软件学报ISSN 1000-9825, CODEN RUXUEW E-mail: jos@iscas.ac.cn

*Journal of Software*, [doi: 10.13328/j.cnki.jos.000000] http://www.jos.org.cn

©中国科学院软件研究所版权所有. Tel: +86-10-62562563

自治移动机器人空间永恒探索算法的符号模型检测方法[[1]](#footnote-1)\*

蔡晓伟1,2, 张 民1,2

1(华东师范大学 上海市高可信计算重点实验室,上海 普陀 200062)

2(华东师范大学 计算机科学与软件工程学院,上海 普陀 200062)

通讯作者: 张民, E-mail: zhangmin@sei.ecnu.edu.cn

摘 要: 随着物联网技术的发展,自治移动机器人(autonomous mobile robots)在网络中的作用也日益重要,利用形式化的方法验证自治移动机器人行为的正确性逐渐成为新的研究热点.当前主要的验证方法多以初始状态已知为前提且面临状态爆炸问题.本文以自治移动机器人空间永恒探索算法为例,提出自治移动机器人符号模型检测方法,该方法不依赖某个具体的初始状态,且适用于不同的同步模型.同时,借助符号模型检测的高效性,有效避免了状态爆炸问题.利用nuXmv[5][9]符号模型验证工具对机器人探索算法在三种同步模型[1]:完全同步模型FSYNC(Full-synchronous model)、半同步模型SSYNC(semi-synchronous model)、异步模型ASYNC(Asynchronous model)进行建模并利用LTL公式定义算法的性质,最终实现算法的形式化验证.验证结果表明在假设初始状态未知的条件下依然可验证性质不被满足并找到反例.同时,实验数据表明了符号模型检测对自治移动机器人算法形式化验证的可行性与高效性.

关键词: nuXmv;移动机器人;空间探索;符号模型检测;

中图法分类号: TP311

中文引用格式: 蔡晓伟,张民. 自治移动机器人空间永恒探索算法的符号模型检测方法. 第二届全国形式化方法与应用会议(FMAC 2017).http://www.jos.org.cn/1000-9825/ 0000.htm

英文引用格式: Xiaowei Tsai, Min Zhang. On Symbolic Model Checking of Mobile Robots Perpetual Exploration Algorithm. 2nd National Symposium on FMAC, 2017 (in Chinese).http://www.jos.org.cn/1000-9825/0000.htm

On Symbolic Model Checking of Mobile Robots Perpetual Exploration Algorithm

CAI Xiaowei1,2, ZHANG Min1,2

1(Shanghai Key Lab of Trustworthy Computing, ECNU, Shanghai 200062, China)

2(MoE International Joint Lab of Trustworthy Software (ECNU), Shanghai 200062, China)

**Abstract**: With the development of Internet of Things technology, the role that autonomous mobile robots play in the network is becoming increasingly important. Formal verification of the correctness of autonomous mobile agents has become a new research topic. Most of the exising approaches either suffer state-explosion problem or rely on concrete initial states, which however are usually not undefined and have to be enumerated to make verification complete. In this paper, we propose a symbolic model checking approach to the formal verification of a typical automonous mobile agent system called mobile robot perpetual exploration system. For the symbolicity feature of the model checking, the verifcation does not rely on specific initial states and meanwhile state-explosion problem can be avoided. We use the state-of-the-art symbolic model checker nuXmv to verify the mobile robot perpetual exploration algorithm under three different scheduling modes called full synchronous model (FSYNC), semi-synchronous model (SSYNC) and asynchronous model (ASYNC). Experimental results show that even without providing specific initial states a counterexample can be found in our approach for the perpetual exploration property, which concides with the existing verification result which is obtained by model checking with specific initial states. Meanwhile, the experimental data shows the feasibility and efficiency of symbolic model checking in the formal verification of autonomous mobile agent systems.

**Key words**: nuXmv; LTL; Mobile robots; space exploration; symbolic model checking

随着物联网技术的飞速发展,自治移动机器人(Automonous mobile robots)在网络中的作用日益重要.自治移动机器人可以在没有任何中央调度的情况下,通过互相协调合作完成任务,用于一些特殊环境,如未知区域的地图构造、环境监测、城市搜索、在人无法进入的危险区域作业,监视、未知空间的探索等.

自主机器人空间探索协议是一种典型的自主移动机器人协议,通过定义自主机器人的行为,实现物理空间的探索.在机器人自主空间探索协议中,机器人之间无法通过外部表征进行识别,所有机器人是完全相同的且都执行相同的探索算法.机器人具备以下几个特征:1、无记忆存储,即不能存储过去完成的动作;2、无方向传感器,即本身无法辨别方向,并且没有偏好移动方向;3、无通讯功能,即机器人之间不能发送和接收消息;4、有视觉传感器,即可以通过视觉传感器获取空间其他机器人的位置信息.空间模型由原始的连续二维欧几里德空间模型,逐渐演化成为有限位置的离散空间模型.离散空间使用图来描述,图的结点代表空间的位置,边表示机器人可以由一个结点到相邻结点的路径.离散空间模型简化了机器人模型,更加关注机器人的数量和空间位置之间的关系.

机器人的移动可以分为三个阶段:观察(Look),计算(Compute),移动(Move).在观察阶段,机器人可以获取图的快照信息,这些快照信息记录着其他机器人在图上的位置.在收集到其他机器人位置信息之后,机器人进入计算阶段,依据收集到的信息计算决定是否移动.移动阶段完成之前计算阶段做出的移动策略.原始模型中,部分机器人同步执行观察、计算、移动,并且这个三个阶段具有原子性,这种模型下,调度策略有两种:完全同步调度模型FSYNC (fully-synchronous model)和半同步调度模型SSYNC (semi-synchronous model).随后,Flocchini et al提出了异步调度模型ASYNC (asynchronous model),该模型中,每个机器人观察,计算,移动三个阶段不再具有原子性,每个机器人在异步执行移动阶段可能会使用过时的快照信息做出移动决策.自主机器人空间探索协议有两个变体,包括探索终止(exploration with stop)和永恒探索(perpetual exclusive exploration).探索终止即所有的机器人最终都会在某个位置停止探索;而永恒探索表示所有的机器人将不停地探索所有可能的节点.

自主机器人空间探索协议核心问题是根据具体的物理空间如何定义自主移动机器人的行为以保证其完成预设的任务.针对该问题,目前主要借助手动推演与形式化验证技术以保证自治移动机器人算法或协议满足一定的性质.然而手动推演不仅过程冗长复杂,推演过程中也容易出现错误. 尤其对于异步调度模型,存在快照过时的情况,根本无法使用手动推演的方式进行推演验证.形式化方法因其自动性,严谨性与高效性逐渐被用于各种自治移动机器人协议的验证. 如Béatrice 等人使用DiVinE和ITS工具实现了移动机器人算法的验证[1].Ha等人使用Maude重写逻辑语言实现移动机器人永恒探索算法的验证[2].然而已知的这些方法多针对某种特定的调度模型进行建模,并通过人为设定一个具体的初始状态进行验证.然而自治移动机器人算法或协议中系统的初始状态多是未知的.尽管理论上可以通过穷举所有可能的初始状态对系统进行一一验证,然而在实际操作中不仅效率低下并容易发生遗漏.

本文以自主机器人永恒空间探索协议为例,提出一种符号模型检测方法用于自治移动机器人算法或协议的建模与验证.在三种典型的调度策略下,利用nuXmv工具分别对机器人自主空间探索算法进行建模并利用LTL公式对机器人的移动行为进行定义,利用nuXmv提供的基于BDD方法及SMT方法实现了协议的符号模型检测,分别验证不同场景下(不同环节点个数与机器人个数)协议是否满足永恒探索的需求.当出现不满足情况时,nuXmv给出不满足的状态路径作为反例,分析不满足时的具体原因.验证结果与Béatrice 和Ha等人的验证结果相同.相比Béatrice和Ha等人的验证方法,本文提出的方法具有如下优点: 1、基于符号模型检测的方法不依赖于某个具体的初始状态,可以在不提供初始状态的前提下对协议的性质进行验证;2、符号模型检测建模方法具有模块化特性,主要模块可在不同的调度模型下重复使用,大大简化了建模的复杂性;3、符号模型检测的高效性可有效避免验证过程中的状态爆炸问题.

文章结构:第一章移动机器人永恒探索算法和调度策略介绍;第二章nuXmv探索算法在不同调度策略下的建模;第三章探索算法的模型检测与结果分析;第四章本文工作与相关工作的对比分析;第五章本文工作总结与未来工作的展望.

# 移动机器人探索算法和调度策略的介绍

本节介绍机器人在图上移动过程,机器人在不同的调度策略下探索算法的差异.

## 探索空间定义

探索空间一般可抽象为一个无向连通图,图的每个结点表示空间上机器人可达的位置,边表示机器人可以通过的路径,机器人沿着该条路径到达相邻的空间位置.每个结点在同一时间只能有一个机器人,黑色结点说明该空间位置在此刻有一个机器人,白色结点说明该空间位置上没有机器人.

|  |  |
| --- | --- |
| 图 1无向连通图表示探索空间 | 图 2机器人移动三阶段 |

图1给出了一个简单的离散探索空间的无向连通图表示,在该探索空间上有两个机器人,它们分别位于两个黑色的空间结点上.类似于计算机网络的网络拓扑结构,探索空间结构也有总线拓扑结构、星型拓扑结构、环形拓扑结构、树形拓扑结构等.

## 机器人移动三个阶段

机器人移动分为三个阶段,分别是观察(look)、计算(compute)和移动(move).在观察阶段,机器人通过视觉传感器获取环境中其他机器人的位置的快照信息.计算阶段根据观察阶段获取的位置快照信息计算得出移动决策,即机器人是否移动,若移动是沿着那条路径进行.移动阶段机器人的动力装置依据移动决策作出相应的移动.如图二所示,这三个阶段是按照观察、计算、移动再到观察重复进行的.

## 机器人移动调度策略

假设在一个探索空间中存在个机器人,使用集合表示.空间所有位置进行编号,使用集合表示所有位置结点的编号.机器人与位置之间的关系使用函数表示,机器人的位置,且满足对于任意两个机器人,

集合Rob中只有被随机调度器选中的机器人,才执行移动.那么被选中的机器人非空集合Rob,对于,执行观察、计算、移动,不做相关动作,这就是调度策略.目前有三种调度策略,分别是完全同步调度策略FSYNC,、半同步调度策略SSYNC与完全异步调度策略ASYNC.

在半同步调度策略中,非空集合,任意机器人同步执行观察、计算、移动,完成这三步之后,就是完成一个移动阶段.进入下一个移动阶段之前,调度器又会随机选中一组机器人,重复执行上述的移动阶段,也就是每个移动阶段选中的集合Sched都是随机的.如图三,算法表示半同步调度策略具体执行过程.其中synchronous块,表示同步块中所有机器人同步完成移动阶段.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| FSYNC-SCHEDULE(Rob) | | | | | SSYNC-SCHEDULE(Rob) | | | | | ASYNC-SCHEDULE(Rob) | | | | |
| 1 | while | | | | 1 | while | | | | 1 | foreach r in Rob | | | |
| 2 |  | synchronous{ | | | 2 |  | choose Sched from Rob | | | 2 |  | synchronous { | | |
| 3 |  |  | foreach r in Rob{ | | 3 |  | synchronous { | | | 3 |  |  | foreach r in Rob{ | |
| 4 |  |  |  | r.look | 4 |  |  | foreach r in Sched{ | | 4 |  |  |  | r.look |
| 5 |  |  |  | r.compute | 5 |  |  |  | r.look | 5 |  |  |  | r.compute |
| 6 |  |  |  | r.move | 6 |  |  |  | r.compute | 6 |  |  |  | r.move |
| 7 |  |  | } | | 7 |  |  |  | r.move | 7 |  |  | } | |
| 8 |  | } | | | 8 |  |  | } | | 8 |  | } | | |
| 9 |  |  | | | 9 |  | } | | | 9 |  | | | |
| (a) 完全同步调度策略 | | | | | (b) 半同步调度策略 | | | | | (c) 完全异步调度策略 | | | | |

表 1 **完全同步,半同步与异步调度策略**

完全同步调度策略是半同步调度策略中很特殊的一种,每个移动阶段选中的机器人集合,所有机器人都被调度器选中.完全同步调度策略算法如图四所示,每个移动阶段被调度器选中的是全部机器人,同步执行移动阶段.

完全异步调度策略,每个机器人异步执行移动阶段,也就是说某个机器人还在观察阶段,其他机器人或许在执行计算或者移动.在异步调度策略中,机器人会使用过时的快照信息,做出移动策略.如图五所示,每个机器人都各自执行观察、计算、移动,机器人之间是并行执行移动阶段.

# 移动机器人环形空间永恒探索算法

探索空间结构有总线拓扑结构、星型拓扑结构、环形拓扑结构、树形拓扑结构,在此以环形拓扑结构为研究对象.首先介绍环形拓扑结构环上的机器人视觉快照、匹配移动算法获取移动决策、移动决策的执行.在此基础上,将介绍永恒探索移动算法的相关概念.

## 机器人视觉快照

图3给出一个简单的环形拓扑结构探索空间,沿着环的顺时针,给每个位置结点进行递增编号,所有位置的编号组成的集合为Pos,黑色结点表示该位置有一个机器人,白色结点表示该位置没有机器人,下面给出结点上是否有机器人的定义.



图 3 环形拓扑结构探索空间

**定义1 (结点机器人数)** 结点机器人,j表示结点的编号.当结点j上有k个机器人时,当结点上没有机器人时,.

在环形拓扑结构的空间中,机器人可以顺时针观察,也可以逆时针观察.那么每个机器人的观察位置快照就有顺时针快照和逆时针快照两种.

**定义2 (****机器人快照)** 结点上机器人通过视觉传感器获取位置快照信息,表示快照方向,+表示顺时针,-表示逆时针,表示结点位置.在拥有n个结点的环形拓扑结构探索空间上,结点j上机器人顺时针和逆时针的快照如下:

如图3中结点0为例,其顺时针和逆时针快照如下:

这种机器人快照表示方法有一定弊端,当环形拓扑结构探索空间结点数n很大时,这种表示方式很不方便.后续提出了一种F-R的定义方式,这里是可以表示连续无机器人结点个数和连续有机器人结点个数.表示连续m个结点无机器人,表示连续n个结点有机器人.那么顺时针序列1和逆时针序列1转化成F-R的定义方式为:

## 环形拓扑结构空间移动机器人探索算法

环形拓扑结构空间的探索模式主要有两种,即环形空间探索停止(Ring exploration with stop)和环形空间永恒探索(Perpetual ring exploration).本文重点研究环形空间永恒探索算法.

### 最小移动算法

最小移动算法是指环形拓扑结构空间结点数为n ≥ 10,环形拓扑结构空间上机器人数r = 3,并且n和r数量关系互质的情况下,确保机器人的移动满足环形空间永恒探索.移动算法可分为两个阶段,分别是稳定阶段(Legitimate phase)和收敛阶段(Convergence phase).

最小移动算法中的稳定阶段的移动规则定义如下:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Legitimate phase | | | | | |
| RL1:: |  |  |  | → | r.Back |
| RL2:: |  |  |  | → | r.Front |
| RL3:: |  |  |  | → | r.Front |



图 4 最小移动算法的稳定阶段

如图4所示,C1中机器人A快照顺时针快照 ,匹配移动算法RL1做出后退的决策,其他机器人B、C此时无匹配移动算法,保持静止,只有机器人A从结点n-1移动到结点n-2.相同,在C2中机器人B从结点0移动到结点n-1,C3中机器人C从结点3移动到结点2.最后如C4所示,对于机器人A、B、C而言,在C1和C4状态下所获取的快照是一样的,也就是说,整个系统会重复上述C1到C4的过程.此时,系统被称为进入了稳定阶段.稳定阶段的系统状态称为稳定状态.系统进入稳定状态,就会永远停留在稳定阶段.

系统除了稳定状态之外,还有其他的状态,在这些状态下,系统通过一定的规则逐渐到达某个稳定状态,进而系统进入稳定状态.系统从非稳定状态到达稳定状态的过程,被称为收敛阶段.以最小移动算法为例,其收敛阶段的移动规则定义如下:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Convergence phase | | | | | |
| RC1:: | 4 ≤ x ≤ z | ⋀ |  | → | r.Front |
| RC2:: | x ≠ y , x ˃ 0 | ⋀ |  | → | r.Doubt |
| RC3:: | 0 ˂ x ˂ z ˂ y ⋀ (x,y) ≠ (1,2) | ⋀ |  | → | r.Front |
| RC4:: |  |  |  | → | r.Back |
| RC5:: |  |  |  | → | r.Back |

### 环形空间永恒探索算法的性质

在环形拓扑结构空间中,要求在任意时刻每个结点至多只有一个机器人,并且对于任何一个机器人无论其在初始状态位于环上哪个节点,都可以无限次的访问环上的任意一个节点.这两个性质分别被称为非冲撞性与不终止性.另外,无论机器人如何移动,机器人在环上的顺序是不变的,即任意一个机器人的左右邻居永远是不变的.

* 非冲撞性:同一时刻每个结点上,至多有一个机器人,两个机器人不能同时经过同一条边.
* 非互换性:.按照某一方向,如顺时针或逆时针,如果机器人A在机器人B的前面,那么在随后任意可达的状态中机器人A则一直会在机器人B的前面.
* 非终止性:每个机器人对都可以对环上任意一个结点无限次地的访问.

# 基于nuXmv的空间永恒探索算法的建模

使用nuXmv对移动机器人空间永恒探索算法分别根据完全同步调度策略FSYNC、半同步调度策略SSYNC、完全异步调度策略ASYNC三种调度策进行建模,根据建立的模型验证机器人移动算法是否满足永恒探索的性质.本章节主要以最小移动算法是否是永恒探索算法为例,详述建模与验证过程.

## nuXmv的简单介绍

nuXmv是经典符号模型检测工具nuXmv的后续版本,用于有限状态和无限状态系统(尤其是同步系统)的验证分析,支持CTL及LTL等时序逻辑公式的符号模型检测.对于有限状态系统,采用当前最先进的SAT求解算法,而对于无限状态系统,通过基于SMT可满足性求解利用MathSAT5求解工具实现性质的验证.

nuXmv继承了nuXmv模块化建模特性.每个模块(Module)可描述一个状态迁移系统.模块主要包含三个部分:变量(variables),约束(constraints)和规范(specification).变量用于定义系统状态,约束用于定义状态转移关系及模型的一些约束条件,而规范表示利用CTL或者LTL逻辑公式定义的将要验证的系统性质.对于包含多个具有相同行为实体的系统,可以定义一个参数化的模块描述实体的行为,然后通过模块实例化描述系统的整体行为.因此,nuXmv这种模块化的特性使其非常适用于自治移动机器人系统的建模.此外,nuXmv符号模型检测方式可以在不提供系统初始状态的条件下对系统进行验证.而自治移动机器人系统大多不对机器人的具体的初始位置进行设定,而只定义一些必要的约束,如每个节点上至多只有一个机器人等.同时该特性还可以有效避免系统的状态爆炸问题.因此,从建模与验证的角度分析,nuXmv都非常适用于自治移动机器人系统的形式化验证与分析.

## 移动机器人空间永恒探索算法的建模

### 移动机器人的建模

通过一个参数化的模块对移动机器人建模，模块的主要部分定义如图5所示.模块带有三个参数p1,p2和p3,分别表示机器人自身机器左右两个邻居机器人的位置.需要邻居机器人作为参数的原因是机器人的移动需要考虑其相邻机器人的位置.机器人的状态用两个变量phase和move表示.前者表示机器人的移动状态，即查看-计算阶段(lc)或移动阶段(m),而后者表示机器人将要移动的距离,取值可为-1,1和0,分表表示向后,向前移动一个节点和不发生移动.

下面重点以变量move的值得计算和机器人位置p1的变化为例,介绍机器人移动过程的建模.

图 5 利用NuXmv对移动机器人的定义模块

1 **MODULE** robot(p1,p2,p3) -- p1,p2,p3表示机器人及其两个邻居的位置

2 **VAR**

3 phase : {lc,m}; -- lc表示look&computing, m表示move

4 move : -1..1; -- 变量move的值可为-1, 0和1

5 **ASSIGN**

6 **init**(phase) := lc; -- 变量phase的初始值为lc, move为0

7 **init**(move) := 0; -- 注意机器人的初始位置并未确定

8 **next**(phase) := -- 以下6行表示变量phase在后续状态中的值

9 **case**

10 phase = lc : m; -- 若当前值为lc, 则在下一个状态为m

11 phase = m : lc; -- 若当前值为m, 则在下一个状态为lc

12 TRUE : phase; -- 否则,phase的值不变

13 **esac**;

14 **next**(move) := -- 变量move在下一步的值.仅以一种情况为例

15 **case**

16  phase = lc & ((((p2 - p1) >= 0)? (p2 - p1 - 1):

17 (p2 - p1 - 1 + 10)) = 0) & ((((p3 - p2) >= 0)? (p3 - p2 - 1):

18 (p3 - p2 - 1 + 10)) = 2):-1

19 …

20 **esac**;

21 **next**(p1) := -- 机器人在下一个状态是的位置

22 **case**

23 phase = m & (move + p1) <= 0 : 10 + (move + p1);

24 phase = m & (move + p1) > 0 & (move + p1) <= 10: (move + p1);

25 phase = m & (move + p1) > 10 : (move + p1) - 10;

26 TRUE : p1;

27 **esac**;

28 **FAIRNESS**

29 Running1

如图5中15-17行代码所示，使用NuXmv描述移动算法RC1顺时针方向的匹配，在p1位置上的机器人r从自身开始，按照顺时针方向，依次计算p1到p2、p2到p3之间的间隔个数。对应匹配RC1中机器人的之间的间隔个数，以此作为机器人r顺时针匹配移动算法RC1的依据。逆时针方向的匹配则是计算p1到p3、p3到p2之间的间隔个数作为匹配依据.相邻机器人A到机器人B间隔个数使用表示,其中ℱ表示方向，+表示顺时针，-表示逆时针。机器人r的移动量move使用表示，下面分别介绍和的计算方法。

在环形拓扑结构空间中,每个机器人在空间每个时刻都有一个位置,可以根据机器人位置计算机器人之间的间隔个数,这样可以知道每个机器人顺时针和逆时针的快照F-R序列. 如图6中所示,环形拓扑结构空间都按照顺时针方向进行位置编号,位置a处是机器人A,位置b处是机器人B,当结点编号到n-1时,下一个位置是0,那么计算机器人A到机器人B之间连续没有机器人的结点数量时,就要考虑机器人A和机器人B之间是否包含位置为0的情况.顺时针时,b>a说明不包含位置为0的情况,b<a说明包含位置为0的情况.逆时针时,说明包含位置为0的情况,说明包含位置为0的情况.



图 6顺时针和逆时针快照获取

其中,表示顺时针,表示逆时针.n是环形探索空间的位置结点的个数.利用上述公式,计算相邻机器人之间的无机器人的位置个数,进一步可知每个机器人视觉快照F-R序列.

所有机器人都执行相同的移动算法,机器人通过位置信息计算获取F-R快照序列.对于一个机器人r来说,其每次发生移动之后,其位置结点编号c(r)发生变化.后续机器人获取快照信息也会随着位置的变化而变化.那么,在机器人模型中,需要传入其他机器人的位置信息,设置统一的移动算法.机器人模型的核心就是移动算法的表示.机器人移动有前进和后退,而前进和后退是以顺时针和逆时针的机器人F-R快照作为参照物.机器人在编号序的位置结点上的移动,就是其所在位置编号值+1或者-1.

假设机器人r的F-R快照满足移动算法中的某条规则L,记为,则根据规则L可确定r的移动方向,用变量表示,从而计算出r的移动量.计算公式如下:

对于环形拓扑结构空间,如果同一机器人r在位置的顺时针和逆时针快照F-R序列相同,即机器人r的快照F-R序列是对称的.若r的快照F-R匹配算法中某条规则L,则根据规则L得出的移动方向无论是前进或者后退,上述公式中的两个条件均成立,即机器人的移动量可为1或者 -1,此时则随机选择一个值.若r的快照F-R不能匹配算法中任何一个移动规则时,.

机器人进入移动阶段,其下一个位置结点编号为.机器人在新的位置,改变了整个系统中机器人的快照F-R序列,机器人根据新的快照F-R序列,做出对应的移动决策.

### 三种调度策略的建模

对于完全同步调度策略和异步调度策略，利用上节对移动机器人定义的模块进行实例化.假设只考虑三个机器人的情况.用r1,r2和r3分别表示三个机器人,变量p1,p2和p3表示三个机器人在环上的位置.三个机器人的位置关系为从顺时针的方向r1在r2之后,r2在r3之后.则r1可被定义为机器人模块的一个实例,如表2(a)所示.机器人r2与r3的实例化与r1类似.而异步调度策略的建模只需要在机器人模块实例化时加上关键字**process**,如表2(c)所示**.**该关键字的作用是声明实例之间的执行以异步方式进行[[2]](#footnote-2).

|  |  |  |
| --- | --- | --- |
| r1 : robot(p1,p2,p3);  r2 : robot(p2,p3,p1);  r3 : robot(p3,p1,p2); | r1 : robot’(p1,p2,p3);  r2 : robot’(p2,p3,p1);  r3 : robot’(p3,p1,p2); | r1 : **process** robot(p1,p2,p3);  r2 : **process** robot(p2,p3,p1);  r3 : **process** robot(p3,p1,p2); |
| 1. 完全同步调度模型 | 1. 半同步调度模型 | 1. 异步调度模型 |

表 2 三种调度策略的建模

对于半同步调度策略,每个机器人在调度过程中只有选中和非选中两种情况.因此在移动机器人定义模块给机器人设置一个调度变量*dispatcher*,可以有*chosen*(选中)与*steady*(非选中)两种取值情况.在nuXmv中,在声明变量时可指定变量的取值范围.通过关键字VAR声明变量即其取值范围,代码如下:

声明中指出的取值为.在使用时,不给赋初始值,系统初始状态值也是随机的,即 的初始值可以是也可以是.下一个移动过程开始前,都会执行一次调度,每次调度是随机的,即对于每个机器人来说,下一个移动过程中的取值,在取值范围之内都是随机的.使用nuXmv的*next*函数,对机器人下一个移动过程中,的取值进行定义:

*next*函数中,指明的可取值范围,也就是说,下一个移动状态时,随机取其中之一.机器人在匹配移动算法时,若变量取值为,才做移动决策的计算.

新的移动机器人定义的模块被命名为robot’ ,与robot具有相同的三个参数.为描述半同步调度策略，只需要利用模块robot’初始化机器人实例即可,如表2 (b)所示.

## 永恒探索性质LTL公式定义

根据上节对移动机器人空间永恒探索算法的建模,算法的无冲撞性,非互换性和非终止性可分别用如下如下LTL公式定义:

(非冲撞性)

(非互换性)

(非终止性)

上述三个性质都以移动机器人空间永恒探索算法满足公平性为前提,即任何一个机器人都可以被无限次的调度,公平性可用如下LTL公式描述:

(公平性)

# 验证结果与分析

在完全同步调度策略(FSYNC,Fully Synchronous)、半同步调度策略(SSYNC,Semi Synchronous),验证最小移动算法是满足对环形拓扑结构空间的永恒探索,在完全异步调度策略(ASYNC,Asynchronous)下,不满足对环形拓扑结构空间的永恒探索.

完全同步调度策略验证结果如下:

|  |  |  |
| --- | --- | --- |
| 位置结点数n | 验证结果 | 执行时间(min) |
| 10 | 满足 | 06:55:13 |
| 11 | 满足 | 54:50:19 |
| 13 | 未知 | 时间超过1小时 |

半同步调度策略验证结果如下:

|  |  |  |
| --- | --- | --- |
| 位置结点数n | 验证结果 | 执行时间(min) |
| 10 | 满足 | 01:27:10 |
| 11 | 满足 | 12:28:36 |
| 13 | 满足 |  |

完全异步调度策略验证结果如下:

|  |  |  |
| --- | --- | --- |
| 位置结点数n | 验证结果 | 执行时间(min) |
| 10 | 不满足 |  |
| 11 | 不满足 |  |
| 13 | 不满足 |  |

使用NuSMV符号化模型检测方法对最小机器人移动算法，分别在完全异步模型、半同步模型、异步模型下建模，成功获得判定结果，此结果与其他文献的结论是一致的。最小机器人移动算法在完全同步模型和半同步模型下满足对环形空间永恒探索性质，而在异步模型中不满足，根据实验结果给出的反例状态路径，可以得出其原因是异步过程中，机器人使用过时的快照信息做出的移动决策会导致相邻的机器人发生碰撞，此时同一个位置结点上有两个机器人，后续所有机器人都没有与移动算法相匹配，机器人都不能移动，此时详尽过程与Ha Thi Thu Doan的文章中所获得反例原因是一样的，在他的文章中定义此种情况为死锁状态。

本文在Béatrice Bérard、Ha Thi Thu Doan取得的研究成果的基础上，实现了移动机器人符号化模型检测方法，并在Béatrice Bérard、Ha Thi Thu Doan的基础上有所进步，NuSMV中使用LTL描述机器人任意初始状态，验证机器人移动算法是否是永恒探索算法，这种验证方式更加完备，贴近永恒探索算法初始条件，在Ha Thi Thu Doan也提及了机器人位置初始转态，在他的文献中验证了固定某一初始状态下，实现移动算法的验证，对于每种初始状态，都手动定义，在位置结点数量和机器人数较少的时候，这种方式还比较容易操作，但是面对结点数量和机器人数量增多时，这种方法很容易遗漏初始状态，按照排列组合，每增加一个结点或者机器人数，初始状态可能是指数级的增长。使用NuSMV符号化模型检测方法在验证最小机器人移动算法时，模型为完全异步模型、半同步模型，若设定机器人数初始位置状态，电脑配置cpu intel E3-1231 v3 内存16G的情况下，一般计算时间15s以内。

# 总结以及未来工作

本文提出了自治移动机器人系统的符号化模型检测方法,以移动机器人空间永恒探索算法为例,介绍了算法在完全同步,半同步及异步调度模式下的形式化建模与验证过程,分别验证了算法的不冲突性,不终止性及不交换性等主要性质.验证结果表明在不提供初始状态的条件下通过符号化模型检测的方法同样可以找到算法在异步调度模式下不满足不终止性的反例.

References:

1. Bérard B, Lafourcade P, Millet L, et al. Formal verification of mobile robot protocols. Distributed Computing, 2013:1-29.
2. Doan H T T, Bonnet F, Ogata K. Model Checking of a Mobile Robots Perpetual Exploration Algorithm. International Workshop on Structured Object-Oriented Formal Language and Method. Springer, Cham, 2016:201-219.
3. Blin L, Milani A, Potop-Butucaru M, et al. Exclusive Perpetual Ring Exploration without Chirality. Lecture Notes in Computer Science, 2010, 6343:312--327.
4. Baldoni R, Bonnet F, Milani A, et al. On the Solvability of Anonymous Partial Grids Exploration by Mobile Robots. Principles of Distributed Systems. Springer Berlin Heidelberg, 2008:428-445.
5. Cavada R, Cimatti A, Dorigatti M, et al. The nuXmv Symbolic Model Checker. Computer Aided Verification. Springer International Publishing, 2014:334-342.
6. Suzuki I, Yamashita M. Distributed Anonymous Mobile Robots: Formation of Geometric Patterns. Sirocco'96, the, International Colloquium on Structural Information & Communication Complexity, Siena, Italy, June. DBLP, 1999:313-330.
7. Suzuki I, Yamashita M. Erratum: Distributed anonymous mobile robots: formation of geometric patterns. Siam Journal on Computing, 1999, 28(4):1347-1363.
8. Flocchini P, Prencipe G, Santoro N. Distributed Computing by Oblivious Mobile Robots. Morgan & Claypool Publishers, 2012.
9. Cimatti A, Griggio A, Schaafsma B J, et al. The MathSAT5 SMT Solver. International Conference on TOOLS and Algorithms for the Construction and Analysis of Systems. 2013:93-107.
10. Clarke E M, Mcmillan K L, Hartonas-Garmhausen V. Symbolic Model Checking. International Conference on Computer Aided Verification. Springer-Verlag, 1996:419-427.
11. Flocchini, Paola, Ilcinkas, et al. Computing Without Communicating: Ring Exploration by Asynchronous; Oblivious Robots. Algorithmica, 2013, 65(3):562-583.
12. Kamei S, Lamani A, Ooshita F, et al. Asynchronous Mobile Robot Gathering from Symmetric Configurations without Global Multiplicity Detection. Lecture Notes in Computer Science, 2011, 6796:150-161.
13. Flocchini P, Prencipe G, Santoro N, et al. Gathering of asynchronous robots with limited visibility. Theoretical Computer Science, 2005, 337(1):147-168.
14. Devismes S, Lamani A, Petit F, et al. Optimal Grid Exploration by Asynchronous Oblivious Robots. Symposium on Self-Stabilizing Systems. Springer Berlin Heidelberg, 2012:64-76.
15. Bonnet F, Défago X, Petit F, et al. Brief Announcement, Discovering and Assessing Fine-Grained Metrics in Robot Networks Protocols. Stabilization, Safety, and Security of Distributed Systems. Springer Berlin Heidelberg, 2012:282-284.
16. Rozier K Y, Vardi M Y. LTL satisfiability checking. International Journal on Software Tools for Technology Transfer, 2010, 12(2):123-137.

1. \* 基金项目: 国家自然科学基金(00000000, 00000000); 南京大学计算机软件新技术国家重点实验室开放课题(KFKT00000000)

   Foundation item: National Natural Science Foundation of China (00000000, 00000000); State Key Laboratory for Novel Software Technology (Nanjing University)开放课题 (KFKT00000000)

   收稿时间: 0000-00-00; 修改时间: 0000-00-00; 采用时间: 0000-00-00; jos在线出版时间: 0000-00-00

   CNKI在线出版时间: 0000-00-00 [↑](#footnote-ref-1)
2. 注：nuXmv主要针对同步系统的验证,不再支持关键字process.使用关键字process将调用nuXmv早期版本nuSmv的模型检测功能. [↑](#footnote-ref-2)