

# 基于变维度状态空间的增量启发式路径规划方法研究

张浩杰<sup>1,2</sup> 龚建伟<sup>1</sup> 姜岩<sup>1</sup> 熊光明<sup>1</sup> 陈慧岩<sup>1</sup>

**摘要** 在移动机器人路径规划中需要考虑运动几何约束, 同时, 由于它经常工作于动态、时变的环境中, 因此, 还必须保证路径规划算法的效率。本文提出了一种基于变维度状态空间的增量启发式路径规划方法, 该方法既能满足移动机器人的运动几何约束, 又能保证规划算法的效率。首先, 设计了变维度状态空间, 在机器人周围的局部区域考虑运动几何约束组织高维状态空间, 其他区域组织低维状态空间; 然后, 基于变维度状态空间, 提出了一种增量启发式路径规划方法, 该方法在新的规划进程中可以使用以前的规划结果, 仅对机器人周围的局部区域进行重搜索, 从而能保证算法的增量性及实时性; 最后, 通过仿真计算和机器人实验验证了算法的有效性。

**关键词** 变维度状态空间, 运动几何约束, 增量, 启发式路径规划

**引用格式** 张浩杰, 龚建伟, 姜岩, 熊光明, 陈慧岩. 基于变维度状态空间的增量启发式路径规划方法研究. 自动化学报, 2013, 39(10): 1602–1610

**DOI** 10.3724/SP.J.1004.2013.01602

## Research on Incremental Heuristic Path Planner with Variable Dimensional State Space

ZHANG Hao-Jie<sup>1,2</sup> GONG Jian-Wei<sup>1</sup> JIANG Yan<sup>1</sup> XIONG Guang-Ming<sup>1</sup> CHEN Hui-Yan<sup>1</sup>

**Abstract** Path planning with kinodynamic constraints is often required for mobile robots. Meanwhile, the robot often operates in dynamic and time-varying environments, so the effectiveness of the planner should be guaranteed. In this paper we present an incremental heuristic path planner based on variable dimensional state space. The proposed planner not only considers the robot's kinodynamic constraints into planning, but also guarantees the effectiveness. First, the variable dimensional state space is designed. The high-dimensional state space is organized around the robot and the low-dimensional state space elsewhere. Then, an incremental heuristic path planner is proposed based on the variable dimensional state space. The incremental and anytime properties of the planner are guaranteed by reusing previous computation. Finally, the effectiveness of the planner is confirmed by simulation and robot experiments.

**Key words** Variable dimensional state space, kinodynamic constraints, incremental, heuristic path planning

**Citation** Zhang Hao-Jie, Gong Jian-Wei, Jiang Yan, Xiong Guang-Ming, Chen Hui-Yan. Research on incremental heuristic path planner with variable dimensional state space. *Acta Automatica Sinica*, 2013, 39(10): 1602–1610

移动机器人是指具有一定的感知能力, 可以在有障碍物的环境中面向目标自主运动, 从而完成一定任务的自主系统。由于工作环境的复杂性, 对这些自主系统进行路径规划时, 必须考虑它们的运动几何约束。但是, 运动几何约束的引入将导致路径规划变为一个在高维状态空间进行路径搜索的问题, 状态空间维数的增加必然将导致路径规划时间和计算机内存需求的增加, 尤其是当环境尺寸增加时, 这个

收稿日期 2012-07-27 录用日期 2012-11-30

Manuscript received July 27, 2012; accepted November 30, 2012

国家自然科学基金(51275041, 90920304)资助

Supported by National Natural Science Foundation of China (51275041, 90920304)

本文责任编辑侯增广

Recommended by Associate Editor HOU Zeng-Guang

1. 北京理工大学机械与车辆学院智能车辆研究中心 北京 100081 2.

中国北方车辆研究所 北京 100072

1. Intelligent Vehicle Research Center, School of Mechanical Engineering, Beijing Institute of Technology, Beijing 100081 2. China North Vehicle Research Institute, Beijing 100072

问题尤其突出, 难以满足在线实时规划的需求。比如, 图1所示是采用Anytime repairing A\* (ARA\*) 算法<sup>[1]</sup>对随机生成的50幅1000×1000栅格大小的地图分别在2D ( $x, y$ ) 和3D ( $x, y, \theta$ ) 状态空间进行路径规划的结果, 图中横坐标表示启发值比例因子 $\varepsilon$ , 其值从1.0增长到3.0, 递增梯度为0.2。从图1(a)和图1(b)这两幅图显示的平均规划时间和平均扩展节点数目对比关系可以看出, 当状态空间的维数增加一维时, 相应的规划时间和扩展节点数目将大幅地增加。尤其当 $\varepsilon = 1.0$ , 这时ARA\* 算法将实施最优路径规划, 在这两个状态空间进行路径规划的结果相差最大。

目前, 在路径规划问题中考虑移动机器人运动几何约束的路径规划方法主要分为两类。一类是考虑机器人的运动几何约束而在高维状态空间进行路径规划, 这类方法通常只能获得次优路径<sup>[2–3]</sup> 或者

只能解决有限区域范围内的路径规划问题<sup>[4]</sup>. 另一类方法主要用于解决大尺寸环境下的路径规划问题, 这类规划方法通常将整个规划进程分两层进行: 一层是在 2D ( $x, y$ ) 状态空间进行全局规划获得全局路径; 另一层是在移动机器人周围的局部区域设计全局路径跟踪控制器或者是在高维状态空间进行局部路径规划<sup>[5-7]</sup>, 从而跟踪全局路径. 很显然, 由于在进行第一层规划时, 忽略了移动机器人的运动几何约束, 这种规划方法必然将降低规划时间, 使得在线实时规划成为可能. 然而, 由于这两层规划之间假设条件的不一致性, 这种分层的规划方法通常不具有完备性, 且容易导致机器人陷入局部极小点而无法到达目标点.

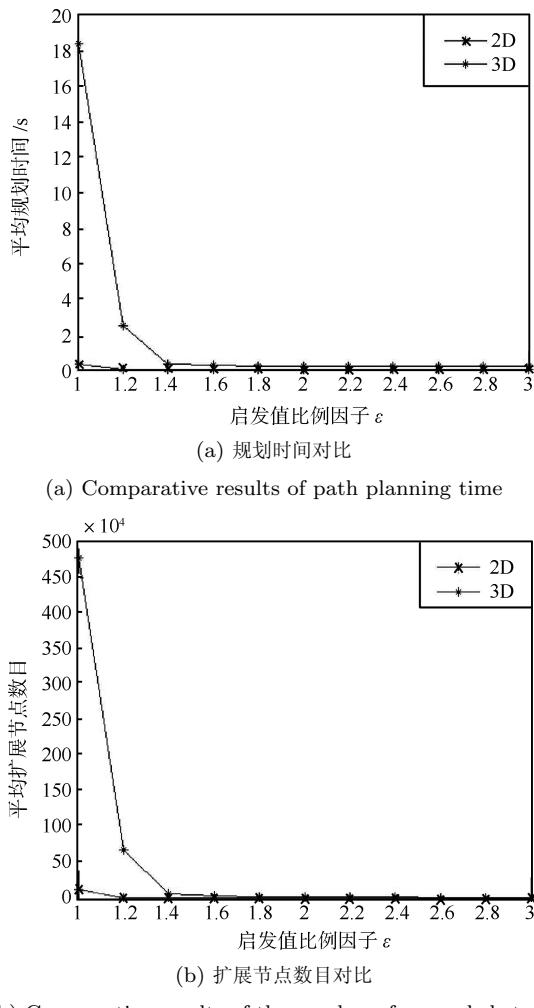


图 1 ARA\* 算法在 2D ( $x, y$ ) 和 3D ( $x, y, \theta$ ) 状态空间规划结果

Fig. 1 The path planning results with 2D ( $x, y$ ) and 3D ( $x, y, \theta$ ) state space using ARA\* algorithm

增量性可以有效地提高路径规划算法的效率, 所谓路径规划算法的增量性是指当环境改变时, 算

法可以在当前的路径搜索进程中利用以前进行路径规划时的搜索结果组织新的搜索, 而不必从起点到终点重新搜索路径, 从而达到加速当前的路径搜索<sup>[8]</sup>. 文献 [9-10] 提出了一种考虑移动机器人的运动几何约束, 将全局规划和局部规划融合在同一规划进程中的路径规划方法, 该方法通过设定一定的时间周期不断重启规划进程进行路径规划, 然而, 它不具有增量性. 在本文中, 我们扩展了这种规划方法, 通过不断地在地图中的关键点和移动机器人周围的局部区域引入高维状态空间, 合理地定义高、低维状态空间边界处的节点连接关系, 动态地组织变维度状态空间, 提出了一种基于变维度状态空间的增量启发式路径规划方法, 并保证所提出的路径规划方法具有完备性和增量性. 所提出的路径规划方法基于以下思想: 仅在移动机器人周围的局部区域需要组织高维状态空间进行高维规划, 从而保证所得路径的平滑性及可执行性, 而其他大部分区域可以组织低维状态空间进行低维规划, 从而保证算法的效率.

## 1 路径规划问题定义

### 1.1 符号说明

为了表述方便, 在本文中引入了一些符号作为标记. 这些符号的表示及其含义见表 1 所示.

表 1 符号说明

Table 1 Symbol description

| 符号                          | 含义        |
|-----------------------------|-----------|
| $\mathbf{R}^n$              | $n$ 维平面空间 |
| $S^n$                       | $n$ 维旋转空间 |
| $\mathcal{W}$               | 工作空间      |
| $G$                         | 状态空间      |
| $\mathcal{A}$               | 移动机器人模型   |
| $\mathcal{O}$               | 障碍物区域     |
| $\mathcal{C}$               | 移动机器人构型空间 |
| $\mathcal{C}_{\text{obs}}$  | 障碍物构型空间   |
| $\mathcal{C}_{\text{free}}$ | 自由空间      |

### 1.2 路径规划的状态空间表示

在移动机器人路径规划研究中, 机器人所有可能运动姿态的集合, 称之为机器人的构型空间, 以符号  $\mathcal{C}$  表示, 同样, 障碍物的构型空间以符号  $\mathcal{C}_{\text{obs}}$  表示. 由于机器人既可平移, 又可转动, 因此, 它共有 6 个自由度, 分别以位置量  $(x, y, z)$  和转动量  $(\phi, \theta, \psi)$  表示, 其中,  $\phi$ 、 $\theta$  和  $\psi$  分别为侧倾角、俯仰角和横摆角. 由于所研究的机器人经常工作于平坦地面, 因此, 可以对机器人的运动进行简化, 仅考虑机器人沿

$x, y$  方向的平移及绕  $z$  轴的横摆运动, 忽略沿垂直方向的位移、俯仰角和侧倾角, 即  $z = \phi = \theta = 0$ . 基于这一假设, 以符号  $v$  表示机器人的平移速度, 符号  $\omega$  表示其横摆角速度, 它应满足如下的运动学约束条件:

$$\begin{cases} \dot{y}\cos\theta - \dot{x}\sin\theta = 0 \\ \dot{x} = v\cos\theta \\ \dot{y} = v\sin\theta \end{cases} \quad (1)$$

如果以  $\mathbf{z} = [x, y, \theta]^T$  表示状态矢量,  $\mathbf{u} = [v, \omega]^T$  表示控制矢量, 可得机器人系统的状态转移方程表达式:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \cos\theta \\ \sin\theta \\ 0 \end{bmatrix} v + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \omega \quad (2)$$

由此可知,  $\mathcal{W} = \mathbf{R}^2$ ,  $\mathcal{A} \in \mathbf{R}^2$ . 因此, 机器人的构型空间可表示为

$$\mathcal{C} = \mathbf{R}^2 \times \mathcal{S}^1 \quad (3)$$

如果以  $q$  表示机器人在构型空间中的姿态, 而以  $\mathcal{A}(q) \in \mathcal{W}$  表示  $q$  投影到工作空间  $\mathcal{W}$  中的状态, 则障碍物的构型空间  $\mathcal{C}_{\text{obs}}$  和自由空间  $\mathcal{C}_{\text{free}}$  可分别以下列集合表示:

$$\mathcal{C}_{\text{obs}} = \{q \in \mathcal{C} | \mathcal{A}(q) \cap \mathcal{O} \neq \emptyset\} \quad (4)$$

$$\mathcal{C}_{\text{free}} = \mathcal{C} \setminus \mathcal{C}_{\text{obs}} \quad (5)$$

依据式 (1)~(5), 可以将路径规划问题抽象为给定移动机器人的模型  $\mathcal{A}$ 、障碍物区域  $\mathcal{O}$ 、机器人的构型空间  $\mathcal{C}$  及起始状态  $s_{\text{start}}$ 、终止状态  $s_{\text{goal}}$  ( $s_{\text{start}}, s_{\text{goal}} \in \mathcal{C}_{\text{free}}$ ), 利用相关的路径规划算法搜索获得一条最优路径  $\pi = \{s_0, s_1, \dots, s_i, \dots, s_k\}$ , 满足条件  $\forall s_i, s_i \in \mathcal{C}_{\text{free}}$ , 且  $s_0 = s_{\text{start}}, s_k = s_{\text{goal}}$ .

通常, 路径规划问题的求解可转换为在离散的状态空间搜索获得最优路径, 而状态空间则是离散状态及其状态之间连接关系的集合. 如果以四元组  $[S, E, S_b, G_b]$  表示路径规划问题的状态空间, 其中,  $S$  是状态空间中节点状态的集合, 而  $E$  是节点状态之间连接关系的集合,  $S_b$  ( $s_{\text{start}} \in S_b$ ) 和  $G_b$  ( $s_{\text{goal}} \in G_b$ ) 分别是路径规划的起始状态和终止状态的边界约束条件. 对于任意给定的两个节点  $s_i, s_j \in S$ , 为其分配节点之间的平移消耗为  $c(s_i, s_j)$ . 如果以  $c(\pi)$  表示路径消耗, 则满足机器人运动几何约束的路径规划问题可以降阶为在状态空

间中找到节点序列, 满足条件:

$$c(\pi) = \min \sum_{i=1}^k c(s_{i-1}, s_i) \quad (6)$$

## 2 变维度状态空间

### 2.1 变维度状态空间的总体思想

对于状态空间中的离散节点状态, 如果它考虑移动机器人的运动几何约束, 所组织的状态空间将变为一个高维状态空间, 状态空间维数的增加势必影响路径规划算法的效率; 相反, 如果不考虑移动机器人的运动几何约束, 比如仅考虑节点状态的位置  $(x, y)$  信息进行路径规划, 所获得的路径不便于执行. 因此, 为了既保证路径规划算法的效率, 又保证能够得到平滑、易于执行的路径, 本文提出了一种变维度状态空间. 变维度状态空间的基本思想是: 在机器人当前位置的局部区域组织高维状态空间进行高维规划, 其他区域组织低维状态空间进行低维规划, 如图 2 所示.

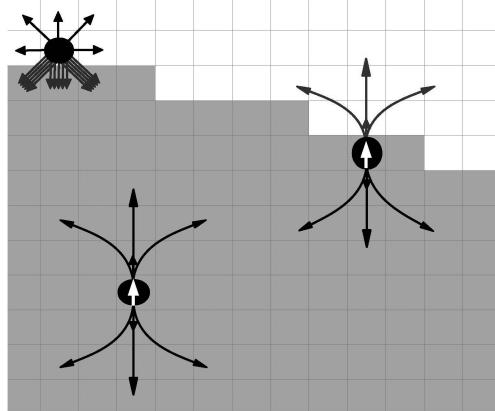


图 2 变维度状态空间

Fig. 2 Variable dimensional state space

在图 2 中, 浅灰色区域为高维状态空间, 白色区域为低维状态空间. 在高维状态空间中, 基于式 (2) 中表示的机器人的状态转移方程, 并采用 Howard 等提出的轨迹生成算法<sup>[11]</sup>, 可以确定节点之间的 Lattice  $(x, y, \theta)$  连接关系<sup>[12]</sup>. 这些节点连接关系可离线生成, 并用于生成机器人的基本运动单元. 在低维状态空间中, 节点连接关系由 8-连接栅格确定. 由于移动机器人在每次移动过程中仅仅执行全局路径的一小段, 而大部分路径尚未执行. 因此, 可以在移动机器人周围的局部区域考虑它的运动几何约束组织高维状态空间实施高维规划, 保证当前的可执行路径较为平滑且易于执行, 而其他大部分区域基于低维状态空间实施低维规划, 保证算法的

效率.

## 2.2 变维度状态空间的边界处理

在组织变维度状态空间时, 存在的难点是如何处理高、低维状态空间边界处的节点连接关系. 如果以  $G_H = (S_H, \varepsilon_H)$  和  $G_L = (S_L, \varepsilon_L)$  分别表示变维度状态空间中的高维状态空间和低维状态空间, 其中,  $S_H$  和  $S_L$  表示相应状态空间中节点的集合,  $\varepsilon_H$  和  $\varepsilon_L$  表示相应状态空间中节点之间连接关系的集合. 对于这两个不同维状态空间边界处的节点, 从高维状态空间中的节点到低维状态空间中的节点的匹配采用多对一的节点连接关系, 而从低维状态空间中的节点到高维状态空间中的节点的匹配采用一对多的节点连接关系, 如图 3 所示. 在图 3 中, 状态空间中的节点以小立方块表示, 低维状态空间  $G_L$  中的节点连接关系由 8-连接栅格确定, 而高维状态空间  $G_H$  中的节点连接关系由 Lattice 运动单元确定.

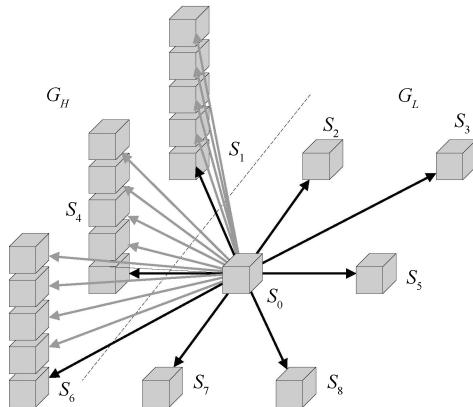


图 3 变维度状态空间边界处节点关系

Fig. 3 The connection relation between states on the edge of variable dimensional state space

**定义 1.** 函数  $\lambda$  表示将高维状态空间中的节点投影到低维状态空间中, 相应地,  $\lambda^{-1}$  表示将低维状态空间中的节点投影到高维状态空间中, 则对于任一节点  $s_h \in S_H$  及  $s_l \in S_L$ ,  $\lambda$  和  $\lambda^{-1}$  应满足以下条件:

$$\lambda(s_h) = s_l \quad (7)$$

$$\lambda^{-1}(s_l) = s_h \quad (8)$$

$$\lambda(\lambda^{-1}(s_l)) = s_l \quad (9)$$

$$\lambda^{-1}(\lambda(s_h)) \neq s_h \quad (10)$$

由式 (7) ~ (10) 可知, 映射函数  $\lambda$  是多对一的匹配, 而  $\lambda^{-1}$  是一对多的匹配.

对于高、低维状态空间边界处的节点, 建立一对多的节点连接关系. 比如, 在图 3 中,  $s_0 \in S_L$ , 它可通过 8-连接栅格与节点  $s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8$

建立连接, 如图 3 中黑色箭头所示, 其中,  $(s_2, s_3, s_5, s_7, s_8) \in S_L, s_1, s_4, s_6 \in S_H$ . 由于节点  $s_1, s_4, s_6$  属于高维状态空间, 因此, 对应于同一位置坐标应有多个节点, 如图 3 中同一位置叠加的多个小立方块所示, 这些叠加的节点具有相同的位置信息和不同的航向信息, 因此, 我们增加了节点  $s_0$  与这些同一位置坐标下的多个节点的额外连接关系, 灰色箭头连接表示在高、低维状态空间边界处建立的额外节点连接关系.

基于上述两个状态空间  $G_H, G_L$  及边界处节点连接关系的定义, 通过合并高维状态空间  $G_H$  和低维状态空间  $G_L$ , 组织了变维度状态空间  $G_{HL} = (S_{HL}, \varepsilon_{HL})$ , 其中,  $S_{HL}$  表示变维度状态空间中节点的集合,  $\varepsilon_{HL}$  表示变维度状态空间中的节点连接关系的集合. 在  $G_{HL}$  中, 高维状态空间由  $1, 2, \dots, N$  个以点  $(x_n, y_n)$  为重心的区域  $R_n$  组成. 当新的高维状态空间  $R_i$  引入时, 可通过使用映射函数  $\lambda^{-1}$  移除  $R_i$  中所有的低维状态, 替之以高维状态从而完成重新组织变维度状态空间.

## 3 基于变维度状态空间的增量式规划方法

### 3.1 Anytime dynamic A\* (AD\*) 算法

AD\* 算法是基于传统 A\* 算法的改进算法, 与 A\* 算法不同, 它在估价函数中对启发值增加了一个比例因子  $\varepsilon$  ( $\varepsilon \geq 1$ ), 其修正后的估价函数可以表示为

$$f(n) = g(n) + \varepsilon h(n) \quad (11)$$

式中,  $g(n)$  表示从起始节点到当前节点  $n$  的真实消耗, 而  $h(n)$  表示从当前节点  $n$  到目标节点的估算消耗, 也称为启发值.

AD\* 算法在进行路径搜索过程中一共维持了 OPEN 表、CLOSED 表和 INCONS 表三个表. OPEN 表中存储待被搜索的节点, CLOSED 表中存储已经被搜索过的节点, 而 INCONS 表中存储由于启发值比例因子  $\varepsilon$  改变或者节点之间平移消耗改变后导致出现的状态不一致节点 ( $v \neq g$ ). AD\* 算法搜索节点的先后顺序由节点的  $f$  值决定, 通过将搜索过程中出现的状态不一致的节点重新移至 OPEN 表中, 实现了算法的实时性和增量性, 关于该算法的理论分析参见文献 [13].

### 3.2 基于变维度状态空间的规划算法思想

高维状态空间随着机器人的移动而移动, 以符号  $G_{H1}$  和  $G_{H2}$  表示移动过程中旧、新的高维状态空间, 符号  $G_L$  表示低维状态空间, 符号  $G_F$  表示这两个高维状态空间的边界缓冲区. 因此, 在移动过程中必然导致两个高、低维状态空间边界处的部分节

点的状态空间所属关系发生改变,也就是说节点可能从属于低维状态空间变为属于高维状态空间,或者从属于高维状态空间变为属于低维状态空间,这时,需要重新组织变维度状态空间,其组织过程以图4表示。

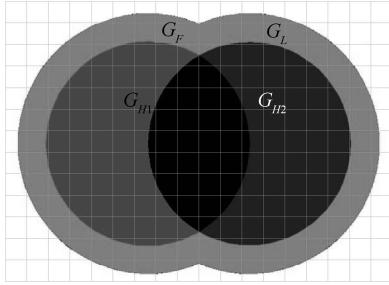


图4 机器人移动后的变维度状态空间重新组织

Fig. 4 Re-organization of variable dimensional state space after robot moved

在重新组织变维度状态空间的过程中会建立一些新的节点状态空间所属关系及节点连接关系,同时,这些节点间的平移消耗也会由于变维度状态空间的重新组织而发生变化。因此,路径规划算法必须意识到这些平移消耗的改变是由于传感器数据更新或者是变维度状态空间的重新组织造成的,从而针对不同的情况采用不同的方法重新建立节点的连接关系。对于那些在变维度状态空间重新组织过程中状态空间所属关系改变的节点,采用算法1中的`convert_vertex`函数更新这些节点在状态空间中的节点连接关系。函数`convert_vertex`仅适用于 $G_{H1}$ 、 $G_{H2}$ 及 $G_F$ 中的节点,而对于 $G_L$ 中的节点,它不做任何处理。

#### 算法 1. `convert_vertex`

- 1) for all  $s \in G_{H1} \vee G_{H2}$
- 2)  $g(s) = \infty, v(s) = \infty$
- 3) UpdateSetMembership( $s$ )
- 4) end for
- 5) for all  $s \in G_F (s \neq s_{start})$  do
- 6)  $g(s) = \infty$
- 7)  $bs(s) = \arg \min_{s' \in \text{succs}(s)} (v(s') + c(s', s))$
- 8)  $g(s) = v(bs(s)) + c(bs(s), s)$
- 9) UpdateSetMembership( $s$ )
- 10) end for
- 11) for all  $s \in G_F$  and  $v(s) \neq \infty$  do
- 12) for each  $s' \in \text{preds}(s)$
- 13) if  $s' \in G_{H1} \vee G_{H2}$  then
- 14)  $g(s') = \infty$
- 15)  $bs(s') = \arg \min_{s'' \in \text{succs}(s')} (v(s'') + c(s'', s'))$
- 16)  $g(s') = v(bs(s')) + c(bs(s'), s')$
- 17) UpdateSetMembership( $s'$ )
- 18) end if
- 19) end for

算法1的实施流程是:在移动机器人每次移动后,针对 $G_{H1}$ 和 $G_{H2}$ 中在上一次规划过程中生成的节点,`convert_vertex`函数将它们的 $g$ 和 $v$ 值赋初值为无穷大,并将这些节点从 $OPEN$ 表中移除,正如机器人从未探测过 $G_{H1}$ 和 $G_{H2}$ 区域。而对于 $G_F$ 中已经生成的节点 $s$ ,将其 $g$ 值赋初值为无穷大,并在 $s$ 的父节点的集合 $\text{succs}(s)$ 中重新寻找它当前的最好父节点,这个最好的父节点用符号 $bs(s)$ 表示,然后,根据节点 $s$ 的 $g(s)$ 和 $v(s)$ 之间的关系更新其在 $OPEN$ 表或者 $INCONS$ 表中的位置。对于 $G_F$ 中已经被扩展的节点 $s$ ,从它的子节点集合 $\text{preds}(s)$ 中找出属于 $G_{H1}$ 或者 $G_{H2}$ 的子节点 $s'$ ,将 $s'$ 的 $g$ 值赋初值为无穷大,并在 $s'$ 的父节点的集合 $\text{succs}(s')$ 中重新寻找它当前的最好父节点,同样把 $s'$ 的最好的父节点以符号 $bs(s')$ 表示,然后,依据 $g(s')$ 和 $v(s')$ 之间的关系更新 $s'$ 在 $OPEN$ 表或者 $INCONS$ 表中的位置。

`convert_vertex`函数仅在移动机器人周围的局部区域内对那些在重新组织变维度状态空间过程中受影响的节点进行处理,更新它们在 $OPEN$ 表或者 $INCONS$ 表中的位置关系,从而为下次规划过程做好准备。由于它对大部分低维状态空间中的节点并不做任何处理,因此,它并没有破坏上次规划完成时已经建立的搜索树,并可以利用之前的规划结果,仅对局部区域进行重搜索,实现算法的增量性,从而避免了每次遇到障碍物后都从起点状态到终点状态进行重新进行规划。

图5是使用`convert_vertex`函数在重新组织变维度状态空间过程中更新节点连接关系的一个简单例子。为了直观起见,变维度状态空间采用 $6 \times 8$ 个栅格大小。在这个例子中, $G_H$ 表示高维状态空间,如图5中的灰色长方形区域所示,它共有20个栅格点,以灰色的圆点表示,而节点之间的连接关系由Lattice运动单元确定。 $G_L$ 表示低维状态空间,它里面的栅格点用黑色小方块表示,而节点之间的连接关系由8-连接栅格确定。灰色和黑色箭头分别表示高维状态空间和低维状态空间中节点之间的连接关系。图5(a)中所示是使用后向Anytime dynamic A\* (AD\*)算法<sup>[12]</sup>在启发值比例因子 $\epsilon = 1.0$ 时,从目标状态 $s_g$ 搜索至起始状态 $s_r$ 所得的规划结果,白色空心节点是在这次搜索过程中被扩展的节点,实心节点是在这次搜索过程中还没有被扩展的节点,也就意味着这些节点还存储在 $OPEN$ 表中待扩展,所获得的路径用带网格图案的线条表示。接下来,高维状态空间 $G_H$ 移动后,变维度状态空间中共有20个栅格点改变了状态空间所属关系,10个栅格点从低维状态空间移入高维状态空间,同样,有

10个栅格点从高维状态空间移出到低维状态空间, 如图5(b)中带“ $\times$ ”符号的节点。对于那些改变状态空间所属关系及仍然在高维状态空间中的节点, 采用算法1中的`convert_vertex`函数删除它们在原先所属的状态空间中的节点连接关系, 比如, 已扩展的节点 $s_1$ 和 $s_2$ 在 $G_L$ 状态空间中的节点连接关系及 $G_H$ 状态空间中已经建立的节点连接关系将被删除, 如图5(b)中虚线箭头所示。同时, 由于 $s_1$ 和 $s_2$ 已移至 $G_H$ , 可通过低维状态空间到高维状态空间的映射函数 $\lambda^{-1}$ 获得它们相对应的高维状态, 并把它们以高维状态重新扩展。图5(c)所示是 $G_H$ 移动完成后重新组织变维度状态空间, 然后, 等待利用规划算法进行重新搜索。在这次规划过程中, 节点 $s_1$ 将使用 $G_H$ 中设定的节点连接关系重新被扩展。最后, 图5(d)所示是重新进行路径规划后的结果, 在图5(d)中同样以带图案的线条表示获得的新路径, 白色空心节点则表示在这次规划过程中被重新扩展的节点。从这个简单的例子可以看出, 在重新组织变维度状态空间的过程中, 所提出的`convert_vertex`函数可以实现重新建立状态空间所属关系改变的节点的连接关系, 而对 $G_L$ 中状态空间所属关系没有改变的节点不做任何处理。在机器人移动后使用`convert_vertex`函数重新建立节点及其之间的连接关系, 重新组织变维度状态空间, 为增量式规划算法的实施奠定了基础。

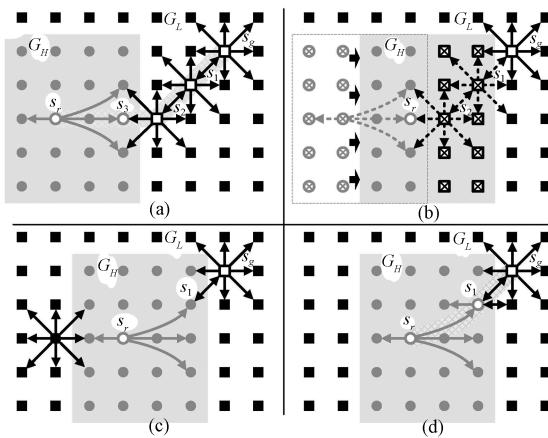


图5 重新组织节点连接关系

Fig. 5 Re-organization connection relation  
between states

### 3.3 基于变维度状态空间的规划算法实施

依据上述设计的变维度状态空间及机器人移动后变维度状态空间的重新组织方法, 可以使用任何基于地图的规划算法搜索变维度状态空间获得最优路径。在本文中我们以AD\*规划算法为基础, 提出了一种基于变维度状态空间的增量启发式路径规划方法, 如算法2所示。在算法2中,

`COMPUTEPATH`函数实施了AD\*算法进行路径规划(第4行), 随后, 当机器人移动后(第6行), 使用函数`convert_vertex`重新组织变维度状态空间(第7行)进行重新规划。算法重复执行上述流程, 直至机器人运动至目标状态。

### 算法2. 增量启发式路径规划方法

```

1) while  $s_{curr} \neq s_{goal}$  do
2)   update map with sensor data
3)    $i = i + 1$ 
4)    $\pi^i = \text{COMPUTEPATH}(s_{curr}, s_{goal}, G_{HL})$ 
5)   end while
6)    $s_{curr} = \pi^i(1)$ 
7)   convert_vertex( $G_{HL}$ )

```

## 4 实验结果

为了验证所提出的基于变维度状态空间的增量启发式路径规划方法的有效性, 我们随机生成了50幅地图进行仿真计算。同时, 基于Pioneer 3-AT轮式移动机器人搭建了实验平台, 设计了实验场景进行了规划方法的机器人验证实验。

### 4.1 仿真计算

在仿真计算阶段, 随机生成的50幅地图尺寸大小为 $2000 \times 2000$ 个栅格, 栅格分辨率为 $20\text{ cm} \times 20\text{ cm}$ 。设定机器人的尺寸占据 $4 \times 10$ 个栅格, 高维状态空间是以机器人为中心, 半径 $r = 50$ 个栅格大小的圆形区域。设定传感器的探测范围为以机器人为中心的 $30 \times 30$ 个栅格大小的方形区域, 该区域以外的环境未知。

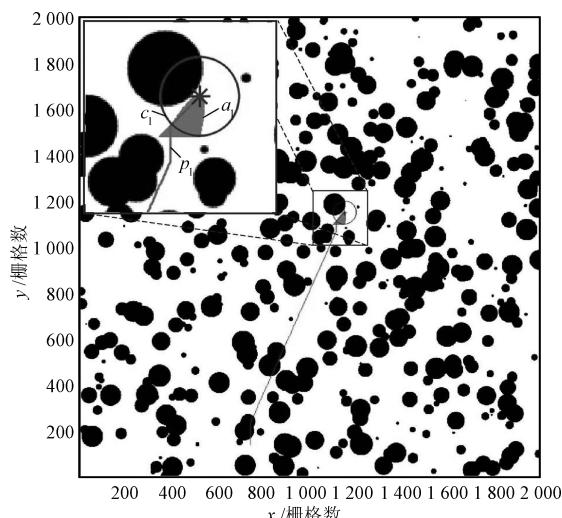
为了验证所提出的路径规划方法具有增量性和高效性, 设定AD\*算法的启发值比例因子 $\varepsilon = 1.0$ , 这时使用AD\*算法进行单次路径规划时, 它将变为A\*算法。因此, 我们分别采用A\*算法在3D $(x, y, \theta)$ 状态空间和本文所提出的规划方法在变维度状态空间 $(3D(x, y, \theta) \setminus 2D(x, y))$ 进行路径规划。整个仿真计算在装有Linux系统的计算机上运行, 该计算机的配置为Intel(R) Core(TM) i7 CPU, 主频为2.93 Hz, 仿真计算的结果见表2所示。从表2中可以看出, 所提出的基于变维度状态空间的增量启发式规划方法与A\*算法相比, 在整个路径规划过程中的规划时间及扩展的2D状态和3D状态数目的总和都有明显减少, 规划速度约是A\*算法的42倍, 而所得路径的真实消耗并没有增加太多。

图6所示是在这次仿真计算中机器人移动前、后两次规划时的场景及机器人最终行驶的路径, 由于在传感器探测范围以外的区域是未知的, 对于这些未知的区域, 在规划过程中假定为可通行区域, 因此, 图6(a)和(b)中出现路径穿越障碍物是正常的。

表 2 仿真计算结果  
Table 2 Simulation results

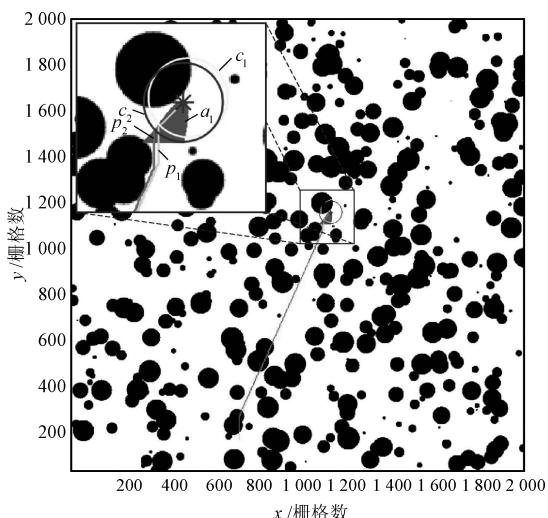
| 算法    | 初始规划时间 (s) | 平均规划时间 (s) | 2D 状态扩展数量 | 3D 状态扩展数量 | 路径消耗 (mm) |
|-------|------------|------------|-----------|-----------|-----------|
| 新算法   | 0.84       | 0.14       | 64 062    | 7 241     | 275 823   |
| A* 算法 | 35.56      | 5.87       | 0         | 2 328 883 | 257 108   |

图 6(a) 所示是机器人移动前的规划结果, “\*”表示它当前时刻的起始位置, 圆形区域  $c_1$  为高维状态空间, 其他区域为低维状态空间, 路径  $p_1$  表示获得的规划结果。由于这次的规划过程也是基于之前的规划结果, 因此, 在这次规划过程中仅对高维状态空间内的局部区域中的节点进行重新扩展, 如图 6(a) 中的区域  $a_1$  所示。当机器人移动一步至图 6(b) 所示的起始位置后, 基于变维度状态空间的增



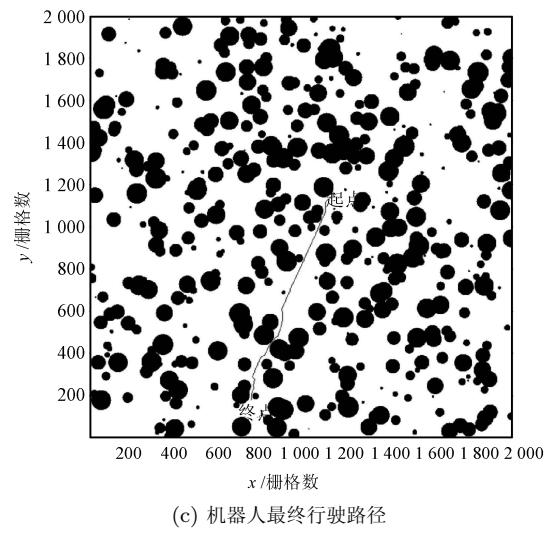
(a) 移动前规划结果

(a) The path planning results before moving



(b) 移动后规划结果

(b) The path planning results after moving



(c) 机器人最终行驶路径

(c) The path that the robot traveled finally

图 6 基于变维度状态空间的增量启发式规划算法规划结果

Fig. 6 The path planning results with variable dimensional state space using incremental heuristic search

量启发式规划方法仅对状态空间所属关系发生改变的节点进行更新, 从而重新建立了节点之间的连接关系。图 6(b) 中, 圆形区域  $c_1$  为上次规划时的高维状态空间, 圆形区域  $c_2$  为这次规划时的高维状态空间, 路径  $p_1$  表示上次规划获得的结果, 路径  $p_2$  表示当前时刻规划获得的结果, 从图 6(b) 中可以看出, 新的规划过程仅是对节点状态空间所属关系发生变化的局部区域进行了重搜索, 如图 6(b) 中区域  $a_1$  所示, 而对其他区域并没有进行重搜索, 这点正是体现了所提出的基于变维度状态空间的增量启发式规划方法的优越性, 它可以使用之前的规划结果, 从而保证算法的规划效率。

#### 4.2 机器人实验

为了验证所提出的路径规划方法在实际应用中的有效性, 我们搭建了机器人实验平台进行真实场景下的实验。在本次实验中, 实验平台采用 Pioneer 3-AT 机器人, 如图 7 所示。实验平台的硬件主要由笔记本电脑、UTM-30LX 激光雷达、遥控手柄和 Pioneer 3-AT 机器人组成, 实验平台的软件系统运行在 ROS (Robot operating system) 下。在实验中, 机器人的工作模式分为遥控模式和自主模式两种。在遥控模式下, 操作者可通过遥控手柄控制机器人运动, 而在自主模式下, 机器人需要根据已知和探测到的新的环境信息进行自主避障运动。

实验场景为北京理工大学车辆实验楼三层走廊, 在这个场景下, 首先, 在遥控模式下, 利用手柄驱动机器人四处运动, 结合离线的 SLAM 算法建立了环

境地图; 然后, 将该地图作为全局路径规划的先验地图, 在该地图上可以设定路径规划的起始状态和终止状态; 最后, 在自主模式下, 采用所提出的基于变维度状态空间的增量启发式路径规划方法进行路径规划。而在机器人执行规划所得路径的过程中, 人作为动态障碍物在走廊中移动, 从而测试基于变维度状态空间的增量启发式路径规划方法在动态未知环境下的路径修复能力, 实验结果如图 8 所示。



图 7 Pioneer 3-AT 实验平台  
Fig. 7 Pioneer 3-AT platform

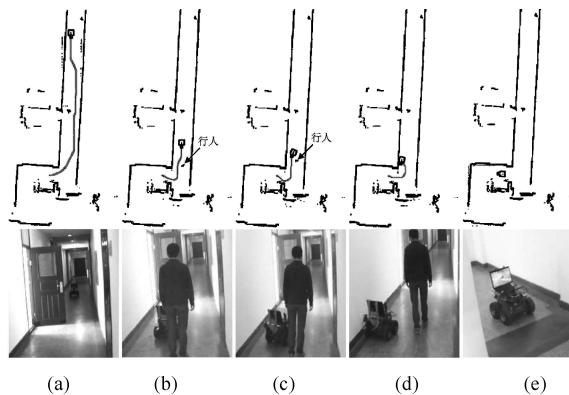


图 8 实验结果

Fig. 8 Experiment results

在这个真实场景的实验中, 我们设定机器人的局部地图更新范围为以它为中心  $20 \times 20$  个栅格大小的方形区域, 栅格的分辨率为  $20\text{ cm} \times 20\text{ cm}$ 。当动态障碍物出现在这个区域的时候更新全局地图; 否则, 不更新全局地图。图 8(a) 所示是最开始时的路径规划结果, 此时, 走廊中还没有动态障碍物出现。机器人跟踪在图 8(a) 中获得的路径, 随着机器人的逐渐移动, 当运动至图 8(b) 所示位置时, 这时人作为动态障碍物出现在机器人的探测范围后, 规划算法更新了全局地图并重新修复了路径。图 8(c) 和图 8(d) 所示为规划算法根据动态障碍物的位置变化不断地修复路径。而图 8(e) 显示了机器人最终到达了目标点。在整个规划过程

中, 规划算法的效率很高, 机器人不需要停下来等待生成新的路径, 它的运动连续并且未与动态障碍物碰撞, 更多关于这次实验的结果及视频可参看 <https://sites.google.com/site/haojie9527/video>。

## 5 结论

在路径规划中考虑移动机器人的运动几何约束的情况下, 为了保证规划算法在大尺寸环境下的执行效率, 本文提出了一种基于变维度状态空间的增量启发式路径规划方法。该方法在移动机器人周围的局部区域考虑机器人的运动几何约束组织高维状态空间, 其他区域仅考虑位置信息  $(x, y)$  组织低维状态空间, 通过在高、低维状态空间边界处设定一对多的节点连接关系, 组织了变维度状态空间。所提出的规划方法使用  $\text{AD}^*$  算法在设计的变维度状态空间进行路径规划, 并在机器人移动后重新组织变维度状态空间, 保证了规划方法的实时性和增量性。最后, 通过仿真计算和机器人实验证实了所提出的规划方法有效地提高了规划速度, 并且规划获得的路径易于机器人执行。

## References

- Likhachev M, Gordon G J, Thrun S. ARA\*: anytime  $A^*$  with provable bounds on sub-optimality. In: Proceedings of the 17th Annual Conference on Neural Information Processing Systems. New York, USA: IEEE, 2003
- Bekris K E, Kavraki L E. Greedy but safe replanning under kinodynamic constraints. In: Proceedings of the 2007 IEEE International Conference on Robotics and Automation. New York, USA: IEEE, 2007. 704–710
- Petti S, Fraichard T. Safe motion planning in dynamic environments. In: Proceedings of the 2005 IEEE/RSJ International Conference on Intelligent Robots and Systems. New York, USA: IEEE, 2005. 2210–2215
- Likhachev M, Ferguson D. Planning long dynamically feasible maneuvers for autonomous vehicles. *International Journal of Robotics Research*, 2009, **28**(8): 933–945
- Brock O, Khatib O. High-speed navigation using the global dynamic window approach. In: Proceedings of the 1999 IEEE International Conference on Robotics and Automation. New York, USA: IEEE, 1999. 341–346
- Kelly A J. An Intelligent, Predictive Control Approach to the High-speed Cross-country Autonomous Navigation Problem, Technical Report, CMU-RI-TR-95-33, Robotics Institute, Carnegie Mellon University, 1995
- Philippson R, Siegwart R. Smooth and efficient obstacle avoidance for a tour guide robot. In: Proceedings of the 2003 IEEE International Conference on Robotics and Automation. New York, USA: IEEE, 2003. 446–451
- Mills-Tettey G A, Stentz A, Bernardine Dias M. DD\* lite: efficient incremental search with state dominance. In: Proceedings of the 21st National Conference on Artificial Intelligence. Palo Alto, USA: AAAI, 2006. 1032–1038

- 9 Zhang H J, Chen H Y, Jiang Y, Gong J W, Xiong G M. Variable dimensional state space based global path planning for mobile robot. *Journal of Beijing Institute of Technology*, 2012, **21**(3): 328–335
- 10 Zhang H J, Butzke J, Likhachev M. Combining global and local planning with guarantees on completeness. In: Proceedings of the 2012 IEEE International Conference on Robotics and Automation. New York, USA: IEEE, 2012. 4500–4506
- 11 Howard T M, Kelly A. Optimal rough terrain trajectory generation for wheeled mobile robots. *International Journal of Robotics Research*, 2007, **26**(2): 141–166
- 12 Pitovraiko M, Knepper R A, Kelly A. Differentially constrained mobile robot motion planning in state lattices. *Journal of Field Robotics*, 2009, **26**(3): 308–333
- 13 Likhachev M, Ferguson D I, Gordon G J, Stentz A, Thrun S. Anytime dynamic A\*: an anytime, replanning algorithm. In: Proceedings of the 2005 International Conference on Automated Planning and Scheduling. California, USA: IEEE, 2005. 262–271



**张浩杰** 北京理工大学智能车辆研究所博士研究生。2008年获得中南大学交通设备信息工程系学士学位。主要研究方向为自主平台的路径规划技术。

E-mail: haojie.bit@gmail.com

**(ZHANG Hao-Jie)** Ph. D. candidate at the Intelligent Vehicle Research Center, Beijing Institute of Technology.

He received his bachelor degree from the School of Traffic and Transportation Engineering, Central South University in 2008. His main research interest is path planning for autonomous system.)



**龚建伟** 北京理工大学智能车辆研究所副研究员。2002年获得北京理工大学机械电子工程系博士学位。主要研究方向为智能车机器人的路径规划及控制技术。本文通信作者。

E-mail: gongjianwei@bit.edu.cn

**(GONG Jian-Wei)** Associate professor at the Intelligent Vehicle Research Center, Beijing Institute of Technology. He received

his Ph. D. degree from the School of Mechatronical Engineering, Beijing Institute of Technology in 2002. His research interest covers planning and control for autonomous mobile robots. Corresponding author of this paper.)

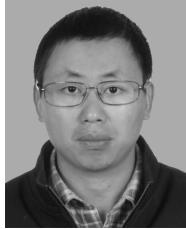


**姜岩** 北京理工大学智能车辆研究所讲师。2010年获北京航空航天大学车辆工程系博士学位。主要研究方向为智能车辆的路径跟踪、计算机视觉技术。

E-mail: piuzuzu@gmail.com

**(JIANG Yan)** Assistant research professor at the Intelligent Vehicle Research Center, Beijing Institute of Technology.

He received his Ph. D. degree from the School of Transportation Schence and Engineering, Beihang University in 2010. His research interest covers path following and computer vision for intelligent vehicles.)



**熊光明** 北京理工大学智能车辆研究所副教授。2005年获北京理工大学机械电子工程系博士学位。主要研究方向为智能车机器人的控制及视觉技术。

E-mail: xiongguangming@bit.edu.cn

**(XIONG Guang-Ming)** Associate professor at the Intelligent Vehicle Research Center, Beijing Institute of Technology.

He received his Ph. D. degree from the School of Mechatronical Engineering, Beijing Institute of Technology in 2005. His research interest covers autonomous mobile robots and computer vision.)



**陈慧岩** 北京理工大学智能车辆研究所教授。主要研究方向为智能车辆技术、车辆信息技术。

E-mail: Chen.h.y@bit.edu.cn

**(CHEN Hui-Yan)** Professor at the Intelligent Vehicle Research Center, Beijing Institute of Technology. His research interest covers intelligent vehicle and information technologies.)