 $J(t + k|t) \in \mathcal{J} = J_c, J_g, J_v$ 为根据任务不同选择的损失函数， $\mathcal{C}_{conn}$ 为已经连接的对， $H_c \leq H_m$ 来降低计算维持联结状态的开销以进一步加速计算（因为联结本身的易保持性）。

最终完整的 MPC 框架如下三个算法组成：

---

**Algorithm 1** Augment Connection Pairs

---

**Input:**  $\mathcal{C} = \{(C_i, C_j), \dots\}$ : set of connection pairs

**Output:**  $\tilde{\mathcal{C}}$ : augmented set of connection pairs

**Initialize:**  $\tilde{\mathcal{C}} = \{\}$

```

1: function AUGMENTPAIRS( $\mathcal{C}$ )
2:   for  $(C_i, C_j)$  in  $\mathcal{C}$  do
3:      $(C_i, C_j).status = decoupled$ 
4:      $(C_i, C_j).type = anchor \text{ or } knob$ 
5:      $(C_i, C_j).anchor\_index = \text{getAnchorIndex}(C_i, C_j)$ 
6:      $(C_i, C_j).head = (C'_i, C'_j)$  based on  $anchor\_index$ 
7:      $\tilde{\mathcal{C}}.append((C_i, C_j))$ 
8:   return  $\tilde{\mathcal{C}}$ 
```

---

1 算法的作用在于完成连接对的初始化，将一些必要信息存储进统一的数据结构中，并开辟变量保存连接对的状态，为后续记录连接对状态变化做准备。

---

**Algorithm 2** Update Connection Pair Lists

---

**Input:**  $C_{conn}$ : connected pairs,  $C_{active}$ : active pairs,  $\tilde{C}$ : augmented pair list,  $\epsilon$ : threshold

**Output:**  $C_{conn}$ : connected pairs,  $C_{active}$ : active pairs

```
1: function UPDATEPAIRS( $C_{conn}, C_{active}, \tilde{C}, \epsilon$ )
2:   for ( $C_i, C_j$ ) in  $C_{active}$  do
3:      $a = (C_i, C_j).anchor\_index$ 
4:     if ( $C_i, C_j$ ).status is decoupled then
5:       if ( $C_i, C_j$ ).head in polygon( $R_{\sim a}, \epsilon$ ) then
6:         ( $C_i, C_j$ ).status  $\leftarrow$  head_aligned
7:         continue
8:       if ( $C_i, C_j$ ).status is head_aligned then
9:         if  $C_i$  in polygon( $R_j, \epsilon$ ) then
10:          ( $C_i, C_j$ ).status  $\leftarrow$  head_inserted
11:          continue
12:         if ( $C_i, C_j$ ).head not in polygon( $R_{\sim a}, \epsilon$ ) then
13:          ( $C_i, C_j$ ).status  $\leftarrow$  decoupled
14:         if ( $C_i, C_j$ ).status is head_inserted then
15:           $C_{active}.remove(C_i, C_j)$ 
16:           $C_{conn}.append(C_i, C_j)$ 
17:   return  $C_{conn}, C_{active}$ 
```

---

2 算法负责检查和更新连接对状态。

---

**Algorithm 3** Align Connection Pairs

---

**Input:**  $T$ : target configuration,  $x$ : robot states at time  $t$

**Output:**  $u$ : control input

**Initialize:**  $C_{conn} = \{\}$ ,  $C_{active} = \{\}$

```
1: function ALIGNCONNECTIONPAIRS( $T, x$ )
2:    $C_{goal} \leftarrow assignConnectionPairs(T, x)$  if not assigned
3:    $\tilde{C} \leftarrow augmentPairs(C_{goal})$ 
4:    $C_{active} \leftarrow assignActivePair(\tilde{C}, x)$  without conflict
5:    $u \leftarrow MPC(x, C_{conn}, C_{active})$ 
6:    $C_{conn}, C_{active} \leftarrow updatePairs(C_{conn}, C_{active}, \tilde{C}, \epsilon)$ 
7:   return  $u$ 
```

---

3 算法在上述内同的基础上完成对目标构型的对齐, 输出最终的控制信号  $u$ , 并更新连接对状态, 为多次运动和维护连接对状态做准备。

## 2、实验结果

本文的 MPC 框架验证实验分为模拟验证和实际验证两个部分

1)、在模拟仿真中, 为了检验计算性能, 研究团队设定参数  $H_m = 3, 5, 10$ ,

$H_c = 3$ ,  $\Delta t = 0.1s$ , 并分别在组装 2, 4, 6, 8 个机器人的任务上进行了重复测试, 结果如下图 7b。为了验证 MPC 可以有效结合软锚机构, 研究团队在不同的处置偏置上测试了 100 次, 以一分钟内完成连接为成功作为标准, 得到了成功率曲线和平均连接时长曲线。可以看到多数情况下成功率可以达到 80%, 平均时间不超过九秒。

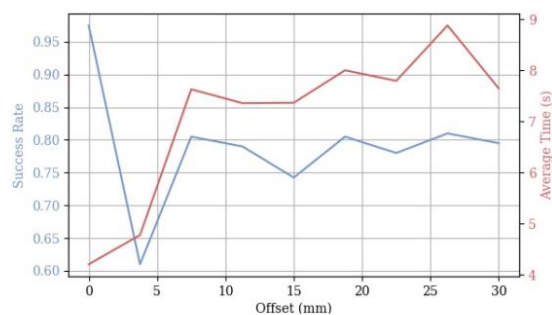
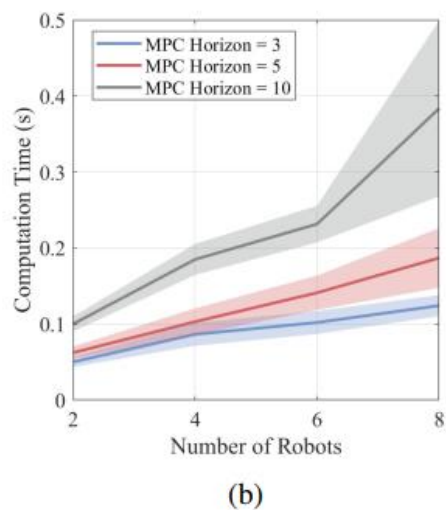
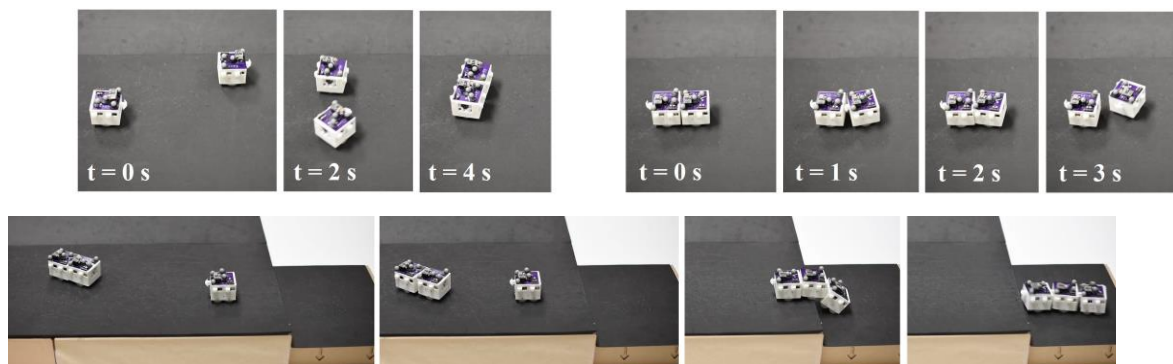


Fig. 10: The success rate and average time of two robots coupling with a given offset distance.

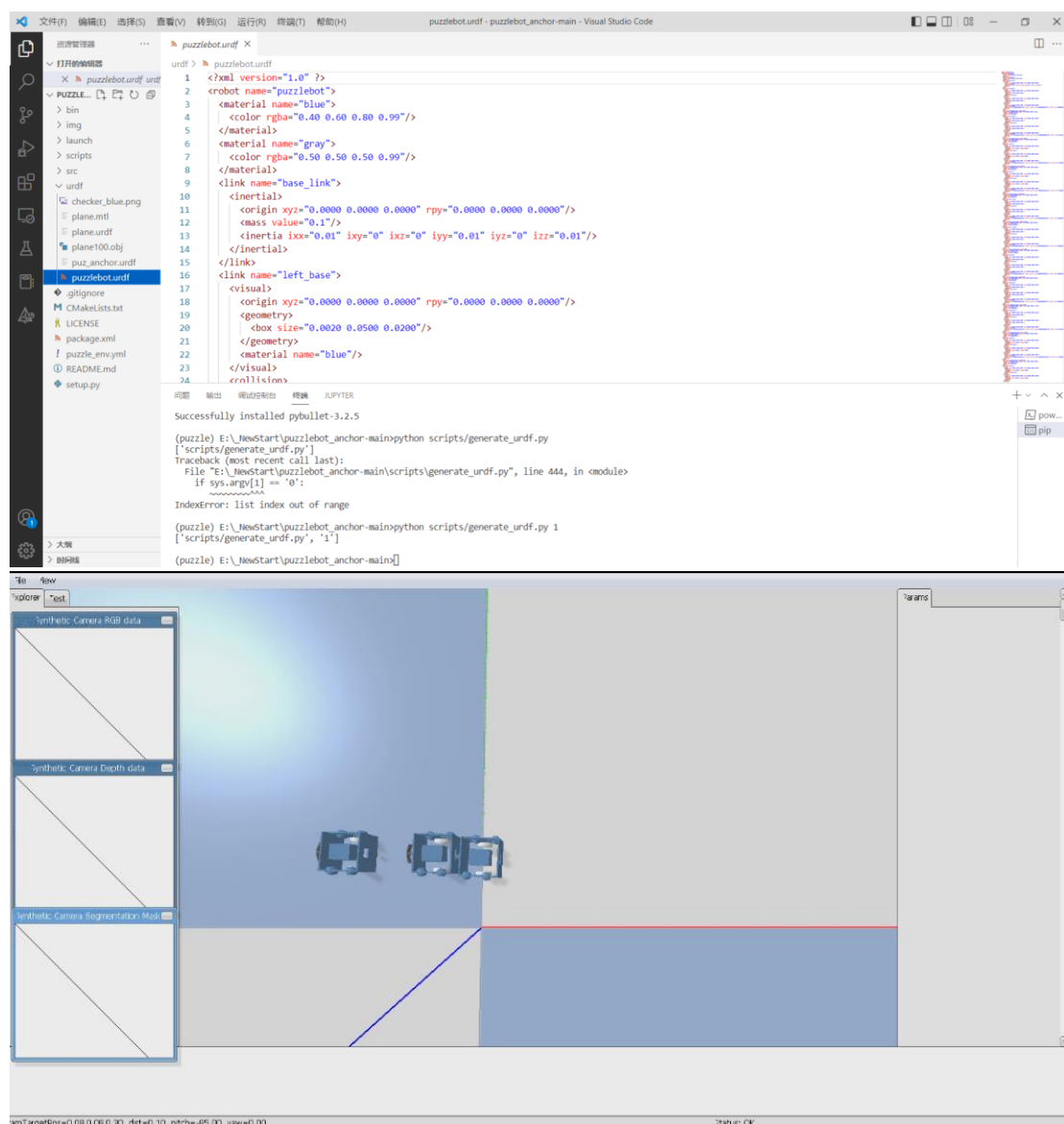
2)、在实际环境的测试中，研究人员集成了一个 10HZ 工作的中央 MPC 规划节点和一个 30HZ 的高频 PID 控制器来操控机器人，可以成功完成拼接和下楼任务。见下图 9。



#### 四、代码复现

幸运的是，作者提供了源代码，在作者的 github 中，提到已经在 Ubuntu3.7 进行了全面测试，在 Macbook 做了部分测试。

我们选择在 win10 系统下进行模拟仿真，也成功得到了测试结果。



我们的复现代码、测试结果和仿真视频已上传至 [https://github.com/AI-Santiago/puzzlebot\\_anchor](https://github.com/AI-Santiago/puzzlebot_anchor)。同时，本报告和 PPT 也已同步上传，并进行了匿名化处理。

## 五、感想总结

在阅读这篇论文的时候，我们首先遇到了很多困难。

比如，因为缺少相关基础，我们在阅读了相关综述和参考文献列表后的部分文章后，才对论文有了初步的认识。又经过反复阅读，才整理清楚自己的理解，并形成了这篇报告。

又比如，为了直观了解实验结果，我们复现了代码，解决诸多 bug 后，终于看到了理想的如论文所述的结果。

幸运的是，我们付出良多，也收获良多。



## 参考文献

- [1]Feczko, J., Manka, M., Krol, P., Giergiel, M., Uhl, T. & Pietrzyk, A. (2015)Review of the modular self reconfigurable robotic systems. In: 2015.10th International workshop on robot motion and control, RoMoCo 2015, Poznan, Poland. New York: IEEE. <https://doi.org/10.1109/RoMoCo.2015.7219733>
- [2] Morten Winkler Jorgensen, Esben Hallundbk Ostergaard, and Henrik Hautop Lund. Modular atron: Modules for a self-reconfigurable robot. In 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)(IEEE Cat. No. 04CH37566), volume 2, pages 2068 - 2073. Ieee, 2004.
- [3] Haruhisa Kurokawa, Kohji Tomita, Akiya Kamimura, Shigeru Kokaji, Takashi Hasuo, and Satoshi Murata. Distributed self-reconfiguration of m-tran iii modular robotic system. The International Journal of Robotics Research, 27(3-4):373 - 386, 2008.
- [4] John W. Romanishin, Kyle Gilpin, Sebastian Claici, and Daniela Rus. 3D M-Blocks: Self-reconfiguring robots capable of locomotion via pivoting in three dimensions. In 2015 IEEE International Conference on Robotics and Automation (ICRA), pages 1925 - 1932, May 2015. doi: 10.1109/ICRA.2015.7139450.
- [5] Tarik Tosun, Jay Davey, Chao Liu, and Mark Yim. Design and characterization of the EP-Face connector. In 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pages 45 - 51, October 2016. doi: 10.1109/IROS.2016.7759033.
- [6] David Saldana, Bruno Gabrich, Guanrui Li, Mark Yim, and Vijay Kumar. Modquad: The flying modular structure that self-assembles in midair. In 2018 IEEE International Conference on Robotics and Automation (ICRA), pages 691 - 698. IEEE, 2018.
- [7] Luis A Mateos, Wei Wang, Banti Gheneti, Fabio Duarte, Carlo Ratti, and Daniela Rus. Autonomous latching system for robotic boats. In 2019 International Conference on Robotics and Automation (ICRA), pages 7933 - 7939. IEEE, 2019.
- [8] Masahiro Shimizu and Akio Ishiguro. An amoeboid modular robot that exhibits real-time adaptive reconfiguration. In 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems, pages 1496 - 1501. IEEE, 2009.
- [9] Sha Yi, Zeynep Temel, and Katia Sycara. Puzzlebots: Physical coupling of robot swarms. In 2021 IEEE International Conference on Robotics and Automation (ICRA), pages 8742 - 8748. IEEE, 2021
- [10] Akshay Thirugnanam, Jun Zeng, and Koushil Sreenath. Duality-based convex optimization for real-time obstacle avoidance between polytopes with control barrier functions. In 2022 American Control Conference (ACC), pages 2239 - 2246. IEEE, 2022.

- [11] Akshay Thirugnanam, Jun Zeng, and Koushil Sreenath. Safety-critical control and planning for obstacle avoidance between polytopes with control barrier functions. In 2022 International Conference on Robotics and Automation (ICRA), pages 286–292. IEEE, 2022.
- [12] Yi S, Sycara K, Temel Z. Reconfigurable Robot Control Using Flexible Coupling Mechanisms. arXiv preprint arXiv:2306.16265. 2023 Jun 28.