

文章编号: 1002-0446(2007)02-0097-09

基于宏行为的侦察机器人事务执行机制研究^{*}

唐鸿儒^{1,2}, 宋爱国¹, 章小兵¹

(1 东南大学仪器科学与工程系, 江苏 南京 210096 2 扬州大学信息工程学院, 江苏 扬州 225009)

摘要: 为了便于操作员在侦察现场进行事务规划, 提出了基于宏行为的事务描述方法和事务执行机制. 按照机器人对环境变化的响应方式的不同, 将机器人的行为分成基本行为、组合行为和宏行为, 给出了描述这些行为的方法和一个宏行为的设计实例. 引入了基于 RS 模型的行为合成运算符合成机器人行为的方法, 从而可以形式化地构建侦察机器人的事务描述. 实现了能够解释和执行事务描述的、基于宏行为的事务执行器. 仿真和实验结果表明提出的任务描述方法可行, 事务执行器能够可靠地完成事务描述所规定的任务.

关键词: 半自主侦察机器人; 宏行为; 事务描述; 事务执行机制

中图分类号: TP24 **文献标识码:** B

Mission Execution Mechanism of Reconnaissance Robot Based on Macro Behavior

TANG Hong-ru^{1,2}, SONG Aiguo¹, ZHANG Xiaobing¹

(1 Department of Instrument Science and Engineering, Southeast University, Nanjing 210096 China;

2 College of Information Engineering, Yangzhou University, Yangzhou 225009 China)

Abstract In order for the operator to plan reconnaissance mission conveniently at field, the mission description method and mission execution mechanism of reconnaissance robot based on macro behavior is brought forward. The behaviors of reconnaissance robot are classified into elementary behaviors, combinational behaviors and macro behaviors according to the ways of responding to the environment changes during task execution. The description methods for these behaviors are presented and a design example of macro behavior is shown. The formal mission description method is introduced which uses behavior composition operators based on RS (Robot Schema) model to assemble elementary behaviors, combinational behaviors and macro behaviors, and a mission executor based on macro behaviors which can interpret and execute the mission specification is realized. Simulation and experimental results show the validity of the presented mission specification method and the mission executor can perform the specified tasks reliably.

Keywords semi-autonomous reconnaissance robot; macro behavior; mission description; mission execution mechanism

1 引言 (Introduction)

侦察机器人通常在未知或者非结构化的环境下运动和操作, 这就需要机器人能够快速响应无法预知或不可预见的紧急事件, 因而, 常常将基于行为的体系结构作为侦察机器人体系结构的基础. 1986 年, Brooks 提出了包容式体系结构^[1], 它是第一个也是最著名的基于行为的体系结构的例子. 目前, 已经实现了许多基于不同体系结构的、具有一定反应能力的自主机器人^[2,3], 同时开发出多种行为协调机

制, 如协作行为选择机制^[4,5]、竞争行为选择机制^[6]、时间顺序协调方法^[7]等等. 关于基于行为的移动机器人的设计方法也有不少研究, 如 Kosecka 等人提出了一种任务描述语言 (Task Description Language TDL), 用它可以将特定的任务描述为网络形式^[8]. Scheutz 和 Andronache 提出用一种通用的体系结构实现动态行为选择策略, 从而以统一的方式集成现有的行为选择机制^[9]. Colombetti 等人提出了行为工程的概念, 其目标是提供自主机器人的开发方法和工

^{*} 基金项目: 国家自然科学基金资助项目 (60475034); 霍英东教育基金资助项目 (91055).

收稿日期: 2006-04-06

具^[10]。

虽然自主系统和人工智能的研究已经有多年,但是研制的机器人在未知的环境中完成复杂任务时仍然遇到许多困难,导致要使得机器人自主能力越高,就要求任务的专门化程度越高,以及需要了解更多的环境细节^[11]。有些情况下,虽然机器人具有复杂的自主行为,但仍然不知道在非结构化的环境中如何自主地完成简单的工作,因此复杂多样的应用环境需要高度柔性的机器人系统。但是,目前的机器人一般设计用来完成特定的工作,他们缺乏在非结构化和变化的环境中生存的能力,这时就需要人与机器人协作。因此,研究具有局部自主能力的、能够通过人机交互进行遥操作的半自主侦察机器人是重要的研究方向。半自主侦察机器人就是具有局部自主环境建模、自主检障和避障、局部自主导航等能力的移动机器人,它能够自主地完成操作员规划好的任务,而复杂的环境分析、任务规划、全局路径选择等工作则交由操作员完成,通过操作员与机器人的协同来完成指定的任务。

移动机器人的事务规划是伴随着移动机器人的发展而发展的, Lyons 于 1989 年提出了一个计算模型 RS(Robot Schema)^[12],接着又于 1993 年提出基于 RS 模型的行为合成方法,用于基于行为的机器人事务规划^[13]。MIT 的 Maes 于 1989 年提出传播活动网络 SAN 的概念^[14], Bagchi 等人将 SAN 用于不确定环境中的任务规划和动态的行为选择^[15]。卡内基—梅隆大学机器人研究所开发的事务描述语言 SAUSAGES 将机器人事务描述为一系列操作状态和状态变迁^[16]。乔治亚技术学院移动机器人实验室开发的配置描述语言 CDL,采用递归合成的方法描述机器人事务,实现了事务描述与体系结构、机器人相互独立^[17]。一些成功的、具有机器人事务规划功能的体系结构和软件系统也各具特色,获得了很好的应用效果,如 JPL 的 WITS^[18-19]、乔治亚技术学院的 Aurora^[20]和 MissionLab^[17]、Murphy 和 Rogers 实现的 TeleVia-SFX^[21]、Fong 等人开发的 PdaDrive^[22]等等,其中最具特色的要算 WITS 和 MissionLab 了, WITS 允许科学家坐在自己的办公室内利用火星车传回的图像选择科学研究目标和针对这一目标的研究活动,规划火星车的移动路径。MissionLab 采用了可视化的事务配置编辑器支持事务配置的构建,已经应用到多种机器人上,它允许不熟悉编程语言的用户利用基本行为进行行为组和机器人事务的图形化构造。尤其是 DARPA 发起的 MARS(Mobile Auto-

nomous Robot Software) 2020 项目,研究无人侦察机(UAV)和无人地面车辆(UGV)的机器人队列监督和控制^[23],这些机器人队列将依靠通信来协调, MARS 2020 将实现事务描述、事务规划、行为协调和事务执行功能。MARS 2020 是美国国防部提出的未来作战系统 FCS 的组成部分。

可以看出,国外移动机器人的研究和应用已经具有很好的理论基础,可以在多种环境中工作的机器人已经进入实用阶段,许多机器人监控和事务规划系统逐步成熟。我国在反恐、防爆机器人研究方面也取得了一些初步成果^[24-26],上海交通大学等单位研制了国内首台 ZXPJ01 型消防机器人,北京航空航天大学开发了 RT3-EOD、RAPTOR 排爆机器人,中科院沈阳自动化所研制出“卫富”和“灵蜥”危险作业机器人,哈尔滨工业大学面向未来战场需求和反恐作战需要,研制了模块化、多功能的地面无人作战平台,中科院沈阳自动化所研制了轮、腿和履带复合型自主式移动机器人 Climber^[27],等等。这些机器人在行走机构、模块化、机械手等方面各有特色。

由于侦察机器人所处的环境多种多样,现场可能是明亮的或者昏暗的,潮湿的或者干燥的,可能具有强电磁干扰,空气中可能充满粉尘,操作员可能蹲在碎石堆上、穿着厚厚的防护服或戴着大手套操作机器人,这就要求侦察机器人的事务规划接口显示直观,事务规划命令输入简单方便,并具有一定的自动检查错误和容错能力。为此,提出了基于宏行为的事务执行机制以便操作员在更高的抽象层次上规划机器人事务、减轻工作强度、提高规划效率。提出了宏行为的概念和宏行为的表示方法,并提出了基于 RS 模型和宏行为的事务描述方法,给出了基于宏行为的事务执行器的实现方法。仿真和实验结果表明,利用宏行为描述机器人事务方便快捷,基于宏行为的事务执行器能够成功地完成规划的事务。

2 行为和行为协调机制 (Behaviors and behavior coordination mechanism)

自从 Brooks 引入行为控制以来,产生了许多行为控制方法,一般认为行为控制中的行为比较简单,且可以并发执行。复杂任务可以通过组合若干个简单行为产生复杂的紧急响应来实现。组合行为是通过一定形式的仲裁或者融合机制来解决多个行为试图同时控制同一个执行器的冲突问题。由于行为是面向任务的,行为设计者可以利用任务结构的特点,来简化系统的设计和行为的实现。这里,我们按照行

为对外部事件的反应方式将行为分为基本行为、组合行为和宏行为三类,下面分别叙述.

2 1 基本行为

基本行为 EB (Elementary Behavior) 是直接 与传感器和执行机构连接的、具有特定用途的功能模块. 基本行为可以用端口自动机来描述^[28], 它包括一个输入接口表、一个输出接口表、一组内部参数和一个行为体. 行为体是一个循环执行的程序, 程序指令同步地从输入端口读数据, 按照一定的算法进行运算, 然后向输出端口写数据, 也可以存取内部变量作为输入参数, 基本行为可以采用下面的形式化定义:

```

Elementary Behavior: = [ behavior name ⟨Name⟩,
  Input-Port-List: ( ⟨IpList⟩),
  Output-Port-List: ( ⟨OpList⟩),
  Parameters-List: ( ⟨ParamList⟩),
  Behavior: ( ⟨BehaviorBody⟩) ]

```

其中: *Name* 是行为名; *IpList* 和 *OpList* 分别是用 ⟨Port-name⟩: ⟨Porttype⟩ 定义端口的输入和输出端口表; *ParamList* 是用 ⟨Param name⟩: ⟨Param type⟩ 定义内部变量的变量表; *BehaviorBody* 是行为描述体. 上述参数除行为名外, 其它用“()”括起来的部分若不需要可以省略.

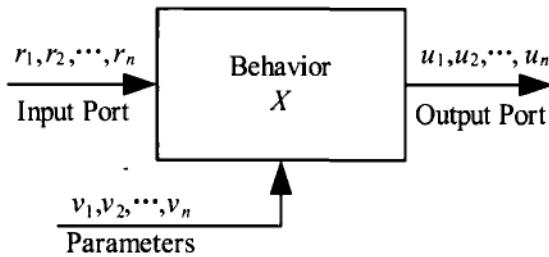


图 1 基本行为符号

Fig 1 Symbols for elementary behavior

基本行为也可以用图 1 所示的符号表示. 基本行为就像半自主侦察机器人功能的构件, 数量越多, 功能越强. 移动侦察机器人常用的行为可能包括: 检障和避障行为、墙壁或者障碍物跟踪行为、点到点运动行为、目标寻找行为和跟踪行为等等. 例如, 如图 2 所示的墙壁跟踪模块 (EB_MoveAlongWall) 用来控制机器人沿着墙壁向前移动, 且与墙壁保持适当的距离, 它利用侧面的距离传感器感知墙壁的距离 d_i 作为输入端口参数, 然后计算机器人的方向与墙壁的夹角 θ 再根据 θ 计算机器人的转动速度 ω 作为输出端口参数. EB_MoveAlongWall 的内部参数表包括 *side*

和 *standoff* 两个参数, *side* 指示机器人跟踪左侧或者右侧的墙壁, *standoff* 则表示机器人必须保持与墙壁的最小距离, 修改 *side* 和 *standoff* 可以获得更好的行为适应性和快速响应性.

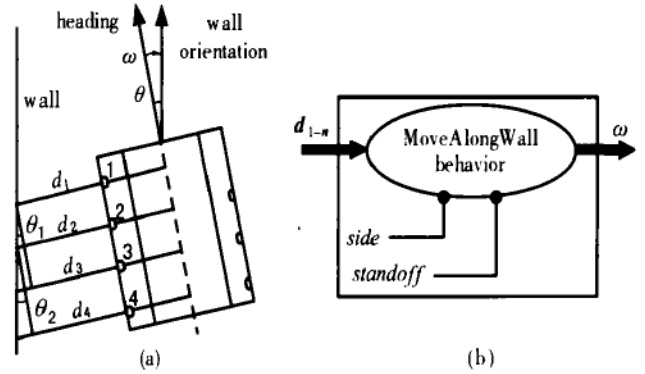


图 2 基本行为 EB_MoveAlongWall 的示意图

Fig 2 Schematic diagram of EB_MoveAlongWall

下面章节中用到的其它基本行为如下: EB_Swerve 是前方遇障碍物时的转向行为, EB_GotoPoint 是控制机器人以一定的速度朝着指定目标点方向运动的行为. EB_OutsideCorner 是引导机器人围绕外墙角运动的行为. EB_Recover 是控制机器人在堵转情况下自动解脱的行为. EB_Forward 是控制机器人以恒定的速度向前运动的行为.

2 2 组合行为

组合行为 CB (Combinational Behavior) 由若干个基本行为和一个行为协调器组成, 由行为协调器协调各个基本行为的工作和输出控制量, 组合行为一般设计用来完成一些简单任务. 可以用 (1) 式表示组合行为:

$$CB(EB_1, EB_2, \dots, EB_n, \Sigma) \tag{1}$$

其中: EB_1, EB_2, \dots, EB_n 为预定义的基本行为, Σ 为行为协调器. 组合行为也可以用图 3 所示的示意图表示, 各个基本行为的输出为 O_1, O_2, \dots, O_n , 通过行为协调器仲裁或者融合后形成统一的输出控制量 O , 写成 $O = \Sigma(O_1, O_2, \dots, O_n)$. 大多数系统中, 行为协调器利用外部传感器数据或者表示机器人航位自推测位置的内部数据作为输入“事件”, 这些事件导致了行为输出的融合或者行为之间的离散切换. 文 [4] 中 Pirjanian 将行为协调机制分成仲裁机制和命令融合机制两类. 其中, 仲裁机制又包括基于优先级的仲裁、基于状态的仲裁和赢者通吃机制, 命令融合机制则包括投票法、叠加法、模糊推理法和多目标法.

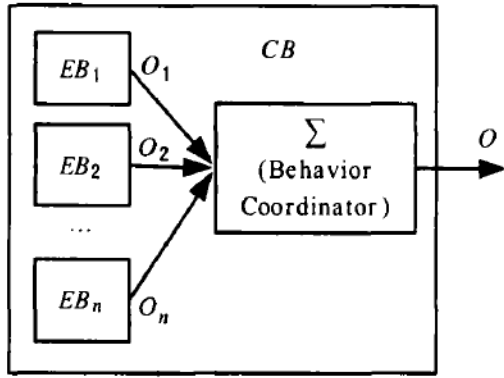


图 3 组合行为示意图

Fig 3 Schematic diagram of combinational behavior

图 4给出了能够引导机器人在有障碍环境中沿着左侧墙壁运动的组合行为 $CB_MoveAlongLeftObstacle$ 的示意图,包括了 5个基本行为,它们是堵转自恢复行为 $EB_Recover$ 、前方有障碍物右转行为 $EB_Swerve(R)$ 、左侧遇外墙角左转行为 $EB_OutsideCorner(L)$ 、左侧墙壁跟踪行为 $EB_MoveAlongWall(L,standoff)$ 和恒速向前移动行为 $EB_Forward$ 每个行为的输出是输出运动速度 (v)或者转动速度 (ω)命令,行为协调器采用基于优先级的仲裁机制.图 4中 5个基本行为的优先级从上到下依次降低,其中 $EB_Recover$ 具有最高优先级, $EB_Forward$ 具有最低优先级.具有高优先级的基本行为输出抑制低优先级的行为输出.在这个组合行为中所有基本行为同步执行,但只有那些需要控制机器人运动和转动的基本行为才会输出速度或者转动速度命令.仿真和实验证明,利用这个组合行为,机器人能够成功完成某些指定的任务,如图 5(a)为机器人在这个行为的控制下实现搜索房间的仿真移动轨迹.图 5(b)为机器人在这个行为的控制下实现沿房屋外围侦察的仿真运动轨迹.

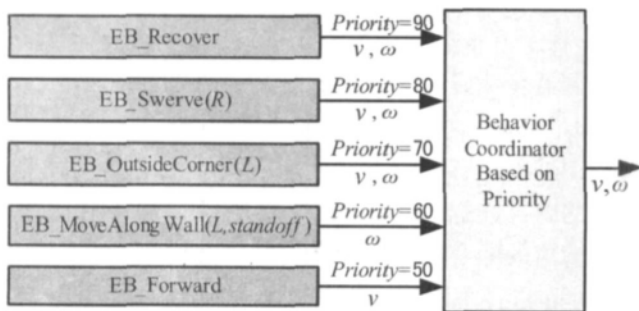


图 4 组合行为 $CB_MoveAlongLeftObstacle$ 示意框图

Fig 4 Schematic diagram of $CB_MoveAlongLeftObstacle$

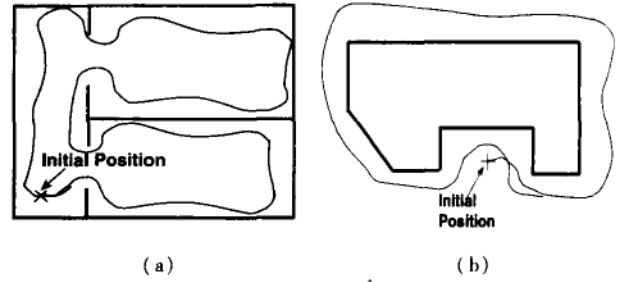


图 5 $CB_MoveAlongLeftObstacle$ 控制下的运动轨迹

Fig 5 Movement trajectory under the control of $CB_MoveAlongLeftObstacle$

2.3 宏行为

2.3.1 宏行为表示

宏行为 $MB(Macro\ Behavior)$ 是为了满足某个特定任务,利用组合行为构建的具有推理能力的高级智能行为.宏行为由若干个组合行为和一个推理机制组成,推理机制能够根据任务执行的不同阶段决定各个组合行为模块的激活或者终止.

例如,面向任务目标的宏行为可以利用有限状态自动机(FSA)作为宏行为的推理机制,它根据机器人当前所处的环境状况,通过FSA来选择对应的组合行为投入工作来响应环境的变化.FSA提供了形式化的语法,使得行为的表示和推理变得直观和容易实现.在FSA中,每个状态对应一个组合行为,这样,当机器人处于某个状态时,就创建并执行与此状态对应的组合行为实例.

基于FSA的宏行为可以用(2)式表示为:

$$MB(\Gamma, \Omega) \tag{2}$$

其中, Γ 为一个用五元组表示的 FSA:

$$\Gamma = (Q, \Sigma, \delta, q_0, F) \tag{3}$$

这里: Q 是状态集, $Q = \{q_0, q_1, \dots, q_n\}$, 表示 FSA 中有 $n+1$ 个状态; Σ 是事件集, $\Sigma = \{e_1, e_2, \dots, e_m\}$, 表示 FSA 中引起变迁的 m 个事件; δ 是变迁集, 表示状态与事件之间的关系, $\delta \subseteq (Q \times \Sigma \times Q)$, 用三元组 (q, e, q') 表示状态变迁, 它表示当机器人处于 q 状态时, 若发生事件 e , 则机器人转向 q' 状态, 但 $q \notin F$; q_0 是初始状态, $q_0 \in Q$; F 是结束状态, $F = F_s \cup F_u$, F_s 是成功状态集, F_u 是不成功状态集, 且有 $F_s \cap F_u = \emptyset$ 和 $F_s \neq \emptyset$.

Ω 是对应于 Q 状态集中每个状态的组合行为集, 写成 $\Omega = \{CB_0, CB_1, \dots, CB_n\}$, 其中 CB_0, CB_1, \dots, CB_n 为在状态 q_0, q_1, \dots, q_n 时分别调用的组合行为.

2.3.2 $MB_MoveToGoal$ 宏行为

为了使机器人在具有“陷阱”的复杂障碍环境中鲁棒地到达指定的目标位置,我们设计了一个宏行

为 $MB_MoveToGoal$ 表示为:

$$MB_MoveToGoal(\Gamma, \Omega) \quad (4)$$

其中: $\Gamma = (Q, \Sigma, \delta, q_0, F)$, 它表示如图 6所示的有限状态自动机 FSA $Q = (S_0, S_1, S_2, S_3, S_4, S_5)$, 分别表示机器人处于 initial-state、move-to-point、move-along-left-obstacle、move-along-right-obstacle、achieved-goal和 unachieved-goal状态; 初始状态 $q_0 = S_0$; 成功状态集为 $F_s = S_4$, 不成功状态集为 $F_u = S_5$.

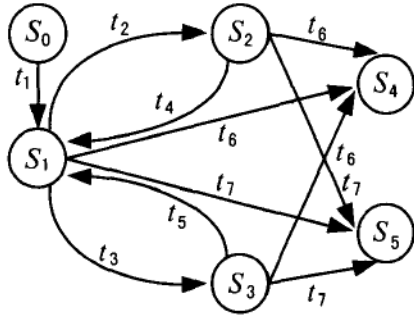


图 6 向指定目标位置移动的宏行为的 FSA

Fig 6 FSA of macro behavior moving to a goal position

事件集 $\Sigma = \{e_1, e_2, \dots, e_7\}$, 表示 FSA 中有 7 个事件, 如表 1 所示. $MB_MoveToGoal$ 的变迁集 δ 可以用表 2 描述, 例如, 机器人处于 S_1 状态时, 若发生事件 e_2 , 则表中对应的 $t_2 \rightarrow S_2$ 表示发生变迁 t_2 , 机器人转向 S_2 状态.

表 1 $MB_MoveToGoal$ 的事件表

Table 1 Events in $MB_MoveToGoal$

事件	描述
e_1	发出“start”命令, 启动宏行为.
e_2	$\theta_{current} < -10^\circ$ 且 $ \theta_{current} > \theta_{last} $, 表示机器人左前方有障碍物, 并且机器人的朝向与目标方向的夹角逐渐增大. $\theta_{current}$ 表示机器人当前朝向与目标方向的夹角, θ_{last} 表示上一个控制周期的夹角角度.
e_3	$\theta_{current} > 10^\circ$ 且 $\theta_{current} > \theta_{last}$, 表示机器人右前方有障碍物, 且机器人的朝向与目标方向的夹角逐渐增大.
e_4	$0^\circ < \theta_{current} < 90^\circ$ 且 $d_{current} < d_{mir}$, 表示目标位置在机器人右前方, 并且当前机器人与目标点之间的距离 $d_{current}$ 小于上次变迁 t_2 发生时记录下来的距离 d_{mir} .
e_5	$-90^\circ < \theta_{current} < 0^\circ$ 且 $d_{current} < d_{mir}$ 且, 表示目标位置在机器人左前方, 并且当前机器人与目标点之间的距离 $d_{current}$ 小于上次变迁 t_3 发生时记录下来的距离 d_{mir} .
e_6	机器人成功到达目标位置.
e_7	机器人在预定的时间内未能成功到达目标位置.

注: 定义机器人朝向与目标方向的夹角为 θ 变化范围为 $-180^\circ \sim 180^\circ$, 当机器人朝向到目标方向是逆时针时 θ 为正, 顺时针时 θ 为负.

Ω 是对应于 Q 中 $S_0 \sim S_5$ 的组合行为集. $\Omega = \{Null, CB_1, CB_2, CB_3, Null, Null\}$, $Null$ 表示对应的状态无行为, CB_1, CB_2 和 CB_3 对应的组合行为分别为 $CB_MoveToPoint, CB_MoveAlongLeftObstacle$ 和 $CB_MoveAlongRightObstacle$. $CB_MoveToPoint$ 是控制机器人在具有少量障碍物的简单环境中移向目标位置的组合行为, 但是可能存在“死循环”现象^[12, 13], 它包括的基本行为有: $EB_Recover, EB_Swerve()$ 和 $EB_GoToPoint$. $CB_MoveAlongLeftObstacle$ 为图 4 表示的组合行为. $CB_MoveAlongRightObstacle$ 与 $CB_MoveAlongLeftObstacle$ 类似, 只是跟踪右侧的障碍物.

表 2 $MB_MoveToGoal$ 的变迁表

Table 2 Transitions in $MB_MoveToGoal$

	S_0	S_1	S_2	S_3	S_4	S_5
e_1	$t_1 \rightarrow S_1$					
e_2		$t_2 \rightarrow S_2$				
e_3		$t_3 \rightarrow S_3$				
e_4			$t_4 \rightarrow S_1$			
e_5				$t_5 \rightarrow S_1$		
e_6		$t_6 \rightarrow S_4$	$t_6 \rightarrow S_4$	$t_6 \rightarrow S_4$		
e_7		$t_7 \rightarrow S_5$	$t_7 \rightarrow S_5$	$t_7 \rightarrow S_5$		

注: 在变迁 t_2 和 t_3 发生时需要记录下机器人与目标位置的瞬时距离, 并存入 d_{mir} 中, 用于后继事件的判断.

2.3.3 测试宏行为 $MB_MoveToGoal$ 的仿真和实验

首先利用 Pioneer Simulator 对 $MB_MoveToGoal$ 进行了一系列的仿真测试, 仿真中采用了如图 7 所示的两个环境模型, 图中粗实线表示墙壁或者障碍物. 在图 7(a) 中, 机器人的起始位置在 S , 机器人朝向 X 轴方向, 目标位置设定为 G . 机器人开始执行移向目标位置任务时处于 S_1 状态, 并由 $CB_MoveToPoint$ 控制, 在向目标位置移动过程中, 在 A 点处检测到机器人左前方有一堵墙壁, 机器人切换到跟踪左侧墙壁移动状态, 由组合行为 $CB_MoveAlongLeftObstacle$ 接管控制机器人. 它成功地到达“陷阱”环境的外侧, 避免了文 [29, 30] 中所述的“死循环”. 当机器人到达 B 点时, 机器人与目标点之间的距离小于在 A 点时机器人与目标位置的距离, 且目标位置在右前方, 机器人切换回 S_1 状态, 并直接朝着目标行驶. 图 7(b) 给出了机器人在另一个环境中向目标位置移动的轨迹, 图中的点 A, B, C, D 对应于变迁 t_3, t_5, t_2, t_4 发生的地方. 其它几个在更复杂环境中移向目标的仿真实验也都能成功完成.

仿真中使用的行为和协调器程序直接应用在 Pioneer3 AT 机器人实验中, 环境布置和仿真中所用

的环境基本一致, 在我们实验室进行的 4 个实验中机器人都能成功地抵达目的地, 且未发生任何碰撞.

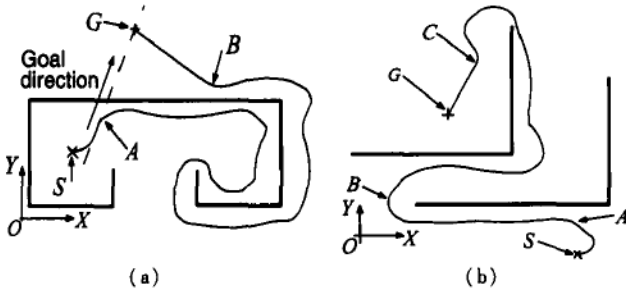


图 7 宏行为 MB_MovToGoal仿真的运动轨迹

Fig 7 Movement trajectories in simulation under the control of MB_MovToGoal

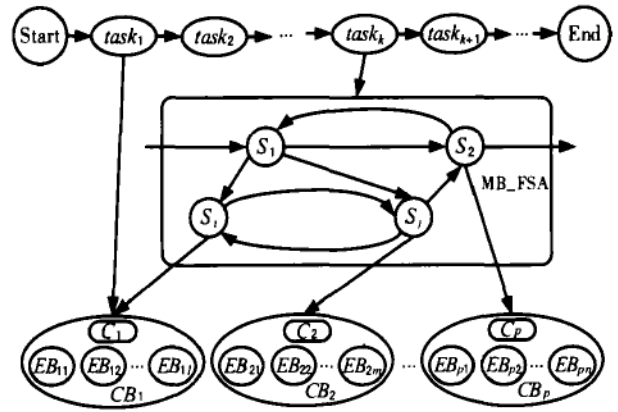


图 8 基于任务完成行为的事务分解图

Fig 8 Decomposition of mission based on task-achieving behaviors

3 事务分解和描述 (Mission decomposition and description)

3.1 事务分解

侦察机器人经常执行地形侦察、化学物质泄漏侦察、灾难受害者救援等任务, 对于机器人来说, 这些任务可以被分成若干个顺序执行的子任务, 由机器人执行^[7]. 可以采用图 8 所示的基于任务完成的方法进行事务分解, 一个事务的执行可以分解为若干个顺序执行的任务 $task_1, task_2, \dots, task_k, task_{k+1}, \dots$ 来实现. 对于简单的子任务, 机器人利用一个组合行为就可以自主完成, 如图 8 中的 $task_1$ 就可以用组合行为 CB_1 自主完成, 而图中的 $task_k$ 是一个复杂的任务, 需要调用预定义的、由 MB_FSA 协调的若干个组合行为的宏行为才能实现. 另一些任务则可能需要两个或者两个以上的组合行为或者宏行为并发运行才能够完成, 也可能需要一个甚至多个组合行为或宏行为的循环执行才能完成. 可以看出, 对于一个机器人来说, 预定义的基本行为、组合行为和宏行为的种类和数量越多, 机器人能够独立完成事务的能力越强.

但是, 对于半自主侦察机器人来说, 由于它的工作环境一般情况下是未知的非结构环境, 分解事务和选择合适行为模块的工作只能由操作侦察机器人的操作员来完成. 又因为操作员不能够在到达侦察现场之前规划侦察机器人的侦察事务, 而要求操作员根据现场实时情况进行事务规划, 这就对于机器人事务规划的事务规划器在操作的直观性、便捷性和灵活性等方面提出很高的要求. 为此, 我们设计了具有图形化操作界面的事务规划器, 它提供了预定义的机器人基本行为、组合行为和宏行为供操作员选择, 还提供合成这些行为的合成运算符用于合成实现某一特定任务的指令.

3.2 RS合成运算符

形式化地描述侦察机器人的事务规划是规划器的设计基础, 基于行为的机器人事务描述一般应该具有描述任务的顺序、条件、并发、循环执行的能力. 这里, 我们采用基于 Lyons 的 RS 模型^[14]定义的进程合成运算符作为基于行为的侦察机器人事务描述的行为合成运算符. RS 模型共有 6 个合成运算符, 简要叙述如下.

顺序运算符“;”表示两个或者多个行为可以依次顺序执行, 如事务 $T = A; B$ 表示执行事务 T 时, 先执行 A , A 结束后执行 B , 而不管 A 是正常完成还是异常终止. 如 $T = MoveToGoal(Goal1); MoveToGoal(Goal2)$ 表示机器人先运动到目标位置 $Goal1$, 然后再运动到目标位置 $Goal2$.

并发运算符“|”表示两个或者多个行为并行执行, 如 $T = A | B$ 表示执行事务 T 时 A 和 B 并发执行, 当 A 和 B 都结束时, 事务 T 才结束. 如 $T = LookFor(RedBall) | LookFor(YellowBall)$ 表示机器人寻找红球和黄球, 只有两个球都找到时停止.

条件运算符“:”可以用来表示有条件的行为网络, 如事务 $T = A : B$ 表示执行事务 T 时, 先执行 A , 当 A 成功结束后再执行 B , 否则不执行行为 B . 如命令机器人先寻找红色的球, 然后再移向它, 可以表示为: $T = LookFor(RedBall) : MoveToTarget(RedBall)$.

禁止运算符“#”类似于并发运算符, 如事务 $T = A \# B$ 表示执行事务 T 时 A 和 B 并发执行, 但当任意一个行为结束时, 其它的行为也结束. 如命令机器人寻找道路旁的一个危险标志, 则事务可以表示为: $T = FollowRoad() \# LookFor(DangerSign)$.

同步循环运算符“:;”是对上面 4 个运算符递归

定义, 如 $T = A; B$ 可以递归定义为 $A; B = A (B; (A; B))$, 它类似于“while-loop”循环. 如命令机器人捡垃圾, $LookFor(trash); PickUp(trash)$.

异步循环运算符“:”也是对上面 4 个运算符的递归定义, 如 $T = A: B$ 可以递归定义为 $A: B = A (B | (A: B))$, 但是, 异步循环运算符仅仅在每次条件满足时启动一组并发执行的行为, 它不是简单地循环.

3.3 事务描述

从图 8 可以看出, 我们首先将一个事务分解为若干个能够顺序执行的任务, 每个任务可以由一个基本行为、或者一个组合行为、或者一个宏行为完成, 也可以利用上述 6 种 RS 运算符合成多个行为的表达式来实现. 这里, 我们用 EB_set 表示预定义的机器人基本行为集, 用 CB_set 表示组合行为集, 用 MB_set 表示宏行为集, 定义 $B = EB_set \cup CB_set \cup MB_set$ 简称为机器人的行为集合. 用 Λ 表示 RS 运算符集合, $\Lambda = (“;”, “|”, “:”, “#”, “:;”, “::”)$. 定义 $Bexp$ 是用 Λ 中的合成运算符和 B 中的行为构造的、用来完成特定任务的行为组合表达式. 这样, 对应于图 8 中的每个任务, 定义执行此任务的行为组合表达式为一条任务执行语句 $TaskStatement$ 根据前面的定义, $TaskStatement$ 可以是 B 中的任意一个行为, 也可以是 $Bexp$ 表达式, 即: $TaskStatement = Bexp$, 或者 $TaskStatement \subset B$. 若干条顺序执行的任务执行语句 $TaskStatement$ 就构成了事务执行程序, 我们称这种事务执行程序为事务描述. 事务描述可以用 (5) 式表示:

$$SEQ(T, P) \tag{5}$$

其中: T 表示组成事务的子任务集合, $T = (task_1, task_2, \dots, task_n)$. $P = (TaskStatement_1, TaskStatement_2, \dots, TaskStatement_n)$ 是分别对应于任务 $(task_1, task_2, \dots, task_n)$ 的任务执行语句.

例如, 命令机器人从当前位置经过路径点 $A(x_1, y_1)$ 行驶到 $B(x_2, y_2)$ 位置, 然后寻找 B 点周围的危险标志, 直到找到为止. 我们将这个事务分成 3 个任务, 则这一事务可以描述如下:

//Search for danger sign

MoveToGoal(A); //移动到 A 点位置.

MoveToGoal(B): //移动到 B 点位置, 成功则转下一步.

(TurnAround(360) # SearchFor(Danger Sign)). //在 B 点旋转 360°, 并寻找周围墙上的危险标志, 找到或转一圈后未找到, 停止.

4 事务执行器 (Mission executor)

为了执行上述的事务规划, 设计了图 9 所示的基于宏行为的事务执行器, 用以执行式 (5) 表示的事务描述, 事务执行器包括一个事务描述解释器、任务执行管理器和任务执行器三个部分. 操作员规划的事务描述传送到机器人后, 由事务描述解释器逐条解释, 分解并转换成机器人的基本行为、组合行为或者宏行为, 然后传送给任务执行管理器.

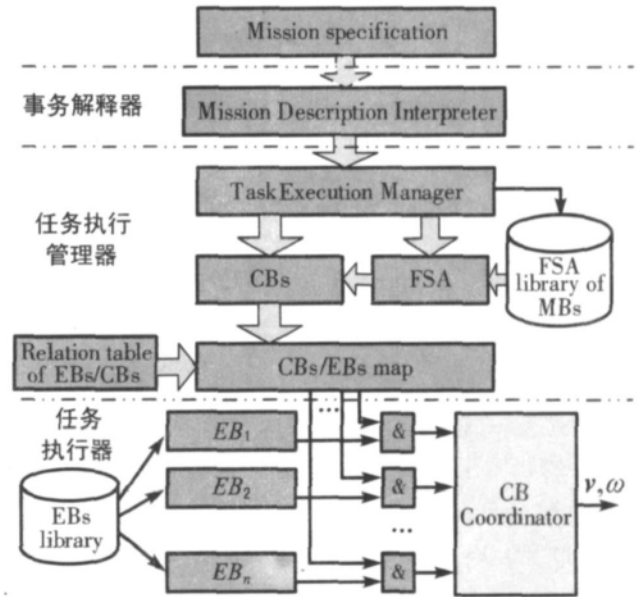


图 9 基于宏行为的事务执行器框图
Fig 9 Diagram of mission executor based on macro behavior

任务执行管理器则负责调用行为和管理行为的执行. 例如, 当前的任务执行语句是宏行为, 则任务执行管理器从宏行为的 FSA 库中调用并执行相应宏行为的 FSA 模块, 根据 FSA 当前的状态选择需要激活的组合行为, 再查询组合行为和基本行为的关系表, 确定需要激活的基本行为. 同样, 若当前的任务执行语句是用并发运算符“|”或者禁止运算符“#”合成的行为, 则当前任务需要激活的组合行为可能为两个或两个以上, 由任务执行管理器协调它们的执行.

任务执行器是任务的实施者, 它负责协调被激活的基本行为的并发和输出工作, 激活的基本行为输出可以通过“&”门输入到组合行为协调器, “&”门的选通信号由组合行为/基本行为映射器输出. 这里, 我们实现的组合行为协调器采用基于优先级的仲裁机制. 例如, 假设当前机器人执行组合行为 $CB_MoveAlongLeftObstacle$ 则与基本行为 $EB_Recover$, $EB_Swerve(R)$, $EB_OutsideCorner(L)$, $EB_MoveAlongWall(L, standoff)$ 和 $EB_Forward$ 相对应的“&”

门的选通信号为“1”，允许这些基本行为的输出传送到组合行为仲裁器，再按照这些基本行为的优先级确定最后的输出。

5 仿真和实验结果 (Results of simulation and experiment)

首先利用 Pioneer Simulator对事务执行器进行测试，采用了图 10所示的环境模型。例如，命令机器人从起始位置 *S* 移动到目标位置 *G*，再利用视频摄像头寻找 *G* 点周围的一个危险标志 (假定为一个红色标志)。在这样的环境中，由于环境比较复杂，即使使用前面所述的宏行为 *MB_MoveToGoal* 能够克服图 10 底部的“陷阱”环境，但当机器人到达 *P* 点时，机器人也将沿着图中的虚线所示的方向移动。为了使得机器人能够快速到达目标位置，在图形化事务编辑器上利用前面实现的行为和行为合成符来描述这个事务，可以方便地解决上述问题。可以先命令机器人从起始位置运动到 *A* 点，再运动到 *B* 点，然后运动到目标位置 *G*，接着执行搜索红色标志的任务。假设机器人的起始坐标位置为 *S* (3000 3000)，且机器人朝向 *X* 轴方向，*A* 点坐标为 (4500 5500)，*B* 点坐标为 (4500 8500)，*G* 点坐标为 (9000 8000)，则事务描述如下：

```

MB_MoveToGoal( 4500 5500); //移动到 A 点
MB_MoveToGoal( 4500 8500); //移动到 B 点
MB_MoveToGoal( 9000 8000); //移动到 G 点,

```

若成功到达 *G* 点则转下一步。

(*CB_TurnAround*(360) # *EB_SearchFor*(Red-Block)). //在 *G* 点利用组合行为 *CB_TurnAround*() 旋转 360°，并利用基本行为 *EB_SearchFor*() 寻找周围墙上的红色标志块，找到标志时停下；若转一圈后仍然未找到，也停止。

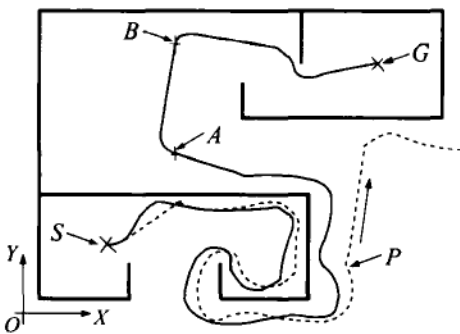


图 10 执行基于宏行为的事务描述的运动轨迹
 Fig 10 Movement trajectory during executing mission description based macro behavior

将上述事务的前 3 条任务执行语句作为一个事务描述发送给上节介绍的事务执行器执行，事务执行器连接到 Pioneer Simulator 仿真器上，可以看到机器人能够成功地经过 *A*、*B* 点到达 *G* 点。机器人的移动轨迹如图 10 中的粗实线表示。

接着将同样的事务执行器软件直接下载到 Pioneer3 AT 机器人中进行实验。Pioneer3 AT 机器人带有一个 Canon VC-C4 型的一体化 PTZ 摄像头、一个视频采集卡并安装了 Carnegie Mellon 大学开发的 ACTS (ActiMedia Color Tracking System)，实验场地布置得类似于图 10 所示的环境。将上述完整的具有 4 条任务执行语句的事务描述发送给机器人，机器人能够成功地沿着指定的路径到达目的地，然后原地转动，寻找红色的标志，当找到红色标志后立即停下来，且调整摄像头的俯仰角使得红色标志处于视频图像的中央位置。图 11 为部分实验场景照片。

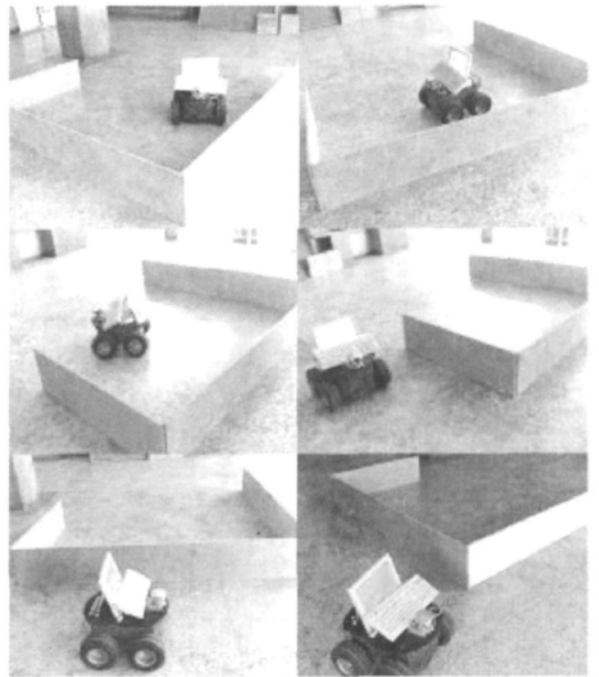


图 11 实验场景照片
 Fig 11 Snapshots of experimental scenes

6 结论 (Conclusion)

侦察机器人的事务规划必须在侦察现场进行，因此，方便、快捷的操作方式和清晰、直观的显示形式是事务规划器的设计要点，基于宏行为的半自主侦察机器人的事务执行机制研究就是为了给操作员在更高的抽象层次上规划机器人事务、减轻工作强度、提高规划效率。本文介绍的宏行为是基于基本行为和组合行为的、面向具体的任务目标而设计的，解

决了基于行为的机器人不能明确表达事务目标的矛盾. 提出的基于 RS 模型的行为合成符, 用于行为的合成, 从而构造出形式化的、能够完成具体任务的任务执行语句, 若干条顺序执行的语句就构成了一个机器人的事务描述. 提出并实现的基于宏行为的事务执行器既保留了基于行为的体系结构对环境的快速响应性, 又可以方便面向事务目标的任务执行语句的执行. 事务执行器基于开放式的体系结构和模块化的开发方法, 可以方便加入新的基本行为、组合行为和宏行为, 可以方便实现动态的行为协调机制. 仿真和实验都证明了基于宏行为的事务执行机制方案可行, 能够可靠地执行完成给定的任务.

参考文献 (References)

- [1] Brooks R A. A robust layered control system for a mobile robot[J]. IEEE Journal of Robotics and Automation, 1986, RA-2(1): 14-23
- [2] Arkin R C. Motor schema-based mobile robot navigation[J]. The International Journal of Robotics Research, 1989, 8(4): 92-112
- [3] Gat E, Desai R, Ivlev R, et al. Behavior control for robotic exploration of planetary surfaces[J]. Robotics and Autonomous Systems, 1994, 10(4): 490-503
- [4] Pirjanian P. Behavior Coordination Mechanisms: State-of-the-Art[R]. USA: University of Southern California, 1999.
- [5] Yen J, Pfluger N. A fuzzy logic based extension to Payton and Rosenblatt's command fusion method for mobile robot navigation[J]. IEEE Transactions on Systems, Man, and Cybernetics, 1995, 25(6): 971-978
- [6] Foka A F, Trahanias P E. Predictive autonomous robot navigation[A]. Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems[C]. Piscataway, NJ, USA: IEEE, 2002, 490-495
- [7] Arkin R C, MacKenzie D. Temporal coordination of perceptual algorithms for mobile robot navigation[J]. IEEE Transactions on Robotics and Automation, 1994, 10(3): 276-286
- [8] Kosecka J, Christensen H I, Bajcsy R. Experiments in behavior composition[J]. Robotics and Autonomous Systems, 1997, 19(3-4): 287-298
- [9] Scheutz M, Andronache V. A re-architectural mechanism for dynamic changes of behavior selection strategies in behavior-based systems[J]. IEEE Transactions on Systems, Man, and Cybernetics Part B: Cybernetics, 2004, 34(6): 2377-2395
- [10] Colmbetti M, Dorigo M, Borghi G. Behavior analysis and training: A methodology for behavior engineering[J]. IEEE Transactions on Systems, Man, and Cybernetics Part B: Cybernetics, 1996, 26(3): 365-380
- [11] Philipp A, Christensen H I. Behavior coordination in structured environments[J]. Advanced Robotics, 2003, 17(7): 657-674
- [12] Lyons D M, Arbib M A. A formal model of computation for sensory-based robotics[J]. IEEE Transactions on Robotics and Automation, 1989, 5(3): 284-293
- [13] Lyons D M. Representing and analyzing action plans as networks of concurrent processes[J]. IEEE Transactions on Robotics and Automation, 1993, 9(3): 241-256
- [14] Maes P. How to do the right thing[J]. Connection Science, 1989, 1(3): 291-323
- [15] Bagchi S, Biswas G, Kawamura K. Task planning under uncertainty using a spreading activation network[J]. IEEE Transactions on Systems, Man, and Cybernetics Part A: Systems and Humans, 2000, 30(6): 639-650
- [16] Gowdy J. SAUSAGES: between Planning and Action[R]. USA: Carnegie Mellon University, 1994
- [17] Mackenzie D C, Arkin R C, Cameron J M. Multiagent mission specification and execution[J]. Autonomous Robots, 1997, 4(1): 29-52
- [18] Backes P G, Tharp G K, Tso K S. The Web interface for tele-science (WITS)[A]. Proceedings of the 1997 IEEE International Conference on Robotics and Automation[C]. Piscataway, NJ, USA: IEEE, 1997, 411-417.
- [19] Backes P G, Rabideau G, Tso K S, et al. Automated planning and scheduling for planetary rover distributed operations[A]. Proceedings of the 1999 IEEE International Conference on Robotics and Automation[C]. Piscataway, NJ, USA: IEEE, 1999, 984-991.
- [20] Arkin R C, Balch T. AuRA: principles and practice in review[J]. Journal of Experimental and Theoretical Artificial Intelligence, 1997, 9(2-3): 175-189
- [21] Rogers E, Murphy R R, Stewart A, et al. Cooperative assistance for remote robot supervision[A]. Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics[C]. Piscataway, NJ, USA: IEEE, 1995, 4581-4586
- [22] Fong T, Thorpe C, Baur C. Collaboration, dialogue, and human-robot interaction[A]. Proceedings of the 10th International Symposium of Robotics Research[C]. Berlin, Germany: Springer-Verlag, 2001, 255-266
- [23] Anon. The Mobile Robot Laboratory in the College of Computing at the Georgia Institute of Technology. Mars2020[EB/OL]. <http://www-static.cc.gatech.edu/ai/robotlab/mars2020/>, 2004-05.
- [24] 王挺. 我国危险作业机器人研究开发取得新进展[J]. 机器人技术与应用, 2005, 12(7): 11-15
- [25] 王野, 王田苗, 裴葆青, 等. 危险作业机器人关键技术综述[J]. 机器人技术与应用, 2005, 12(7): 23-31
- [26] 李磊, 叶涛, 谭民, 等. 移动机器人技术研究现状与未来[J]. 机器人, 2002, 24(5): 475-480
- [27] 王挺, 王越超, 赵忆文. 多机构复合智能移动机器人的研制[J]. 机器人, 2004, 26(4): 289-294
- [28] Steenstrup M, Arbib M A, Manes E G. Port automation and the algebra of concurrent processes[J]. Journal of Computer and System Sciences, 1983, 27(1): 29-50
- [29] Chee B Y, Lang Y T, Tse W T. Fuzzy mobile robot navigation and sensor integration[A]. Proceedings of the Fifth IEEE International Conference on Fuzzy Systems[C]. New York, NY, USA: IEEE, 1996, 7-12
- [30] Zhu A, Yang S X. A fuzzy logic approach to reactive navigation of behavior-based mobile robots[A]. Proceedings of the IEEE International Conference on Robotics and Automation[C]. Piscataway, NJ, USA: IEEE, 2004, 5045-5050

作者简介:

- 唐鸿儒 (1964-), 男, 博士生, 副教授. 研究领域: 机器人控制技术.
- 宋爱国 (1968-), 男, 教授, 博士生导师. 研究领域: 机器人传感与控制技术.
- 章小兵 (1972-), 男, 博士生, 讲师. 研究领域: 机器人控制技术.