

群体机器人自组装的分布式控制方法与实验研究^①

徐 敏^{②*} 魏洪兴^{③**} 蔡颖鹏^{**}

(* 厦门理工学院电子与电气工程系 厦门 361024)

(** 北京航空航天大学机械工程及自动化学院 北京 100191)

摘 要 基于自组装模块化机器人平台 Sambot,提出了一种分布式的自组装控制方法,并用 5 个 Sambot 机器人,成功实现了 4 足构型的自组装实验。用此方法可使每个对接机器人仅依赖红外传感器的局部感知和通信能力,通过漫游、导航、自主对接行为实现与种子机器人的自主对接。这种自组装控制方法可以直接扩展到多个 Sambot 执行,以构建任意构型的集合体机器人。

关键词 自组装,群体机器人,自主对接,分布式控制,基于行为的控制

0 引 言

受社会性昆虫的群体行为启发,近年机器人学术界关于群体机器人的自组装控制研究已形成研究热点。自组装是指在无人参与的情况下,一群智能体部件自动组合成一种新的形态或结构的过程^[1]。在机器人领域,自组装为多机器人协作提供了一种非常实用的行为模式。一群具有相同功能的模块或个体机器人通过相互连接与分离,可以构建具有比单个模块或个体机器人更强的运动、感知和作业能力的集合体机器人^[2]。实现个体机器人不能胜任的任务。

当前在群体机器人研究领域,只有较少的研究平台具有自组装的功能,而且由于设计目标不同,通过自组装建立的多机器人结构并不具备类似自重构机器人的整体运动能力^[3-5]。而在自重构机器人领域,组成自重构机器人的基本模块一般不能自主运动或者只具有非常有限的自主运动能力,而且通常都是把模块排列到预定的位置完成构型的重构^[6-9]。在自组装控制算法研究方面,Chris^[10]、Jonathan^[11]、Hristo^[12]等进行了基于规则的自组装控制算法的研究,但仅进行了仿真分析,没有进行实际平台的实验研究。Grady^[13]等人基于 Swarm-bot 平台,提出了一种称为 SWARMORPH 的分布式自组装控制方法,实现了给定目标构型的自组装控制。本文基于一种新

型的自组装模块化群体机器人平台 Sambot,提出了一种分布式的自组装控制方法,自组装从种子机器人开始,采用有方向的自组装控制,决定对接机器人的连接方向;对接机器人仅依赖自身的红外传感器实现探测、导航与自主对接,从而实现多个 Sambot 机器人自组装控制,构建给定的机器人构型,我们使用 5 个 Sambot 机器人成功地实现了 4 足构型的自组装构建。

1 Sambot 机器人平台

1.1 总体结构

Sambot 机器人是一种结合自重构机器人与自组装机器人优点的群体机器人平台,每个 Sambot 机器人都是一个完全自主的移动机器人,类似群体机器人平台中的个体机器人。多个 Sambot 机器人通过自组装,可以构成一个集合体机器人(robotic structures),如蛇形机器人或 4 足爬行机器人,这种集合体机器人具有类似自重构机器人的重构和整体运动能力。

如图 1 所示,Sambot 由上面的主动对接面板与下面的自主移动本体两部分组成。主动对接面板上有一对主动对接卡扣,能够允许其它 Sambot 在一定偏差范围内从前、后、左、右 4 个被动对接面上的对接卡槽进行自主对接及锁紧。同时,主动对接面板还可以围绕本体的中心轴实现 $\pm 150^\circ$ 范围的转动。

① 863 计划(2009AA043901)资助项目。

② 男,1963 年生,学士,副教授;研究方向:嵌入式技术与智能控制;E-mail: xuminxx26@163.com

③ 通讯作者,E-mail: weihongxing@buaa.edu.cn
(收稿日期:2010-02-08)

当对接时,主动对接面板向前(或向后)旋转 90° ,使主动对接面板与其它 Sambot 的被动对接面进行对接。对接完成后,多个 Sambot 机器人通过对接面上的电气触点实现 CAN 总线通信,用于实现多个 Sambot 对接后的整体运动控制。自主移动本体底部有两个对称的驱动轮,通过差分驱动实现 Sambot 的自主移动。

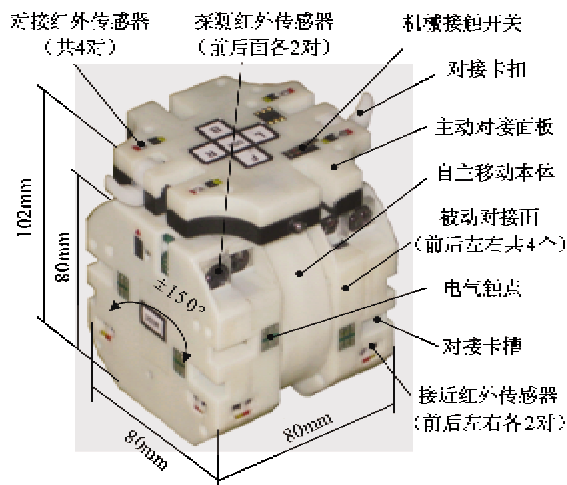


图1 Sambot 机器人的总体结构平台

Sambot 安装了多个红外传感器用于探测障碍及实现对接导引。在本体前后两个对接面上部各装有两对探测红外传感器,通过红外反射探测前后的障碍物,探测距离范围为 $0 \sim 160\text{mm}$,用于实现主动避障。在本体前、后、左、右4个面下部各安装了两对接近红外传感器,通过与其它机器人的对应位置的对接红外传感器对射实现红外通信,对射范围为 $0 \sim 300\text{mm}$ 。该接近红外传感器有两种用途,一是用于实现两个 Sambot 自主对接过程中的红外导引,二是在自组装过程中作为对接方向,打开红外发射,引导对接机器人到达对接位置,实现给定方向的自主对接。在主动对接面板上安装了4对对接红外传感器,在进行对接时,对接机器人的主动对接面板向前(或向后)旋转 90° ,使上面(或下面)两对对接红外传感器分别与被动对接面下方的接近红外传感器一一对应,通过红外通信,引导两个机器人实现自主对接。

1.2 对接机构

Sambot 的自主对接机构应具有远距离对接引导和允许有一定的对接偏差,同时,还应具有连接分离快速可靠、结构强度高的特点。

Sambot 选择了一种采用蜗轮蜗杆驱动的对接卡

扣与另一个机器人的对接卡槽锁紧实现对接。如图2所示,对接卡扣内部的机械传动路线如下:电机的输出端经过一对齿轮减速后传递给转轴,转轴两端分别连接两个蜗杆,通过蜗轮蜗杆驱动左、右两个卡扣同步旋转,实现对接和分离。

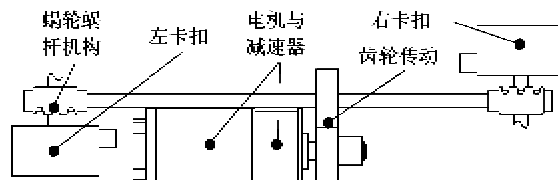


图2 对接卡扣的传动示意图

为保证连接的可靠性和灵活对接,对接卡槽采用非对称的错位布置,可以允许其它机器人从前、后、左、右4个面同时进行对接,而不产生干涉。如图3所示,在 Sambot 前、后、左、右4个面均设计有与卡扣对应的卡槽。当机器人 A 和机器人 B 已经通过卡扣1对接完毕(加粗线条表示模块 A 和 B 的接触面),此时机器人 C 要与机器人 B 通过卡扣2进行对接,为了避免卡扣2在旋转进入 B 的卡槽时遇到阻碍,在 A 的主动对接面板对应位置也开有卡槽,从而实现了多个机器人可以与从前、后、左、右4个方向与其同时进行对接。

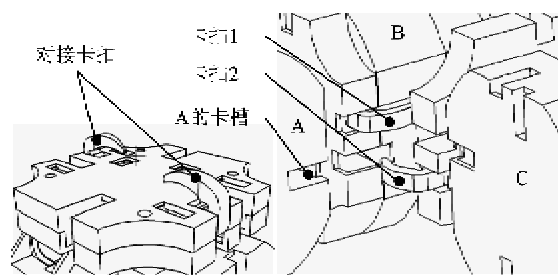


图3 对接卡槽的分布

2 多 Sambot 自组装的形态分析

多个 Sambot 机器人通过自组装,可以构成链式自重构机器人。我们利用 3D 软件建立了多个 Sambot 机器人组成的构型模型,以检验模块之间连接的可能性以及是否存在碰撞干涉等,为实验条件下构型的建立提供基础。按照机器人关节的分布关系,我们将多个 Sambot 组成的机器人构型分为线性构型和多足构型两种。如图4所示,线性构型包括直线与环形两种,环形构型由多个 Sambot 通过前后串

行连接组成。事实上,直线性构型是一种平面 2D 构型,而环形构型可视为一种 3D 构型,即多个 Sambot 模块首先组成直线构型,然后直线构型的首、尾 Sambot 模块在其它模块带动下,进行面对面的旋转,实现首尾 Sambot 模块的相互对接,形成一个环形构型。

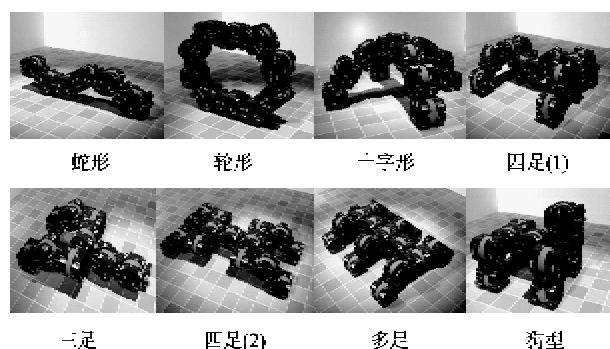


图 4 多 Sambot 的构型仿真图

此外,更多的 Sambot 还可以构成三足、四足、多足等多种机器人构型。其中狗型构型为 3D 构型,需要在平面构型的基础上,通过 Sambot 模块的搬运和举升实现,而其它构型为平面构型,可以通过多个 Sambot 自组装直接构建。

同时,我们在微软机器人工作室 (Microsoft Robotics Studio, MSRS) 这一平台上建立了 Sambot 的仿真环境,可以实现多个 Sambot 的运动和控制系统仿真。MSRS 是微软提供的机器人开发平台,它以受欢迎的 .NET Framework 为基础,为机器人开发人员提供了一个标准且一致的方法来开发各种机器人平台的应用程序。图 5 上部两个图分别是 Sambot 在 MSRS 环境下建立的机构结构线框图与实体图,这里

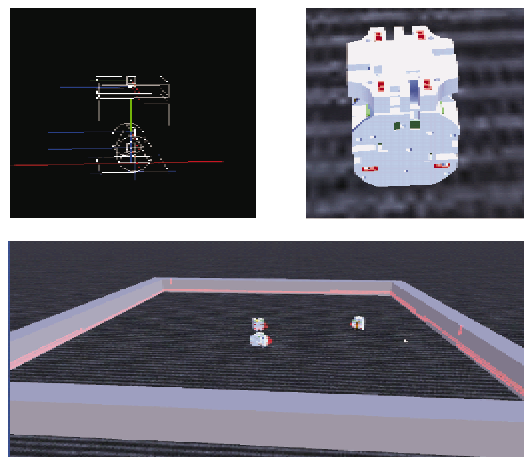


图 5 基于 MSRS 平台建立的 Sambot 仿真环境

Sambot 被简化成由双轮、主动对接面板与对接卡扣等部件组成的结构。图 5 下部的图给出了多个 Sambot 机器人在 MSRS 环境中自主运动的仿真结果。

由于基于 MSRS 建立的 Sambot 仿真环境能够执行与 Sambot 实体相同的控制算法,从而它为多 Sambot 自组装控制算法的验证提供了有利的条件。

3 分布式自组装控制方法

3.1 自组装控制原理

构型给定的群体机器人的自组装任务,是从一个称为种子 (Seed) 的机器人开始的。Seed 机器人的选择没有特殊的要求,可以从分布的多个 Sambot 中随机选择一个。Seed 保存目标构型及组成构型后的整体运动序列。环境中的其它机器人 (称为 Docking-SA) 根据构型指令不断地与 Seed 对接。其中,种子机器人的主要功能包括:(1)决定构型指令;(2)现有构型与目标构型比较;(3)控制构型增长。

根据 Sambot 的结构特点,Sambot 一共可以有 4 种构型指令: $Configuration_Inst(i)$, $i = (F, B, L, R)$ (分别表示 Sambot 的前、后、左、右 4 个对接面)。执行每种构型指令的方法是打开对应面的两个接近红外传感器,我们称这个动作为打开对接方向 (称为 Docking-Direction),等待 Docking-SA 机器人与自己进行对接。如执行 $Configuration_Inst(F)$ 指令,表示打开 Seed 前对接面下方的左、右两个接近红外传感器。

Docking-SA 与 Seed 进行自主对接的控制策略如下:在自组装的初始状态,种子机器人位于平台中心,保持静止不动。并且根据构型指令打开对应被动对接面的对接方向 Docking-Direction,等待 Docking-SA 与自己对接。由于 Sambot 仅具有有限的红外感知能力和姿态感知能力,Docking-SA 根据自身四周的探测红外传感器与接近红外传感器提供的信息寻找 Seed 机器人,然后围绕 Seed 做逆时针导航运动,直到到达 Seed 的对接方向 Docking-Direction,实现与 Seed 的自主对接。

考虑到多机器人之间的红外干扰,这里假设对接是串行的,每次只能有一个机器人执行对接任务。即每次在实验平台上只加入一个 Sambot 作为 Docking-SA,直到该 Docking-SA 与 Seed 成功对接后,再加入新的 Docking-SA。

3.2 Sambot 的基本运动模式

Sambot 本身仅具有有限范围的感知能力和和局部通信能力,有以下约束:

(1)每个机器人没有全局坐标信息,不能进行全局导航控制。

(2)每个机器人仅具有局部感知能力。这里 Sambot 可以通过探测红外传感器探测目标,并用接近红外传感器进行导航。

(3)每个机器人只有局部通信能力(Sambot 具有基于 ZigBee 的全局无线通信,但我们没有使用)。

Docking_SA 采用了基于行为(Behavior-based)的控制方法,即控制器由一系列的行为组成,每个行为都用于实现某个特定的目标。而每种行为,又是一系列运动模式(Motor Schema)的组合。Sambot 建立了下列基本运动模式:

(1)Rotate(ω , Φ):以角速度 ω 自转角度 Φ , $\Phi \in (-180^\circ, 180^\circ)$ 。

(2)SetMoveSpeed (V_l , V_r):设置左右轮的速度, $-4000 \leq (V_l, V_r) \leq 4000$ 。

(3)Forward(v , L):以速度 v 向前移动距离 L , $L \geq 0$ 。

(4)Backward(v , L):以速度 v 向后移动距离 L , $L \geq 0$ 。

(5)ARC(v , R , Ψ):以速度 v , 以当前点为起始点,以 R 为半径,移动角度为 Ψ 的圆弧轨迹, $\Psi \in (-180^\circ, 180^\circ)$ 。

(6)DockBoard_Rotate(ω , Φ):主动对接面板以角速度 ω 自转角度 Φ , $\Phi \in (-180^\circ, 180^\circ)$ 。此动作可用于自组装完成后的集合体机器人整体运动控制。

(7)Lock():对接卡扣锁紧。

(8)Unlock():对接卡扣打开。

(9)Align():用于对接过程中的对齐调整。在保证主动对接面板下方的两个接近红外传感器收到信息的前提下,低速向前方移动。

Sambot 主要根据自身的红外传感器检测周围的环境,决定机器人的行为。为方便计算,将 Sambot 的传感器定义如下:

(1)探测红外传感器(Detecting IR):前后面上方各两对,用于检测前方的 Seed 和其它机器人。

(2)接近红外传感器(Approaching IR):前、后、左、右面下方各两对,用于引导机器人寻找 Docking_Direction。

(3)对接红外传感器(Docking IR):主动对接面板上、下各两对,用于在对接过程中检测 Seed 机器人的 Docking_Direction,引导实现自主对接。

(4)机械接触开关:当两个机器人接触时,位于主动对接面板上的机械接触开关被压下,触发对接卡扣

执行锁紧动作。

(5)陀螺仪传感器:给出机器人当前姿态, $\beta \in (0 \sim 360^\circ)$ 。

3.3 Sambot 的控制算法设计

Docking_SA 采用状态机控制,包括三个主要状态:漫游,导航和自主对接。

(1)漫游:每个 Docking_SA 加入实验平台后,执行漫游行为。该行为是打开前方避障红外传感器,按随机轨迹进行漫游。这里实验台的 4 个边缘为 40mm (高于接近红外传感器,而低于探测红外传感器),可以用机器人下面的接近红外传感器检测工作台边缘。当漫游到工作台边缘时,接近红外传感器给出信息,机器人执行反向运动。当前方探测红外传感器有输入信息后,表示前方发现 Seed 机器人,Docking_SA 进入导航行为。漫游行为的具体控制算法如下:

漫游控制算法 Wander()

```

1. While Seed is not found
2. {
3.   Initialization: Open detecting and approaching IR
4.   Forward until detect a side
5.   If Docking_SA approach one side
6.   Then
7.     Turn around to be parallel with the side
8.     Move along the side to one corner of the platform
9.     Turn around to the center of the platform
10.    Move to the center of the platform
11.    Turn a circle to search for Seed
12.    If it fail to find Seed, break
13.   Else DSA must approach one corner of the platform
14.   Then
15.     Turn around to the center of the platform
16.     Move to the center of the platform
17.     Turn a circle to search for Seed
18.     If it is failed to find Seed, break
19. }
```

(2)导航:导航的目的是通过 Sambot 有限的红外感知能力,将 Docking_SA 导航到 Seed 的 Docking_Direction 位置。实现方法是使 Docking_SA 围绕 Seed 作逆时针运动,Docking_SA 每运动一个体长(80mm),向右自转探测 Seed,保证 Seed 在 Docking 的红外探测传感器检测范围之内。在此过程中,Docking 始终打开与 Seed 对应侧面的接近红外传感器,直到接收到信息,表示已到达 Seed 的 Connect_Direction 位置,即进入对接行为。导航行为的具体算法如下:

导航控制算法 Navigation()

```

1. While approaching IR don't detect Docking_Directions
2. {
3.   If left detecting IR sensor doesn't detect Seed
4.   Then
5.     Turn left to detect Seed
6.     Turn right of 30°;
7.     Forward 80mm
8.   Else left detecting IR sensor detect Seed
9.     Turn right until left detecting IR doesn't detect Seed
10.    Turn right of 30°
11.    Forward 80mm
12. }
    
```

(3)自主对接:进入对接行为后,Docking_SA 转向 Seed,打开主动对接面板上对接红外传感器,进行自主对接。Docking_SA 在保持主动对接面板上左、右两个对接红外传感器同时接收到红外信息的前提下向前移动,直到机械接触开关被压下,表示与 Seed 接触,此触发对接卡扣执行锁紧动作,从而完成自主对接。自主对接行为的控制算法如下:

自主对接控制算法 Docking ()

```

1. Docking ()
2. {
3.   Adjust the location by moving in the current direction
4.   to make sure both two approaching IR sensors on the
5.   left side detect IR signal from Docking_Direction
6.   Turn 90° to towards Docking_Direction
7.   Forward and adjust moving direct by two approaching
8.   IR sensors on the front side;
9.   If succeed to touch with Docking_Direction
10.  Then
11.    Lock();
12.  Else
13.    Wander();
14. }
    
```

当两个机器人连接并锁紧后,Docking_SA 与 Seed 的电气触点接触,双方发送握手信号确认完成 CAN 总线通信连接。之后 Docking_SA 向 Seed 发送自己的基本信息。Seed 可以通过 I/O 信号判断是 Docking_SA 与自己的哪一个对接面进行连接,然后更新构型信息。

这样,每个 Docking_SA 都执行完全相同的控制程序,而 Seed 决定整个自组装过程。每次 Docking_SA 与 Seed 对接后,Seed 将现有构型信息与目标构型进行比较,决定下次对接方向 Docking_Direction,直至达到目标构型。

4 实验

我们演示了 5 个 Sambot 通过自组装达到目标构型的实验,实验在一个 600mm×600mm 的实验平台上进行。首先将 Seed 放置在实验平台中心位置。每次在平台中增加一个 Docking_SA 实现自组装。

除了每次自组装过程中的 Docking_SA 是手工加入到实验平台中的,整个自组装过程都是自主进行的。其中 Seed 保存了目标构型的信息,而 Docking_SA 在自组装过程中,没有目标构型信息,也没有任何全局坐标信息,仅通过内部的状态机控制,执行漫游、导航、对接、锁紧等行为,直到找到 Seed 的对接方向 Docking_Direction,执行自主对接。

我们进行了十字交叉四足构型的自组装实验,自组装过程序列如图 6 所示。其中图中的箭头指出了一下次的对接方向 Docking_Direction。对于四足构型,Docking_SA 可以选择从前、后、左、右 4 个方向与 Seed 进行对接。包括手工加入新 Docking_SA 的时间,整个四足构型的自组装共耗时 4 分 50 秒,我们分别统计每个 Docking_SA 漫游、导航和自主对接行为消耗时间,如表 1 所示。

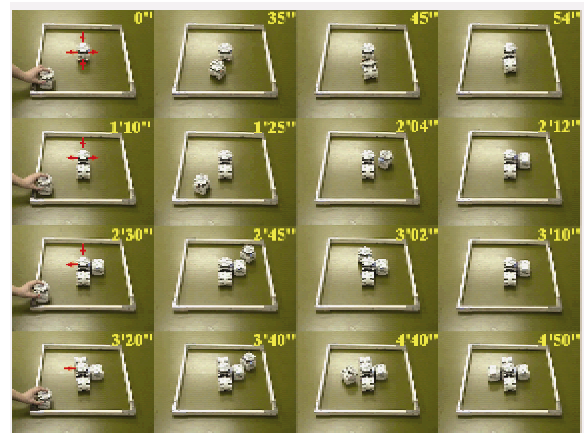


图 6 四足构型的自组装序列

表 1 自组装过程中每个 Docking_SA 漫游,导航和自主对接行为消耗时间

Docking_SA 序号	漫游耗时 (s)	导航耗时 (s)	自主对接 耗时(s)
1	35	10	9
2	15	39	8
3	15	17	8
4	20	60	10

可以发现,自主对接行为消耗时间较少,而且每个 Docking_SA 所消耗的时间基本接近,表明只要

Docking_SA 到达 Docking_Direction 位置,自主对接的成功率较高。每个 Docking_SA 漫游时间具有一定随机性,但由于实验平台为 600mm×600mm,故所消耗时间相对较少,即表示 Docking_SA 可以很快找到 Seed,而导航时间由于 Docking_SA 进入导航行为时的位置距对接方向 Dockingt_Direction 的位置距离不同,有较大差异。

由于我们的自组装控制方法运用时 Docking_SA 执行基于行为的控制器,与目标构型的模块数目和构型的复杂程度无关,因此无需修改即可应用于更多数量机器人的自组装控制。

5 结 论

本研究基于 Sambot 平台,提出了一种用于群体机器人的分布式自组装控制方法,并用此方法,成功地进行了 5 个 Sambot 机器人的 4 足构型的自组装实验。研究表明:(1)通过采用一种新的卡扣卡槽对接机构,Sambot 可以从前、后、左、右 4 个方面同时进行对接,而不产生干涉;(2)多个 Sambot 可以构建直线与多足等多种机器人构型,基于 MSRS 建立的仿真环境能够实现与 Sambot 实体相同的控制算法。

下一步工作拟采用智能算法,实现多个机器人的并行自组装控制,同时开展基于中央模式发生器(CPG)的集合体机器人构型运动控制方法研究。

参考文献

- [1] Whitesides GM, Grzybowski B. Self-assembly at all scales. *Science*, 2002, 295(5564): 2418-2421
- [2] Groß R, Dorigo M. Self-assembly at the macroscopic scale. *Proceedings of the IEEE*, 2008, 96(9): 1490-1508
- [3] Groß R, Dorigo M. Evolution of solitary and group transport behaviors for autonomous robots capable of self-assembling. *Adaptive Behavior*, 2008, 16(5): 285-305
- [4] Groß R, Bonani M, Mondada F, et al. Autonomous self-assembly in swarm-bots. *IEEE Transactions on Robotics*, 2006, 22(6): 1115-1130
- [5] Tuci E, Groß R, Vitot T, et al. Cooperation through self-assembly in multi-robot. *ACM Transactions on Autonomous and Adaptive Systems*, 2006, 1(2): 115-150
- [6] Shen W M, Moll B, Rubenstein M, et al. Multifunctional behaviors of reconfigurable superbot robots. In: *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, New York, USA, 2007. 2548-2548
- [7] Rubenstein M, Payne K, Will P, et al. Docking among independent and autonomous CONRO self-reconfigurable robots. In: *2004 IEEE International Conference on Robotics and Automation*, New Orleans, USA, 2004. 2877-2882
- [8] Yim M, Duff D G, Roufas K D. A modular reconfigurable robot. In: *Proceedings of the 2000 IEEE International Conference on Robotics and Automation*, San Francisco, USA, 2000. 514-520
- [9] Yoshida E, Murata S, Kamimura A, et al. A self-reconfigurable modular robot: reconfiguration planning and experiments. *The International Journal of Robotics Research*, 2002, 21(10-11): 903-915
- [10] Jones C, Mataric M J. From local to global behavior in intelligent self-assembly. In: *Proceedings of the 2003 IEEE International Conference on Robotics and Automation*, Taiwan, China, 2003. 721-726
- [11] Kelly J, Zhang H. Combinatorial optimization of sensing for rule-based planar distributed assembly. In: *Proceedings of the 2006 IEEE/RSJ International Conference Intelligent Robots and Systems*, Beijing, China, 2006. 3728-3734
- [12] Bojinov H, Casal A, Hogg T. Multiagent control of self-reconfigurable robots. *Artificial Intelligence*, 2002, 142(2): 99-120
- [13] O'Grady R., Christensen A L, Dorigo M. SWARMORPH: multirobot morphogenesis using directional self-assembly. *IEEE Transactions on Robotics*, 2009, 25(3): 738-743

The research on distributed control of self-assembly in swarm robots

Xu Min*, Wei Hongxing**, Cai Yingpeng**

(* Electronic and Electrical Engineering Department, Xiamen University of Technology, Xiamen 361024)

(** School of Mechanical Engineering and Automation, Beihang University, Beijing 100191)

Abstract

Based on the Sambot platform, the authors presented a distributed control method for the self-assembly in swarm robots, and with the method, implemented the experiment on the self-assembly for the quadruped configuration consisting of 5 Sambots. When using the control method, the autonomous docking of each robot with the seed robot just depends on the local perception and the communication ability of the infrared sensors, and the robots' behaviors of wandering, navigation and autonomous docking. The experimental results show that the control method for self-assembly can be used to realize the autonomous construction of robotic structures consisting of multiple Sambot.

Key words: self-assembly, swarm robot, autonomous docking, distributed control, behavior-based control