

第七章 自动导引车

7.1 概述

自动导引小车是一种以电池为动力，装有非接触式导向装置的无人驾驶自动运输车(见图7-1、7-2)。其主要功能是，在计算机控制下，通过复杂的路径将物料按一定的停位精度输送到指定的位置上。



7.1.1 AGV的发展现状

20世纪50年代中期，Barret公司设计出无人驾驶卡车，也就是AGV的最早雏形。后来，美国物料搬运研究所将其定义为AGV，它是可充电的无人驾驶小车，可根据路径和定位情况编程，而且行走的路线可以改变和扩展。据报导，到1960年时，欧洲就安装了各种型号不同水平的自动搬运车系统，使用了13,000多台AGV。1959年AGV开始用于仓库自动化和工厂作业中。日本也从这时开始引进AGV技术。日本是使用这种车辆最多的国家，在20世纪80年代末，拥有各种类型的自动搬运车超过一万台，其生产厂家达47家，广泛应用于汽车制造、机械、电子、钢铁、化工、医药、印刷、仓储、运输业和商业上。

20世纪70年代，AGV作为生产组成部分进入了生产系统，并得到了迅速发展。1973年，瑞典VOLVO公司在KALMAR轿车厂的装配线上大量采用了AGV进行计算机控制装配作业，扩大了AGV的使用范围

7.1.2 AGV在AS/RS中的作用

控制台通过计算机网络接受立体仓库管理系统下达的AGV输送任务，通过无线局域网通讯系统实时采集各AGV、拆箱机器人的状态信息。根据需求情况和当前AGV运行情况，将调度命令传递给选定的AGV。AGV完成一次运输任务，在托盘回收站待命，等待下次任务。

各立体库出货口和拆箱机器人处均有光导通讯装置。对运行中的AGV，控制台将通过无线局域网通讯系统与AGV交换信息，实现AGV间的避碰调度、工作状态检测、任务调度。在立体仓库和拆箱机器人处通过光导通讯与AGV交换任务和状态，完成移载。

自动驾驶系统完成AGV的导引。充电系统由充电器和充电控制器组成，完成在线快速自动充电。AGV接受控制台的任务，完成运输。地面移载设备可实现AGV的自动移载、加载、交换空托盘。图7-4为在青岛海尔自动化立体仓库中移载用的激光导引AGV。

7.1.3 AGV的组成

AGV由以下各部分组成：车体、蓄电池、车上充电装置、控制系统、驱动装置、转向装置、精确定位装置、移载机构、通信单元和导引系统。

1. **车体。** 由车架和相应的机械电气结构如减速箱、电机、车轮等组成。车架常采用焊接钢结构，要求有足够的刚性。
2. **蓄电池与充电装置。** 常采用24V或48V直流工业蓄电池为动力。
3. **驱动装置。** 驱动装置是一个伺服驱动的变速控制系统，可驱动AGV运行并具有速度控制和制动能力。它由车轮、减速器、制动器、电机及速度控制器等部分组成，并由计算机或人工进行控制。速度调节可采用脉宽调速或变频调速等方法。直线行走速度可达1m/s，转弯时为0.2~0.5m/s，接近停位点时为0.1m/s。

4. 转向装置。

AGV常设计成三种运动方式：只能向前；能向前与向后；能纵向、横向、斜向及回转全方位运动。转向装置的结构也有三种：

- (1) **铰轴转向式三轮车型**。车体的前部为一个铰轴转向车轮，同时也是驱动轮。转向和驱动分别由两个不同的电动机带动，车体后部为两个自由轮，由前轮控制转向实现单方向向前行驶。其结构简单、成本低，但定位精度较低(见图7-5)。
- (2) **差速转向式四轮车型**。车体的中部有两个驱动轮，由两个电机分别驱动。前后部各有一个转向轮(自由轮)。通过控制中部两个轮的速度比可实现车体的转向，并实现前后双向行驶和转向。这种方式结构简单，定位精度较高(见图7-6)。
- (3) **全轮转向式四轮车型**。车体的前后部各有两个驱动和转向一体化车轮，每个车轮分别由各自的电动机驱动，可实现沿纵向、横向、斜向和回转方向任意路线行走，控制较复杂，见图7-7。

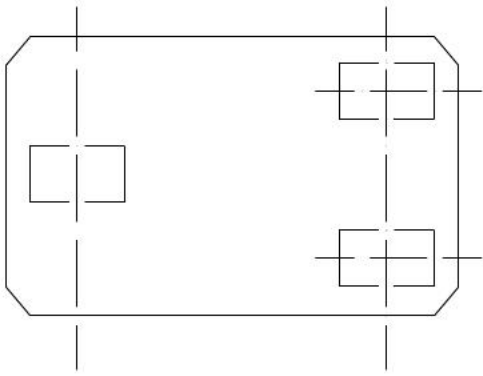


图7-5 铰轴转向式三轮车型

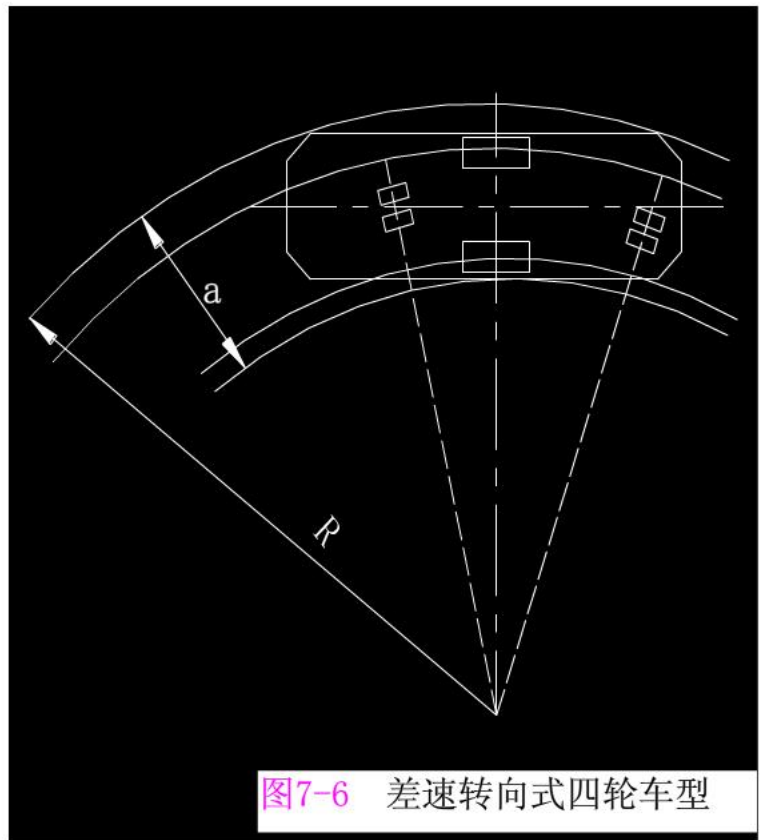


图7-6 差速转向式四轮车型

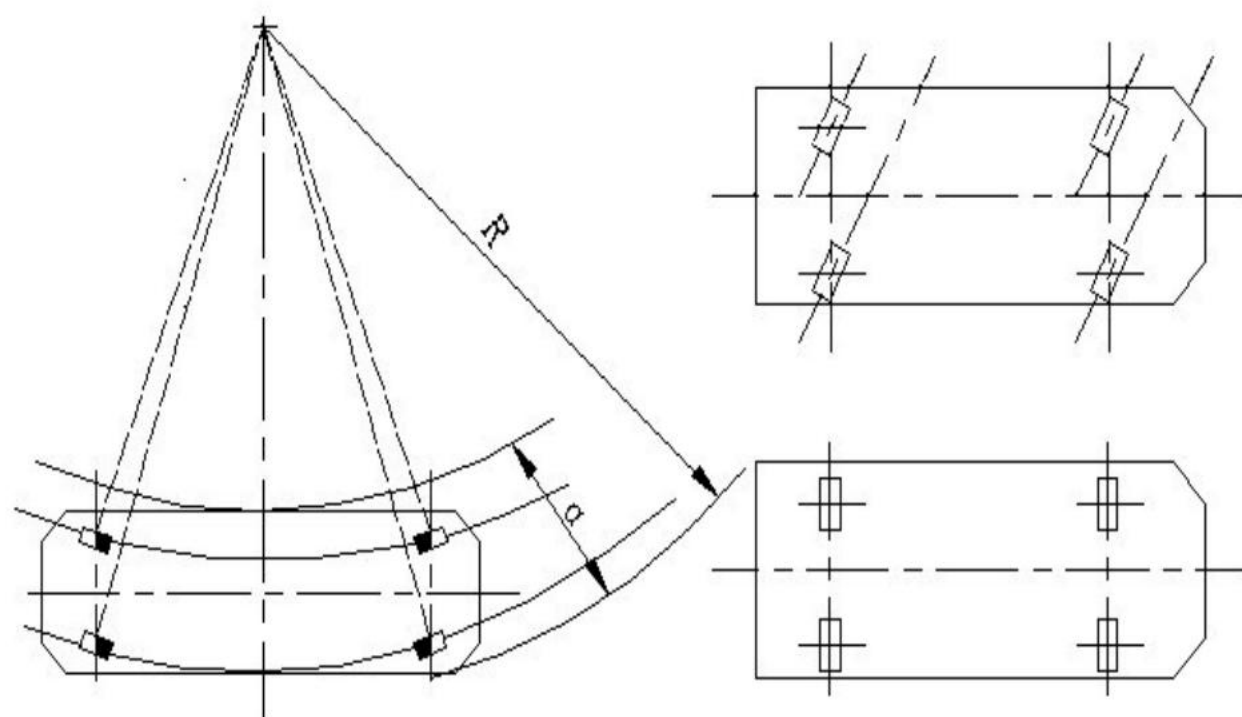


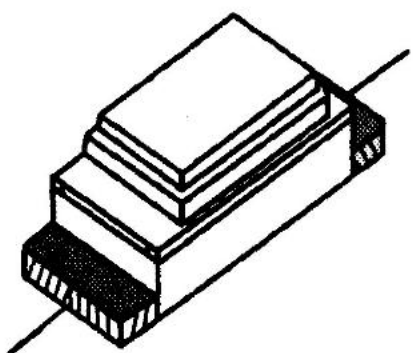
图7-7 全轮转向式四轮车型

5. 控制系统。 AGV控制系统包括车上控制器和地面(车外)控制器，均采用微型计算机，通过通信进行联系。

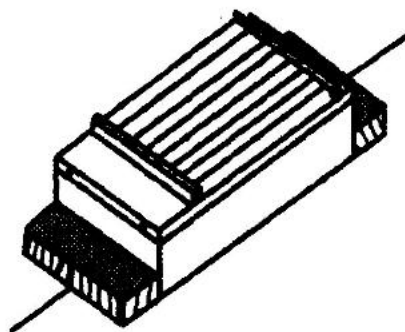
- 输入AGV的控制指令由地面(车外)控制器发出，存入车上控制器(计算机)；AGV运行时，车上控制器通过通信系统从地面站接受指令并报告自己的状态。车上控制器可完成以下监控：手动控制、安全装置启动、蓄电池状态、转向极限、制动器解脱、行走灯光、驱动和转向电机控制与充电接触器的监控等。
- 控制台与AGV间可采用**定点光导通讯和无线局域网通讯两种通讯方式**。采用无线通讯方式时，控制台和AGV构成**无线局域网通讯网**，控制台和AGV在网络协议支持下交换信息。无线通讯要完成AGV的调度和交通管理。
- 在出库站和拆箱机器人处移栽站都设有红外光通讯系统，其主要功能是完成移栽任务的通讯。
- AGV充电可以采用在线自动快速充电方式

6. 移载装置。AGV用移载装置来装卸货物，即接受和卸下载荷。常见的AGV装卸方式可分为被动装卸和主动装卸两种。

- 被动装卸方式的小车自己不具有完整的装卸功能，而是采用助卸方式，即配合装卸站或接收物料方的装卸装置自动装卸。常见的助卸装置有滚柱式台面[图7-8(b)]和升降式台面[图7-8(a)]两种。采用滚柱式台面的环境要求是站台必须带有动力传动辊道，AGV停靠在站台边，AGV上的辊道和站台上的辊道对接之后同步动作，实现货物移送。升降式台面的升降台下设有液压升降机构，高度可以自由调节。为了顺利移载，AGV必须精确停车才能与站台自动交换。

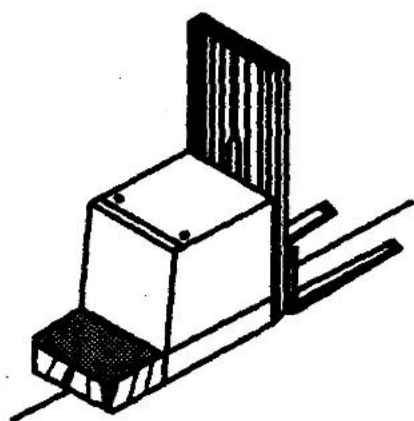


(a) 升降式台面

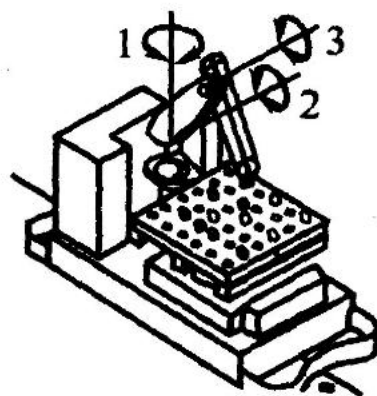


(b) 滚柱式台面

- 主动装卸方式是指自动小车自己具有装卸功能。常见的主动装卸方式有单面推拉式、双面推拉式、叉车式[图7-9(a)]和机器人式[图7-9(b)]四种。



(a) 叉车式



(b) 机器人式

图7-9 主动装卸装置

7. 安全装置。 为确保AGV在运行过程中自身安全，现场人员及各类设备的安全，AGV将采取多级硬件、软件的安全措施。在AGV的前面设有红外光非接触式防碰传感器和接触式防碰传感器——保险杠。AGV安装醒目的信号灯和声音报警装置，以提醒周围的操作人员。一旦发生故障，AGV自动进行声光报警，同时无线通讯通知AGV监控系统。

(1) 障碍物接触式缓冲器。 障碍物接触式缓冲器是一种强制停车安全装置，它产生作用的前提是与其它物体相接触，使其发生一定的变形，从而触动有关限位装置，强行使其断电停车。

(2) 障碍物接近传感器。 非接触式检测装置是障碍物接触式缓冲器的辅助装置，是先于障碍物接触式缓冲器发生作用的安全装置。为了安全，障碍物接近传感器是一个多级的接近检测装置，在预定距离内检测障碍物。在一定距离范围内，它会使AGV降速行驶，在更近的距离范围内，它会使AGV停车，而当解除障碍物后，AGV将自动恢复正常行驶状态。障碍物接近传感器包括激光式，超声波式，红外线式等多种类型

- 如日本产的红外线传感器能检测搬运车的前后方向、左右方向的障碍物，也能在二段内设定慢行和停止，也即2m内减速、1m内停车。发射的光频率数有4种或8种，能防止各搬运车间的相互干扰。

(3) 装卸移载货物执行机构的自动安全保护装置。AGV的主要功能是解决物料的全自动搬运，故除了其全自动运行功能外，还有移载货物的装置。移载装置的安全保护装置包括机械和电气两大类。如位置定位装置、位置限位装置、货物位置检测装置、货物形态检测装置、货物位置对中结构、机构自锁装置等结构。

7.1.4 AGV的典型产品

7.2 AGV的导引方式

AGV的导引方式，所谓AGV导引方式是指决定其运行方向和路径的方式。它不同于前面所说的一般通信。

常用的导引方式分两大类：车外预定路径方式和非预定路径方式。

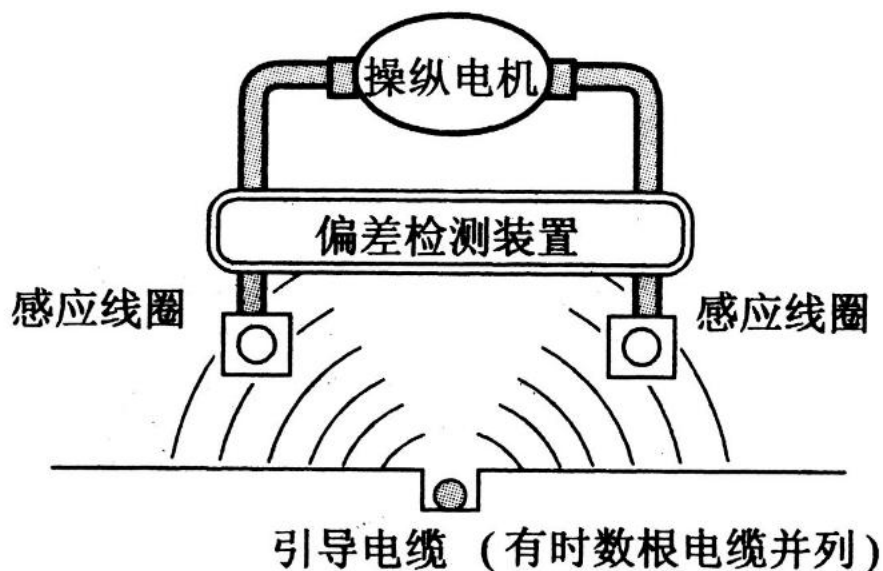
- 车外预定路径导引方式是指在行驶的路径上设置导引用的信息媒介物，AGV通过检测出它的信息而得到导向的导引方式，如电磁导引、光学导引、磁带导引(又称磁性导引)等。
- 非预定路径(自由路径)导引方式其一是是指在AGV上储存着布局上的尺寸坐标，通过识别车体当前方位来自主地决定行驶路径的导引方式，又称车上软件一编程路径方式；其二是激光导引。

AGV导引技术的研究十分活跃，具体体现为：

- ①路径的设定更加灵活机动。
- ②路径变更简单易行。
- ③提高对路面或环境变化的适应能力。
- ④精确地实时检测位置和方位值，提高引导性能。
- ⑤赋予感知和回避障碍的性能(智能)。
- ⑥具有人机对话功能
- ⑦更强的信息通信功能。
- ⑧系统尽量不依赖于中央计算机。
- ⑨多辆**AGV**协调工作。

• 为此，需要解决好以下几项关键技术：

- ①高精度且廉价的位置、方向检测手段。
- ②信息通信手段。
- ③图像处理和图像识别技术以及自动转换器的实用化。
- ④系统总体技术(特别是多辆**AGV**群控技术)。



7.2.1 预定路径方式

1. 车外连续标记

(1) 电磁导引

这是目前AGV采用最广泛的一种导引方式(见图7-10)。它需要在地面开槽(约51mm宽，15mm深)埋设电缆，接通低压、低频信号，在电线周围产生磁场。车上需安装有两个感应线圈，并使其分别位于此导引线的两侧。导向线中电流约为200~300mA，频率为2kHz~35kHz。

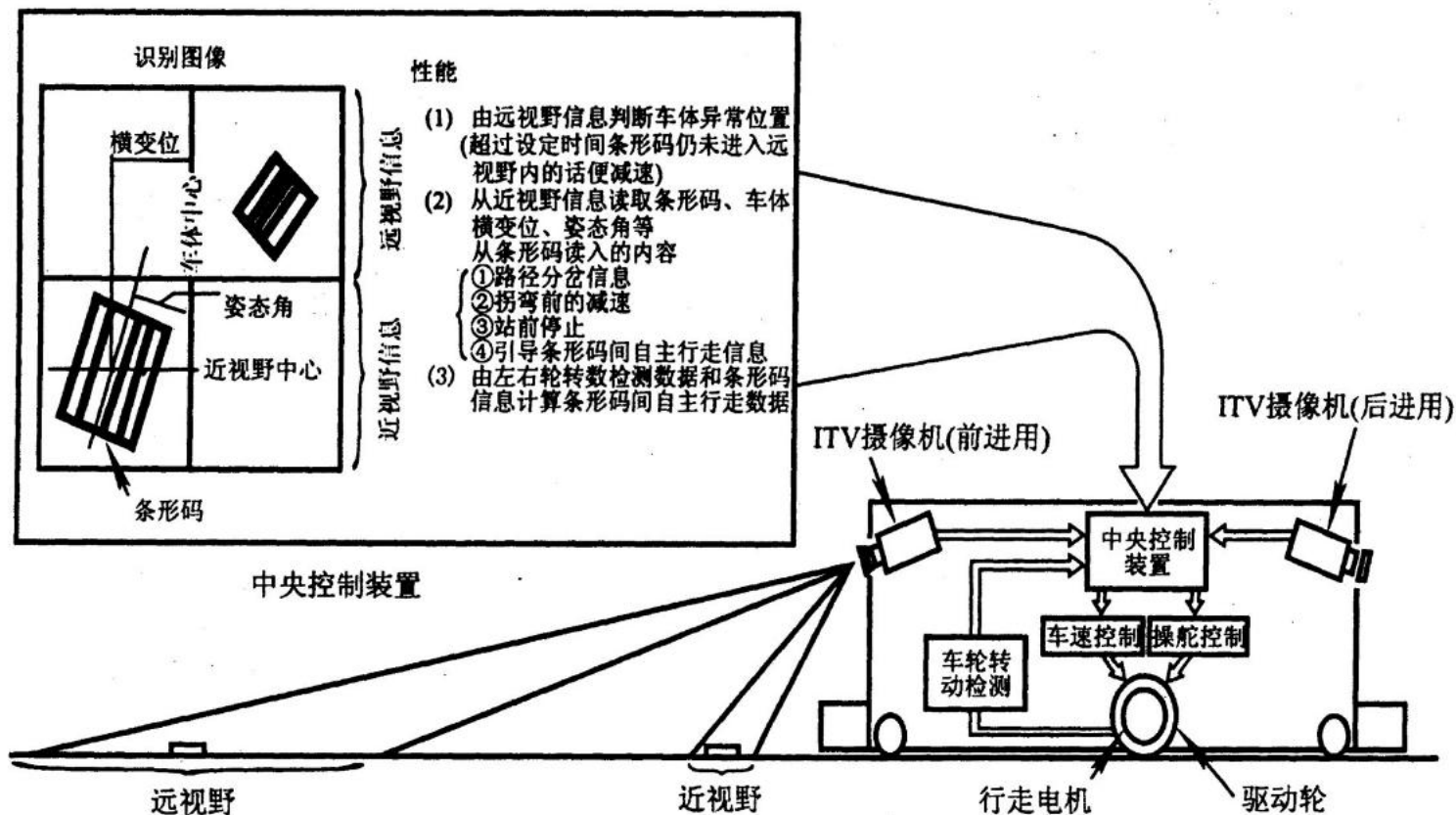
(2) 反光带或磁带导引

- 1) 反射式导向。如图7-11所示，这种引导方式是在地面上连续铺设一条用发光材料制作的带子，或者用发光涂料涂抹在规定的运行路线上，在车辆的底部装有检测反射光传感器，通过偏差测定装置到驱动转向电机来不断调整车辆前进的方向
- 2) 磁性式导向。是在地面上连续铺设一条金属磁带，而在车辆上装有磁性传感器，检测磁带的磁场，通过磁场偏差测定控制驱动转向电机来调整车辆行驶方向。

2. 车外间断标志

标志跟踪方式中有一种称为视觉引导法，即在所经路径上断续地设有若干引导标志或反射板(也可是玻璃球)，小车据此自动识别和判断路径(见图7-12)。

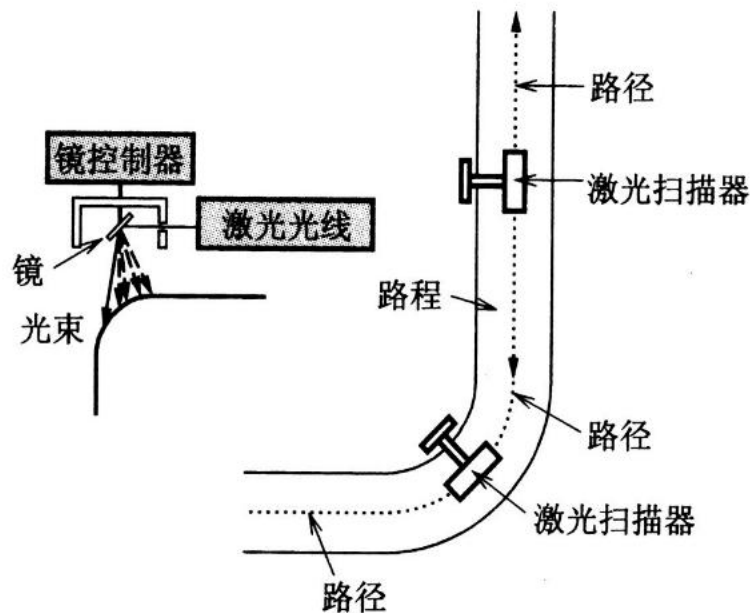
引导的标志除条形码外，还可用圆形、方形、箭头等图形，



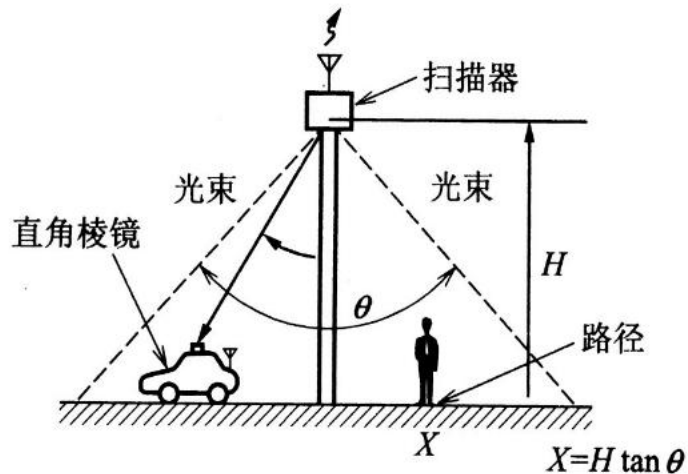
7.2.2 非预定路径

1. 激光导引

(1)光扫描导引。沿着路径从高处用光束进行扫描，计算机根据光信息(扫描角度以及扫描装置标号)，精密检测出AGV现在的位置(图7-13)。这种方法路径变换容易，扫描方式最简单。

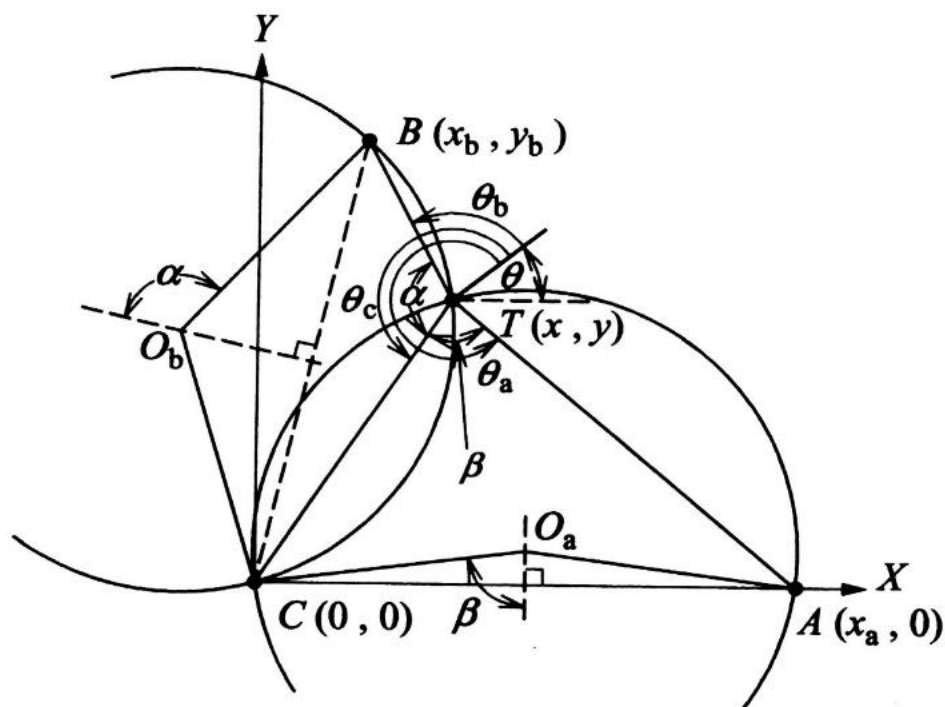


(a) 从高处用光束扫描路径



(b) 移动物体的 X 坐标可由 H 和 θ 求得

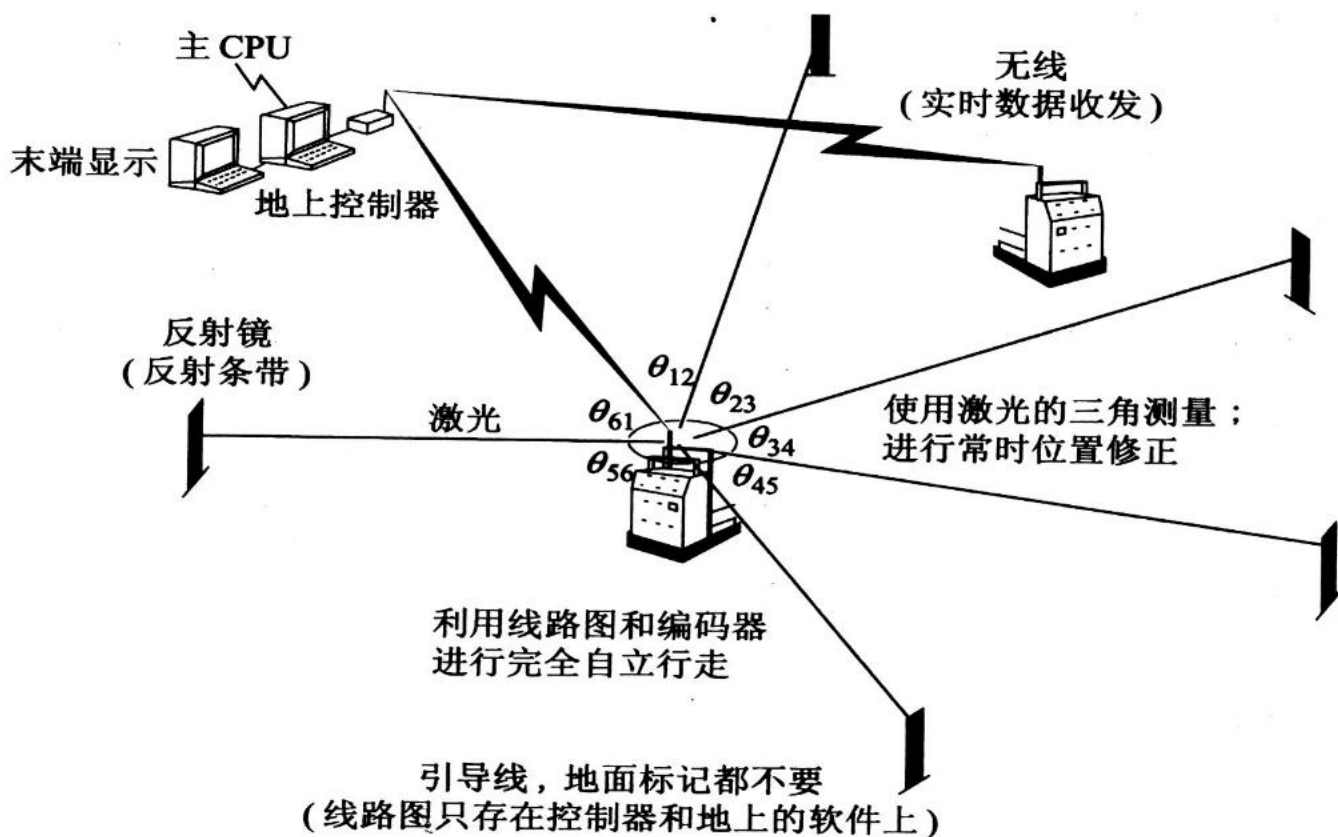
(2)信标方式(激光导航)。这种方式是在路径上或沿着路径设置多个标记，**标记本身主动发出信号提供有关位置信息**。信标方式是**从当前位置**寻找若干个信标，然后根据其方向和有关信标的位置信息，利用三角测量原理计算出现在的位置。



设信标 A . B . C 的坐标分别为 $(x_a, 0)$, (x_b, y_b) , $(0, 0)$ 则机器人的位置 T 点的坐标和方向 θ 可由两个圆的交点求出。

图7-14 利用激光导航方式的高精度位置检测原理

标记可用再现反射体(直角棱镜等), 扫描射线优先采用激光, 也可采用红外线。根据求得的两组两个再现反射体之间的开度角计算出现在的位置、方位, 这种方法称为激光导航法。标记设置简单、廉价, 精度也非常好。



图中参数的计算式如下：

$$y = Kx$$

$$\theta = \pi + \tan^{-1} \frac{y}{x} - \theta_c$$

式中，

$$K = \frac{x_a - x_b - y_b \cos \alpha}{y_b - x_b \cos \alpha - x_a \cos \beta}$$

代替检测多个信标方向的方法，是通过从一个地方的信标发出扫描激光光束，用AGV上若干个传感器来检测的方法(称为激光信标方法)。采取这种方法，在作为移动物体的AGV上设置的3个传感器，如果都接收到激光站(即使是一个)的光就能检测出位置，即便存在很多AGV，如果某AGV上的3个传感器能接收到信标发出的光，这个AGV就能进行高精度的位置检测。这种位置检测法是强噪声检测方法，可作为用于自立分散型群控AGV的高精度位置检测。

2. 数字地图引导

把路径画在数字地图上，作为人与机器的对话式系统，非常容易接近。此外，利用中央计算机的指令把路径的设定作为串行数据给出的方法，对复杂、交叉路径多的路线特别有效。是适合于控制复杂、多种、多量AGV的方法，是使工厂内的物流系统高度自动化所必须的。

7.2.3 智能引导

智能引导方式有**示教式**（初级智能）和**路径规划**两种。

示教式：当AGV沿着示教的路径行走一次，即记住行走路线。它实际上还可学会新的行走路径，并通知主控计算机所学到的东西。主控计算机可通知其他的AGV关于这条新的路径的信息。

7.3 路径规划

AGV智能化的新发展在于自主回避障碍物并达到目的地的路径规划。

首先，影响路径规划的是AGV的自由度数 and 地图的有无。为了便于理解，以2自由度AGV为例，分别考虑有地图时(环境已知时)和没有地图时(环境未知时)的路径·动作规划。这里，假定AGV只考虑两个位置自由度(X、Y轴上的位置)，不考虑姿态方面的一个自由度(绕中心的回转)

- 由于地图是由AGV和障碍物的模型，所以有地图时的路径规划称为基于模型的路径规划(Model-based Path-Planning)。基于模型的路径规划称为离线路径规划。
- 没有地图时的路径规划，AGV用外部传感器(视觉、超声波传感器、光传感器等)得到一面回避障碍一面到达目的地的路径，由此称为基于传感器的路径规划(Sensor-based Path-Planning)。基于传感器的路径规划又称为在线路径规划。

基于模型的路径规划

首先说明为了快速选择最佳(最短)路径，应采用怎样的数据结构来表现地图。

- 最佳(最短)路径由于接近障碍物，如果有位置误差，AGV与障碍物碰撞的可能性很高。下面要说明的是，为防止碰撞，除了最佳性以外更重视安全性的方法，即为了选择离障碍物足够远的安全路径，应采用怎样的数据结构来表现地图。这里由于采用一种OR表。

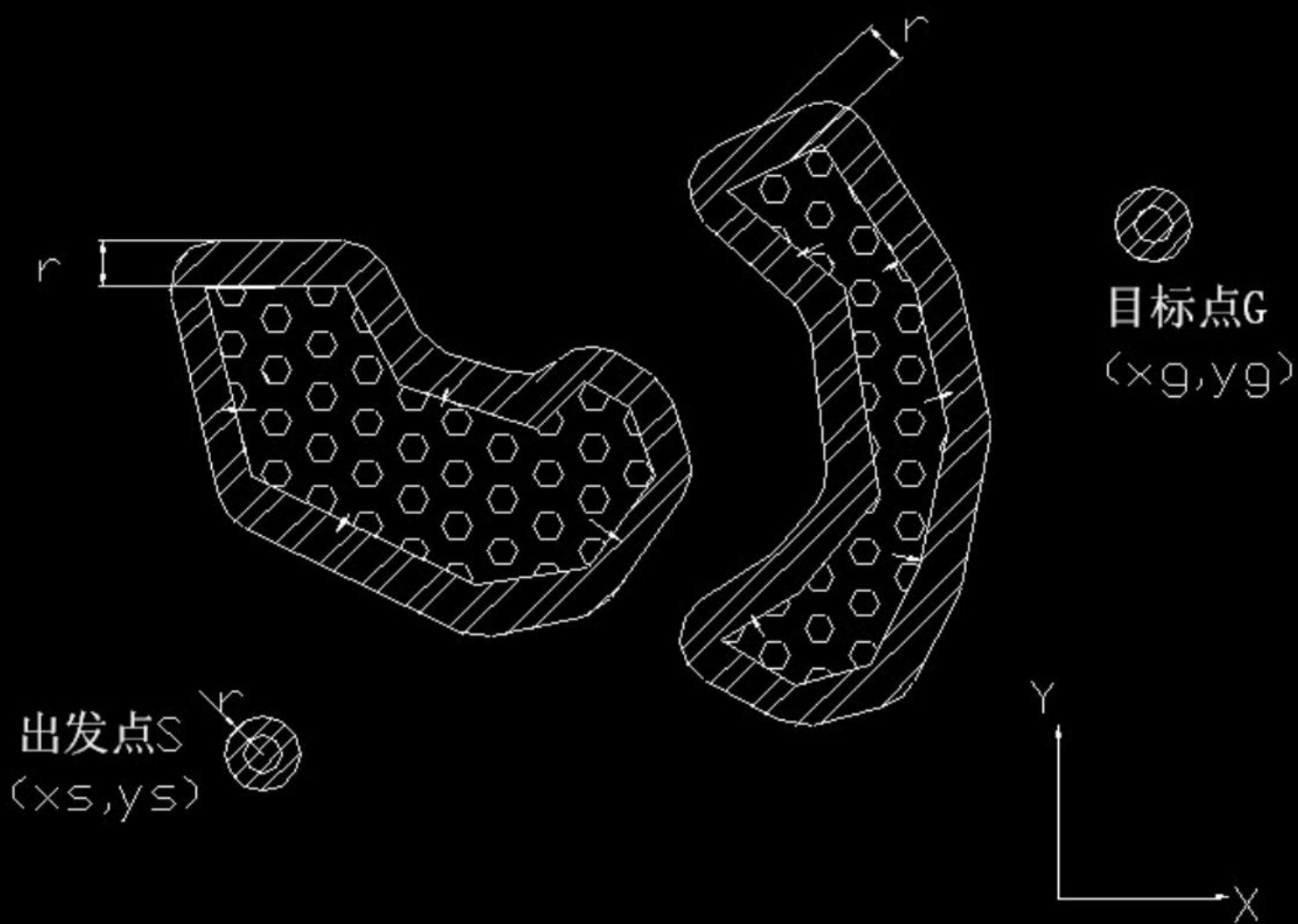
基于传感器的路径规划

AGV用传感器一面检测障碍物一面进行回避并最终到达目的地的路径规划算法，即使存在位置误差，AGV也不会迷失确定的路径，而最终到达目的地附近。

7.3.1 路径规划的模型

1. 几何模型

AGV有2个位置自由度(X、Y轴上的位置)和1个姿态自由度(绕中心的旋转)。这里，取可全方位移动的AGV为例加以说明。由于AGV能全方位移动，所以可忽略AGV的方向(姿态的自由度)。这样，就能用以最大回转长度为半径的圆表示AGV。在此基础上，可以把障碍物的几何尺寸径向扩张一个AGV圆的半径，同时把AGV缩成一个点(图7-16)。由此，在存在扩张了的障碍物的地图(XY平面)上，可以规划成为几何点的AGV的路径。



2. 数学模型

首先，说明基于模型的路径规划。

为了快速选取路径，用所谓图的数据结构表示规划的数学模型(俗称“地图”)。所谓图就是用弧连接节点的数据结构，节点意味着AGV的位置，弧意味着两个位置间的移动(图7-17)。将移动的几何距离、工作量或时间加权折算得出两个位置间移动的模式费用。把模型费用希望值作为费用赋予该两个位置间的弧上。

当所有节点间移动的工作量不变时，弧上赋予的费用可以直接用几何距离标记。弧记忆进入节点和输出节点，总是回到原来的地方(程序上称为“指针返回”)。而且，如果能在两个方向移动则用无向弧，只能单方向移动的用有向弧。

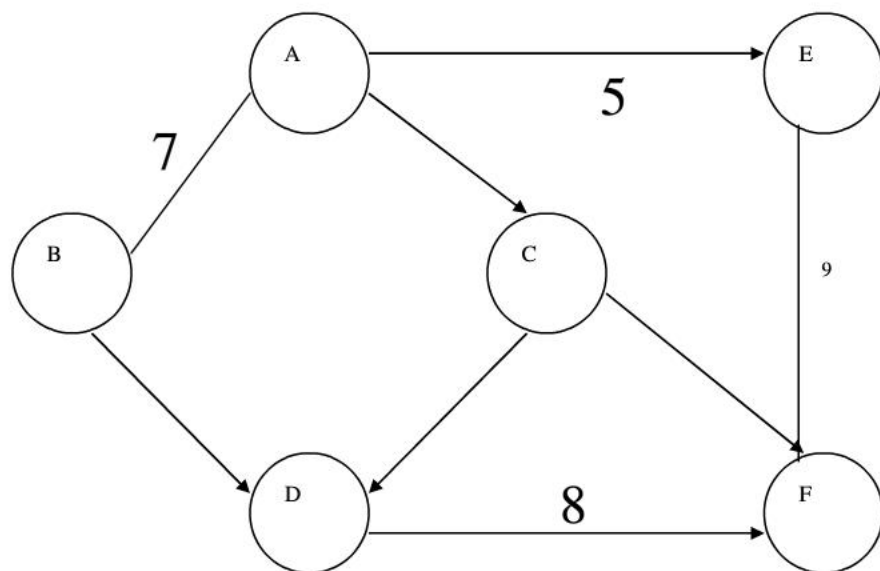


图7-17 图(节点和弧)

7.3.2 基于模型的路径规划

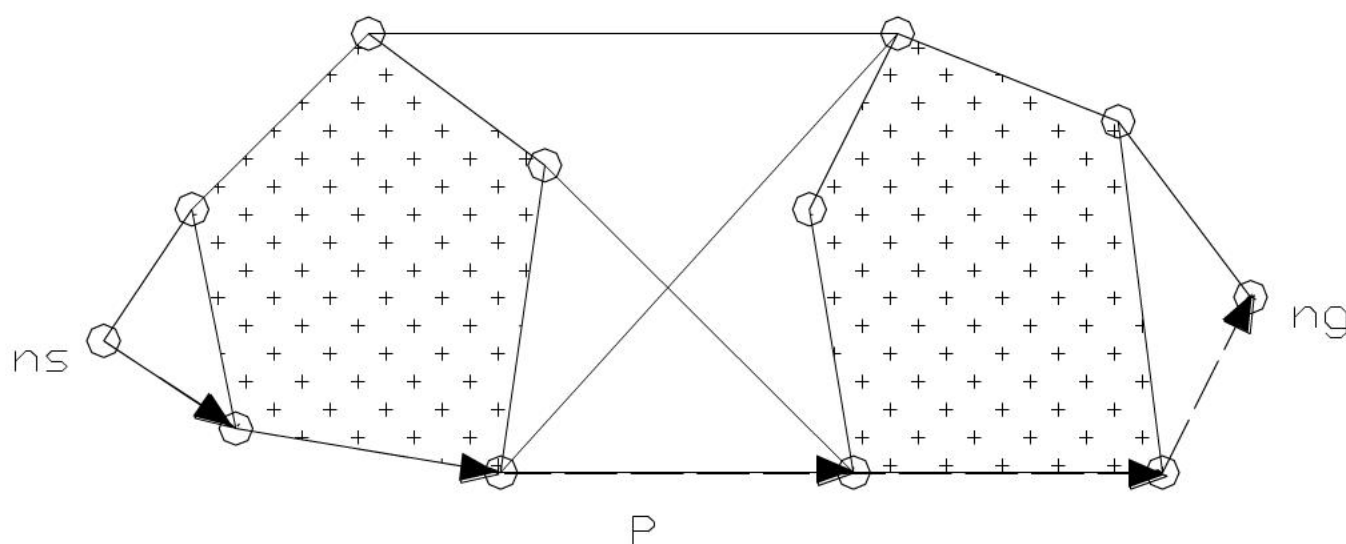
这里介绍两个典型的图。一个是管理从起始节点 ns 到目标节点 ng 的最短路径的切线图(Tangential Graph)，另一个是连接这些节点的安全路径，即管理尽量离开障碍物路径的Voronoi图(Voronoi Graph)。无论哪一种图都是由节点和弧构成的，用节点表示起始点、经过点、目标点；用无向弧表示其间的路径，其上附加有作为费用的欧几里得距离。

最后，无论哪个图，都是用算法A选出任意路径，用算法A*选出最佳(满足)路径。

1. 切线图

切线图用障碍物的切线表示弧。由此可选择从起始节点 ns 到目标节点 ng 的最佳(最短)路径。即切线图是把障碍物边界切线化得到的。从起始节点 ns 开始，过两相邻节点向障碍物边界作切线，每两条切线的交点形成辅助节点。由主节点(起始点、经过点、目标点)、辅助节点和节点间连线组成了切线图。

- 首先把对应起始点S和目标点G的两个节点 ns 和 ng 标注在新的切线图上，
- 然后用算法A*选出最佳(最短)路径P，
- 最后，使点AGV沿着路径P进行PTP(Point-To-Point)控制和CP(Continuous Path)控制，把AGV引导到目的地。

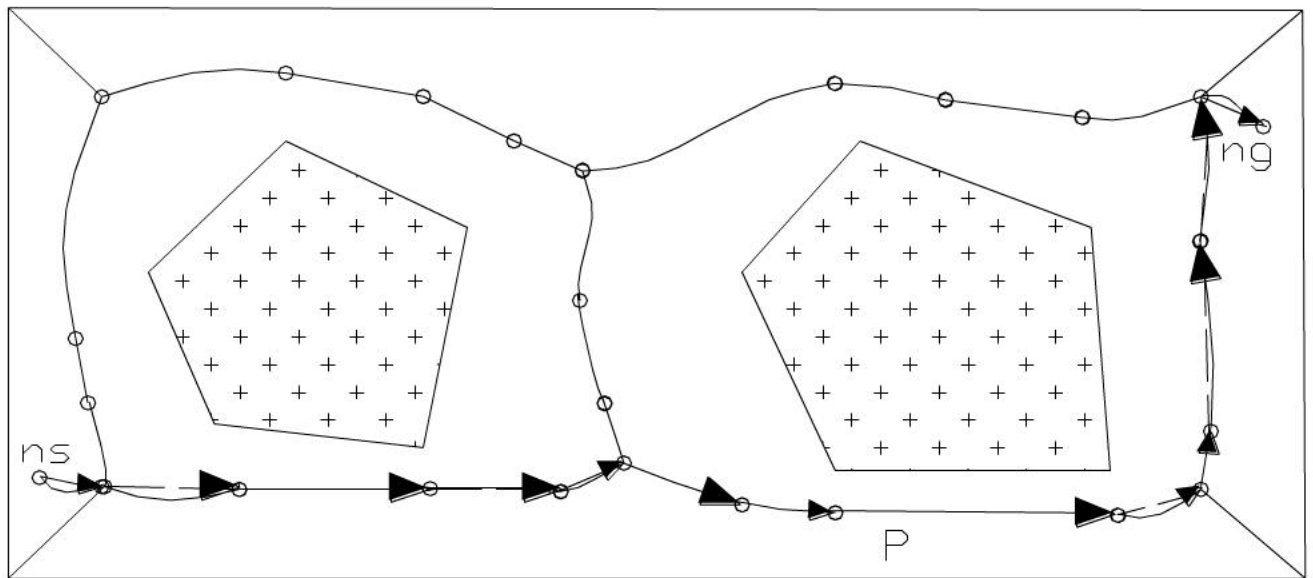


如果在这种控制过程中产生位置误差，机器人碰撞障碍物的可能性会较高，因为AGV几乎接近障碍物行走

2. Voronoi图

Voronoi图可用弧表示距两个以上障碍物和墙壁表面等距离的点阵，用节点表示它们的交叉位置。

首先把对应起始点S和目标点G的起始节点 n_s 和目标节点 n_g 标注在图上，然后用搜索算法A*选出安全路径P，最后，使点AGV沿着路径P进行PTP控制和CP控制，把AGV引导到目的地。由于选择的是安全路径，所以，即使产生位置误差，AGV也能够在离障碍物足够远的路径上走行



3.搜索算法A* (A)

这里要介绍的是，把前面所说的切线图 and Voronoi图作为搜索图G，选出从起始节点 n_s 到目标节点 n_g 的最佳(或满足)路径的算法A*(或选出任意路径的算法A)。

算法A*(或A)一面计算节点 n 的费用 $f(n)$ ，一面搜索图G。费用 $f(n)$ 是从起始节点 n_s 经由当前节点 n 到目标节点 n_g 的最小费用(最短距离)的估价函数。可用下式计算：

$$f(n) = g(n) + h(n)$$

式中， $g(n)$ 是起始节点 n_s 和当前节点 n 之间的现时点上的最小费用(最短距离)；而 $h(n)$ 是当前节点 n 和目标节点 n_g 之间的最小费用 $h^*(n)$ 的估计值，称为启发式函数值。

OPEN是管理以后扩展节点的明细表，所有节点按费用 $f(n)$ 递增顺序排列，CLOSED是管理已扩展节点的明细表。

通常A*(或A)等搜索算法，从节点 n 扩展的所有节点 n' 中，把必要的节点同费用 $f(n')$ 都标注OPEN上(参看算法的第⑤步)，这个操作称为“扩展节点 n ”。

算法A*(A)的程序流程说明:

- ① 把起始节点 ns 代入OPEN。
- ② 如果OPEN是空表, 由于路径不存在, 所以算法终止。
- ③ 从OPEN取出先头(费用 f 最小)的节点 n , 并把它移到CLOSED。
- ④ 如果节点 n 是目标节点 ng , 则顺次返回到来自的节点上(程序上是追寻(返回)指针)。然后, 若是到达起始节点 ns , 则终止算法, 得到一个路径。
- ⑤ 如果不是这样, 扩展节点 n , 把指针从其子孙节点 n' 返回到节点 n (记住从哪来的)。然后, 对所有的子孙节点 n' 做以下工作:
 - 节点 n' 如果不在OPEN或CLOSED表中, 则它就是新的搜索节点。因此, 首先计算估计值 $h(n')$ (从节点 n' 到节点 ng 的最短距离的估计值)。其次, 计算评价值 $f(n')=g(n')+h(n')$ (这里 $g(n')=g(n)+c(n, n')$, $g(ns)=0$, $c(n, n')$ 是连接节点 n 和 n' 弧的费用)。然后, 把节点 n' 同估计值 $f(n')$ 都代入OPEN。
 - 若节点 n' 存在于OPEN或CLOSED表中, 则它就是已被搜索的节点。于是, 把指针换到带来最小值 $g(n')$ 的路径上(变更来自的地方)。然后, 在这个指针发生替换时, 若节点 n' 存在于CLOSED中, 则把它返回到OPEN后, 再计算值 $f(n')=g(n')+h(n')$ 。
- ⑥ 返回到②

算法A*(A)的程序流程

- 估计值 h 比真值 h^* 小或相等时，上述的算法变为 A^* ，可选出从起始节点 ns 到目标节点 ng 的最佳路径(总计费用最小的路径)。
- 若估计值 h 比真值大， h^* 算法则变为 A ，可选出从起始节点 ns 到目标节点 ng 的满足要求的路径(总计费用不是最小的路径)。
- 因此，机器人的路径规划多用从当前地点 (X_p, Y_p) 到目的地 (X_g, Y_g) 的平方范数 $\sqrt{(X_g - X_p)^2 + (Y_g - Y_p)^2}$ 定义估计值。这个估计值 h 常常比 h^* 小，成为算法 A^* ，用它可选择最佳路径。

图7-20的搜索图G存在估计值 h (在节点上用括号给出)比真值 h^* 大的节点。例如，节点A和H的估计值 h 是8和4，但到达目的地的最小真值是7和2。由于这个费用评价过大，存在于最短路径上的节点H等可以忽略，算法错过了费用8的最佳路径，最终得到费用9的满足要求的路径。

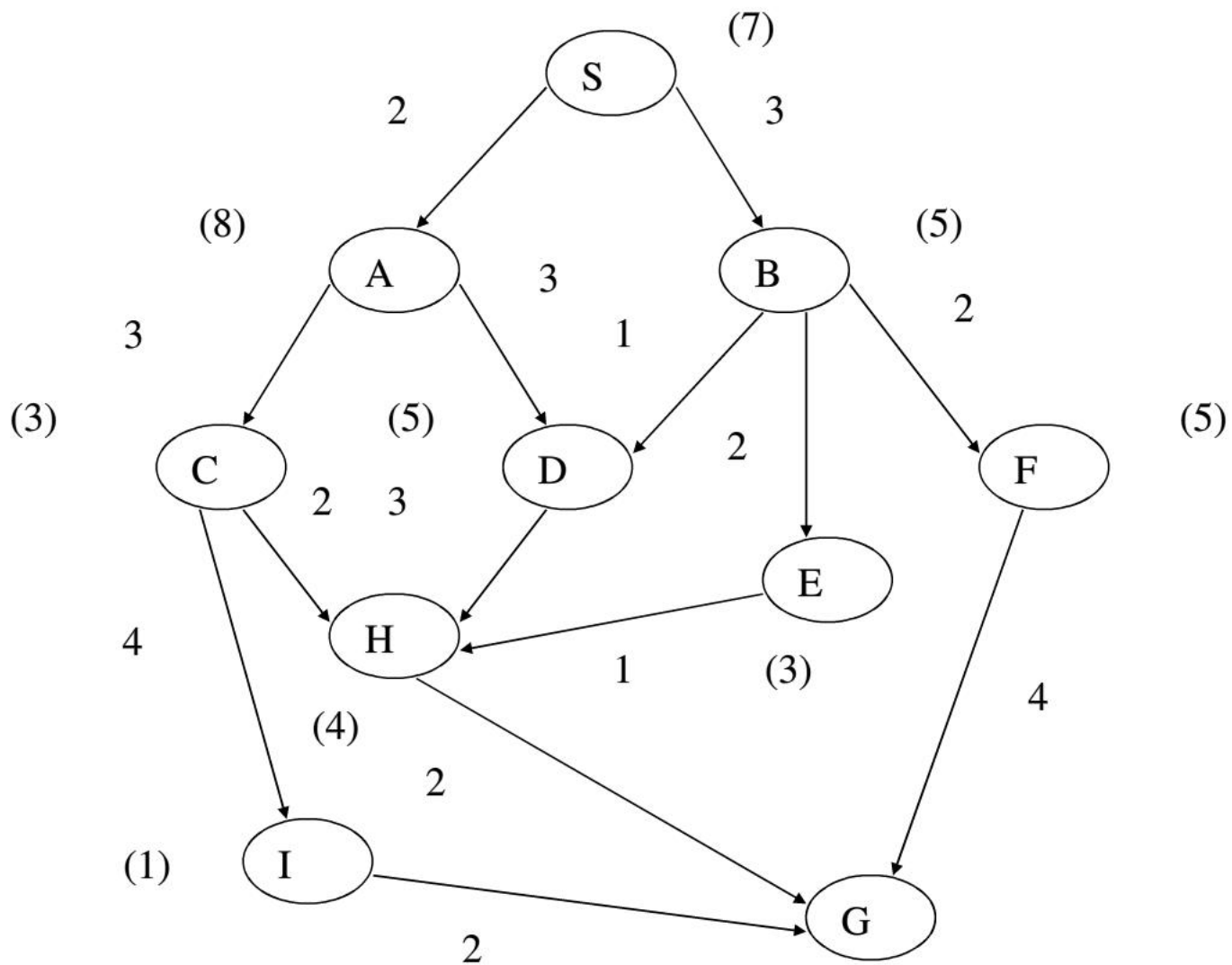
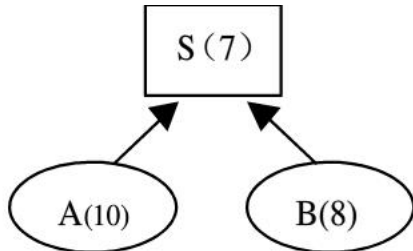


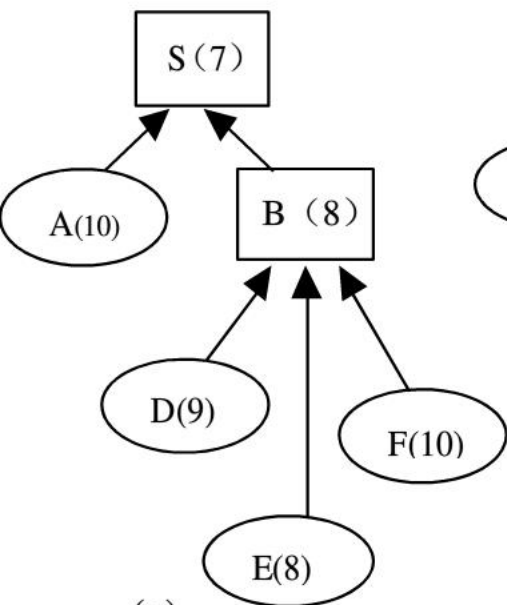
图7-20 搜索图G(有的地方估计值比真值大)



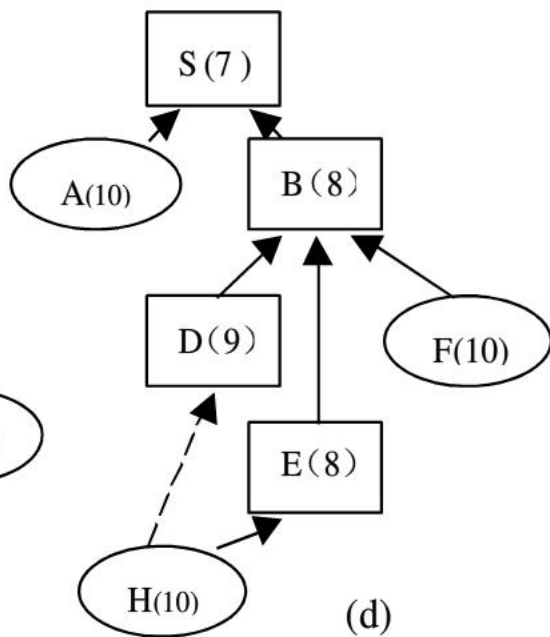
(a)



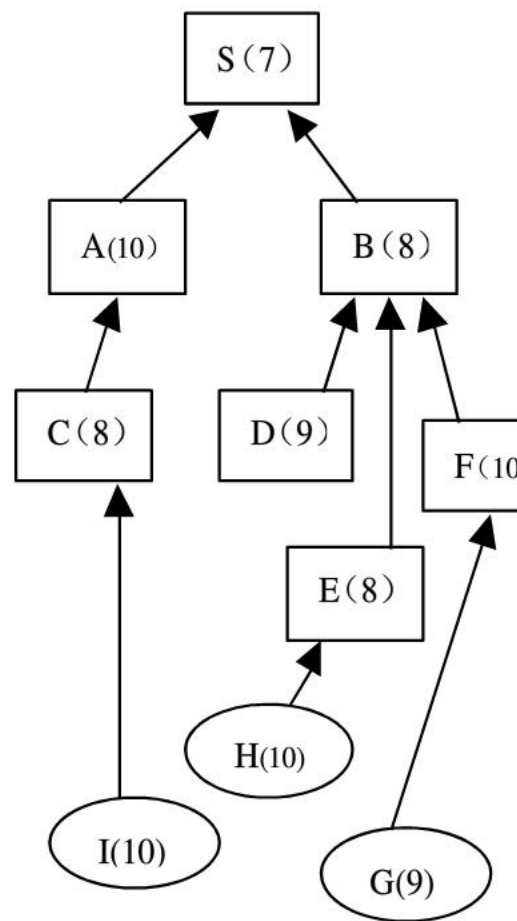
(b)



(c)



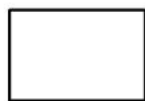
(d)



(e)



属于 OPEN 表的节点

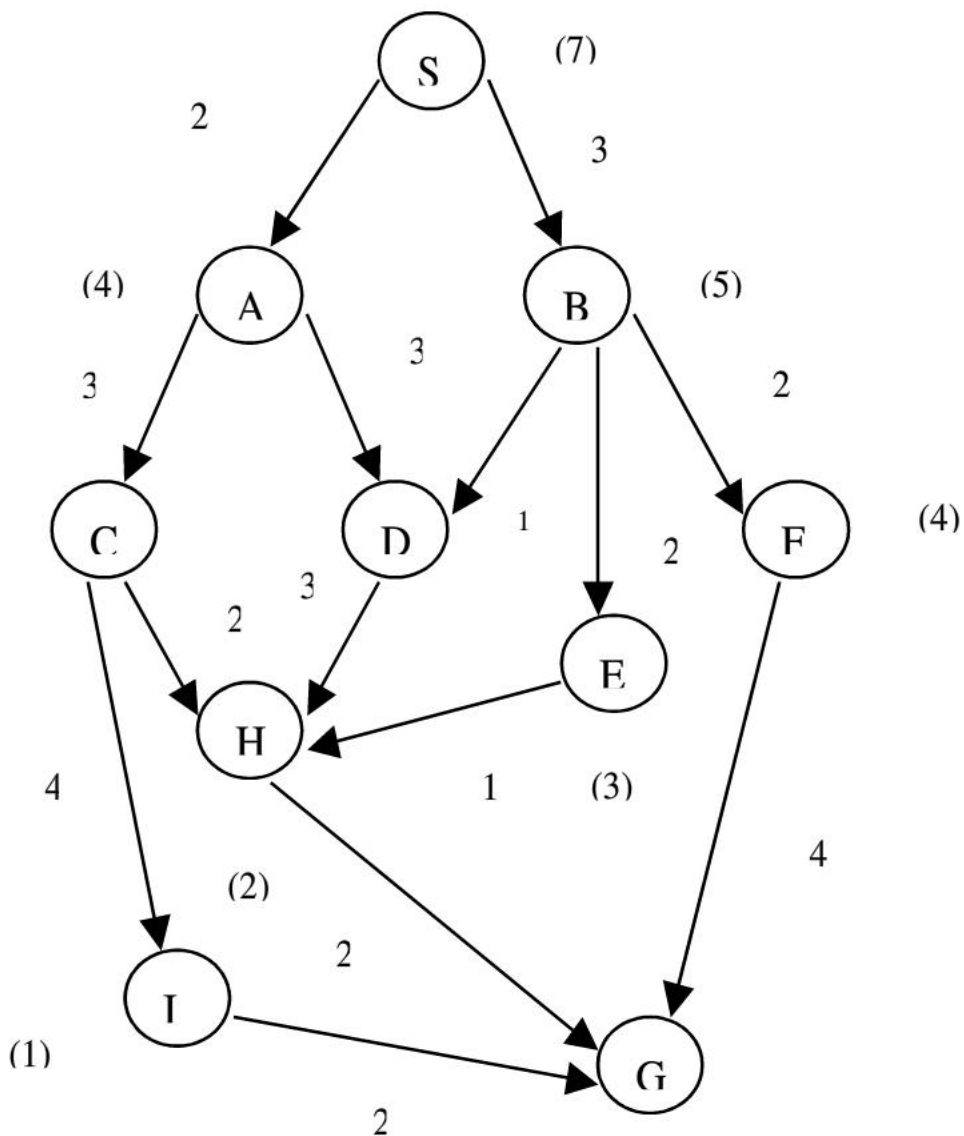


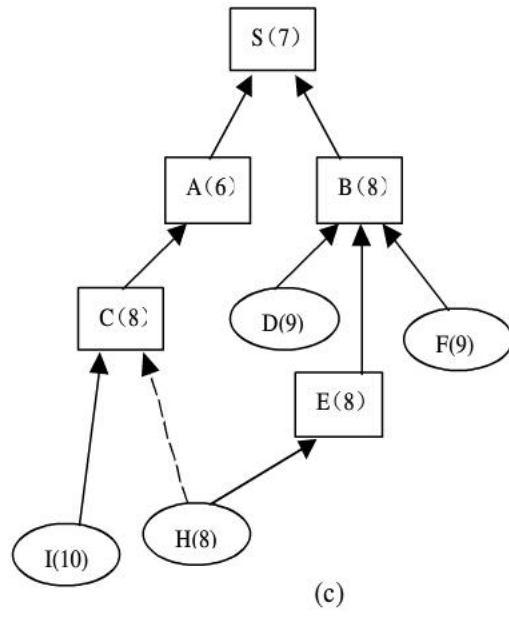
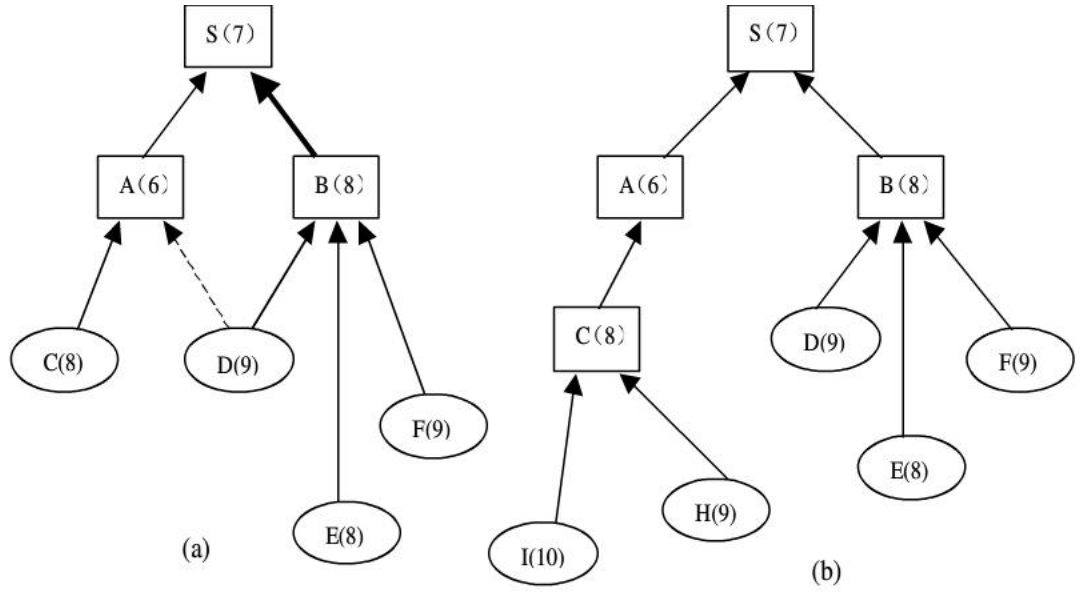
属于 CLOSE 表的节点

- I. 起始节点S被代入OPEN[图7-21(a)], 子节点A和B被代入OPEN[图7-21(b)]。起始节点S扩展后移到CLOSED,
- II. 由于节点A、B的评价值分别为10、8, 所以选中扩展节点B。B的子节点D、E、F, 评价值分别为9、8、10, 全都代入OPEN, 扩展后节点B被移到CLOSED[图7-21(c)]。
- III. 节点E评价值最小, 选中扩展节点E。E的子节点H同评价值10都代入OPEN, 扩展后节点E被移到CLOSED。
- IV. OPEN中当前评价值最小的子节点是D, 所以选中节点D。D的子节点H被再次搜索, 节点D被移到CLOSED。注意到节点H的值 g , 由于过去的费用6(经由节点E、B返回到节点S)比新的费用7(经由节点D、B返回到S)小, 所以不更换指针[图7-21(d)]。指针仍在E。由于当前OPEN上存在评价值 f 都为10的三个节点A、H、F, 此时须对这三个节点A、H、F都分别向下扩展。

- V. 用中断连接节点A，扩展节点C同评价值8都代人OPEN，节点A被移到CLOSED。然后，C的两个扩展节点I、H中，选中值 f 最小的节点I（实际上，ACH路径在先已经被否定），节点I同评价值10都代人OPEN，节点C同评价值8都被移到CLOSED。节点I的扩展节点即为目标，OPEN保持
- VI. 节点H的扩展节点即为目标，OPEN保持
- VII. 节点F的扩展节点即为目标，OPEN保持
- VIII. 当前OPEN上三个节点I、H、F的评价值都为10,其后的扩展节点都是目标G。计算分别经三个节点F、H、I到目标G的评价值,经节点F到目标G的评价值最小。所以用中断连接扩展节点F，节点G连同评价值9都代人OPEN，节点F移到CLOSED[图7-21(e)]。

在图7-22的搜索图G上，所有节点的估计值 h (在节点上用括号给出)常常比真值 h^* 小或相等。



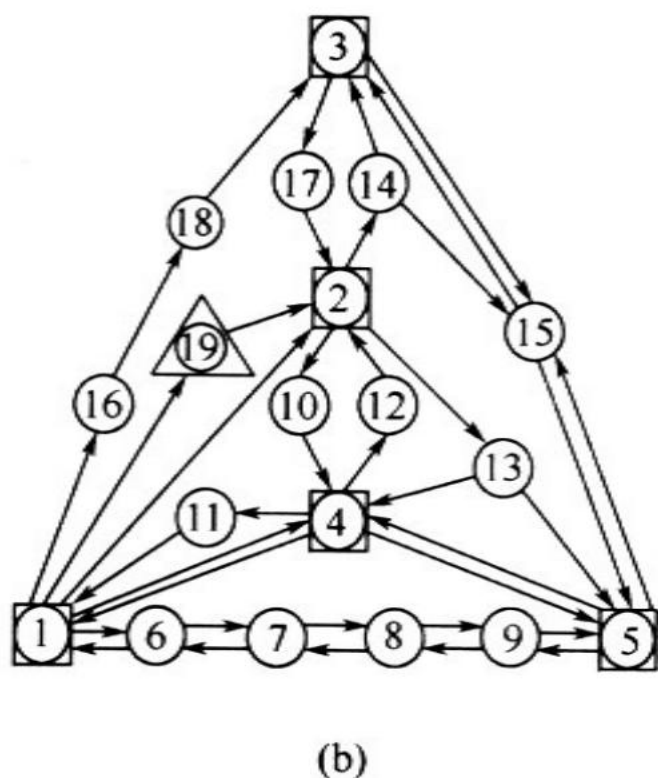
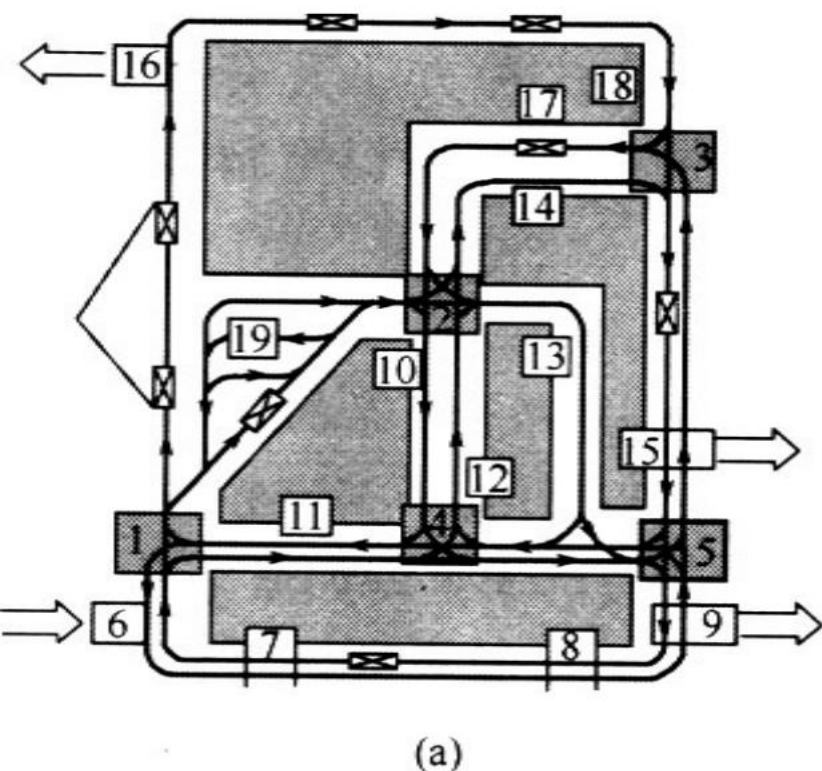


- I. 起始节点S被代入OPEN，子节点A、B连同评价值 f 8被代入OPEN，起始节点S扩展后移到CLOSED。
- II. 由于节点A、B的评价值 f 分别为6、8，所以节点A扩展子节点C、D后移到CLOSED，节点B、C、D连同各自的评价值8、9都代入OPEN。
- III. 用中断连接扩展节点B，节点B扩展子节点E、F。子节点E、F连同评价值8、9都代入OPEN，节点B扩展后代入CLOSED。再次搜索节点D。这时，如果注意到节点D的值 g ，由于新的费用值4(经由节点B返回到节点S)比过去的费用5(经由节点A返回到节点S)小，所以要更换指针，重新计算的评价值 f 变为9[图7-23(a)]。
- IV. 仍然用中断连接扩展节点C，节点C扩展子节点I、H。子节点I、H连同评价值10、9都代入OPEN，节点C移到CLOSED [图7-23(b)]，

- V. 当前OPEN上五个节点I、H、D、E、F。所以选中评价价值 f 最小的扩展节点E。然后，把节点E代入CLOSED，再次搜索节点H。
- VI. 这时，注意到节点H的值 g ，比起过去的费用7(经由节点C、A返回节点S)来新的费用6(经由节点E、B返回到节点S)要小，所以更换指针，重新计算H的评价价值 f 为8[图7-23(c)]
- VII. 当前OPEN上扩展值 f 最小的节点H，其后的扩展节点是目标G，节点G以评价价值8代入OPEN，节点H代入CLOSED
- VIII. 最后，节点G如果选择作为值 f 最小的节点，则把指针返回到节点H、E、B、S，最终得到费用8的最短路径。这里，由于节点H的估计值 h 与真值 h^* 相比过小，所以这个最佳路径上的节点必须调整。

4. AGV的路径规划实例

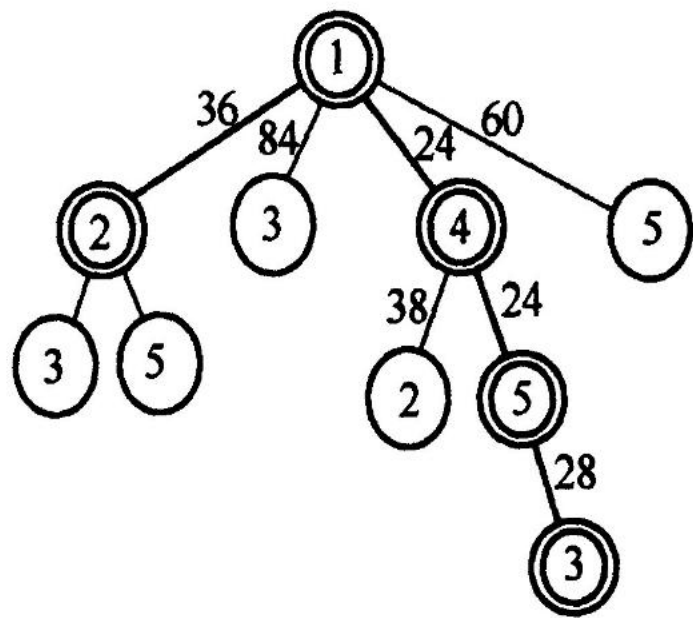
Dijkstra最短路径确定法。图7-24中的节点1~5是AGV运行路线上的岔口，6~18是货物装卸点，6是系统的入口，9、15、16是系统的出口，19是AGV停车处。在图7-24(b)中，将AGV可能的运行方向用箭头进行了表示。



(1) 因为节点1~5是找出6~18货物装卸点间最短距离的关键点，故先只考虑系统中的各岔口即节点1~5间的最短路径。为此，将相应箭头所表示的距离填入下面的距离矩阵。因节点1、3之间无直接连接的箭头，但两者之间可通过16、18连在一起，所以 $C_{1,3}=C_{1,16}+C_{16,18}+C_{18,3}=84$ 。 $C_{3,4}=\infty$ 表示所连两点间无直接联系，之间的联系要通过岔口节点。由此得到系统的距离矩阵(表7-2)。

表7-2 距离矩阵

		目的地				
		1	2	3	4	5
出发点	1	0	36	84	24	60
	2	∞	0	44	38	34
	3	∞	40	0	∞	28
	4	24	38	∞	0	24
	5	60	28	28	24	0



(2)搜索过程

- I. 最短路径法首先要求找到搜索的起始点 i ，起始点 i 必须是通过一个单箭头就能和其余节点相连最多的节点。很显然，图7-24(b)中的节点1符合这个要求，将此节点作为第一层。
- II. 在找出由该 i 点出发的最短距离 $C_{i, k}$ 后，令 $d_{i, k}=C_{i, k}$ 再以 k 节点作为第2次搜索的起点。
- III. 根据距离矩阵中与 k 节点对应的行，计算由起始点出发，经此 k 点到其余节点的最短距离，若到节点 j 的距离最短，则计算 $d_{i, k}+C_{k, j}$
- IV. 若 $d_{i, k}+C_{k, j}>C_{i, j}$ 再取 $i \rightarrow j$ 路径的 j 点作为第3次搜索的起点。重复上面搜索过程（将 j 替换 k ）
- V. 若 $d_{i, k}+C_{k, j}<C_{i, j}$ 令 $d_{i, j}=d_{i, k}+C_{k, j}$ ，再取 $i \rightarrow k \rightarrow j$ 路径的 j 点作为第3次搜索的起点。如此往返进行，直到所有节点均被选上为止。

最短路径法的整个搜索过程见表7-3，最短路径计算结果见图7-25，细节见表7-4。

表7-3 最短路径法的搜索步骤

步骤	基点	节点顺序	$C_{1,j}$	$d_{1,j}$	下一个基点
1	1		节点 2, 3, 4, 5 未选	$d_{14} = C_{14} = 24$	4
		1→2	$C_{12} = 36$		
		1→3	$C_{13} = 84$		
		1→4	$C_{14} = 24$		
		1→5	$C_{15} = 60$		
2	4		节点 2, 3, 5 未选	$d_{12} = C_{12} = 36$	2
		1→4→1	1 点已选为基点		
		1→4→2	$d_{14} + C_{42} = 62 > C_{12} = 36$		
		1→4→5	$d_{14} + C_{45} = 48 < C_{15} = 60$		
3	2		节点 3, 5 未选	$d_{15} = C_{152} = 48$	5
		1→2→3	$d_{12} + C_{23} = 80 < C_{13} = 84$		
		1→2→4	4 点已选为基点		
		1→2→5	$d_{12} + C_{25} = 70 > C_{15} = 48$		
4	5		节点 3 未选	$d_{13} = C_{13} = 76$	3
		1→4→5→3	$d_{15} + C_{53} = 76 < C_{13} = 80$		

表7-4 最短路径一览表

起点	目 的 点				
	1	2	3	4	5
1	1	1→2	1→4→5→3	1→4	1→4→5
2	2→4→1	2	2→3	2→4	2→5
3	3→4→5→1	3→2	3	4→5→3	3→5
4	4→1	4→2	4→5→3	4	4→5
5	5→4→1	5→2	5→3	5→4	5

7.3.3 基于传感器的路径规划

- (1) 基于传感器的路径规划的指导思想：离障碍物一定距离时，实施**变向避障**，同时**单调减少**到达目的地的距离，**从而**保证AGV到达目的地
- I. 首先AGV向目的地直进，若是AGV受到障碍物阻碍，记录碰撞点 H_i ，同时AGV顺时针或反时针转过障碍物。
 - II. 如果目的地的方向有空位(脱离点)，记录该脱离点 L_i ，若是脱离点 L_i 比碰撞点 H_i 更接近目的地，AGV就离开障碍物向目的地直进(图7-26)。
 - III. 若是脱离点 L_i 比碰撞点 H_i 更远离目的地，AGV就返回碰撞点 H_i ，反向**搜索**比碰撞点 H_i 更接近目的地的脱离点
 - IV. 按照这个程序，脱离点 L_i 和碰撞点 H_i 单调接近目的地G
 - V. AGV可看成点机器人。点机器人可能与障碍物碰撞的领域 R_i 单调减少(领域 $R_1 \rightarrow R_2 \rightarrow R_3 \rightarrow R_4$)。由此，AGV最终到达目的地G。

算法Class2的程序:

- ① 初始设定 $L_0=S=R$, $i=0$ 。
- ② 点机器人 R 向目的地直进。
 - 如果点机器人到达目的地, 由于得到了解, 算法终止;
 - 如果向量 RG 被障碍物妨碍, 取 $i=i+1$, 把当前地点作为碰撞点 H_i , 执行③。
- ③ 点机器人 R 顺时针或反时针方向绕过障碍物进行追寻。
 - 如果点机器人到达目的地 G , 由于得到了解, 算法终止;
 - 如果欧几里得距离 $|RG|$ 比 $|H_iG|$ 小, 并且向量 RG 不受障碍物妨碍, 把当前地点作为脱离点 L_i , 回到②;
 - 若点机器人 R 返回到碰撞点 H_i , 由于不存在解, 所以算法终止。

