



(12)发明专利申请

(10)申请公布号 CN 111433697 A

(43)申请公布日 2020.07.17

(21)申请号 201880071257.2

(74)专利代理机构 广州嘉权专利商标事务所有
限公司 44205

(22)申请日 2018.08.31

代理人 黄晓升

(30)优先权数据

102017120218.8 2017.09.01 DE

(51)Int.Cl.

G05D 1/02(2020.01)

(85)PCT国际申请进入国家阶段日

2020.04.30

(86)PCT国际申请的申请数据

PCT/EP2018/073497 2018.08.31

(87)PCT国际申请的公布数据

W02019/043171 DE 2019.03.07

(71)申请人 罗博艾特有限责任公司

地址 奥地利林茨

(72)发明人 R·福格尔 H·阿特斯

C·弗罗伊登塔勒 I·克尼特尔

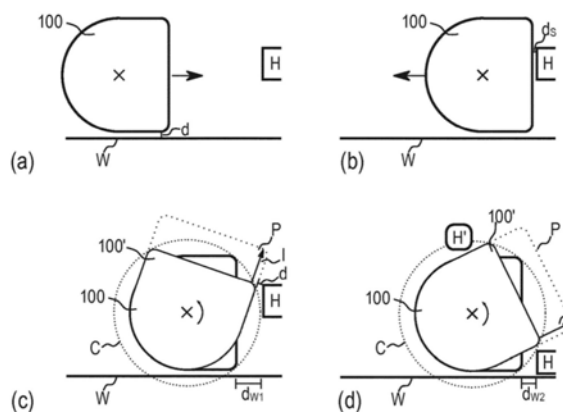
权利要求书6页 说明书27页 附图8页

(54)发明名称

用于自主移动机器人的运动规划

(57)摘要

描述了一种用于控制自主移动机器人的方法,该自主移动机器人可以在第一轮廓跟随模式和至少一个第二轮廓跟随模式下工作,其中,在每个轮廓跟随模式下,在机器人沿着轮廓运动期间,机器人与该轮廓之间保持基本上恒定的距离。根据一个示例性实施例,该方法包括以下步骤:启动第一轮廓跟随模式,其中,机器人沿着第一行驶方向跟随轮廓;检测死胡同状况,在所述死胡同状况下,在不发生碰撞的情况下在第一轮廓跟随模式下继续跟随所述轮廓是不可能的;启动第二轮廓跟随模式,其中,机器人沿着第二行驶方向跟随轮廓;以及确定结束第二轮廓跟随模式所需满足的标准,以及在机器人在第二轮廓跟随模式下工作期间连续地评估所述标准。



1. 一种用于控制自主移动机器人的方法,所述自主移动机器人可以在第一轮廓跟随模式和至少第二轮廓跟随模式下工作,其中,在每个轮廓跟随模式下,在所述机器人(100)沿着轮廓(W,V)运动期间,所述机器人与所述轮廓(W,V)之间保持基本上恒定的距离;所述方法包括以下步骤:

启动所述第一轮廓跟随模式,其中,所述机器人沿着第一行驶方向跟随所述轮廓(W,V);

检测死胡同状况,其中在所述死胡同状况下,在不发生碰撞的情况下在第一轮廓跟随模式下继续跟随所述轮廓(W,V)是不可能的;

启动第二轮廓跟随模式,其中,所述机器人沿着第二行驶方向跟随所述轮廓(W,V);以及

确定结束第二轮廓跟随模式所需满足的标准,,并且在所述机器人在第二轮廓跟随模式下工作期间连续地评估所述标准。

2. 根据权利要求1所述的方法,

其中,所述轮廓跟随模式的特征在于至少两个参数,其中,所述至少两个参数包括行驶方向、轮廓跟随距离(d)以及可选地以下参数之一:所述机器人的面向所述轮廓的侧面、安全距离、为了识别即将发生的碰撞而被考虑的机器人形状、沿着轮廓进行运动所依据的规则,以及

其中,两个不同的修正跟随模式的区别在于至少一个参数。

3. 根据权利要求1或者2所述的方法,其中,所述检测死胡同状况包括:

检测,在不发生碰撞的情况下所述机器人(100)沿着所述轮廓(W)的运动以及机器人(100)的旋转是不可能的,其中,在所述检测中特别考虑存储在所述机器人的地图中的、与位置相关的信息。

4. 根据权利要求1至3中的任一项所述的方法,

其中,如果在第二轮廓跟随模式下再次检测到死胡同状况,那么启动第三轮廓跟随模式,并且

其中,确定结束第三轮廓跟随模式所需满足的标准,并且在所述机器人在第三轮廓跟随模式下工作期间连续地评估这个标准。

5. 根据权利要求4所述的方法,

其中,所述第三轮廓跟随模式与所述第二轮廓跟随模式的区别在于以下参数:所述机器人的面对所述轮廓(W)的侧面。

6. 根据权利要求1至5中的任一项所述的方法,

其中,如果所述第二轮廓跟随模式由于满足确定的标准而被结束,那么继续进行所述第一轮廓跟随模式。

7. 根据权利要求1至6中的任一项所述的方法,其中,所述轮廓(W)通过虚拟的障碍物形成,所述虚拟的障碍物实际上是不存在的,但是包含在所述机器人的地图中。

8. 根据权利要求1至7中的任一项所述的方法,

其中,所述第二轮廓跟随模式结束所满足的标准包含执行特定的运动。

9. 根据权利要求8所述的方法,其中,所述特定的运动包括以下运动中的至少一项:

旋转一定角度以及

平移运动一定距离,尤其沿着向前方向平移运动一定距离。

10.根据权利要求1至9中的任一项所述的方法,其中,所述评估所述标准包括以的内容:

根据可预先确定的规则自动地规划不发生碰撞的机器人运动;

执行规划的机器人运动;

检查,是否可以不发生碰撞地执行规划的机器人运动。

11.根据权利要求10所述的方法,

其中,所述不发生碰撞的机器人运动的规划考虑了存储在所述机器人地图中的、与位置相关的、涉及到障碍物的信息。

12.根据权利要求10或者11所述的方法,

其中,根据可预先确定的规则自动地规划机器人运动包括以下内容:

规划旋转和随后的平移运动,使得在执行所述运动之后,障碍物的一个点与所述机器人之间相距特定的距离,特别是相距轮廓跟随距离。

13.根据权利要求12所述的方法,

其中,以大于一个可预先确定的最小角度的角度进行所述旋转。

14.根据权利要求1至13中的任一项所述的方法,

其中,所述确定结束所述第二轮廓跟随模式所需满足的标准,或者评估所述标准,是在考虑存储在所述机器人的地图中的、与位置有关的信息的情况下进行的。

15.根据权利要求1至14中的任一项所述的方法,

其中,在执行所述第二轮廓跟随模式期间更新结束所述第二轮廓跟随模式所需满足的标准。

16.一种用于在轮廓跟随模式下控制自主移动机器人的方法,在所述轮廓跟随模式下,所述机器人(100)以轮廓跟随距离基本上跟随轮廓(W,V);所述方法在所述轮廓跟随模式下包括:

根据至少一项可预先确定的标准评估至少三个不同的基本运动,以及

基于其评估结果执行所述三个基本运动之一,

其中,所述三个基本运动中的第一基本运动是所述机器人(100)的纯平移运动,

其中,所述三个基本运动中的第二基本运动包含所述机器人(100)朝向所述轮廓(W)的旋转,以及

其中,所述三个基本运动中的第三基本运动包含所述机器人(100)远离所述轮廓(W)的旋转。

17.根据权利要求16所述的方法,

其中,在对所述基本运动中的至少两个基本运动进行相同评估的情况下,选择将所述机器人引导至更接近所述轮廓(W)或者远离所述轮廓(W)较少距离的那个基本运动。

18.根据权利要求16或者17所述的方法,其中,在评估一个基本运动时,考虑在此之前执行的基本运动。

19.根据权利要求18所述的方法,其中,在评估时考虑,在执行所述第二基本运动之后,不应选择所述第三基本运动,并且反之亦然。

20.根据权利要求16至19中的任一项所述的方法,

其中,对所述基本运动的评估考虑以下标准中的至少一个标准:

所述基本运动能够不与障碍物(H)发生碰撞;

在所述运动期间和/或之后所述机器人到障碍物(H)的距离;和

在执行相应的基本运动之后,另一个基本运动例如平移运动不发生碰撞的可能性。

21.根据权利要求20所述的方法,其中,所述障碍物(H)可以是不同类型的,并且在所述评估中考虑所述障碍物(H)的类型。

22.根据权利要求21所述的方法,其中,第一类型的障碍物(H)包括由所述机器人的传感器单元(120)检测到的障碍物(H),并且第二类型的障碍物是实际上不存在的但是包含在所述机器人的地图中的虚拟的障碍物。

23.根据权利要求16至22中的任一项所述的方法,其中,所述第二基本运动和第三基本运动包括在静止状态下的旋转。

24.根据权利要求16至23中的任一项所述的方法,所述方法进一步包括:

检测,根据可预先确定的标准,所述三个基本运动均不能执行,

其中,如果已经检测出所述三个基本运动均不能执行,那么所述机器人更改行驶方向和/或所述机器人的面对轮廓的侧面和/或所述评估标准。

25.根据权利要求24所述的方法,

其中,所述机器人(100)具有优选的行驶方向,并且在将所述行驶方向更改到与所述优选的行驶方向相反的行驶方向上时,确定一个评估标准,在满足所述评估标准的情况下,再次更换到所述优选的行驶方向。

26.根据权利要求16至25中的任一项所述的方法,所述方法进一步包括:

其中,所述至少三个基本运动由多个参数限定,并且所述参数借助于优化方法,尤其是机器学习方法来确定。

27.根据权利要求26所述的方法,其中,借助于机器学习方法,至少部分自动地确定所述参数,使得所述机器人在某些可预先确定的情况下执行期望的、可预先确定的运动模式。

28.一种用于利用机器人使用区域的第一地图来控制自主移动机器人(100)的方法,其中,所述第一地图至少包含关于障碍物(H)的位置的数据;所述方法包括:

在假定所述机器人(100)的简化的虚拟形状(101)下,在所述第一地图中规划到达目标点的路径(P);

使所述机器人沿着规划的路径(P)运动;

在沿着所述规划的路径(P)运动期间,借助于所述机器人的传感器单元(120)探测在所述机器人(100)环境中的障碍物(H),

在关注实际的机器人形状下确定,由于障碍物(H)而不能够在不发生碰撞的情况下行驶所述规划的路径(P),

在关注实际的机器人形状下,继续所述机器人(100)的运动。

29.根据权利要求28所述的方法,

其中,所述机器人(100)的简化的虚拟形状(101)由一个圆来表示,所述机器人(100)可以围绕其中心旋转,而所述中心不运动。

30.根据权利要求29所述的方法,

其中,选择所述圆的半径,使得当所述机器人(100)围绕其中心旋转时,所述机器人

(100)的外轮廓的至少两个点在所述圆上运动,例如使得所述圆的半径对应于所述机器人的一半宽度。

31.根据权利要求30所述的方法,

其中,所述机器人的至少一部分位于所述圆的外部。

32.根据权利要求28至31中的任一项所述的方法,

其中,在所述机器人(100)沿着所述规划的路径运动期间,将探测到的障碍物(H)记录到第二地图中,其中,障碍物的位置和/或范围的准确性在所述第二地图中比在所述第一地图中更大。

33.根据权利要求32所述的方法,

其中,确定由于障碍物而不能在不发生碰撞的情况下行驶所述规划的路径(P),以及述继续所述机器人(100)的运动是以所述第二地图为基础的。

34.根据权利要求28至33中的任一项所述的方法,其中,在假定所述机器人(100)的简化的虚拟形状(101)下,在所述第一地图中规划到达目标点的路径(P)包括以下内容:

在假设所述机器人(100)的简化的第一虚拟形状(101)下,规划到达所述目标点的不发生碰撞的路径(P),其中所述简化的第一虚拟形状完全包含所述机器人(100)。

35.根据权利要求34所述的方法,其中,在假定所述机器人(100)的简化的虚拟形状(101)下,在所述第一地图中规划到达目标点的路径(P)进一步包括:

如果在假设第一虚拟形状下的路径规划中没有找到不发生碰撞的路径,那么在假设所述机器人(100)的第二虚拟形状下,重新规划到达所述目标点的路径(P),其中,所述机器人的第二虚拟形状不完全包括所述机器人。

36.根据权利要求34所述的方法,

其中,所述机器人(100)的第一虚拟形状是圆形形状。

37.根据权利要求35所述的方法,

其中,所述机器人(100)的第一虚拟形状和第二虚拟形状都具有圆形形状和/或所述第一虚拟形状对应于刚好还围住所述机器人并且所述机器人可以围绕其中心(在静止状态下)旋转的最小的圆。

38.根据权利要求28至33中的任一项所述的方法,其中,在假定所述机器人(100)的简化的虚拟形状(101)下,在所述第一地图中规划到达目标点的路径(P)包括以下内容:

在假设所述机器人的第一虚拟形状(100)下,规划到达所述目标点的第一路径,

在假设所述机器人(100)的至少一个与所述第一虚拟形状不同的第二虚拟形状下,规划到达所述目标点的至少一个第二路径,

根据可预先确定的标准评估所述第一路径和所述至少一个第二路径,

基于所述评估,从所述第一路径和所述至少一个第二路径中选择出用于后续的机器人运动的路径(P)。

39.根据权利要求28至38中的任一项所述的方法,其中,在假定所述机器人(100)的简化的虚拟形状(101)下,在所述第一地图中规划到达目标点的路径(P)包括以下内容:

分配用于行驶所述机器人使用区域的区域的成本函数值;和

针对所述机器人(100)的简化的虚拟形状(101)确定最低成本的路径,

其中,在所述成本函数值中包括所述路径到障碍物之间的距离。

40. 根据权利要求39所述的方法, 其中, 特别地, 所述成本取决于障碍物对所述机器人的旋转自由度的限制。

41. 根据权利要求39或者40所述的方法,

其中, 基于包含在所述第一地图中的信息来计算所述成本函数值并且其中, 特别地, 所述虚拟形状是不完全包括所述机器人的圆形形状。

42. 根据权利要求28至41中的一项所述的方法, 其中, 在关注实际的机器人形状下, 继续所述机器人(100)的运动包括:

确定在所述规划的路径(P)上的中间目标点;

跟随所述障碍物的轮廓, 直到在所述机器人和所述中间目标点之间的路径是畅通的或者终止条件被满足为止。

43. 根据权利要求42所述的方法,

其中, 所述终止条件考虑了所述机器人与所述规划的路径(P)和/或与所述中间目标点的距离, 并且

在满足所述终止条件之后, 所述机器人(100)从其当前的位置出发朝向所述目标点规划新的路径。

44. 根据权利要求43所述的方法,

在满足所述终止条件之后, 所述机器人(100)使计数器增加, 所述计数器对失败尝试的次数进行计数。

45. 一种用于借助于机器人使用区域的地图来控制自主移动机器人(100)的方法, 其中, 所述地图至少包含关于借助于传感器单元(120)识别的真实障碍物的位置的信息以及关于虚拟障碍物的信息; 所述方法包括:

在所述真实障碍物的附近中控制所述机器人(100), 使得避免与所述真实障碍物的碰撞, 其中, 考虑所述机器人(100)的实际形状, 并且

在所述虚拟障碍物的附近中控制所述机器人(100), 使得避免与所述虚拟障碍物的虚拟碰撞, 其中, 考虑所述机器人(100)的简化的虚拟形状(101)。

46. 根据权利要求45所述的方法,

其中, 所述机器人(100)的实际形状的一些部分位于所述机器人(100)的简化的虚拟形状(101)的外部。

47. 根据权利要求45或者46所述的方法,

其中, 所述机器人(100)的简化的虚拟形状(101)是一个圆, 所述机器人可以绕圆的中心旋转, 而所述中心不运动。

48. 根据权利要求47所述的方法,

其中, 选择所述圆的半径, 使得当所述机器人(100)围绕其中心旋转时, 所述机器人(100)的外轮廓的至少两个点在所述圆上运动, 例如使得所述圆的半径对应于所述机器人的一半宽度。

49. 根据权利要求45至48中的任一项所述的方法,

其中, 所述机器人(100)的简化的虚拟形状(101)使得所述机器人能够在相对于障碍物的任何可能的不发生碰撞的位置上不发生碰撞地旋转。

50. 一种用于在轮廓跟随模式下控制自主移动机器人的方法, 在所述轮廓跟随模式下,

所述机器人(100)以轮廓跟随距离基本上跟随轮廓(W,V),

其中,所述机器人的地图至少包含关于借助于传感器单元(120)识别的真实障碍物的位置的信息以及关于虚拟障碍物的信息,并且所述机器人在这个地图上连续地确定其位置;

其中,在所述轮廓跟随模式下,所述机器人(100)沿着轮廓(W,V)运动;

其中,所述轮廓(W,V)通过真实障碍物的走向和虚拟障碍物的虚拟边界的走向给出。

51.根据权利要求50所述的方法,

其中,基于存储在所述地图中的信息来确定在所述机器人与所述轮廓(W,V)之间的距离。

52.根据权利要求50或者51所述的方法,其中,所述轮廓跟随距离取决于正在被跟随的轮廓是否代表虚拟障碍物。

53.一种用于自主移动机器人的控制单元,所述控制单元被设计用于执行根据权利要求1至52中的任一项所述的方法。

用于自主移动机器人的运动规划

技术领域

[0001] 本描述涉及自主移动机器人的领域,尤其是涉及一般形状的自主移动机器人的运动的规划和实施。

背景技术

[0002] 近年来,自主移动机器人,特别是服务型机器人,越来越多地应用于家庭领域,例如用于清洁或者用于监视住宅。机器人通常具有圆形的形状和驱动单元,该驱动单元使它们能够围绕垂直轴线旋转。这极大地简化了对这些机器人的路径规划(轨迹规划)和控制,因为它们旋转的自由度从不受相邻的障碍物限制。

[0003] 由于对机器人的功能的特殊要求,偏离圆形的(圆盘形的)形状可能是值得期望的。例如可以使机器人的通常为圆形的形状在一侧上变平,使得机器人可以通过该平的侧面与墙壁平行来沿着该墙壁行驶。在该平的侧面上例如可以安装清洁单元(例如刷子),以便可以尽可能靠近墙壁地对该清洁单元进行引导。同样,由于其他的原因,偏离机器人的圆形的结构形状可能是有必要的或者是值得期望的。

[0004] 相对于机器人的基面为非圆形的结构形状可能导致机器人不能够在每种情况下就地转动,即使它的驱动单元原则上允许这样做。如在上面所提的示例中那样,如果机器人通过其平的侧面非常靠近障碍物(例如墙壁)站立,那么机器人将不再能够围绕其垂直轴线任意地旋转,而不与障碍物发生碰撞。为了规划和评估机器人的运动可能性,因此,除了障碍物和机器人在机器人使用区域中的位置之外,还必须考虑机器人的定向。解决该问题的一种方法在于将标准运动模式用于预先确定的情况。但是,这种方法是不灵活的并且容易出错。此外,很难预测自动运动机器人可能进入的所有可能的情况。另一种方法是精确地规划机器人的运动,即位置和定向的变化(其合起来也被称为姿态),以便从起点到达目标点。但是,这比对于圆形的机器人来说要复杂得多,由此,一方面,在实施中的容易出错性增大,而且用于必要的计算的资源消耗(计算时间、处理器性能、内存要求)增加。

[0005] 发明人已经为自己设定了任务,即实现针对任何形状的自主移动机器人的运动的简单而稳健的规划。

发明内容

[0006] 上述任务通过根据权利要求1、16、28、45和50的方法来解决。各种不同的示例性实施例和扩展方案是从属权利要求的主题。

[0007] 描述了一种用于控制自主移动机器人的方法,所述自主移动机器人可以在第一轮廓跟随模式和至少第二轮廓跟随模式下工作,其中,在每个轮廓跟随模式下,在所述机器人沿着轮廓运动期间,所述机器人与所述轮廓之间保持基本上恒定的距离。根据一个示例性实施例,所述方法包括以下步骤:启动所述第一轮廓跟随模式,其中,所述机器人沿着第一行驶方向跟随所述轮廓;检测死胡同状况,在所述死胡同状况下,在不发生碰撞的情况下在第一轮廓跟随模式下继续跟随所述轮廓是不可能的;启动第二轮廓跟随模式,其中,所述机

机器人沿着第二行驶方向跟随所述轮廓;以及确定结束所述第二轮廓跟随模式所需满足的标准,以及在所述机器人在第二轮廓跟随模式下工作期间连续地评估所述标准。

[0008] 此外,描述了一种用于在轮廓跟随模式下控制自主移动机器人的方法,在所述轮廓跟随模式下,所述机器人以轮廓跟随距离基本上跟随轮廓。根据一个示例性实施例,所述方法在所述轮廓跟随模式下包括以下步骤:根据至少一项可预先确定的标准评估至少三个不同的基本运动,以及基于其评估结果执行所述三个基本运动之一。所述三个基本运动中的第一基本运动是所述机器人的纯平移运动,所述三个基本运动中的第二基本运动包含所述机器人朝向所述轮廓的旋转,以及所述三个基本运动中的第三基本运动包含所述机器人远离所述轮廓的旋转。

[0009] 此外,描述了一种用于控制自主移动机器人的方法,所述自主移动机器人具有机器人使用区域的第一地图,其中,所述第一地图至少包含关于障碍物的位置的数据。根据一个示例性实施例,所述方法包括:在假定所述机器人的简化的虚拟形状下,在所述第一地图中规划到达目标点的路径。在一些示例性实施例中,所述方法可以进一步包括:使所述机器人沿着规划的路径运动,在所述机器人沿着所述规划的路径运动期间,借助于所述机器人的传感器单元探测在所述机器人环境中的障碍物,并且最后在关注实际的机器人形状下确定,由于障碍物而不能在不发生碰撞的情况下行驶所述规划的路径,以及在关注实际的机器人形状下,继续所述机器人的运动。

[0010] 此外,描述一种用于借助于机器人使用区域的地图来控制自主移动机器人的方法,其中,所述地图至少包含关于借助于传感器单元识别的真实障碍物的位置的信息以及关于虚拟障碍物的信息。根据一个示例性实施例,所述方法包括:在所述真实障碍物的附近中这样地控制所述机器人,即避免与所述真实障碍物的碰撞,其中,考虑所述机器人的实际形状,并且在所述虚拟障碍物的附近中这样地控制所述机器人,即避免与所述虚拟障碍物的碰撞,其中,考虑所述机器人的简化的虚拟形状。

[0011] 此外,描述了一种用于在轮廓跟随模式下控制自主移动机器人的方法,在所述轮廓跟随模式下,所述机器人以轮廓跟随距离基本上跟随轮廓。所述机器人的地图至少包含关于借助于传感器单元识别的真实障碍物的位置的信息以及关于虚拟障碍物的信息。所述机器人在这个地图上连续地确定其位置,其中,在所述轮廓跟随模式下,所述机器人沿着轮廓运动并且所述轮廓通过真实障碍物的走向和虚拟障碍物的虚拟边界的走向给出。

附图说明

[0012] 下面借助于附图更详细地解释不同的示例性实施例。这些图示不一定是按正确比例绘制的,并且本发明不仅仅限于所示出的方面。相反,重要的是要说明作为基础的原理。在附图中显示:

[0013] 图1示出了自主移动机器人的两个示例,在该机器人中分别有一侧面被变平,使得机器人可以通过其平的侧面非常靠近地沿着障碍物,例如墙壁运动。

[0014] 图2借助于框图示例性地示出了自主移动机器人的结构。

[0015] 图3显示了用于自主移动机器人的外壳形状的不同变型方案并且说明了外壳形状对使机器人运动的可能性的影响。

[0016] 图4借助于流程图示出了一种用于在死胡同状况下控制自主移动机器人的方法。

[0017] 图5借助于四个示意示意图(a)至(d)示出了一种用于在死胡同状况下控制自主移动机器人的示例性的方式。

[0018] 图6示例性地示出了一种用于在死胡同状况下控制自主移动机器人的方法。

[0019] 图7示出了一种用于在更复杂的死胡同状况下控制自主移动机器人的方法的另一个更复杂的示例。

[0020] 图8示例性地示出了各种不同的基本运动。

[0021] 图9借助于简单的示例示出了基本运动的选择。

[0022] 图10示出了基于环境的基本运动的运动。

[0023] 图11示出了与虚拟障碍物的轮廓跟随行驶。

[0024] 图12示出了在障碍物之间的圆形机器人的路径规划;这相当于针对在障碍物之间的一个点的路径规划,这些障碍物分别已经以机器人的半径被增大。

[0025] 图13示出了基于成本的路径规划的示例。

具体实施方式

[0026] 由于对机器人100的功能的特殊要求,与机器人的基本上圆形的、圆盘形的外壳形状相偏离可能是值得期望的或者是必要的。为此图1示出了两个示例。在图1中的示意图(a)示出了用于清洁地板表面的自主移动机器人100(清洁机器人)。机器人外壳的一个侧面被变平,使得机器人100可以通过该平的侧面与墙壁W平行地定向。在图1中的示意图(b)示出了具有自主移动机器人100的另一个示例,该自主移动机器人用于借助于平台来运输物体(服务型机器人),所述平台可以被运动到与工作台T或者工作表面的边缘对齐。所示出的示例清楚地表明,这样设计成的机器人不能够在每种情况下就地围绕其垂直轴线旋转,即使其驱动单元在原理上允许这样地做。在图1中所示的情况下,机器人100不能够在不与障碍物(墙壁W、工作台T)发生碰撞下而旋转。该事实对机器人100的运动规划有影响,因为针对从在机器人使用区域中的一个起点到一个目标点的轨迹的规划,必须附加地考虑机器人的定向。

[0027] 在更详细地讨论用于自主移动机器人的运动规划之前,首先要简要地描述自主移动机器人的结构。图2借助于框图示例性地示出了自主移动机器人100的各种不同的单元(模块),单元或者模块在这种情况下可以是用于控制机器人的独立的组件或者软件的一部分。一个单元可以具有几个子单元。负责机器人100的行为的软件可以由机器人100的控制单元150来执行。在所示的示例中,控制单元150包括处理器155,该处理器被设计用于执行包含在存储器156中的软件指令。控制单元150的一些功能也可以至少部分地借助于外部计算机来执行。这意味着控制单元150所需的计算能力可以至少部分地转移到外部的计算机上,该外部的计算机可以例如经由家庭网络或者经由互联网(云)来访问。

[0028] 自主移动机器人100包括驱动单元170,该驱动单元例如可以具有电动机、传动机构和轮子,由此,机器人100(至少在理论上)可以行进到它的使用区域中的任何点。驱动单元170被设计用于将从控制单元150接收的命令或者信号转换成机器人100的运动。

[0029] 自主移动机器人100此外包括通信单元140,以便建立与人机界面(HMI)200和/或其他外部设备300的通信链路145。通信链路145例如是直接的无线连接(例如蓝牙)、本地无线网络连接(例如WLAN或ZigBee)或者互联网连接(例如到云服务)。人机界面200可以例如

以视觉的方式或者也以听觉的方式向用户输出关于自主移动机器人100的信息(例如电池状态、当前工作指令、地图信息如清洁地图等等)以及接收用于自主移动机器人100的工作指令的用户命令。用于HMI200的示例是平板PC(个人计算机)、智能手机、智能手表和其他的可穿戴设备、计算机、智能电视或者头戴式显示器等等。HMI200可以替代地或者附加地被直接地集成到机器人中,由此机器人100例如可以通过按键、手势和/或语音输入和输出进行操作。

[0030] 用于外部设备300的示例是:计算机和服务器,计算和/或数据被转移到该计算机和服务器上;外部传感器,该外部传感器提供附加的信息;或其他的家用电器(例如其他的自主移动机器人),自主移动机器人100可以与这些其他的家用电器一起工作和/或交换信息。

[0031] 自主移动机器人100可以具有工作单元160,例如用于处理地板表面并且特别是用于清洁地板表面的处理单元(例如刷子、真空吸尘装置)或者用于抓取和运输物品的抓持臂。

[0032] 在一些情况下,例如在远程临场机器人或监视机器人的情况下,可以使用另一个组件来完成规定的任务并且不需要任何工作单元160。因此,远程临场机器人可以具有与HMI耦合的通信单元140,该通信单元例如装备有多媒体单元,该多媒体单元例如包括麦克风、摄像机和屏幕,以便能够在多个在空间上远离的人员之间进行通信。监视机器人在监控行驶下借助于传感器查明不同寻常的事件(例如火、光、未经授权的人员等等)并且例如将其通告控制部门。在这种情况下,代替工作单元160,设置有带有用于监视机器人使用区域的传感器的监视单元。

[0033] 自主移动机器人100包括传感器单元120,该传感器单元具有不同的传感器,例如一个或多个用于探测关于机器人在其机器人使用区域内的环境(environment)的信息的传感器,例如在机器人使用区域内的障碍物或者其他的地标(landmarks)的位置和范围。用于探测关于环境的信息的传感器例如是用于测量与在机器人的环境中的物体(例如墙壁或者其他障碍物等等)之间的距离的传感器,例如光学的和/或声学的传感器,该光学和/或声学的传感器可以借助于三角测量或者对发射出的信号的传输时间的测量来测量距离(三角测量传感器、3D摄像机、激光扫描仪、超声波传感器)。替代地或者附加地,可以使用摄像机来收集关于周围环境的信息。特别是当从两个或者多个位置出发来观察物体时,也可以确定物体的位置和范围。

[0034] 附加地,机器人可以具有传感器,以便检测与障碍物的(通常是无意的)接触(或者碰撞)。这可以通过加速度计(该加速度计例如检测机器人在发生碰撞时的速度变化)、接触开关、电容式传感器或者其他触觉式或者触敏式传感器来实现。附加地,机器人可以具有地面传感器,以识别在地面中的边缘,例如楼梯台阶的边缘。在自主移动机器人的领域中的其他常用的传感器是用于确定机器人的速度和/或走过的路程的传感器,例如用于确定机器人的位置和运动变化的里程表和/或惯性传感器(加速度传感器、旋转速率传感器),以及用于检测轮子与地面之间的接触的轮子接触开关。

[0035] 自主移动机器人100例如可以与基站110相关联,在该基站处机器人例如可以对其能量储存装置(电池)充电。在完成的任务之后,机器人100可以返回到该基站110。如果机器人不再具有要处理的任務,则它可以在基站110中等待新的使用。

[0036] 控制单元150可以被设计成用于提供机器人在其使用区域中独立运动和完成任务所需要的所有功能。为此目的,控制单元150例如包括处理器155和存储器模块156,以便执行软件。控制单元150可以基于从传感器单元120和通信单元140获得的信息来生成用于工作单元160和驱动单元170的控制命令(例如控制信号)。如已经提到的那样,驱动单元170可以将这些控制信号或者控制命令转换成机器人的运动。包含在存储器156中的软件也可以被模块化地设计。导航模块152例如提供用于自动创建机器人使用区域的地图以及用于规划机器人100的运动的运动的功能。控制软件模块151例如提供通用(全局)控制功能并且可以在各个模块之间形成接口。

[0037] 为了使机器人能够自主地完成任务(task),控制单元150可以包括用于在机器人的使用区域中对机器人进行导航的功能,该功能由上面提到的导航模块152提供。这些功能本身是已知的并且除了其它方面以外,可以包括以下功能之一:

[0038] • 提供借助于传感器单元120,例如但不是仅仅借助于SLAM方法收集关于环境的信息来创建(电子)地图(Simultaneous Localization and Mapping,同时定位和地图创建),

[0039] • 管理针对机器人的一个或多个与地图相对应的使用区域的一个或多个地图,

[0040] • 基于通过传感器单元120的传感器确定的周围环境的信息,确定机器人在地图上的位置和方向(姿态),

[0041] • 进行从机器人的当前的姿态(起始点)到目标点的基于地图的路径规划(轨迹规划),

[0042] • 轮廓跟随模式,在该轮廓跟随模式下机器人(100)沿着一个或者多个障碍物(例如墙壁)的轮廓在与该轮廓基本上恒定的距离d下运动。

[0043] 控制单元150可以借助于导航模块152并且基于传感器单元120的信息,例如在机器人的运行期间连续地更新机器人使用区域的地图,例如当机器人的环境发生变化时(障碍物被移动,门被打开等等)。然后,当前的地图可以被控制单元150用于机器人的短期的和/或长期的运动规划。控制单元150在机器人的(目标)运动被实际执行之前针对机器人的该(目标)运动先前计算的那个路径被称为规划范围(planning horizon)。在此处描述的示例性实施例,除其他方面以外,涉及在某些情况下的不同的方法和策略,例如在这样的情况下,其中某些操作被障碍物阻挡并且因此不能够进行实施。

[0044] 一般地,可由机器人100使用的(电子)地图是地图数据的集合(例如数据库),其用于存储关于机器人的使用区域和在该使用区域中与机器人有关的环境的、与位置相关的信息。在这一点上,“与位置相关的”是指将存储的信息分配给在地图中的一个位置或者一个姿态。因此,地图代表了大量的具有地图数据的数据记录项,并且地图数据可以包含任何与位置相关的信息。在这种情况下,与位置相关的信息可以以不同的详细程度和抽象程度进行存储,其中,该详细程度和抽象程度可以与特定的功能相适配。特别地,各单个信息可以冗余地存储。与同一区域相关、但以不同的形式(数据结构)进行存储的多个地图的汇编常常也被称为“地图”。

[0045] 非圆形机器人-导论:图3分别在从下方观察的视图中显示了用于自主移动机器人100的外壳形状的各种本身已知的示例。在所示出的示例中,机器人100分别具有例如用于处理地板表面的工作单元160,例如特别是刷子、真空吸尘单元和/或擦拭单元。

[0046] 此外,机器人100分别具有驱动单元170,该驱动单元具有两个彼此独立地驱动的轮子170R、170L。一般地,移动机器人可以具有优选的运动方向(在没有限制一般性的情况下被定义为向前方向),该优选的运动方向通过箭头标示。该优选的运动方向或者说向前方向可以例如通过工作单元在外壳中或者在外壳上的布置来预先确定,但是也可以通过传感器(例如传感器单元120)的布置来预先确定。例如,用于拾取脏物的清洁单元(例如真空吸尘装置)可以安置在驱动单元170的前面,使得更少的脏物可以进入轮子中。此外,例如可以在驱动单元170的后面安置用于施加清洁液或者用于对地板表面进行擦亮的清洁单元,使得轮子在清洁后的地板表面上不会留下任何污染。例如这样地布置传感器,即它们主要检测在机器人的优选的运动方向上(即,在机器人100的前面)的周围环境。然而,机器人100也可以逆着所述优选的运动方向(也就是说向后)运动(可能有一定的限制)。关于所提到的优选方向(即使在具有(在俯视图中)基本上为圆形形状的机器人的情况下)机器人在特定的位置上的定向也起着一定的作用,并且例如朝向轮廓(例如墙壁)和离开轮廓的旋转运动是可以明确地区别开的,即使机器人的位置(在围绕对称轴线旋转期间)不改变。

[0047] 如果两个被驱动的轮子170R、170L沿着相反方向旋转,那么机器人会就地围绕通过“×”标记的中心点(运动中心,旋转中心)围绕其垂直轴线旋转并且由此执行纯旋转运动(也就是说,没有平移运动分量)。

[0048] 来自图3中的示意图(a)示出了圆形机器人,该机器人的轮子170R和170L被布置在对称轴之一上。这具有的优点是,机器人可以就地围绕其中心旋转。不管障碍物H的位置如何,该旋转都不会受到干扰,因此圆形机器人在围绕其垂直轴线进行适当的旋转之后始终可以沿着其优选的方向(也就是说,向前)行驶。

[0049] 来自图3中的示意图(b)示出了D形机器人。D形状具有的优点在于,可以使用在机器人的整个宽度上延伸的工作单元160。附加地,可以使工作单元160运动到特别靠近障碍物H(例如墙壁)。但是,在这种示出的姿态下,机器人不再能够在不发生碰撞的情况下旋转;在围绕其垂直轴线旋转之前,它必须首先至少向后(与优选的运动方向相反)行驶一点。

[0050] 来自图3中的示意图(c)示出了圆形机器人,该机器人的轮子170R和170L未沿着机器人形状的对称轴线之一进行布置。这具有的优点是,工作单元160可以在机器人的整个宽度上延伸。但是,机器人100的中心点“×”(运动中心)不再与圆形的外壳基面的几何中心相一致,因此在旋转时会发生与障碍物H的碰撞。为了避免这种碰撞,如在来自图3中的示意图(c)中所示的那样,机器人必须首先至少向后行驶一点。在本例中,机器人不能够以简单的方式进行纯粹的旋转运动。围绕运动中心点“×”的旋转会导致几何中心的运动也具有平移的运动分量(特别是围绕运动中心的在圆形轨迹上的运动),该平移的运动分量可能被障碍物阻挡。

[0051] 来自图3中的示意图(d)示出了机器人100的水滴状的外壳形状,其中,外壳的基面具有明显的拐角部,但是其余都是圆形的。这具有的优点在于,工作单元160可以被布置在机器人的拐角部中,并且因此可以将其引导到特别靠近障碍物(例如进入到房间的角落中)。与在D形状的情况下相比,机器人的运动受到限制的程度较小。但是,在这里也存在这样的情况,即在可以围绕垂直轴线不受阻碍地旋转之前,机器人必须至少向后行驶一点。

[0052] 来自图3中的示意图(e)示出了被伸长的、基本上D形的机器人。这具有的优点是,存在更多的空间用于工作单元160,工作单元可以在机器人的整个宽度上延伸。附加地,可

以将工作单元160引导到特别靠近障碍物H例如墙壁。但是,在该位置上,机器人不再能够旋转,并且必须首先至少向后行驶一点。

[0053] 如在图3的示意图(b)至(e)中所示的那样,机器人只能通过沿着向后方向(与优选的运动方向相反)才能从其中操控出来的情况在以下被称为“死胡同状况”。应该指出的是,在图3中所示的机器人只是示例。当然任何其他的形状也是可能的。特别地,形状也可以随着机器人的高度而变化(参照在图1中的示意图(b))。驱动模块170的其他的变型方案(例如链驱动、支腿)也是已知的并且是可能的。

[0054] 轮廓跟随行驶:一种用于自主移动机器人100的路径(轨迹)的局部规划的简单方法在于,机器人直截了当地以一个基本上恒定的轮廓跟随距离 d 跟随一个或者多个障碍物的轮廓(轮廓跟随行驶)。在以下,机器人基于障碍物的轮廓沿着一个轮廓以基本上恒定的距离进行运动的运行模式被称为轮廓跟随模式(contour following mode)或者障碍物跟随模式(obstacle following mode)。由机器人在轮廓跟随模式下执行的运动被称为轮廓跟随行驶(contour following run),并且到轮廓的距离被称为轮廓跟随距离(contour following distance)。轮廓跟随模式的使用本身是已知的,并且例如被用于避开障碍物(例如参见J.Fasola等人,“Fast Goal Navigation with Obstacle Avoidance using a Dynamic Local Visual Mode(使用动态局部视觉模式避障的快速目标导航)”,见:Proc.VII.Simpósio Brasileiro de Automação Inteligente,圣路易斯,2005年9月)。用于执行轮廓跟随行驶的方法尤其可以以基于行为的机器人技术或者反应性机器人技术的概念为基础,其中,当前的传感器测量值(特别是有关障碍物相对于机器人的位置和/或机器人与障碍物之间的距离)可以被直接地转换成用于驱动单元的控制命令。

[0055] 轮廓可以由墙壁的形状、较大的障碍物的形状给出,但是也可以由多个较小的、紧密地聚集在一起的障碍物给出。在这一点上,机器人可以由此坠落的边缘(坠落边缘),例如在楼梯的情况下,也被看作是具有轮廓的障碍物,机器人可以跟随该轮廓。另外,形成轮廓的障碍物可以是标记(例如磁带、电流回路或者导引光束发射器形式的标记),机器人可以借助于相应的传感器检测到这些标记。从这些传感器数据中可以导出一个边界(例如磁带或者电流回路的走向,所发射出的导引光束的走向),机器人不允许自主地行驶通过该边界。该边界也可以用作机器人可以跟随的轮廓。此外,可以在地图数据中记录虚拟障碍物,这些虚拟障碍物标记出机器人不允许自主地行驶的区域(这些区域也被称为限制区域、“禁止进入区域”或者“禁区”)。附加地或者可替代地,虚拟障碍物并且尤其是其虚拟轮廓可以被暂时地用于在设置用于处理的区域中“锁住”或者引导机器人,直到所述处理完成为止。这种虚拟障碍物的虚拟轮廓也可以在轮廓跟随模式中用作机器人可以跟随的轮廓。

[0056] 轮廓跟随距离 d 取决于机器人的尺寸和任务,但在具体的轮廓跟随模式下可以基本上保持恒定。通过较大的距离可以更容易地(以更大的可能性)避免没有预料到的碰撞,例如由于行驶错误造成的没有预料到的碰撞。在用于处理(特别是清洁)地板表面的机器人的情况下,轮廓跟随模式可以用于靠近墙壁和其他的障碍物的处理。结果,这样的机器人可以非常接近障碍物地沿着障碍物行驶,以实现高的表面覆盖范围,并且尤其是实现在拐角和边缘中的彻底的清洁。在这种情况下,用于在家用领域中小型清洁机器人的示例性的值位于2.5毫米和20毫米之间。还具有这样的清洁机器人,其在轮廓跟随行驶期间在机器人的一部分与要跟随的轮廓之间建立并保持直接的接触(也就是说,通过触摸)。与在相对较小

的机器人的情况下相比,对于大型机器人来说,轮廓跟随距离 d 可以明显更大。

[0057] 为了在轮廓跟随行驶期间控制机器人,机器人可以具有用于探测机器人的邻近的环境的传感器(参见图2,传感器单元120)。这些传感器可以例如可靠地确定到障碍物的距离并且尤其是到在距离范围中要跟随的轮廓的距离。例如,这种传感器可以被布置在机器人的面对要跟随的轮廓的那一侧上。

[0058] 替代地或者附加地,在轮廓跟随行驶期间对机器人的控制可以基于地图数据,其中,用于确定机器人以及障碍物的位置和定向(姿态)传感器测量被存储和被进一步处理。基于地图的规划能够实现预测性的轨迹规划和机器人控制,并且还考虑关于刚好不能够被其中一个传感器探测到的障碍物的信息(传感器的“盲点”)。特别地,也可以考虑不能够通过传感器探测的信息,例如记录在地图上的虚拟障碍物(例如限制区域),机器人不得自主地在这些虚拟障碍物上行驶、越过和/或对其处理。在这里描述的示例性实施例中,例如可以基于地图来评估用于从一个轮廓跟随模式改变为另一个轮廓跟随模式(或者终止一个轮廓跟随模式)的标准。例如,用于结束一个轮廓跟随模式的标准可以是,机器人能够在不发生碰撞的情况下朝着目标点的方向上旋转。“机器人能够在不发生碰撞的情况下朝着目标点的方向上旋转”的这个标准例如可以根据机器人的当前的地图数据来进行评判。

[0059] 为了导航和地图创建,通常使用具有较大的有效范围的传感器来识别障碍物,这些传感器尽管可以很好地检测更远的障碍物,但是在近距离范围中常常是不合适的。例如可以使用三角测量传感器,该三角测量传感器可以通过发射结构化的光(例如激光束或者区域放射的激光束)并且检测被障碍物H散射回来的光,来确定到该障碍物H的距离。一般地,距离越小,对到障碍物的距离的测量就越精确。但是可能存在最小距离,在该最小距离下散射回来的光不再能够被传感器接收到,因为该光位于传感器的视野之外。例如可以使用测量发射出的信号(光、声)的传播时间测量值的传感器;一般地,这些传感器也具有用于检测障碍物的最小距离。对于摄像机而言,由于受限制的视野以及受限制的聚焦,在近距离范围中同样会出现问题。

[0060] 通过使用地图数据,尽管传感器装置受到限制,机器人仍然可以靠近障碍物进行导航,而不需要附加的传感器用于轮廓跟随行驶。附加地,可以容易地实现逆着优选的运动方向(也就是说,在向后方向上)的操控,而没有在机器人的后部中使用复杂的附加的传感器。

[0061] 死胡同状况的处理-向后行驶:如在图3中示例性地所示的那样,一般的、非圆形的机器人形状可能导致机器人100不能够始终沿着优选的方向(向前方向)运动,因为由于在机器人的环境中的障碍物,机器人朝着所希望的方向上的旋转运动(尤其是在静止下围绕中心点“ $\times$ ”的旋转运动)可能被封锁。在这种情况下,机器人处于死胡同状况下,应该注意的是,尤其是用于处理地板表面的机器人应该行驶进入这种情况中,以便实现所述处理的最大可能的表面覆盖范围以及对拐角和边缘的有效清洁。这意味着机器人在执行其任务期间的正常运行中将不可避免地一再进行死胡同状况中。

[0062] 从死胡同状况中操控驶出的一种简单的方法在于,正好在机器人(向前)行驶进入死胡同中的那个路径上向后行驶。这就是说,以相反的顺序和以相反的形式再次执行用于驱动单元的最后生成的控制命令,直到满足终止条件(例如机器人可以在静止下旋转)。

[0063] 为了实现上面提到的向后行驶以从死胡同中解脱出来,必须存储关于控制命令

和/或关于机器人的已行驶路径(例如路径点)的附加信息,其结果是存储需求增加。此外,反向的控制信号不一定导致反向的运动。因此,例如对运动的连续的干扰(例如由于驱动单元并且尤其是轮子的打滑和漂移),该运动不一定与在理论上不受干扰的运动成正比,可能导致,为了向后行驶而对驱动装置的反向控制没有产生与先前在向前方向上行驶的相同的轨迹。此外,可能发生这样的情况,即一个已经引起死胡同状况的可移动的障碍物改变了其位置。在这种情况下和其他的情况下,固定地预先确定下来的行驶操纵(向后行驶一定的距离)并不总是导致机器人的“有意义的”行为。

[0064] 为了克服这个问题,可以基于地图信息生成新的控制命令,以便向后操纵机器人。在这种情况下,尤其可以跟随机机器人进入死胡同状况中所跟随的轮廓。一直进行这种操纵,直到确定可以离开死胡同或者已经离开死胡同为止。

[0065] 图4示出了一种用于控制自主移动机器人100以便跟随障碍物的轮廓的可能的的方法。在这种情况下,启动并执行第一轮廓跟随模式(图4,步骤10)。该第一轮廓跟随模式的特征例如在于机器人的面向所述轮廓的那个侧面、应该跟随所述轮廓所沿的方向、以及轮廓跟随距离 d 。在机器人沿着轮廓运动期间,可能会发生以下情况:机器人确定,在第一轮廓跟随模式下沿着轮廓的第一选择方向的机器人的继续运动是不可能的,因为机器人例如位于死胡同状况下(图4,步骤11)。例如通过机器人根据其在地图中的当前位置和在地图中记录的障碍物的位置来确定其运动的可能性,机器人检测死胡同状况。如果向前运动或者旋转运动是不可能的,因为该运动可能导致与障碍物的碰撞,那么这就是所述的死胡同状况。为了从死胡同中导航出来,机器人在第二轮廓跟随模式下逆着第一方向跟随轮廓(图4,步骤13)。在这种情况下,确定了一个标准(图4,步骤12),在满足该标准时第二轮廓跟随模式应该被停止,例如以便恢复在第一轮廓跟随模式下沿着第一选择方向的运动。

[0066] 在图5中的示意图(a)至(d)将借助于示例来说明根据图4的方法。图5的示意图(a)示出了跟随墙壁W(或者另一个障碍物)的轮廓的机器人100,其中,机器人与墙壁W的轮廓保持尽可能恒定的距离 d (轮廓跟随距离)。机器人100一直跟随该轮廓,直到其路径,例如在图5的示意图(b)中所示的,被(例如位于机器人100的前面的)障碍物H封锁。在这种情况下,障碍物H也可以是墙壁的一部分,例如在房间的拐角中就是这种情况。在以下,为了简单起见,轮廓用附图标记W表示。可以理解的是,该轮廓W既可以代表墙壁的轮廓又可以代表一个或者多个其他的障碍物。示例性地,轮廓W可以被想象为房间的墙壁。

[0067] 如果障碍物H(仅还)处于与障碍物H相距安全距离 d_s ,并且机器人100的旋转是不可能的,那么该路径被视为被封锁。可以理解的是,在足够大地选择安全距离 d_s 的情况下,机器人的旋转自由度不受位于机器人前面的障碍物的限制,但是,特别是在用于处理地板的机器人的情况下,安全距离 d_s 被选择得尽可能的小(显著地小于机器人本身的外部尺寸),以便在处理地板表面时实现尽可能好的表面覆盖范围。因此,安全距离 d_s 例如可以这样地来选择,即机器人可以安全地不会发生碰撞地旋转,或者机器人不能够在不发生碰撞的情况下旋转。后者在许多应用中就是这种情况。安全距离 d_s 可以例如小于或者等于轮廓跟随距离 d ($d_s \leq d$)。例如可以完全放弃安全距离 d_s (也就是说, $d_s = 0$ 毫米),以便机器人会一直跟随轮廓W,直到触摸到位于机器人前面的障碍物H。该触摸可以例如借助于触觉传感器(对触摸做出反应的传感器)来检测。

[0068] 为了从这个位置中行驶出来,机器人控制器150改变到第二轮廓跟随模式,在该第

二轮廓跟随模式下,机器人100沿着相反的方向跟随墙壁W的轮廓(参见来自图5中的示意图(b)),直到限定的标准被满足,即直到机器人100如此远地离开障碍物,以至于它可以在不发生碰撞的情况下旋转并且可以沿着原始的方向(向前)跟随新的障碍物H的轮廓为止。因此,第二轮廓跟随模式与第一轮廓跟随模式的不同在于一个参数,即在机器人应跟随轮廓所沿的方向上不同。另外,设置一个标准(例如旋转不再被封锁),在该标准下可以结束第二轮廓跟随模式,例如以便返回到第一轮廓跟随模式或者重新启动第一轮廓跟随模式。其他的轮廓跟随模式可能在其他的参数上有所不同(例如,轮廓跟随距离、要跟随的轮廓所位于的机器人的侧面(左侧或者右侧),等等)。在简单的示例中,特定的轮廓跟随模式由参数:行驶方向(向前或者向后)和轮廓跟随距离,来确定。

[0069] 根据在这里描述的示例,用于结束第二轮廓跟随模式的标准可以是,机器人可以再次在很大程度上自由地运动,并且特别地可以沿着新的障碍物的轮廓继续进行第一轮廓跟随行驶。除了其它方面以外,这意味着机器人的旋转自由度不再被封锁。但是,在这种情况下,先验性地尚不清楚机器人必须旋转多远才能继续进行该轮廓跟随模式。这一点示例性地在图5的示意图(c)和(d)中被可视化。

[0070] 在图5的示意图(c)中,示出了行驶操纵,在该行驶操纵下机器人100从障碍物H的旁边驶过,该障碍物在轮廓跟随行驶期间居中地位于机器人的轨迹上。在这种情况下,机器人必须跟随墙壁W的轮廓向后运动一段距离 d_{w1} ,直到机器人100可以再次自由转动为此。机器人用于围绕中心点“×”旋转所需要的空间由圆C标示。在这种情况下应该注意的是,一旦机器人100已经沿着轮廓W向后稍微运动了一点,机器人就可以旋转。但是,在这种情况下,机器人不能够旋转到从障碍物H旁边行驶通过的程度。

[0071] 图5的示意图(d)示出了一种为了从障碍物H旁边行驶通过的行驶操纵,该障碍物位于靠近要跟随的轮廓W处。为此目的,机器人必须沿着距离 d_{w2} 向后地跟随墙壁W的轮廓,以便能够再次旋转。在这种情况下要往回行驶的距离 d_{w2} 小于来自图5C中的距离 d_{w1} 。同时,机器人的旋转自由度进一步受到第二障碍物H'的限制,该第二障碍物位于回转圆C的内部。但是,尽管有此限制,机器人仍然可以在两个障碍物H、H'之间行驶通过,并且然后继续执行第一轮廓跟随模式。

[0072] 从在图5的示意图(c)和(d)中所示的示例中可以看出,机器人是否可以旋转以及机器人可以旋转多远,这一事实不是用于结束第二轮廓跟随行驶的具有说服力的标准。如果应该继续进行第一轮廓跟随行驶,那么情况尤其如此。

[0073] 用于评判(通过机器人)是否第二轮廓跟随模式可以被结束并且是否先前的轮廓跟随行驶(在第一轮廓跟随模式下)可以有意义地继续进行的可能的标准例如是,机器人在成功旋转之后能够沿着直线向前运动(也就是说沿着第一轮廓跟随模式的运动方向运动)。这在图5的示意图(c)和(d)中通过通道P来标示,在该通道中机器人可以沿着直线运动长度1。在这种情况下,长度1可以是预设的值,或者可以至少部分地基于在旋转期间所转过的角度或者在第二轮廓跟随行驶期间所行驶过的距离 d_{w1} 或者 d_{w2} 来确定。例如,可以这样地来选择长度1,使得机器人100的前轮廓离开回转圆C。长度1也可以选择得短于为了离开回转圆C而所需的长度。这导致,机器人可以导航到更靠近障碍物的地方。但是,这可能导致以下的情况:在返回到第一轮廓跟随模式之后,必须再次中断第一轮廓跟随模式,这可能导致一系列的向前和向后运动。关于第二轮廓跟随模式是否应当终止的标准可以特别地基于地图进

行评估。在这种情况下,假定地图至少在机器人100的本地环境中是足够准确的并且是最新的。

[0074] 在一些示例性实施例中,用于结束第二轮廓跟随模式的标准可能仅仅在于直线向前运动的可能性。例如,该标准可以是,机器人必须能够在一个可预先确定的方向上向前运动机器人在第二轮廓跟随模式下向后行驶的那个距离,加上另一个可预先确定的距离(例如,距离d)。为了能够定向到这个可预先确定的方向上,机器人一般必须旋转。旋转的可能性不一定是用于结束第二轮廓跟随模式的标准的确定的组成部分。在某些情况下,如果机器人例如在第二轮廓跟随模式期间沿着弯曲的轮廓(向后)运动,那么机器人在没有附加的旋转的情况下就可以到达相应的方向。在其中可以不需要旋转的另一个示例是环境的动态变化。因此,用户例如可以去除已经触发第二轮廓跟随模式的那个障碍物H。结果,机器人的向前运动不再受到限制,并且第二轮廓跟随模式可以在不旋转的情况下以直线运动来结束。

[0075] 附加地或者替代地,在评估导致第二轮廓跟随模式结束的标准时,检查导致第一轮廓跟随模式中中断的障碍物H的位置,或者在可能的旋转之后检查另一个障碍物H'的位置。因此,在机器人前方的一个可预先确定的距离内应该没有任何障碍物存在。同时,先前导致死胡同状况的那个障碍物应该在旋转之后相对于机器人被这样地定位,即使得机器人可以在第一轮廓跟随模式下以预定的轮廓跟随距离d跟随该障碍物H的轮廓。这尤其意味着,在向前运动长度l之后,障碍物的轮廓的一部分与机器人相距轮廓跟随距离d中(参见来自图5中的示意图(c))。

[0076] 如先前参考图5的示意图(d)所解释的那样,为了结束第二轮廓跟随模式机器人至少必须能够旋转的角度可以相对较小,例如在1-5度的范围内,或者可以完全放弃旋转。图6,示意图(a)示出了一个示例,在该示例中,除了在机器人前方的障碍物H之外,第二障碍物H'直接地限制了机器人的旋转。这样的障碍例如可以通过以下事实来识别:它们至少部分地位于回转圆C的前部区域S内(例如在前半圆内)。在这样的状况下,总是需要相对较大的旋转,以便机器人可以结束第二轮廓跟随模式并且继续进行第一轮廓跟随模式。例如,为了限制要检查的运动选项,在这里可能有利的是,将在旋转之前和之后在机器人的定向之间必须存在的较大的最小角度作为标准。这个最小角度可以是标准值(例如45°)或者可以依赖于机器人的形状和/或障碍物H'的形状和大小进行选择。

[0077] 因此,用于结束第二轮廓跟随模式的标准的设置可以取决于在机器人的环境中的障碍物的(例如存储在机器人的地图中的)位置。特别地,如果障碍物的至少一个点位于特别是在机器人旁边的可预先确定的区域S中,那么可以确定和使用第一标准,而否则可以确定和使用第二标准。根据这两个标准,例如应该可以使机器人旋转到离开轮廓的一个位置上,其中,至少在第一标准下,旋转的角度可以大于一个可预先确定的最小角度。如果两个标准都包含最小角度,那么根据第一标准的最小角度大于根据第二标准的最小角度。

[0078] 图6的示意图(b)示出了一个示例,在该示例中第二障碍物H'处于与在图5的示意图(d)中相同的位置上。根据在图5的示意图(d)中示出的示例,第一障碍物H靠近轮廓W,从而机器人可以在小的旋转之后在两个障碍物H、H'之间行驶通过。在图6的示意图(b)中,第一障碍物H处于这样的状态下,即不可能进行这种行驶操纵,因为两个障碍物H、H'彼此靠得太近了。因此,如在图6的示意图(a)中所示的示例中那样,也可以确定和使用用于以大的最

小角度结束第二轮廓跟随模式的标准。

[0079] 例如可以基于第一障碍物H的位置、形状和大小来确定是否需要这种最小角度。可替代地,可以放弃确定最小角度(如在图5的示意图(d)中所示的示例中那样)。例如,如果在第二轮廓跟随模式中确定,障碍物H'位于区域S中并且因此封锁了机器人的旋转,那么可以随后设置最小角度。替代地或者附加地,如果在第二轮廓跟随模式中确定,障碍物H由于与机器人之间的距离不再封锁其旋转,但是由于障碍物H'的原因而不能满足用于结束第二轮廓跟随模式的标准,那么可以随后确定大的最小角度。因此可以在以第二轮廓跟随模式行驶期间更新该标准。

[0080] 除了基于关于机器人的周围环境的信息(特别是地图数据)评估可能的运动之外,用于结束第二轮廓跟随模式的标准还可以包括不发生碰撞地执行该规划的运动。这意味着,仅在成功执行运动后第二轮廓跟随模式才会被结束。如果在运动期间发生没有预料到的碰撞,那么第二轮廓跟随模式将立即继续进行沿着墙壁W的轮廓(在向后方向上)对机器人100的控制。关于碰撞的信息将被包括到关于机器人的环境的信息中,并且尤其是包括到地图数据中,并且因此在以后提供用于控制机器人。应该注意的是,一般地,在第二轮廓跟随模式下可以使直到碰撞之前所执行的运动部分再次倒退回去,尽管这没有明确地实施。相反,这是轮廓跟随模式的一种属性,该轮廓跟随模式将机器人100控制到与要跟随的轮廓在很大程度上平行的方向上。

[0081] 应当注意的是,在图5和图6、示意图(a)和(b)中所示的示例中,轮廓W总是显示为直线,并且因此机器人直线地向后移动。一般地,墙壁W(或者另一个障碍物)的轮廓不一定是直的,而是可以包含曲线和拐角,机器人在第二轮廓跟随模式下同样要向后移动地跟随该曲线和拐角。在图6的示意图(c)和(d)中的示例示出了具有非直线的轮廓W的情况,机器人在第一轮廓跟随模式下跟随该非直线的轮廓,直到障碍物H阻挡了对该轮廓跟随行驶的进一步的执行为止(参见图6的示意图(c))。在随后的第二轮廓跟随模式下,机器人沿着轮廓W向后行驶,直到机器人可以旋转到这样的程度,既机器人可以从障碍物H旁边行驶通过为止(用于结束第二轮廓跟随模式的标准)。在此之后,可以继续进行第一轮廓跟随模式,并且机器人跟随障碍物H的轮廓。该方法使在此处所示的方法与其他的方法区别开来,在所述其他的方法中,使用预先确定的运动模式(操纵),例如简单的直线的向后行驶。这种情况在图6的示意图(e)中示出;由于简单的向后移动,在用Z标记的区域中发生碰撞。图6的示意图(f)示出了死胡同状况的另一个示例,在该死胡同状况下,通过简单的预先确定的运动模式例如向后移动和旋转不可能在不发生碰撞的情况下从该死胡同中逃脱出来。另外,机器人可以直接地对由在其环境中的运动(例如人或者动物的运动)引起的动态变化做出反应(例如机器人利用传感器单元120探测该动态变化并将其用于更新它的地图数据)。由此,可以明显更灵活地和更多样性地使用在这里示出的方法。

[0082] 当机器人在第二轮廓跟随模式下跟随墙壁W(或者另一个障碍物)的轮廓时,可能发生这样的情况,即第二轮廓跟随模式同样不允许任何进一步的运动。例如,如果在机器人的三个侧面上存在障碍物,即,特别是其轮廓被跟随的墙壁W、一个阻止进一步向后行驶的障碍物、以及障碍物H',从而用于结束第二轮廓跟随模式的必需的标准是不满足的,那么这一点是可能的。在这种情况下,可以重新更改方向,以便机器人在第三轮廓跟随模式下再次沿着原始方向行驶。为了避免导致进入死胡同的先前的行驶模式在很大程度上相同的重

复,例如可以变换机器人跟随轮廓时所在的侧面。由此机器人与轮廓W脱离,以便例如跟随障碍物H'的轮廓(该障碍物H'阻碍了对用于结束第二轮廓跟随模式所需的标准的满足),并且到达一个位置,该位置例如使第一轮廓跟随模式能够继续执行。可以设置用于结束第三轮廓跟随模式的新的标准。可替代地,可以保留或者采用用于结束第二轮廓跟随模式的先前设置的标准。

[0083] 在这种情况下,这种操作过程基本上对应于先前参照图4描述的方法,与后者的唯一的区别在于,在第一轮廓跟随模式10之前已经发生了另一个轮廓跟随模式。原则上,这种操作过程可以用第四、第五等轮廓跟随模式重复进行,直到机器人找到从死胡同状况中出来的路径为止。一般地,轮廓跟随模式至少在以下的特征之一上彼此相区别:

- [0084] • 跟随轮廓所沿的方向,
- [0085] • 机器人的面向轮廓的侧面,
- [0086] • 用于导航的参数更改,例如轮廓跟随距离 d 、到障碍物的安全距离 d_s 、速度,
- [0087] • 借此避免与障碍物接触(碰撞)的优先级,
- [0088] • 用于确定碰撞而被考虑的机器人形状(例如可以在基于地图的评估中在机器人的虚拟扩大的外壳形状的形式下考虑安全距离),
- [0089] • 用于生成沿着轮廓的运动的规则,以及
- [0090] • 地图数据的解释和评估。

[0091] 通过更改轮廓跟随距离 d 和/或安全距离 d_s (减小或者增大),机器人可以获得更大的运动自由度。类似地,可以通过调整机器人的速度来提高导航的准确性,由此机器人可以例如更容易地导航通过狭窄的地方,或者可以更好地对例如由于地面覆盖物造成的行驶错误(例如摩擦和漂移)做出反应。

[0092] 在改变了行驶方向的情况下,必须相应镜像地观察要观察的机器人形状。例如,在D形机器人的情况下,可以在轮廓跟随行驶期间限制旋转自由度(尤其是根据轮廓跟随距离 d)。在这种情况下,如果平的侧面指向行驶方向,那么朝着轮廓方向上的旋转是可能的或者仅可以在有限的程度上旋转。另一个方面,如果平的侧面指向行驶方向的反方向,那么(在静止下)从轮廓离开的旋转受到限制。这直接导致用于产生沿着轮廓的运动的规则也相应地改变。

[0093] 在一些示例性实施例中,可能发生这样的情况,即机器人100通过避免碰撞的策略找不到从死胡同中出来的出路。这可以例如通过以下情况来识别,即机器人多次不成功地更改轮廓跟随模式(尤其是机器人的方向和/或机器人的面向轮廓的侧面),而不能够满足用于结束相应的轮廓跟随模式的标准。对此的原因例如可能是有错误的传感器和/或地图数据,由此机器人会将在真实环境中的某个点看作是被障碍物封锁了,但是在该点处是可以自由行驶的。在这种情况下,可以放弃避免碰撞的策略,并且由基于接触的行驶策略来替代。

[0094] 机器人在这种情况下触摸到障碍物所处于的点也可以存储在地图数据中并且用于进一步控制机器人。在用于在轮廓跟随模式下控制机器人的方法的一个示例性的实施方式中,第一轮廓跟随模式和第二轮廓跟随模式(以及其他的轮廓跟随模式)可以分别是独立的软件模块。替代地或者附加地,可以在软件模块中实施多种轮廓跟随模式,该软件模块可以以不同设置的参数启动。

[0095] 应该注意的是,机器人也可以在没有先前执行轮廓跟随行驶的情况下处于死胡同状况下。在这种情况下同样有用的是,向后跟随轮廓,直到机器人确定它可以从死胡同中驶出或者已经从死胡同中驶出为止。为此目的,例如一个用于规划机器人的功能的上级控制机构可以启动第一轮廓跟随模式,该第一轮廓跟随模式应该控制机器人沿着轮廓行驶到优选的方向(向前方向)上。在机器人执行运动之前,可能会发生这样的情况,即机器人确定,在该第一轮廓跟随模式下不能够执行任何运动,因此在相反的方向上的第二轮廓跟随模式被启动并且设置并使用用于结束该第二轮廓跟随模式的标准。替代地或者附加地,用于规划机器人的功能的上级控制机构可以直接地启动所述第二轮廓跟随模式并且根据可预先确定的标准再次结束所述第二轮廓跟随模式。

[0096] 图7借助于另一个稍微更复杂的示例来说明用于控制在死胡同状况下的自主移动机器人的方法,该死胡同状况在几何上比在先前的示例中要复杂一些。该示例还清楚表明,诸如执行固定的、预先确定的运动模式之类的简单方法并不总是适合用于解决死胡同状况。在图7中的示意图(a)至(d)示出了在第一轮廓跟随模式下沿着轮廓W运动期间处于多个相继位置的机器人100,其中,轮廓W位于机器人的右侧(也就是说,机器人100的右侧面向轮廓W)。在本示例中,轮廓W具有“弯折部”,并且机器人超出弯折部地跟随该轮廓(参见图7的示意图(b)和(c))。在图7的示意图(d)中所示的情况下,机器人100已经到达不再能够在第一轮廓跟随模式下进行进一步运动的位置。因此,机器人100的控制器150变更到第二轮廓跟随模式,在第二轮廓跟随模式下,行驶方向是向后倒退的。机器人100跟随轮廓W往回移动并且在所提到的轮廓W的弯折部处达到另一个死胡同状况(参见在图7中的示意图(e));不仅向后行驶的继续进行而且较大的旋转(例如 45°)均被封锁了。

[0097] 作为对这种第二死胡同状况的反应,第二轮廓跟随模式也被结束,并且机器人100的控制器150更改到第三轮廓跟随模式,在该第三轮廓跟随模式下,与第二轮廓跟随模式相比,不仅运动方向而且轮廓(该轮廓应该在距离d下被跟随)所在的机器人的侧面都是“颠倒的”(向前运动而不是向后运动,在轮廓的左侧而不是右侧)。机器人的反应在图7中的示意图(f)至(g)中示出;机器人100朝着轮廓旋转到其左侧,并且以轮廓跟随距离d与轮廓对齐,直到向前运动被再次阻挡(图7的示意图(g))。作为对该第三死胡同状况的反应,该第三轮廓跟随模式被结束并且机器人100的控制器150更改到第四轮廓跟随模式,其中,运动方向再次被改变(向后运动、保持在轮廓左侧)。机器人在这种情况下可以快速地在轮廓间距d与轮廓的左侧平行对齐。从在图7中的示意图(h)中所示的状况出发,机器人可以在轮廓的左侧跟随轮廓向后移动(在第四轮廓跟随模式中),直到满足用于结束该轮廓跟随模式的标准为止,在图7的示意图(i)中所示的状况就是这种情况。机器人可以旋转一个(可预先确定的)角度,并且可以在第一轮廓跟随模式(向前运动、轮廓在机器人的右侧)下跟随另一个轮廓(在图7的示意图(j)中垂直延伸的轮廓)。虚线显示了中心“×”所走过的路径。

[0098] 图7的示意图(k)示出了相对于示意图(a)至(j)略微修改的情况,机器人以类似于在示意图(a)至(j)中所示的方式达到了这种情况。在所示的示例中,机器人可以在向后行驶之后(在第四轮廓跟随模式下)沿顺时针方向旋转(以便轮廓再次位于机器人的右侧)并且可以在第一轮廓跟随模式下继续执行轮廓跟随行驶。

[0099] 基本运动:下面说明在轮廓跟随行驶中控制自主移动机器人100的一种可能的形式。为了降低机器人100运动的大量可能性的复杂性,引入了至少三种基本运动,这些基本

运动适合于使机器人在期望的方向上在可预先确定的轮廓跟随距离下沿着轮廓运动。基于关于机器人的环境的信息并且尤其是基于地图数据对这些基本运动进行评估。选择具有最佳评估结果的基本运动。基于所选择的基本运动生成用于驱动单元170的控制命令。这种方法利用了规划的运动的优点并且同时通过仅仅具有较短的规划范围和该规划的快速重复,可以实现对在环境中的变化(例如人或者动物的运动)或者例如由于地板覆盖物(摩擦、漂移)引起的行驶错误做出快速反应。

[0100] 在评估基本运动时可以确定,没有任何基本运动可以被执行或者应该被执行。例如,可以基于地图数据确定,没有任何基本运动均能够在不发生碰撞的情况下被执行。也可以基于其他的选择规则来确定,没有任何基本运动可以被有意义地执行。对此的一个示例是一种在更下面还要更详细地描述的死胡同状况,在该死胡同状况下,第一轮廓跟随模式不允许沿着轮廓在优选的方向上进行任何进一步的运动。

[0101] 为了使机器人从死胡同状况中行驶出来,启动新的轮廓跟随模式,其中,原则上可以使用相同的或者相似的基本运动,但是运动的方向被颠倒过来。用于评估基本运动的规则可以重新确定或者在很大程度上保持不变。如果用于评估基本运动的规则保持不变,那么只需注意使用向后行驶的机器人外壳的轮廓即可(例如在D形机器人的情况下,因此半圆形的侧面处在运动方向上)。

[0102] 图8示出了可能的基本运动。在这种情况下,这些基本运动至少包括:

- [0103] • 第一基本运动:在当前的运动方向上的直线运动,
- [0104] • 第二基本运动:朝着要跟随的轮廓的方向上的旋转,
- [0105] • 第三基本运动:远离要跟随的轮廓的旋转。

[0106] 因此,第三基本运动的旋转的方向与第二基本运动的旋转的方向是相反的。机器人100的哪一侧应该面对要跟随的轮廓,可以通过上级规划机构来确定,通过该上级规划机构可以触发轮廓跟随行驶。替代地或者附加地,在轮廓跟随模式的开始时(例如基于地图信息)可以确定,可以或者应当被跟随的轮廓位于机器人的哪一侧面上。如果机器人应该跟随一个墙壁,那么机器人应该面对(轮廓)的侧面通常是清楚的。如果机器人应该避开障碍物,那么机器人可以在理论上沿着顺时针方向或者沿着逆时针方向绕过障碍物行驶,其中,优选的方向(例如在顺时针方向上)是可以预先确定的,仅在例外情况下才偏离该优选的方向。

[0107] 图8的示意图(a)示出了作为第一基本运动的直线运动。在这种情况下,两个轮子170L、170R向前运动相同的距离。在第一基本运动中要走过的距离可以是一个固定的距离。为了获得更大的灵活性,可以在运动的评估期间确定要走过的距离。在这种情况下,例如可以考虑用于直线运动的最小和/或最大距离。

[0108] 图8的示意图(b)示出了第二或者第三基本运动的可能的变型方案。在这种情况下,轮子170R和轮子170L沿着相反的方向运动,其结果是,机器人围绕其中心点旋转。

[0109] 图8的示意图(c)示出了第二或者第三基本运动的另一种可能的变型方案。在这种情况下,两个轮子中只有一个轮子170L向前运动,而第二轮子170R是静止的。因此,整个机器人围绕第二轮170R旋转。中心点“×”在圆形轨迹上向前运动。

[0110] 图8的示意图(d)示出了第二或者第三基本运动的另一个可能的变型方案。在这种情况下,两个轮子中只有一个轮子170R向后运动,而第二轮子170L是静止的。因此整个机器

人围绕第二轮子170L旋转。中心点“×”在圆形轨迹上向后运动,但是旋转方向与在图8的示意图(b)和(c)中的旋转方向是相同的。

[0111] 通过适当地控制驱动轮子,机器人也可以围绕其他的点旋转,其中,中心点“×”总是在圆形轨迹上运动。通过选择合适的旋转运动,尤其是可以实现工作单元160(没有示出)的运动的期望的特性。例如,在正常情况下可能值得期望的是,机器人100的工作单元160总是向前运动,这可以例如通过在图8的示意图(c)中所示的运动来实现。由此可以在某些应用中,例如在地毯上,例如机器人的刷子在所述地毯上留下了明显的清洁痕迹,实现更漂亮的清洁模式。通过在图8的示意图(d)中所示的运动,在机器人的前面的区域中布置的清洁单元(参见图3的示意图(b))可以稍微向后运动一点,由此可以实现更彻底的清洁。与在图8的示意图(b)至(d)中所示的旋转运动(其用于相应地定义第二或者第三基本运动)相反的旋转可以通过互换用于两个轮子170L、170R的驱动规则(前进/后退)来产生。

[0112] 在第二或者第三基本运动中要转过的旋转角度可以是例如 0.5° 至 5° 之间的固定角度。为了获得更大的灵活性,尤其是为了使机器人平行于要跟随的轮廓进行定向,可以在运动的评估期间确定合适的旋转角度。在这种情况下,例如可以考虑用于运动的最小和/或最大旋转角度。在第二和第三基本运动中使用的旋转运动可以基本上是相同的,其中,只是旋转方向不同。例如,在两种基本运动中,可以使用在图8的示意图(b)中示出的就地旋转。

[0113] 可替代地,第二和第三基本运动可以选择为不同(也就是说,不仅旋转方向是不同的,而且其它运动特征也是不同的)。由此可以更容易地使运动的特性适应于不同的要求。因此,第二基本运动可以包括根据图8的示意图(d)所示的一个小的向后运动,以实现更紧密地围绕小的障碍物如椅子腿行驶(并且因此实现更彻底的清洁)和/或第三基本运动根据图8的示意图(c)所示可以包括一个小的向前运动,以实现平滑的运动,以便平行于墙壁进行定向。因此机器人的合成的运动是一系列单个的基本运动(即,例如多次向右旋转 1° 、向前运动、多次向右旋转 1° 、向前运动等等),如果这些单个的运动一个接一个地执行,那么它们将导致震颤的运动。可以将控制单元150设计成用于使该运动平滑(例如借助于移动平均滤波器)。

[0114] 也可以考虑其他的基本运动。例如,也可以考虑与当前的运动方向(向后)相反的基本运动。为了获得平滑的运动,可以附加地对待建立以用于规划的基本运动、用于实现基本运动的控制命令进行平滑处理。

[0115] 运动的评估:有本身是已知的各种方法用于评估运动,以控制自主移动机器人100。例如,可以基于在机器人的环境中识别到的障碍物来确定“虚拟力”、“虚拟势能”或者“虚拟成本”。它们可以用于评估基本运动,其中,此时选择这样的运动,该运动跟随可预先确定的最佳状态(例如,沿着虚拟力的运动、虚拟势能或者虚拟成本的最小化)。用于评估运动的方法的选择对于执行在此处描述的示例性实施例来说不是重要的。

[0116] 在评估基本运动时,可以发生这样的情况下,即两个或者多个基本运动被相同地评估。在这种情况下,沿着轮廓引导的那个基本运动是优选的。对于在此处所示的运动来说,这意味着,在相同的评估的情况下,优选执行朝着轮廓方向上的旋转(第二基本运动)。在对直线运动(第一基本运动)和远离轮廓的旋转(第三基本运动)的相同评估的情况下,选择沿着轮廓的直线运动(第一基本运动)。

[0117] 在评估基本运动时,可以考虑一个或者多个先前的基本运动。因此,例如可以“禁

止”撤销最后的运动。特别地,如果第二和第三基本运动(朝着轮廓方向上的旋转和远离轮廓的旋转)是在静止状态下的旋转(参见图8中的示意图(b)),那么可以禁止这些基本运动的直接的顺序。可以设置针对基本运动的顺序的其他规则,以实现机器人沿着要跟随的轮廓的更平滑的行驶行为。

[0118] 在评估运动中的一个必不可少的方面是避免碰撞。例如会导致与障碍物的至少一个点发生碰撞的运动通常会被禁止,或者这种运动会造成非常高的成本。附加地,在评估运动时考虑要跟随的障碍物的轮廓的位置可能是有用的。图8在示意图(a)至(d)中显示了四个简化的示例。如在图9的示意图(a)中所示的那样,如果在要跟随的轮廓W和自主移动机器人100之间的距离大于可预先确定的距离 d (轮廓跟随距离),那么机器人应该朝着轮廓的方向旋转(第二基本运动)。如在图9的示意图(b)中所示的那样,如果在要跟随的轮廓W和自主移动机器人100之间的距离大约等于可预先确定的轮廓跟随距离 d (例如在一定的公差范围 $d \pm \epsilon$ 内),那么机器人基本上笔直地、与墙壁平行地运动(第一基本运动)。如在图9的示意图(c)中所示的那样,如果在要跟随的轮廓W和自主移动机器人100之间的距离 d 小于可预先确定的跟随轮廓的距离 d ,那么机器人应该在远离轮廓的方向上旋转(第三基本运动)。

[0119] 如在图9的示意图(d)中所示的那样,机器人100一般不是与轮廓W平行地定向的。相应地,必须以这样的方式实施对机器人100的控制(也就是说,自动地选择基本运动的顺序),使得机器人100在很大程度上与轮廓W平行地定向。为此目的,例如可以确定轮廓W的定向 θ ,并且可以基于轮廓的定向 θ 和机器人的定向选择基本运动,以便实现在可预先确定的轮廓跟随距离 d 下的平行定向(机器人100和轮廓W的定向此时是相同的)。在这一点上,应该再一次指出,轮廓W一般不是直线的,即使该轮廓附图中被简化作为直线示出。

[0120] 轮廓W的定向 θ 例如可以被确定为轮廓的两个点的连接矢量、在多个点上选择的回归线、轮廓的切线或类似物。环境的地图创建可以例如借助于用于特征提取的算法来进行,其中,障碍物(特别是墙壁)的轮廓的一些部分被作为线(或者面)进行记录并存储。轮廓的定向 θ 一般具有自然的方向,该自然的方向例如是由观察障碍物的方向和/或由机器人应该沿着轮廓跟随的方向得出的。如果轮廓被作为不具方向性的物体示出(例如线),那么尽管如此,通过选择机器人的侧面而明确地给出了机器人的与轮廓平行的方向,所述机器人的侧面在轮廓跟随行驶期间应该面向轮廓W(并且因此也通过确定第二基本运动的旋转方向来给出)。

[0121] 为了评估基本运动,可以将机器人的环境划分为各单个的区域。一种可能的划分例如包括在其中为了不发生碰撞地执行基本运动而不允许有障碍物的一个区域、用于分析要跟随的轮廓的一个区域和/或用于分析继续执行运动的可能性的一个区域。将机器人的环境细分为多个区域在图10的示意图(a)至(c)中示例性地示出。

[0122] 图10的示意图(a)示例性地示出了在机器人的环境中的用于评估直线运动(第一基本运动)的各个区域。示例性地示出了作为要跟随的轮廓的、具有拐角的墙壁W。

[0123] 区域I(在示意图(a)中以阴影方式示出)描述了机器人用于进行最小长度 $l_{\min}$ 的直线(向前)运动所需的那个区域。如果障碍物的至少一部分或者一个点位于在该区域中,那么运动不能在不发生碰撞的情况下执行,并且因此将被禁止。

[0124] 区域II是在机器人旁边在要跟随的轮廓一侧上的区域。例如从机器人一侧开始,该区域与轮廓跟随间距 d 一样宽。如果障碍物的至少一部分或者一个点位于在该区域II中,

那么这不一定是针对运动执行的排除标准。但是,例如在评估时,可以检查,是否机器人应该通过第三基本运动来增大与要跟随的轮廓W之间的距离。例如,如果轮廓W明显地突出到区域II中,那么该评估可能例如得出结论是,机器人运动远离轮廓。但是,如果只有一个小拐角或者一个单独的点靠近区域II的边缘处,那么这不应导致偏转运动,以避免左右摇摆的运动。为此目的,例如,在对第三基本运动进行基于成本的评估的情况下,可以考虑可预先确定的基础成本,该基础成本对应于伸入到区域II中的小拐角的成本。例如可以基于轮廓的突入到区域II中的部分的长度和/或面积比例来确定成本。如果要跟随的轮廓的一些部分位于区域II的边缘上,那么这可能会产生红利(例如负成本)。如果没有轮廓位于区域II中并且尤其是位于要跟随的轮廓的边缘区域中,那么这也可能会造成成本。

[0125] 区域III是在其中检查机器人的可能的进一步运动的区域。例如检查,是否机器人可以继续直线地向前运动,而不发生碰撞,以及可以运动多远。这例如通过最大规划范围 $l_{\max}$ 来限制。在该区域III中,机器人可以例如确定距离 $l_{\min} < l < l_{\max}$,机器人可以在不发生碰撞的情况下行驶该距离。在这种情况下,例如可以考虑与位于机器人前方的障碍物之间的安全距离 d_s 。

[0126] 区域IV是在机器人100旁边,在机器人与轮廓背离的一侧上的区域。一般地,没有障碍物位于这里。如果障碍物的至少一部分位于这里,那么可以将该信息用于使机器人100运动穿过在该障碍物和轮廓W之间的通道。

[0127] 图9的示意图(b)示例性地示出了用于执行旋转(第二/第三基本运动)的区域。应该注意的是,为了更好地图示说明,选择了相对较大的旋转。对于实际的机器人控制,可以将旋转选择得明显较小。

[0128] 在图10的示意图(b)中以阴影方式示出的区域I是在静止状态下进行旋转期间(参见图8的示意图(b))由机器人覆盖的区域。该区域在很大程度上取决于机器人的形状。对于圆形对称的机器人来说(参见来自图3中的示意图(a)),该区域是不存在的,因为由于对称性,附近的障碍物对旋转自由度不产生任何限制。对于在图10的示意图(b)中所示的D形的机器人100来说,区域I分解成两个独立的部分,这两个独立的部分由两个拐角(在机器人处的前方左侧和右侧)确定。机器人的后部部分是圆形的,因此在该区域中不存在对旋转自由度的任何限制。如果旋转不是围绕中心点进行,而是围绕另一个点进行(参见图8的示意图(c)和(d)),那么区域I被相应地变大和移动。

[0129] 附加地,对于区域区III,可以将旋转运动结束之后的运动的可能性,例如直线运动,一并包括到评估中。在这种情况下,也可以评估在位于机器人旁边的区域II和IV中的障碍物。例如可以规定,仅当随后的直线运动可以以可预先确定的距离(例如 $l_{\min}$)执行时,才可以判断朝向轮廓W的旋转(第二基本运动)是合适的。旋转的角度和跟随在该旋转之后的平移运动的距离 $l_{\min}$ 可以彼此进行协调。例如,如果机器人远离轮廓的距离比轮廓跟随距离更远,那么机器人应该能够朝着轮廓的方向上旋转,如果机器人与轮廓的距离小于或者等于轮廓跟随距离,那么应该阻止选择朝着轮廓的方向的基本运动(因为在旋转之后就不可能再继续进行直线向前运动)。这种行为可以通过 $l_{\min}$ 和旋转角度的协调来实现。

[0130] 应该注意的是,如果在旋转之后不可能进行直线运动,那么在下一个步骤中可能必须撤销被规划的基本运动(即旋转),这一般是不希望的并且应该避免。在提到的条件下,可以避免这种情况,即在旋转之后应该能够以可预先确定的距离进行直线运动。这并不意

意味着必须实际上立即接着进行该直线运动。替代地或者附加地,机器人也可以检查另外的附加的旋转,类似于图10的示意图(a)所示的示例中的在距离 $l_{\min}$ 上的直线运动。

[0131] 在自主移动机器人的一些应用中,期望机器人必须尽可能少地向后行驶。如果机器人针对每个(基本)运动,尤其是每个直线运动(第一基本运动)检查,是否在执行了通过区域I的直线运动之后可以在不会发生碰撞的情况下进行完全的旋转或者部分的旋转,那么可以减少向后行驶的频率。区域III示例性地显示了这样的区域,在该区域中不允许存在任何障碍物,以便机器人能够围绕中心点旋转。区域III'示例性地显示了这样的区域,在该区域中不允许存在任何障碍物,以便机器人能够围绕在中心点上方的点进行圆周运动(参见来自图8的示意图(c)中的情况)。

[0132] 附加地,例如在清洁机器人的情况下,被处理的区域(即例如由处理单元160覆盖的区域)可以作为地图信息被存储起来,并且可以被用于评估机器人的运动。同时,可以确定要评估的基本运动的处理增益并且用于对其进行评估。例如,通过这种方式可以识别出机器人何时完全沿着轮廓行驶并且再次到达先前已经处理的区域(特别是,但非仅仅是,轮廓跟随行驶的起点)。与(基本)运动相关的处理增益可以例如是那个(尚未被处理的)地板表面,该地板表面在运动期间会被附加地处理。也可以对该区域进行加权(例如根据地板覆盖物或者机器人所在的房间)。

[0133] 例如,机器人可以与一个它先前已经清洁过的、但是必须再次沿着行驶的轮廓之间保持更大的距离。此外,较大的轮廓跟随距离可能会要求较低的导航准确性,以避免没有预料到的碰撞,因此,例如可以扩大机器人的规划范围和/或速度。例如,可以保持与在机器人前方的障碍物之间的更大的距离,由此机器人不再如此频繁地和如此远地行驶到拐角中和其他狭窄的地方(潜在的死胡同)。

[0134] 参数选择:涉及在此处描述的示例的、用于借助于三种或者更多种不同的基本运动和根据简单的可预确定的规则对这些运动进行评估来控制机器人的方法是一种非常强大的工具,借助于该工具,原则上可以在轮廓跟随行驶中生成大量的运动曲线用于各种不同的使用目的。但是,大量的参数(评估规则、旋转运动的旋转点、要覆盖的距离和旋转角度)的选择会迅速变得混乱和复杂。借助于模拟,原则上可以分析具有给定的参数集的机器人的行为并且将其与期望的行为相适配。

[0135] 附加地,诸如机器学习(机器学习)的方法之类的优化方法的使用使得参数的至少部分自动化的确定成为可能。例如,可以预先给定某些场景(诸如墙壁和椅子腿之类的障碍物的不同的布置)并且借助于可预先确定的测量函数进行优化。例如,可以将墙壁附近的被处理的区域最大化或者可以将所需的时间最小化。附加地或者替代地,可以预先给定人所期望的运动模式(例如,基于市场研究来确定)。可以如此地优化参数,以使机器人路径(模拟地和/或在测试中)尽可能接近预先给定的运动模式。

[0136] 不可见的障碍物:如所描述的那样,轮廓跟随行驶可以基于环境的信息以及基于地图数据而在很大程度上不发生碰撞地被规划和执行。特别地,对基本运动或者用于结束轮廓跟随模式的标准的评估可以基于地图来进行。附加地,机器人可以具有合适的紧急情况例行程序(例如由控制单元150执行的软件模块,参见图2),该紧急情况例行程序可以在发生事先未预料到的事件时启动。例如,可以中断规划的运动并且由此可以使机器人100停止,以避免事故或者限制其影响。有关事先未预料到的事件的信息可以例如被接收到地图

数据中并且用于进一步控制机器人。在完成紧急情况例行程序之后,可以继续执行被以这种方式中断的轮廓跟随模式,或者可以重新规划机器人的当前任务。

[0137] 这种事先未预料到的事件例如是检测到一个坠落边缘,如在楼梯中的情况那样,仅在机器人接近该坠落边缘时和/或至少部分地已经驶越该坠落边缘时,该坠落边缘才被相应的传感器识别。用于事先未预料到的事件的另一个示例是触摸到障碍物(例如碰撞)。之所以会发生这种情况,是因为障碍物先前未用导航传感器进行识别和/或未被记录到地图数据中。这可能在低的、透明的或者反光的障碍物的情况下发生。在某些情况下,例如由于地面不佳,也不能够按规划的那样执行行驶操作,由此机器人无意中与先前检测到的障碍物发生碰撞。也可能发生这样的情况,即障碍物运动(例如由于人或者动物的影响)并且由此引起碰撞。

[0138] 除了立即停止机器人之外,还可以在紧急情况例行程序的范围内执行与没有预料到的事件(其已经触发紧急情况例行程序)相适应的标准化的运动。例如至少可以在这样的程度上将最后的运动颠倒过来(反过来),即机器人处于与检测到的坠落边缘相距一个安全的距离下,和/或用于检测碰撞或者触摸障碍物的触觉传感器被再次释放(即不再检测障碍物)。例如,机器人可以向后行驶几厘米。如果在旋转期间发生没有预料到的事件,那么机器人可以朝相反的方向旋转。

[0139] 在结束标准化的运动之后,可以恢复正常的轮廓跟随行驶。可以将没有预料到的事件的原因输入到地图中,以便可以考虑该原因用于进一步评估机器人的运动。这例如是检测到坠落边缘的地点。该地点可以基于机器人的姿态(位置和定向)以及检测到该坠落边缘的传感器的位置(在机器人中)来确定。一般地,这是一个或者多个可被与障碍物的轮廓的点一样处理的点。

[0140] 由于触摸或者碰撞而发生的没有预料到的事件也被记录到地图中。在这种情况下,可能值得期望的是,用于检测碰撞或者触摸的触觉传感器具有相对良好的空间分辨率,从而可以以高的准确性将障碍物被触摸所在的位置记录到地图中。然而,在实际中,触觉传感器通常仅仅具有低的分辨率。在这种情况下,可以将机器人的触觉传感器的外部轮廓的整个部分,障碍物可能已在该部分处产生了所测量的传感器信号,(作为几何图形或者以采样点的形式)记录到地图中。例如,这可以是用于机器人的各个独立区域的接触开关。应当注意的是,从两个接触开关在接近的时间顺序下的触发中,可以导出关于碰撞的地点的附加的信息,该附加的信息可以以适当的方式记录到地图中。

[0141] 由于由此被接收到地图中的信息并不直接地对应于障碍物的位置,因此可能有必要将它们与先前描述的关于障碍物的信息不同地进行处理。这就是说,在评估基本运动时可以考虑障碍物的类型以及通过哪个传感器检测到障碍物(或者障碍物的一部分)的轮廓。例如可以乐观地解释该信息。这意味着,为了评估基本运动,假设可能已经产生传感器信息的是具有最小干扰位置的最小可能的障碍物。这可能导致与障碍物的进一步接触,由此关于未检测到的障碍物的触觉信息的数量增加。由此使机器人可以以摸索式的运动沿着障碍物运动。

[0142] 如已经针对死胡同状况的处理所提到的那样,可能有必要启动一种轮廓跟随模式,在该轮廓跟随模式下故意地冒碰撞的危险。这意味着,不使用或者仅在有限的程度上使用借助于导航传感器检测到的关于机器人的周围环境的信息和/或地图数据。因此,上面描

述的紧急情况例行程序也包括用于对周围环境进行触觉探索的方法,基于该探索,机器人可以在轮廓跟随模式下沿着轮廓运动。

[0143] 在一些实施例中,机器人可以被这样地构造成,即在与障碍物发生实际的碰撞或者接触之前,识别出与检测到的运动中的障碍物或者由于未按照规划执行的行驶操纵而发生碰撞的风险。然后可以立即停止机器人,由此防止碰撞。

[0144] 虚拟障碍物:在评估基本运动时可以以特殊形式考虑的、用于各种不同障碍物的其他示例是由用户引入到环境中的标记,其目的是界定机器人使用区域的、不允许机器人在其上行驶的区域。这样的标记例如是磁带和电流回路,该磁带和电流回路会形成一个可以被机器人检测到的磁场,或者是导引光束发射器,该导引光束发射器发射出可以被机器人检测到的光束(例如红外激光束)。这些标记可以借助于机器人100的传感器单元120的相应传感器来识别(参见图2),并且不会被机器人例如横穿过去。因此,对于机器人来说,这些标记代表了某种障碍物,在对机器人进行导航时可以考虑该障碍物。附加地,这种障碍物的轮廓可以在轮廓跟随行驶中被跟随。

[0145] 由于不可能与(例如磁性的或者光学的)标记发生碰撞,因此在评估基本运动时可以与例如通过距离测量或者借助于摄像机检测到的障碍物不同地对待这些标记。因此,不行驶越过该标记就足够了,而不必限制旋转自由度。因此,例如可以接受的是,D形机器人的一个角部(见图3)在旋转期间在标记上方掠过。

[0146] 尤其是在轮廓跟随模式下使用地图数据来控制机器人的优点是虚拟障碍物的可用性,这些虚拟障碍物在地图中标记出不允许机器人独立行驶和/或横穿通过的区域。这些区域可以例如由用户经由HMI 200输入或者由机器人独立地创建(“学习”)。通过这种方式,机器人可以记住它不打算再次行驶的区域,例如因为在此处不能保证安全的运行。由此,用户可以例如对机器人临时性地或者永久性地封闭某个区域,而不必将物理标记引入到该环境中。与真正的标记相比,这明显是更灵活的并且更少扰人。

[0147] 如适用于通过标记产生的障碍物一样,这同样也适用于这样的纯虚拟障碍物。由于不可能发生真正的碰撞,因此简化的处理就足够了。这只能被设计成用于避免行驶越过虚拟障碍物的虚拟边界,并且特别是防止驶入被封锁的区域。该方法将以举例说明的方式针对虚拟障碍物进行解释。

[0148] 图10示例性地示出了沿着障碍物(例如墙壁)的轮廓W和与轮廓W垂直的虚拟障碍物的轮廓V的轮廓跟随行驶,该虚拟障碍物包含在地图中,但是实际上并不存在。为了评估沿着墙壁的轮廓W的基本运动,将考虑机器人的完整的D形形状。相反,为了评估相对于虚拟障碍物的轮廓V的基本运动,仅考虑机器人10的简化的虚拟形状101。在图11所示的示例中,简化的虚拟形状101是一个圆,该圆的中心是中心点“×”,并且该圆的直径对应于机器人的宽度。通过这种方式,在静止状态下的纯旋转不受虚拟障碍物的限制,同时,通过将用于避免碰撞的常规规则应用于机器人的简化的虚拟形状,避免横向越过虚拟障碍物的轮廓V(也就是说,简化的虚拟形状101与虚拟障碍物的虚拟碰撞)。因此可以这样地选择圆的半径,使得当机器人100围绕其(运动)中心“×”旋转时,机器人100的外壳的外轮廓的至少两个点在该圆上运动。在所示的示例中,圆的半径等于机器人100的外壳的宽度的一半。因此,机器人100的一部分位于圆的外部,该圆代表虚拟的形状101。

[0149] 在图11的示意图(a)所示的示例中,机器人100沿着墙壁的轮廓W运动,直到它到达

一个位置,在该位置上机器人100与虚拟障碍物之间仅还相距安全距离 d_s 。安全距离 d_s 可以与用于其他类型的障碍物的距离相同(参见图5)。或者,该安全距离可以选择得更大或者更小。特别地,可以将与在机器人100的简化形状101前面的虚拟障碍物之间的安全距离选择得与轮廓跟随距离相同。

[0150] 由于这样的事实,即在处理虚拟障碍物时只考虑了机器人100的简化的形状101,因此通过轮廓W和V形成的拐角不会导致死胡同状况;机器人可以在拐角处不受限制地旋转离开墙壁的轮廓W(例如通过作为在静止状态下的旋转的第三基本运动的顺序),由此使机器人100能够与虚拟障碍物的轮廓V平行地定向。如在图11的示意图(b)中所示的那样,在旋转(第三基本运动)期间,机器人的一个角部A突入到虚拟障碍物中。与来自图5中的示例不同,在这里不必要为了沿着相反的方向跟随轮廓W而离开当前的轮廓跟随模式。

[0151] 在图11的示意图(c)中的示例中,机器人与虚拟障碍物的轮廓V完全平行地定向,其中,轮廓V与机器人之间具有距离 d_s 。如果安全距离 d_s 等于轮廓跟随距离 d ,那么机器人现在可以继续跟随虚拟障碍物的轮廓V。如果安全距离 d_s 小于或者大于轮廓跟随距离 d ,那么控制单元150将在轮廓跟随模式下这样地控制机器人,即,使在虚拟轮廓V和机器人100(或者简化的机器人形状101)之间的距离对应于轮廓跟随距离。

[0152] 应当注意的是,轮廓跟随距离和安全距离,尤其是在用于处理地板表面的机器人的情况下,用来避免意外的碰撞。由于与虚拟障碍物发生这样的碰撞是不可能的,因此也可以根据障碍物的类型设置轮廓跟随距离和/或安全距离。特别是对于虚拟障碍物来说,轮廓跟随距离和/或安全距离可以比其他的障碍物设置得较小或者完全设置为零。特别地,用于虚拟障碍物的轮廓跟随距离和/或安全距离为零可以节省一些用于计算和评估的成本。

[0153] 机器人100的虚拟形状101的最大可能的简化在于通过一个单独的点表示机器人。这个点优选地是中心点“×”(运动中心、旋转中心)。例如,可以在轮廓跟随模式下这样地控制机器人,使得被简化成一个点的(点状的)机器人尽可能精确地在虚拟障碍物的轮廓V上运动。特别地,在这种情况下,可以将轮廓跟随距离 d 和安全距离 d_s 设置为零(也就是说不加考虑)。这意味着障碍物的边界描述了刚刚还被允许的机器人的(或者运动中心的)路径,因此针对机器人被封锁的区域没有被行驶。这一点可以在定义被封锁的区域时加以考虑。例如,用户可以通过HMI 200输入应该不被行驶的区域。然后可以基于该输入这样地确定虚拟障碍物的虚拟边界,即机器人可以以其中心点“×”在轮廓跟随模式下可靠地跟随该边界。

[0154] 在例如针对机器人100的简化的虚拟形状101对基本运动进行基于成本的评估的范围内,例如远离虚拟轮廓V的每个运动可以引起成本。与在可自由行驶的区域运动相比,在被封锁的区域内的运动在这种情况下可能会被禁止或者造成更高的成本。这基本上类似于基于轮廓跟随距离进行的评估。

[0155] 对于具有非常细长的形状的机器人(请参见图3的示意图(e))来说,将其简化成圆形形状或者简化成一个点可能会导致机器人的各个部分非常深地突入到虚拟障碍物中(也就是说,突入到被封锁的区域)。在这种情况下,也可以使用另一种虚拟形状来简化导航和对运动的评估。为此,通过由圆弧、直线段或者简单的多边形块(例如抛物线)组成虚拟形状,该虚拟形状可以被尽可能简单地确定。特别地,可以选择这样的圆弧,该圆弧的中心是中心点“×”(运动中心)。

[0156] 例如,可以将虚拟形状101选择为凸形形状(也就是说,可以通过在该形状内的一

个线段连接任意两个点)。例如,可以这样地选择虚拟形状101,即,该虚拟形状被完全包含在真实形状中。因此,位于虚拟形状之外的机器人的区域可以至少暂时地,例如在旋转过程中超出虚拟障碍物的虚拟轮廓。因此允许相对于虚拟障碍物的运动,尤其是旋转,这些运动在真实障碍物的情况下将导致碰撞。

[0157] 例如,可以这样选择虚拟形状101,即不超过在机器人100的真实形状的点 and 机器人的虚拟形状100之间的最大距离。类似于简化成一个点的情况,该虚拟形状可以是一条线,其中该线的一个点是中心点“×”(运动中心)。例如,这是该线的端点之一。那么第二端点可以考虑机器人的细长的形状(参见图3的示意图(e))。

[0158] 应当注意的是,在这里描述的机器人100的简化的虚拟形状101的概念可以被一般化为机器人的完整的三维形状。例如,简化的三维形状是圆锥体或者其他的旋转体。特别地,通过到平面中的适当的投影,可以将三维的问题化为在这里描述的二维的情况。

[0159] 用于非圆形机器人的简化的路径规划:图11示出了用于机器人的路径规划的两个等效的视图,该机器人具有从起点到达目标点的基本上圆形的基面。在图12左边的示意图(a)中所示的情况下,通过数个较小的障碍物H(例如椅子腿)来确定用于机器人100的不发生碰撞的路径。这种情况等同于在图12中右边的示意图(b)中所示的情况,在该情况下可以通过数个障碍物H确定用于点状的机器人100'的路径,其中,这些障碍物H(与在示意图(a)中所示的情况相比)被扩大了机器人100的半径。通过示意图(b)所示的问题是可以直观地解决的,因为未被障碍物占据的任何点都是机器人的可能的位置。

[0160] 在图12中所示的方法原则上对于机器人100的一般的非圆形形状也是可行的。但是,用于扩大障碍物的规则取决于机器人的定向。同时,必须遵守以下的限制条件:运动只能平行于机器人的定向来进行。由此数学公式变得非常复杂并且计算成本高。如果必须附加地考虑机器人和环境的三维形状,如在来自图1的示意图(b)中的示例的情况中那样,那么成本将进一步上升。因此需要一种更简单的方法。

[0161] 用于具有复杂形状的机器人的路径规划的问题可以通过以下方式简化,例如通过将用于机器人的简化“虚拟”形式的已知路径规划方法用于大空间的路径规划,尤其是在很大程度上自由的区域中。在狭窄的地方,可以在关注确切的机器人形状的情况下使用局部规划。例如,为此目的,可以使用在此处描述的用于执行轮廓跟随行驶的方法,以便确定例如通过在图12的示意图(a)中所示的、具有复杂环境的区域的路径。

[0162] 在图1中示出了一种示例性的情况,在该情况下,可以使用路径规划与机器人的简化虚拟形状以及对机器人的完整形式的局部考虑的组合。在两个视图中(参见在图1中的示意图(a)和(b)),机器人以其优选的行驶方向直接地位于墙壁的前面。由此,既不能够进行沿着优选的方向(向前方向)的运动,也不能够进行在静止状态下的旋转。在采用机器人的简化虚拟形状进行路径规划的情况下,这个问题会被忽略,由此大大地简化了必要的算法和计算。当尝试沿着以这种方式规划的路径行驶时,那么机器人会检测到,由于有障碍物(也就是说墙壁)而不能够定向到规划的路径上,因此例如轮廓跟随模式被启动。

[0163] 如上关于对死胡同状况的处理所描述的那样,这将导致机器人逆着优选的行驶方向(即,向后)运动一点,直到机器人可以自由旋转。上级控制机构现在例如可以确定,机器人现在可以沿着规划的路径定向并且跟随该规划的路径,因此轮廓跟随模式被结束。机器人可以以向后行驶的方式远离墙壁而所借助的预先确定的特殊行驶操纵是不需要的。同样

地,用于这种小操纵的、复杂的、考虑机器人的整个轮廓的路径规划也是不需要的。因此,在此处描述的路径规划方法是灵活的、稳健的并且是节省资源的。

[0164] 在这种情况下,机器人的简化虚拟形状特别地对应于一种圆形形状,该圆形形状的中心位于中心点“×”(运动中心)上。由此,可以使用例如在图12中概略描述的用于路径规划的方法。路径规划的结果是路径P,该路径可以被转换为用于机器人100的驱动单元170的相应的控制命令。在这种情况下,基于借助于传感器单元120探索的有关机器人的周围环境的信息,对机器人的控制连续地进行修正。例如,在这种情况下可以预先给定所期望的精度,机器人以该精度跟随路径P。用于沿着路径控制机器人100的方法本身是已知的。例如,可以使用轮廓跟随行驶的形式来跟随路径,类似于如上针对跟随虚拟障碍物的轮廓V所描述的情况(其中机器人形状简化为虚拟点)。

[0165] 路径规划一般以地图数据为基础,该地图数据或多或少地完全描述使用区域。在这些全局地图中,被探测的细节的准确性通常是被降低的,以限制存储需求和/或计算的复杂性。用于这种地图的示例包括:

[0166] • 特征地图,该特征地图可以以点、线和/或面的形式表示障碍物的轮廓,

[0167] • 网格地图(也称为栅格地图),在该网格地图中使用区域的面被划分为单独的网格单元,并且可以为每个网格单元进行标记,以确定该网格单元是否被障碍物占据还是可以在其上自由地行驶,

[0168] • 拓扑地图,该拓扑地图包含以可由机器人行驶的方式将使用区域的特征点和/或特征区域连接起来的信息。

[0169] 对于这些地图来说,用于路径规划的方法本身是已知的并且可以任意地进行组合。

[0170] 为了沿着路径控制机器人,机器人可以具有第二地图或者第二形式的地图数据,该第二地图或者第二形式的地图数据包含更多细节以及关于利用传感器单元120的传感器获取的周围环境的当前信息。特别地,在机器人沿着路径P行驶期间,可以在该第二地图中以高的准确度输入关于周围环境的当前信息。该输入到第二地图中的信息可以在一段时间之后被再次删除,以减少存储需求 and 处理工作量。可替代地,可以在一段时间后通过解释和/或简化来减少第二地图的信息内容,由此同样减少了存储需求 and 处理工作量。

[0171] 在某些情况下,基于第二地图以及基于利用传感器单元120的传感器获取的周围环境的信息,例如可以确定,进一步跟随规划的路径可能会导致与至少一个障碍物H发生碰撞。为此特别考虑了机器人的完整形状。也可以通过检测实际的碰撞来确定,规划的路径由于有障碍物而不能够在不发生碰撞的情况下被行驶。

[0172] 这种可能的即将发生的碰撞的一个原因尤其可以在于,机器人100的简化的虚拟形状101不会与障碍物(虚拟地)碰撞,并且在规划中仅仅考虑了这种简化的形状。尤其是在圆形机器人的情况下也可能要考虑的其他原因可能是:错误的或者不准确的地图数据、在机器人的大空间的运动的规划上的有限准确性、障碍物的位置变化(例如由于椅子的使用)和/或新的障碍物。

[0173] 在检测到这种即将发生的碰撞之后,机器人可以对其做出反应,从而避免碰撞。在这种情况下,例如可以考虑机器人形状的完整轮廓。例如,控制单元150可以在轮廓跟随模式下这样地进行控制,即机器人跟随障碍物的轮廓,直到机器人再次遇到原始规划的路径、

可以到达目标点或者终止条件被满足为止。

[0174] 例如,可以在轮廓跟随模式开始之前设置一个附加的目标点。该附加的目标点可以是原始规划的路径的一部分,以便机器人可以从该点开始继续跟随原始规划的路径。该附加的目标点(只要可能的话)被设置在要避开的障碍物的后面。例如,当在机器人和目标点之间没有障碍物存在并且机器人可以在不发生碰撞的情况下朝向目标点旋转时,则该目标点是可以到达的。

[0175] 一个终止条件例如是,由于目标点位于障碍物中,因此该目标点是不可到达的。另一个终止条件可以是,机器人与目标点之间的距离变为大于一个可预先确定的值,和/或机器人与原始路径之间的距离变为大于一个可预先确定的值。因此,障碍物的轮廓会将机器人非同寻常地引导偏离其原始的路线。最大距离的可预先确定的值例如是机器人的宽度或者是机器人的宽度的两倍。另一个终止条件例如是在轮廓跟随行驶期间所需要的时间和/或所走过的路程。

[0176] 如果满足终止条件,那么机器人将停止并且检查是否有从其当前位置到达目标点的另一个路径。为此目的,在地图数据中记录以下信息:在先前规划的路径上不能够成功地行驶,并且在哪个位置上或者在哪个区域中发生了沿着路径P的运动的中断和终止。特别地,如果一个路径对于机器人的简化形状101是可以通过的而对于机器人的完整形状是不能够通过的,那么将信息存储起来,用于将来的、采用机器人的简化虚拟形状的路径规划。

[0177] 路径P的路径规划可以“以悲观的方式”进行,其中,机器人在基于理想的地图(没有误差或者有限的准确性)进行规划时始终会达到目标。这例如通过选择机器人的简化虚拟形状作为机器人的外接圆来实现。这意味着,机器人的所有的点都完全位于圆中,并且中心对应于中心点(也参见图13,外接圆102),因此,机器人在静止状态下的旋转在路径P的每个点处都是可能的。在这种情况下,机器人的虚拟形状101可以比实际的机器人100较宽,由此,不会行驶通过两个障碍物之间的狭窄通道。

[0178] 替代地或者附加地,路径P的规划可以“以乐观的方式”进行。在这种情况下,例如假设将圆形形状用于简化的虚拟轮廓,该圆形形状的直径对应于机器人的宽度。由此确保了,当路径穿过圆形形状的中心时,则机器人至少在两个障碍物之间通过。应当注意的是,这仅仅适用于理想的地图数据。在实际中,尽管如此,仍然可能发生这样的情况,即,在到达两个障碍物的前方时会发现不存在足够的空间来跟随路径在两个障碍物之间通过。附加地,在机器人的复杂形状的情况下,可能的是,不能够执行必要的旋转以跟随规划的路径P。

[0179] 悲观的方法的缺点是,在某些环境中或者在附属于这些环境的地图中,找不到从起点到达目标点的路径,尽管这在实际上是可能的。乐观的方法的缺点是,在这种情况下找到路径,这些路径在实际中不能够或者很难通过机器人来行驶。通过具体选择简化的虚拟轮廓,可以选择在乐观的方法和悲观的方法之间的任意的分级。

[0180] 在这种情况下,路径规划可以通过所提到的方法(乐观的、悲观的)的适当的组合来进行。例如,可以首先进行悲观的规划。如果悲观的规划是不成功的,那么进行乐观的规划,以检查是否有可能的路径。例如,可以进行悲观的规划和乐观的规划,以将结果彼此进行比较。规划的路径可以例如根据可预先确定的标准进行评估,并且选择具有最好的评估值的路径(例如最低的“成本”)。可预先确定的标准例如可以考虑路径的长度和/或其与障碍物之间的距离。如果悲观的规划导致的路径仅比乐观规划的路径“略微地更长”,那么可

以选择悲观的路径,在这种情况下,“略微地更长”可以是例如0.1米至10米的固定的可能的绕道,和/或是例如1.5倍至3倍长度的固定的系数。如果需要,在比较中可以考虑采用虚拟形状的其他变型方案的其他规划。

[0181] 可替代地,悲观的方法(例如,完全包围机器人的第一虚拟机器人形状)和乐观的方法(例如,不完全包围机器人的第二虚拟机器人形状)可以被组合在一种规划方法中。对此的一个简单示例在图13中示出,其中,所示的情况与在图12的示意图(a)中的情况非常相似。在这种情况下,分别为机器人地图的不同部分区域分配了成本(例如标量成本值),其中,这些成本特别地考虑了机器人100的实际形状,并且例如如果机器人由于位于附近中的障碍物而潜在地在其(围绕其运动中心的)旋转上受到限制,则这些成本将被设置得较高。在来自图13中的示例中,在具有小于或者等于距离 Δr 的位置处的成本等于 K_1 ,在其他的部分区域中等于 K_0 ($K_1 > K_0$),其中,例如值 Δr 是机器人100的“大的”外接圆102的半径(虚拟形状作为最坏情况考虑)和简化的机器人形状101的半径之差。然后,如在图12中所示的那样,实际的路径规划可以基于简化的虚拟机器人形状101(例如圆,该圆的半径对应于机器人的宽度一半,参见图11)进行,这允许减少至对点的路径规划(参见图12)。例如,在这种基于成本的规划中,可以根据到障碍物的距离来确定路径的成本(如所提到的那样,如果靠近障碍物,那么旋转被潜在地受到限制并且因此在地图的相应部分区域中的成本值就较高)。由此,在两个紧密聚集在一起的障碍物之间的路径(乐观的)可以引起比与围绕障碍物绕行的绕道相关联的路径(悲观的)更高的成本。通过选择用于使机器人运动靠近障碍物的成本,可以限定可接受的绕道,并且可以将其视为优化任务的结果。这种方法的优点在于,当乐观的方法导致一个路径时,该方法总是提供一种结果。同时,如此获得的路径总是在位于起点和目标点之间的狭窄地点与为了避开该狭窄地点所需要的绕道之间进行权衡考虑。在根据图13的示例中,成本 K_0 和 K_1 可以是离散值(例如, $K_0 = 0$, $K_1 = 1$),可替代地, K_1 也可以随着到障碍物H的距离的减小而增大。在这种方法中,可以通过以下方式考虑障碍物,即通过将将在被障碍物占据的区域中的成本选择为几乎无限大。

[0182] 如果在机器人100沿着路径P行驶期间检测到与障碍物H的可能的碰撞,那么机器人可以跟随路径行驶,直到驶近障碍物(例如在考虑安全距离下),并且然后直接变换到轮廓跟随模式。可替代地,机器人可以检查,是否存在有从障碍物旁边绕过的、带回到原始的路径P上的避开路径。

[0183] 特别地,可以根据机器人的当前任务来对与障碍物H的可能碰撞进行反应。例如,在用于处理地板表面的自主移动机器人的情况下在处理规划的路径期间,机器人可以尽可能靠近地驶近障碍物并且然后沿着障碍物的轮廓处理表面。另一方面,同一机器人可以在行驶期间一直行驶到被指派的要处理的区域或者基站、不应进行处理的区域。在这种情况下,可以确定围绕障碍物H的避开路径,以便更快地到达目标点(例如被指派的区域、基站)。

[0184] 为了确定避开路径,可以直接考虑机器人100的完整的形状。替代地或者附加地,可以基于机器人100的简化的虚拟形状101来做出关于围绕障碍物的路径是否是完全可能的初步规划。如果路径将在两个紧密聚集在一起的障碍物之间穿过,那么这一点是特别有用的。在这种情况下可以确定,简化的虚拟形状101不能跟随原始规划的路径,因此需要范围更广泛的绕道行驶和与之相关的路径规划。在这种情况下可以确定,简化的虚拟形状101可以跟随原始规划的路径而在两个障碍物之间通过。在这种情况下,在考虑到机器人的完

整的形状下,机器人可以例如使用轮廓跟随模式在障碍物之间运动通过。

[0185] 确定避开路径的结果可以是,可以在一定的距离下安全地绕开障碍物行驶。如果在原本在很大程度上为自由的区域中只有一个单独的障碍物,那么尤其是这种情况。另一个可能的结果是,可以在轮廓跟随模式下避开障碍物(在该轮廓跟随模式下机器人的完整的形状被考虑)。

[0186] 特别是在使用基于地图数据的轮廓跟随模式的情况下,一种用于检查机器人是否可以绕开障碍物行驶并且返回到原始路径P上的变型方案是该轮廓跟随行驶路线的预先计算(或者模拟)。如果有轮廓跟随模式启动的各种不同的可能性(特别是向右或者向左避开),以便找到最快通向目标点的路径,那么,除了其它方面以外,也可以使用这一点。

[0187] 为了规划避开路径,使用地图数据,该地图数据可以以高的准确性描述周围环境。例如,这根据在第二地图中收集的有关环境的信息来完成。为了限制用于存储需求和计算能力的资源消耗量,可以将避开路径的规划限制在小的区域(例如围绕机器人的、半径为0.5至2米的圆)上。

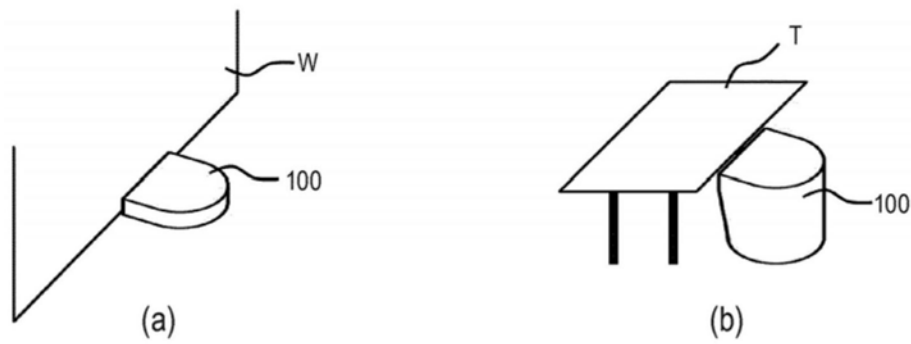


图1

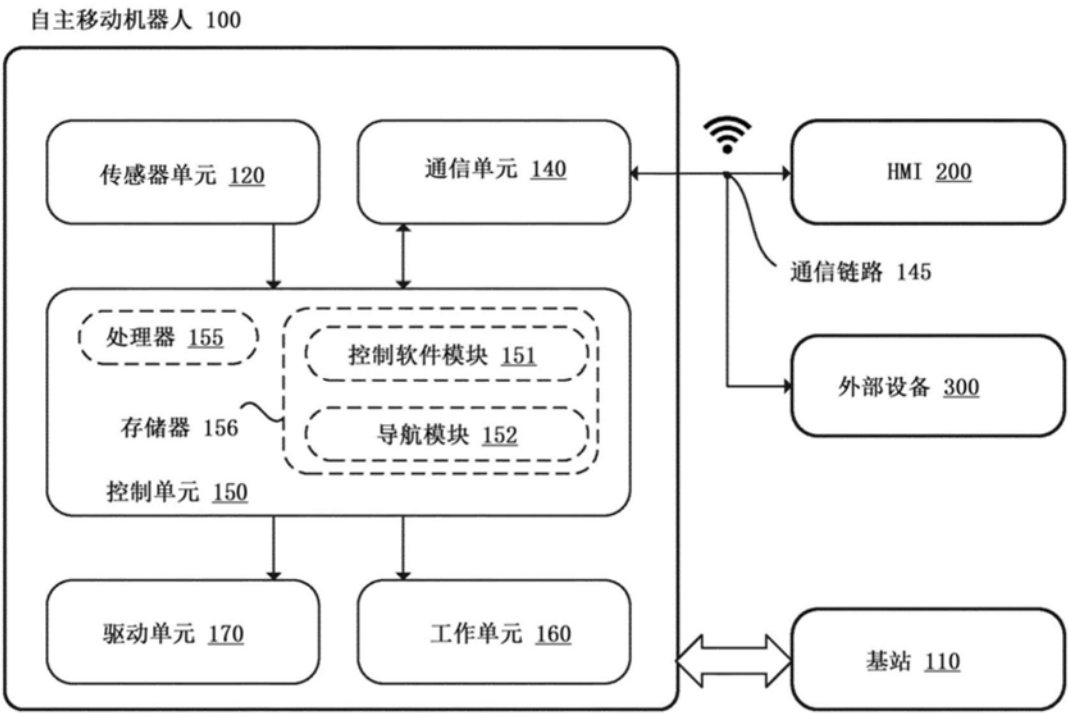


图2

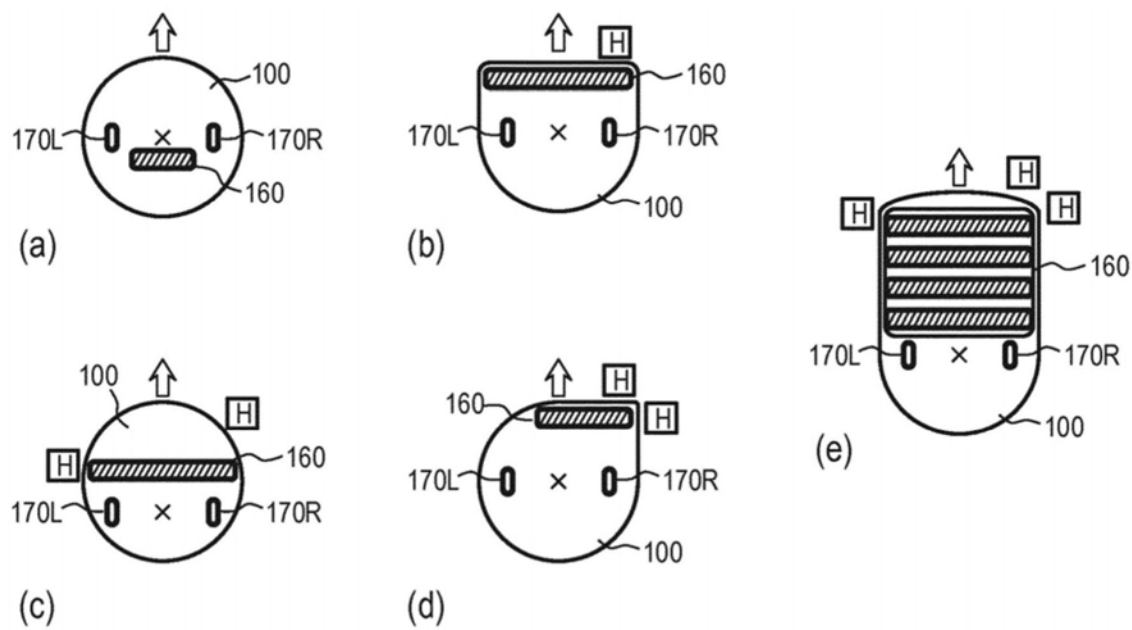


图3

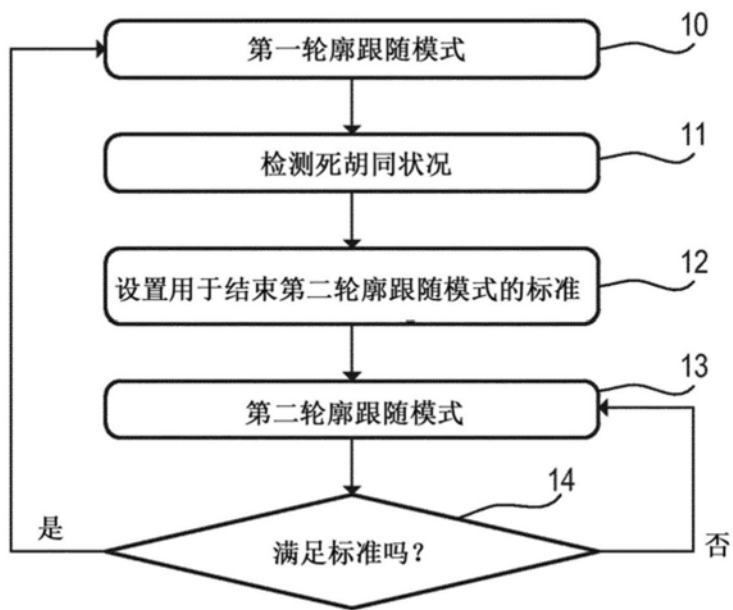


图4

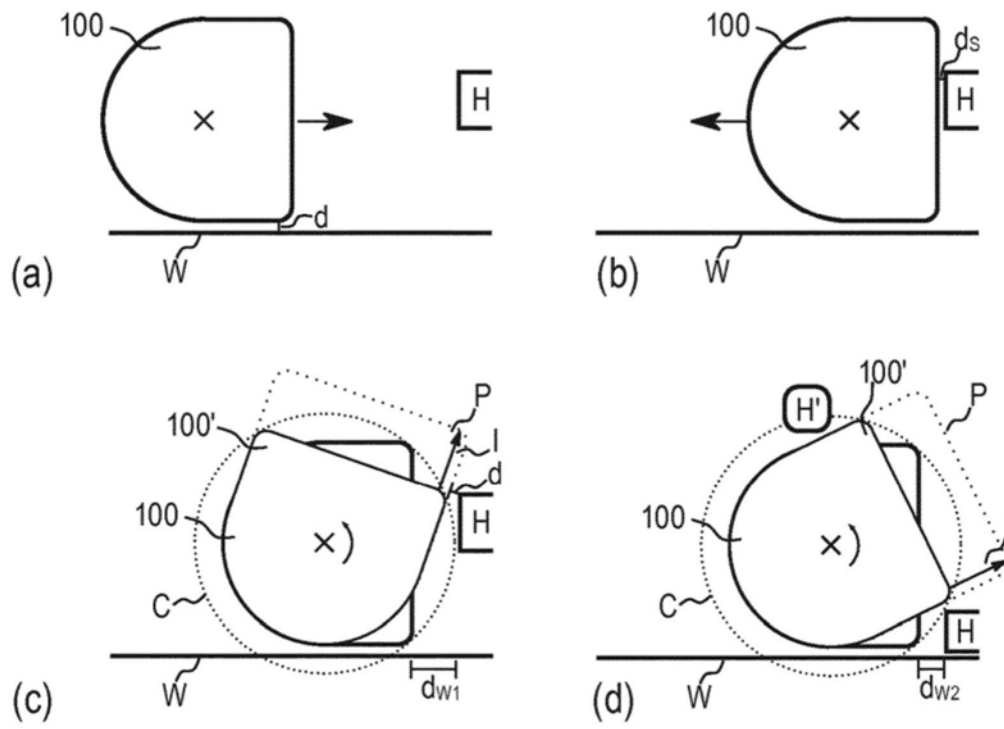


图5

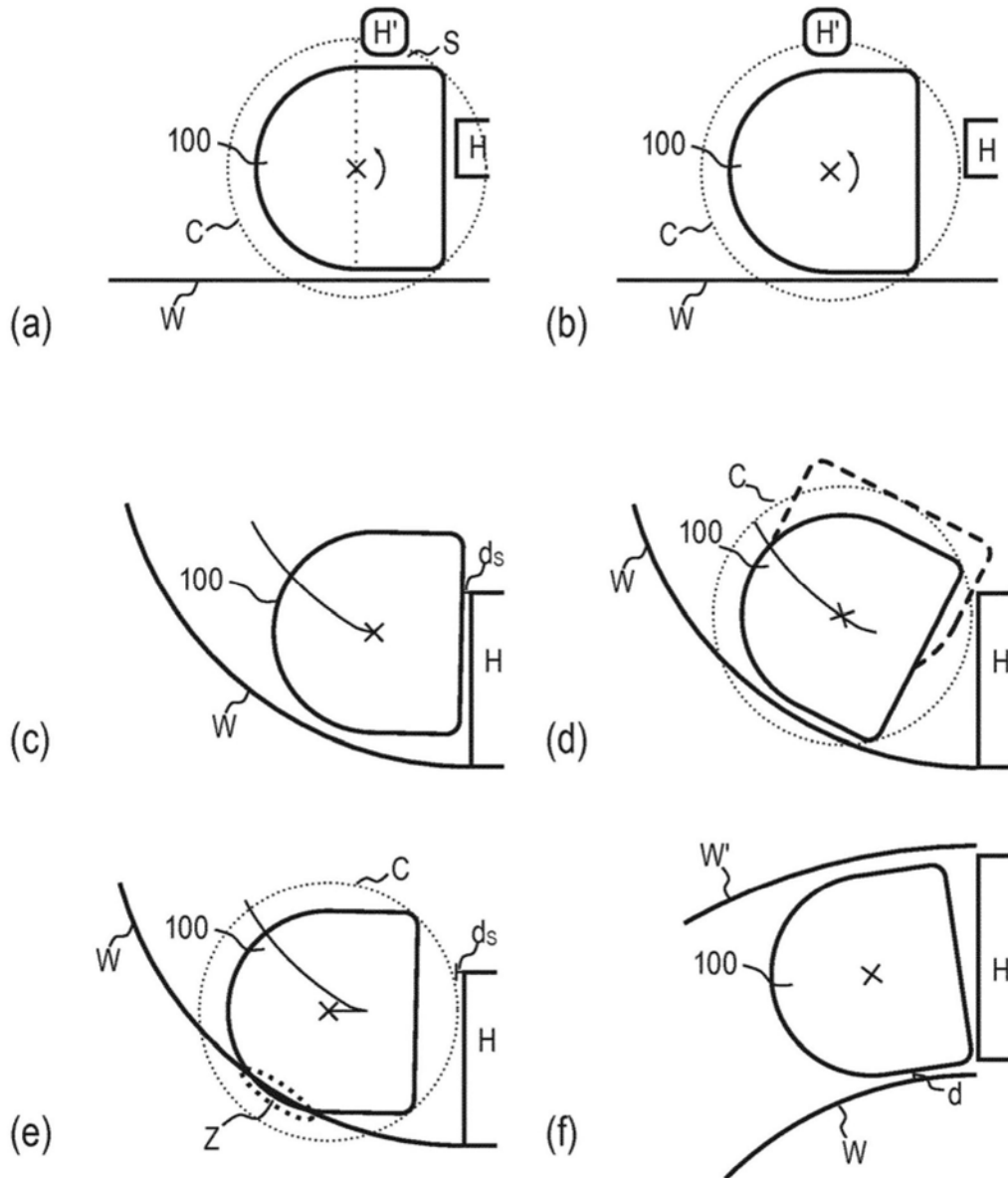


图6

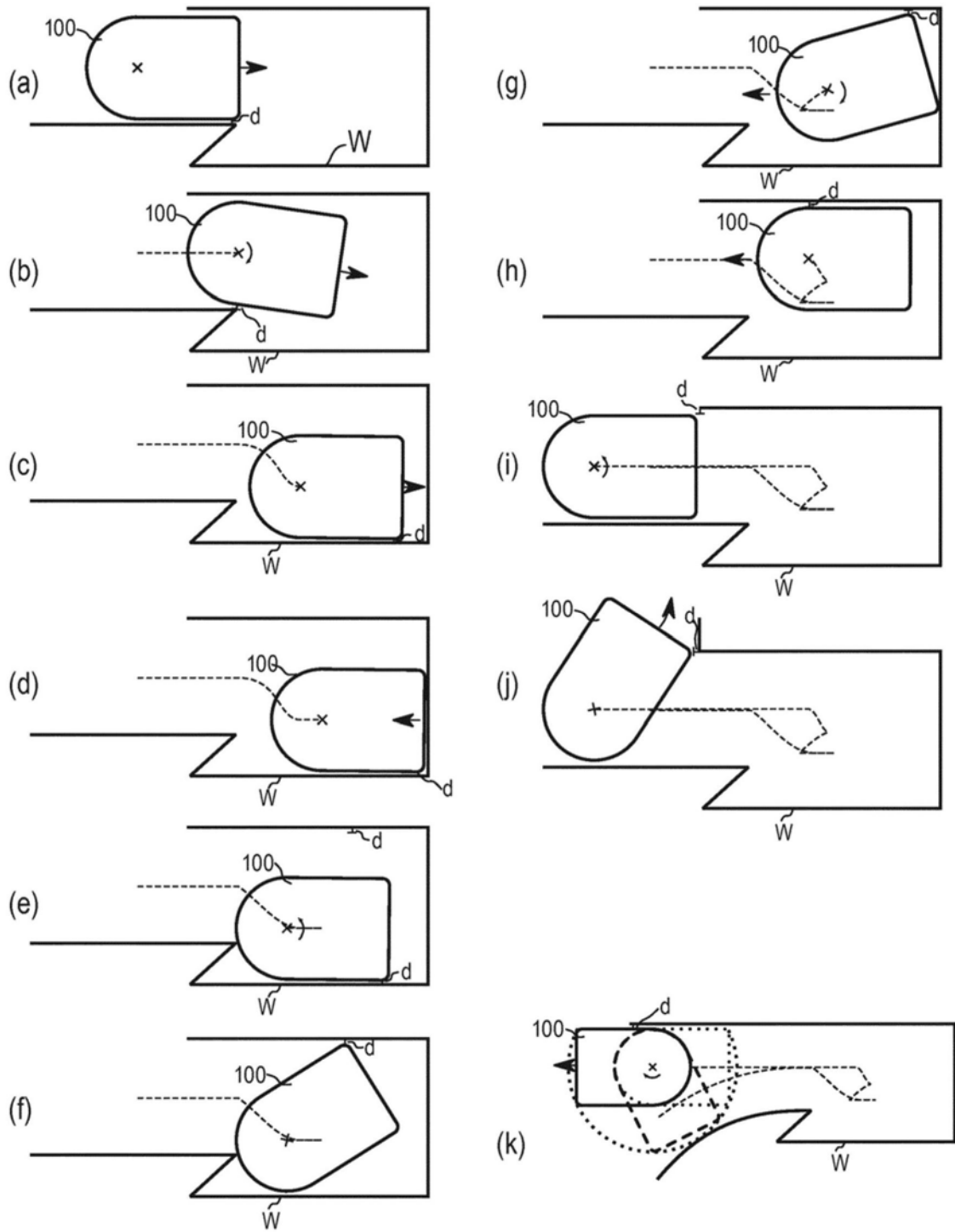


图7

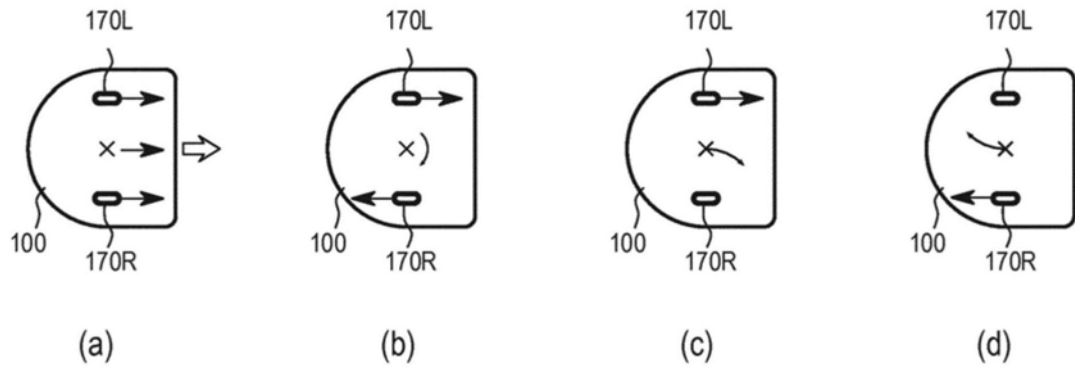


图8

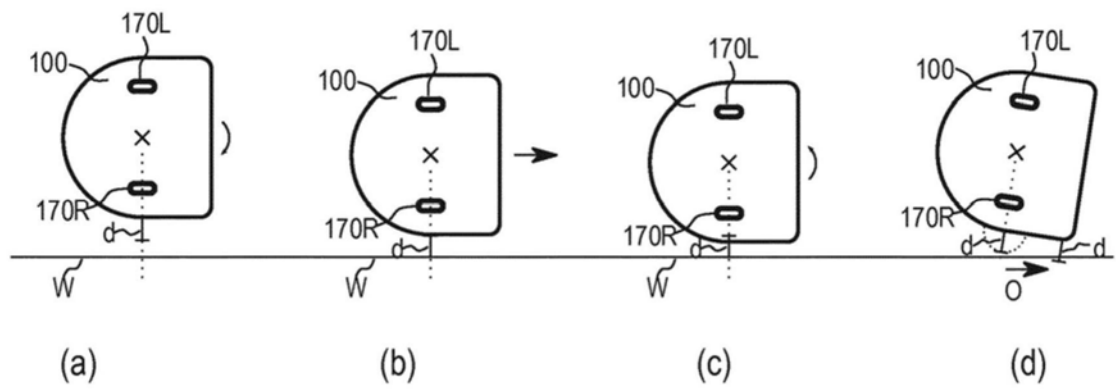


图9

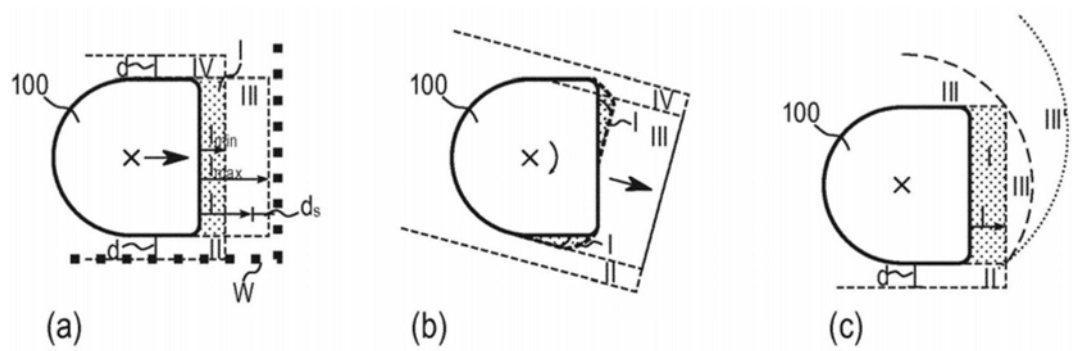


图10

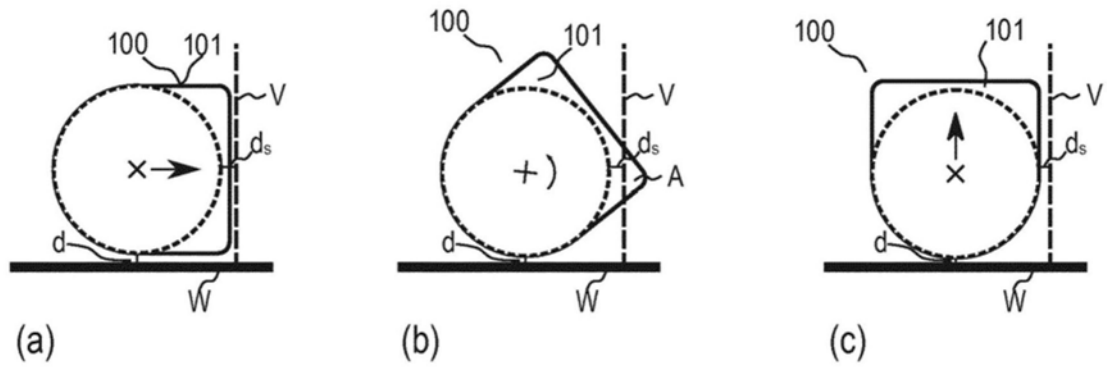


图11

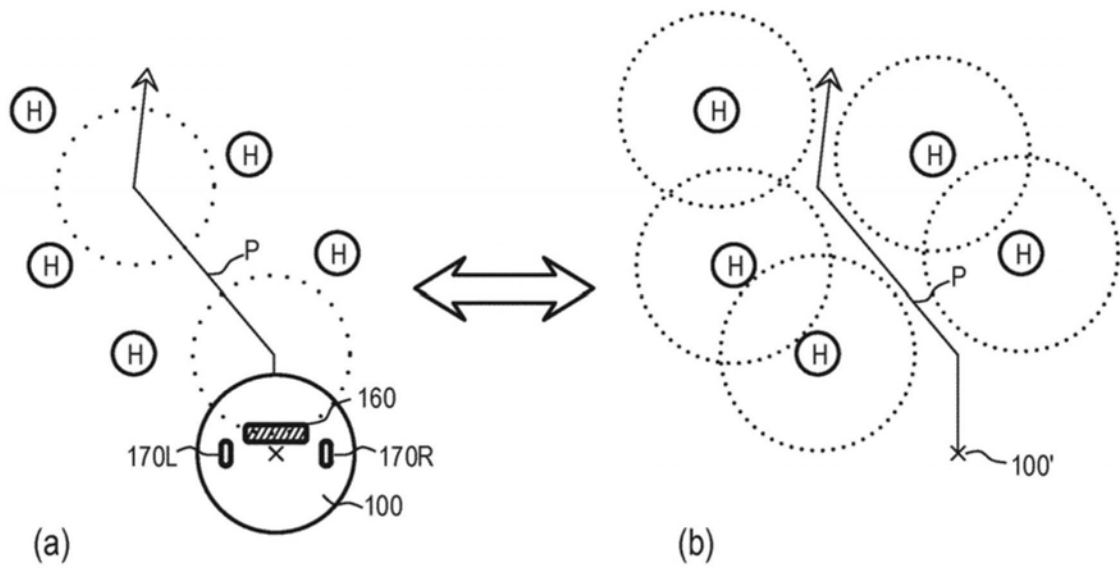


图12

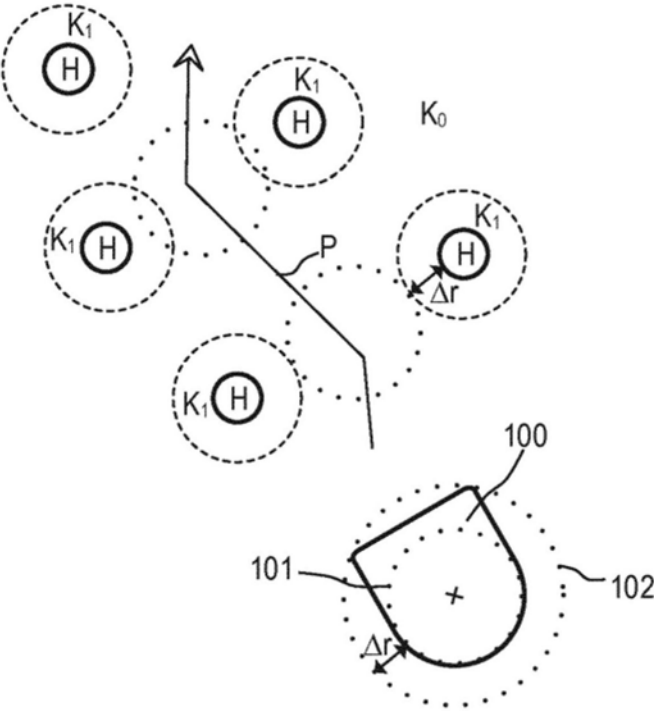


图13