

机器人避障问题

【摘要】

本文主要是对机器人在一个平面区域内的通过不同障碍物到指定目标点进行的研究，首先通过机器人与障碍物的最小安全距离对不同障碍物的禁区进行了划分见图 1，把障碍物划分为有顶点和无顶点两大类。然后证明了机器人在障碍物顶点处转弯路径最优，转弯半径最小路径最优，转弯圆心在障碍物顶点处（圆形障碍物在圆心）路径最优。

问题一对于起点和目标点的路线先用拉绳子的方法确定了可能的最短路线，然后用穷举法确定最佳路径。机器人的行进又分单目标点和多目标点两种情况。

针对单目标点问题，先对只进行一次转弯的过程建立了基本线圆组合结构的解法即模型一。然后对多次转弯问题中的直线路径与圆弧路径的不同的位置关系推导出了计算模型即模型二。对 O-A 是基本的线圆组合，直接用模型一求解得到 O-A 的最短路径长为 471.0372 个单位，所用时间为 96.0178 秒具体情况见文中表 1。对 O-B 和 O-C 都是先用模型二对路线进行基本分割，然后用模型一进行求解得到 O-B 最短路径长为 853.7127 个单位，所用总时间为 179.0851 秒，具体见表 2。得到 O-C 最短路径长为 1087.6 个单位，所用时间为 221.9 秒，具体见表 3。

针对多目标点问题，由于机器人不能直线转向，所以在经过目标点时，应该提前转向，且中间目标点应该在转弯弧上。因此先建立优化模型（模型三）对进行中间目标点处转弯圆弧圆心搜索求解。求出中间目标点转弯圆心后，用把中间目标点的圆心看做“障碍物”的办法把问题转化为单目标点问题。然后利用模型二和模型一进行求解，解得 O-A-B-C-O 的最短路径长为 2812.52 个单位，所用时间为 585.6712 秒，具体见附表 1。

对于问题二，在问题一求出的最短路的基础上，根据转弯半径和速度的关系，在问题一求出的最短路径的模型的基础上，进行路线优化，建立以最短时间为目标的非线性规划模型，求解得最短时间为 94.22825 秒，转弯半径为 12.9886 个单位，转弯圆心坐标为 (82.1414, 207.1387)，具体结果见表 5。

关键词：基本线圆组合 拉绳子法 穷举法 非线性规划 中间目标点转弯圆心

1 问题重述

附图1是一个800×800的平面场景图，在原点O(0, 0)点处有一个机器人，它只能在该平面场景范围内活动。图中有12个不同形状的区域是机器人不能与之发生碰撞的障碍物，障碍物的数学描述如下表：

编号	障碍物名称	左下顶点坐标	其它特性描述
1	正方形	(300, 400)	边长200
2	圆形		圆心坐标(550, 450)，半径70
3	平行四边形	(360, 240)	底边长140，左上顶点坐标(400, 330)
4	三角形	(280, 100)	上顶点坐标(345, 210)，右下顶点坐标(410, 100)
5	正方形	(80, 60)	边长150
6	三角形	(60, 300)	上顶点坐标(150, 435)，右下顶点坐标(235, 300)
7	长方形	(0, 470)	长220，宽60
8	平行四边形	(150, 600)	底边长90，左上顶点坐标(180, 680)
9	长方形	(370, 680)	长60，宽120
10	正方形	(540, 600)	边长130
11	正方形	(640, 520)	边长80
12	长方形	(500, 140)	长300，宽60

在图1的平面场景中，障碍物外指定一点为机器人要到达的目标点（要求目标点与障碍物的距离至少超过10个单位）。规定机器人的行走路径由直线段和圆弧组成，其中圆弧是机器人转弯路径。机器人不能折线转弯，转弯路径由与直线路径相切的一段圆弧组成，也可以由两个或多个相切的圆弧路径组成，但每个圆弧的半径最小为10个单位。为了不与障碍物发生碰撞，同时要求机器人行走线路与障碍物间的最近距离为10个单位，否则将发生碰撞，若碰撞发生，则机器人无法完成行走。

机器人直线行走的最大速度为 $v_0 = 5$ 个单位/秒。机器人转弯时，最大转弯速度为 $v = v(\rho) = \frac{v_0}{1 + e^{10-0.1\rho^2}}$ ，其中 ρ 是转弯半径。如果超过该速度，机器人将发生侧翻，无法完成行走。

请建立机器人从区域中一点到达另一点的避障最短路径和最短时间路径的数学模型。对场景图中4个点O(0, 0)，A(300, 300)，B(100, 700)，C(700, 640)，具体计算：

- (1) 机器人从O(0, 0)出发，O→A、O→B、O→C和O→A→B→C→O的最短路径。
- (2) 机器人从O(0, 0)出发，到达A的最短时间路径。

注：要给出路径中每段直线段或圆弧的起点和终点坐标、圆弧的圆心坐标以及机器人行走的总距离和总时间。

2 问题分析

本问题主要是对机器人在一个平面区域内的通过不同障碍物到指定目标点进行研究，首先通过观察发现障碍物不同，因此我们先要对机器人通过不同的障碍物的情况进行讨论。在行进过程中转弯位置，转弯的半径，转弯圆心的不同都会影响路径的优劣。因此我们要先确定最佳转弯位置、半径和转弯圆心。

对于问题一，分单目标点和多目标点两种情况。针对单目标点问题，从已知点到指定点有多条路径可以选择，可以先用拉绳子的办法确定可能的最短路线，然后用穷举法确定最短路径。由于机器人不能折线转弯，转弯路径由与直线路径相切的圆弧组成，因此可以把路径看成由多个基本线圆组合而成，然后在对基本线圆结构建立求解模型。针对多目标点问题，不能简单的处理为求解每个两点之间的最短路径之和，因为机器人不

能直线转向，所以在经过目标点时，应该考虑提前转向，且中间目标点应该在转弯弧上。因此对于中间目标点处转弯圆弧圆心的确定可以建立优化模型进行搜索求解。求出中间目标点转弯圆心后可以把中间目标点的圆心看做“障碍物”进行研究，这样问题就转化为了单目标点问题。

对于问题二，可以根据转弯半径和速度的关系，在问题一求出的最短路径的模型的基础上，进行路线优化，即建立以最短时间为目标的非线性规划模型，进行求解最短时间。

3 模型假设

为了简化计算，给出如下模型合理性的假设：

- 【1】假设所有障碍物是固定不动的；
- 【2】机器人性能足够好，能准确地沿圆弧转弯；
- 【3】假设机器人在初始时的速度为5个单位每秒；
- 【4】假设机器人在直线切入弧线时的速度是瞬间变化完成的；
- 【5】机器人行走过程中不会意外停止；
- 【6】忽略影响机器人行走非最小转弯半径以及最小安全距离因素。

4 符号说明

为了简化对问题的分析和对数字的处理，做出如下符号规定：

d_m	第 m 条直线路径长度
u_n	第 n 条弧线路径长度
S	最短路径总长度
T	最短路径所用时间
v_0	机器人直线行进速度
v_p	机器人转弯速度

5 模型的建立与求解

5.1 最短路径模型（问题一）

从已知点到指定点有多条路径可以选择，可以先用拉绳子的办法确定可能的最短路线，然后用穷举法确定最短路径。在行进过程中转弯位置，转弯的半径，转弯圆心的不同都会影响路径的优劣。因此我们要先确定最佳转弯位置、半径和转弯圆心。本问题中对于路径的求解分为单目标路径和多目标路径两种，因此在本问题中我们分开讨论。

5.1.1 单目标点模型（只有起点和终点）

由于机器人不能折线转弯，转弯路径由与直线路径相切的圆弧组成，因此对于本问题可以把路径看成由多个基本线圆组合而成。

5.1.1.1 相关推论及证明

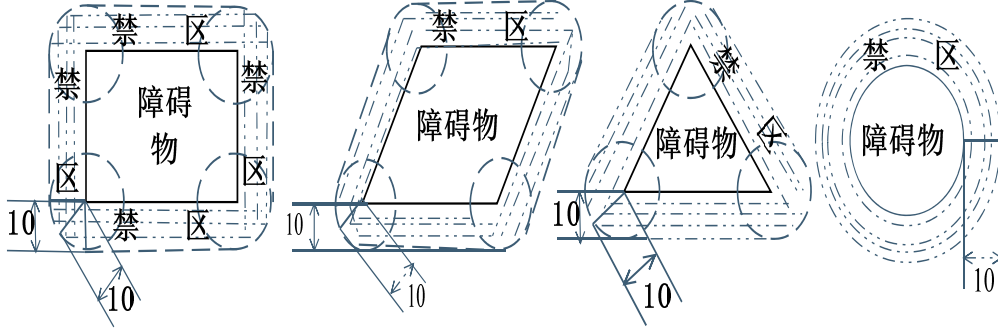
把机器人的行进路线看成一根有弹性的绳子，根据实际情况我们推论：禁区顶点处

变向路径最小，转弯半径最小路径最短，转弯弧圆心在顶点路径最短。

【1】禁区划分

由于机器人行进过程中与障碍物有最小距离的限制，因此我们先画出包络障碍物的禁区，对于有圆形障碍物来说，禁区还是一个圆，对于有顶点的障碍物来说禁区拐角处为一个圆弧，具体如图 1：

图 1 禁区示意图

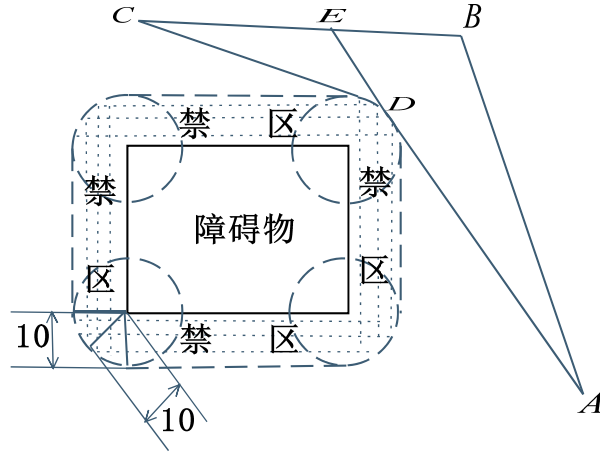


【2】障碍物有顶点

(1) 禁区顶点变向路径最小

如下图 2 所示，机器人从指定的 A 点到 C 点，需要进行变向，从图形可以看出来，在禁区边缘变向总路径会最小，下面我们将进行证明：

图 2 禁区顶点处变向图



假设 D 点为禁区顶点，先不考虑转弯半径等因素，其中 B 为禁区外任意一点，AD 变延长线交 BC 于 E 点。由三角形的任意两边之和大于第三边可以得到：

$$AB + BE > AD + DE$$

$$EC + DE > DC$$

两式相加得到：

$$AB + BE + EC + DE > AD + DE + DC$$

化简得到：

$$AB + BC > AD + DC$$

即，由 A 点到 B 点，选择在顶点 D 处转向，总路径最短。推论得证。

(2) 转弯半径最小路径最短

机器人从 A 点到 B 点需要绕过禁区，在禁区顶点附近（前面已证）转弯。选择的转弯半径越小，得到的路径越短。下面从物理学的角度进行证明：

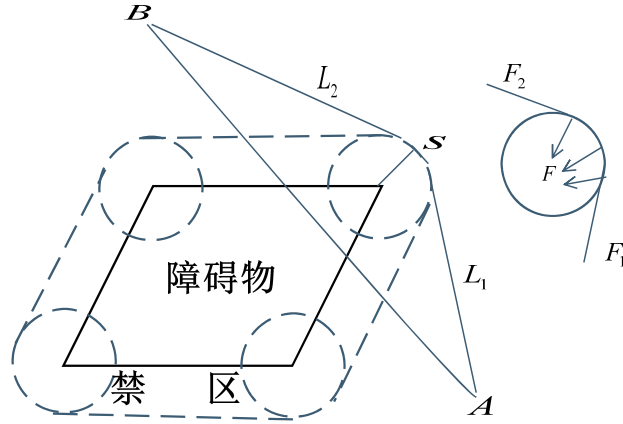
如下图 3 所示，将 A 到 B 的路径看做一条可伸缩的绳子。假设其两点相连时，绳子自然伸长。如线段 AB。由于机器人要绕过禁区，因此拉长绳子绕过禁区，又因为机器人有最小转弯半径为 10，禁区直径也为 10，因此可以把绳子直接绕过禁区边缘。

由于绳子的弹性势能 E_p 与伸长量 ΔL 的关系为：

$$E_p = \frac{1}{2} k \Delta L^2$$

因此绳子伸长量最小时，路径最短。

图 3 机器人过障碍转弯半径



根据最小势能原理^[1]可知，当弹性体平衡时，系统势能最小。即弹性体在自由条件下，有由高势能向低势能转化的趋势。现在将圆环看成也有弹性，在如图所示的条件下为初始状态。圆环受力如图所示，此时圆环有缩小的趋势，随着圆环的缩小系统趋于平衡，弹性绳有最小势能。由能量守恒也可以说明，绳子的弹性势能转化为弹性圆环的弹性势能，于是弹性绳的弹性势能减小。

因此，随着圆环的半径的减小，绳子的势能减小，即最短路径变短。所以最小转弯半径最小路径最短得证。

(3) 转弯弧圆心在顶点路径最短

机器人在禁区顶点附近转弯时，转弯弧的圆心在顶点上，路径最短。

下面进行证明：

如下图 4 所示，圆 O 和圆 O' 均过最远点 D 它们的半径分别为 R 和 R'。其中圆 O 的圆心在垂线上。L₁、S、L₂ 和 L'₁、S'、L'₂ 分别为过 A、B 点向圆 O 和 O' 所引切线段和所夹圆弧长度。则需证明：

$$L_1 + S + L_2 < L'_1 + S' + L'_2$$

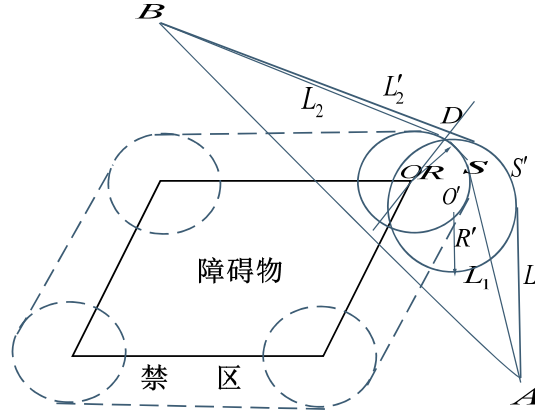
因为两圆圆心不同在过点 D 所引的垂线上，且两圆交于点 D，因此它们只能有两

种关系——相交或相切。若两圆相切，由几何定理知切点、圆心 O 、圆心 O' 三点共线。因点 D 和圆心 O 均在过 D 所作直线 AB 的垂线上，圆心 O' 在垂线外，故两圆不会相切，只能相交。要过点 D 则必有圆 O' 上点到直线 AB 的最大距离大于圆 O 上点到直线 AB 的最大距离。运用上文已证得的结论可得到：

$$L_1 + S + L_2 < L'_1 + S' + L'_2$$

因此推论得证。

图4 机器人过障碍转弯圆心位置



(4) 结论分析

结合 (1) (2) (3) 证明得到结论：

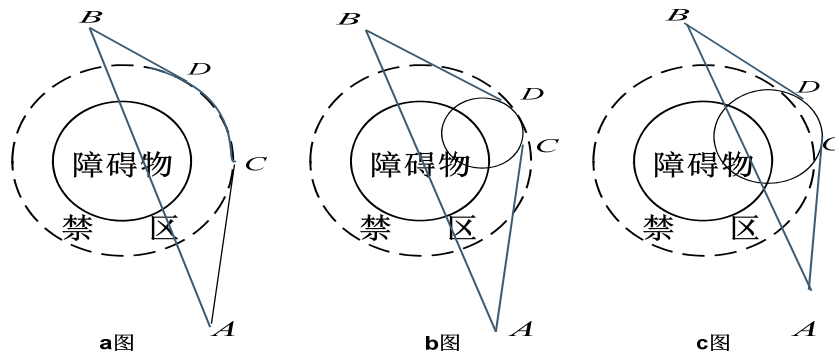
机器人过有顶点的障碍物时，沿以顶点为圆心，最小转弯半径为半径的圆弧转弯路径最短，在本题中机器人的最小转弯半径和机器人与障碍物的最小安全距离相等。因此对于本题来说，机器人沿禁区边缘转弯路径最短。

【3】障碍物为圆的情况

在前面我们已经证明了障碍物有顶点的情况下，机器人过禁区时沿禁区边缘圆弧转弯路径最小。下面就机器人通过圆形障碍物的情况是否符合以上结论进行讨论。

如图 5，分别是机器人通过圆形禁区的 3 种情况，其中 a 图为机器人沿禁区边缘进行转弯的。b 图为机器人沿以圆形障碍物边上一点为圆形，最小转弯半径为转弯半径的弧转弯图，c 图为机器人沿一个刚好绕过禁区的圆弧转弯的情况。

图 5 机器人通过圆形障碍物



由以上用弹性绳的方法进行证明发现，b 图转弯路径最小，但是经过了禁区，因此不可行，另外两种方案 a 图路径明显要比 c 图小。因此机器人过禁区时沿禁区边缘圆弧转弯路径最小得结论同样适用于障碍物位圆的情况。转弯半径为：

$$\rho = r + d \quad (\text{其中 } r \text{ 为障碍物半径, } d \text{ 为机器人距离障碍物的最小安全距离})$$

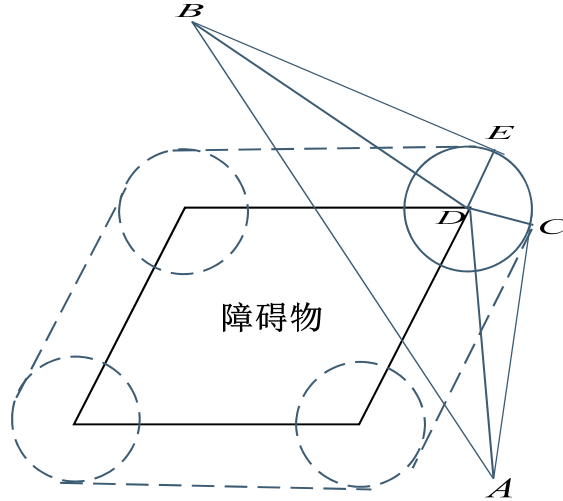
5.1.1.2 模型的建立

经过以上证明得到，起点到目标点中间不管有多少障碍物，最短路径都是由若干个相切的直线和圆弧构成的，前面已经证明机器人经过所有障碍物时，禁区边缘转弯路径最短，因此转弯圆弧半径为危险区域半径。所以在下面模型中经过障碍物转弯时，都以障碍物顶点为转弯弧圆心，最小转弯半径 r 为转弯半径。

【1】途中转弯一次的模型（模型一）

如图 6 所示，已知机器人要从起点 $A(x_1, y_1)$ 点出发绕过障碍物到终点 $B(x_2, y_2)$ 。途中从以障碍物顶点 $D(x_3, y_3)$ 为圆心， r 为半径的弧上转弯，切点为 C 和 E 。需要求 C 点和 E 点坐标，以及 \widehat{ACEB} 的长度。

图 6 机器人过障碍物转弯图



根据两点距离公式：

$$AB = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

$$AD = \sqrt{(x_1 - x_3)^2 + (y_1 - y_3)^2}$$

$$BD = \sqrt{(x_3 - x_2)^2 + (y_3 - y_2)^2}$$

由于 C 点和 E 点是切点，所以

$$DE \perp BE \quad DC \perp AC$$

根据勾股定理：

$$BE = \sqrt{BD^2 - r} = \sqrt{(x_3 - x_2)^2 + (y_3 - y_2)^2 - r}$$

$$AC = \sqrt{AD^2 - r} = \sqrt{(x_1 - x_3)^2 + (y_1 - y_3)^2 - r}$$

◆切点坐标的确定

假设 C 点坐标为 (x_i, y_i) ， E 点坐标为 (x_j, y_j) 。那么根据两点距离公式还有：

$$\begin{cases} BE = \sqrt{(x_2 - x_j)^2 + (y_2 - y_j)^2} \\ DE = \sqrt{(x_3 - x_j)^2 + (y_3 - y_j)^2} \\ AC = \sqrt{(x_1 - x_i)^2 + (y_1 - y_i)^2} \\ DC = \sqrt{(x_3 - x_i)^2 + (y_3 - y_i)^2} \end{cases}$$

综上所述，对于 E 点坐标就有：

$$\begin{cases} \sqrt{(x_3 - x_2)^2 + (y_3 - y_2)^2} - r = \sqrt{(x_2 - x_j)^2 + (y_2 - y_j)^2} \\ \sqrt{(x_3 - x_j)^2 + (y_3 - y_j)^2} = r \end{cases}$$

对于 C 点坐标：

$$\begin{cases} \sqrt{(x_1 - x_3)^2 + (y_1 - y_3)^2} - r = \sqrt{(x_1 - x_i)^2 + (y_1 - y_i)^2} \\ \sqrt{(x_3 - x_i)^2 + (y_3 - y_i)^2} = r \end{cases}$$

解以上方程组即可得到两个切点 C 和 E 的坐标。

◆转弯弧长的确定

在 $\triangle ADB$ 中，根据余弦定理：

$$\angle ADC = \arccos \frac{AD^2 + BD^2 - AB^2}{2AD \times BD}$$

在 $Rt\triangle ACD$ 中：

$$\angle ADC = \arccos \frac{r}{AD}$$

在 $Rt\triangle BED$ 中：

$$\angle BDE = \arccos \frac{r}{BD}$$

对于 $\angle CDE$ 就有：

$$\angle CDE = 2\pi - \angle ADC - \angle BDE - \angle ADB$$

因此弧 \widehat{CE} 的长度就为：

$$\widehat{CE} = r \times \angle CDE$$

◆总路径和总时间的确定

机器人途中转弯一次路径长度就为

$$S = AC + BE + \widehat{CE}$$

所用时间就为

$$T = \frac{AC}{v_0} + \frac{BE}{v_0} + \frac{\widehat{CE}}{v_\rho}$$

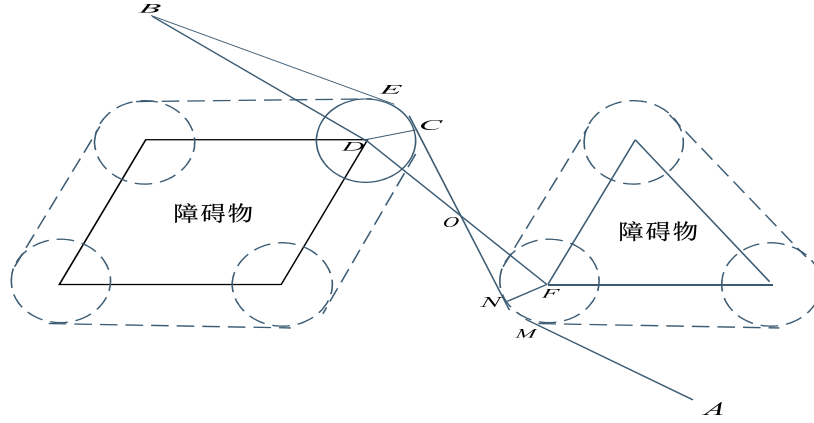
【2】途中转弯多次的模型（模型二）

机器人在从起点到终点的过程中，如果障碍物多的话，机器人需要通过多次转弯才能到达目标点，多次转弯可以分解成由多个两次转弯和一次转弯组成，而在两次转弯的过程中，两个转弯弧度的公切线分为内公切线和外公切线两种。

◆沿内公切线前进的情况

如图 7 所示，机器人从起始点 $A(x_1, y_1)$ 到目标点 $B(x_2, y_2)$ 要经过两个障碍物，途中从障碍物内侧转两次弯，已知两个障碍物转弯处的顶点为 $D(x_3, y_3)$ ， $F(x_4, y_4)$ ，圆 D 和圆 F 的半径均为 r 。

图 7 内公切线路径图



两圆心连线与内切线交点为 $O(x_5, y_5)$ 点。易证：

$$Rt\triangle DCO \cong Rt\triangle FNO$$

因此 O 点为线段 DF 的中点，根据中点坐标公式：

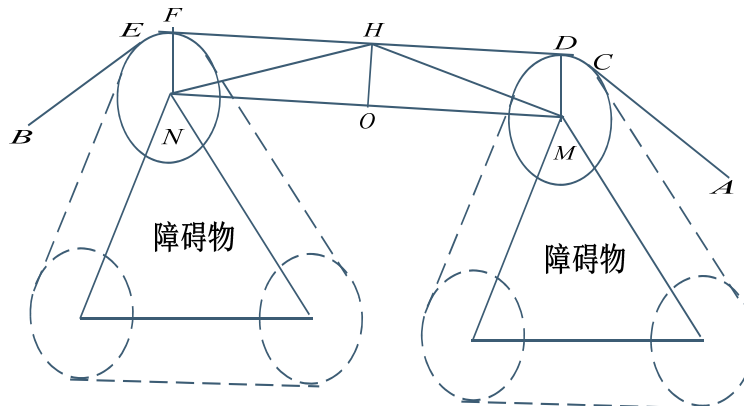
$$\begin{cases} x_5 = \frac{x_3 + x_4}{2} \\ y_5 = \frac{y_3 + y_4}{2} \end{cases}$$

确定 O 点坐标后，用模型一就可以分别求出 \widehat{AMNO} 和 $\widehat{OC\hat{E}B}$ 的长度，以及确定 C 、 E 、 M 、 N 点坐标。

◆沿外公切线前进的情况

如图 8 所示，机器人从起始点 $A(x_1, y_1)$ 到目标点 $B(x_2, y_2)$ 要经过两个障碍物，途中从障碍物外侧转两次弯，已知两个障碍物转弯处的顶点为 $M(x_3, y_3)$ ， $N(x_4, y_4)$ ，圆 M 和圆 N 的半径均为 r 。

图 8 外公切线路径



在两圆心的连线上找出中点 $O(x_5, y_5)$ ，过 O 点作 DF 垂线，交 DF 于 $H(x_a, y_a)$ 点。

$$OM = ON = \frac{1}{2} \sqrt{(x_3 - x_4)^2 + (y_3 - y_4)^2}$$

根据中点坐标公式可得 O 点坐标：

$$\begin{cases} x_5 = \frac{x_3 + x_4}{2} \\ y_5 = \frac{y_3 + y_4}{2} \end{cases}$$

由于 $MD \perp DF, NF \perp DF$ ，所以四变形 $MNDF$ 为矩形，因此 H 也为 DF 的中点，且 $HO=r$ 。

根据勾股定理：

$$HN = \sqrt{OH^2 + ON^2} = \sqrt{\frac{1}{4}(x_3 - x_4)^2 + \frac{1}{4}(y_3 - y_4)^2 + r^2}$$

$$HO = \sqrt{\left(x_a - \frac{x_3 + x_4}{2}\right)^2 + \left(y_a - \frac{y_3 + y_4}{2}\right)^2}$$

因此对于 H 点坐标就有：

$$\begin{cases} \sqrt{\left(x_a - \frac{x_3 + x_4}{2}\right)^2 + \left(y_a - \frac{y_3 + y_4}{2}\right)^2} = r \\ \sqrt{\frac{1}{4}(x_3 - x_4)^2 + \frac{1}{4}(y_3 - y_4)^2} = \sqrt{(x_a - x_4)^2 + (y_a - y_4)^2} \end{cases}$$

解上面方程组可以求出 H 点坐标 (x_a, y_a) ，确定 H 点坐标后，就可以用模型一分别求出路径 $A\hat{C}D\hat{H}$ 和 $H\hat{F}E\hat{B}$ 长度，以及切点 C 、 D 、 E 、 F 坐标。以及所用时间。

5.1.1.3 模型计算

假设机器人从起始点到目标点经过的路径有 m 条直线，长度分别为 d_m ，有 n 条弧线，长度分别为 u_n ，则机器人到达目标点的总路为

$$s = \sum_{m=1}^m d_m + \sum_{n=1}^n u_n$$

途中所用时间为：

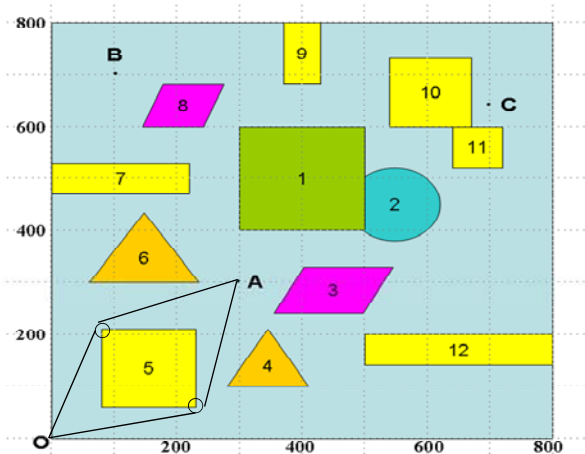
$$T = \frac{\sum_{m=1}^m d_m}{v_0} + \frac{\sum_{n=1}^n u_n}{v_\rho}$$

其中 $v_\rho = \frac{v_0}{1 + e^{10-0.1\rho^2}}$ ， ρ 是转弯半径。

【1】O-A 的最短路径

如图 9 所示，根据前文证明， O - A 点可能的最短路线有两条，从 5 号障碍物左上角处拐弯，或从 5 号障碍物右下角处拐弯。

图 9 O - A 可能最佳路线



两条路线都为基本的线圆组合，用模型一可直接求解，用 MATLAB 程序解得，两条路径分别为 471.0372，505.9853 个单位，因此最佳总路程为从障碍物 5 左上经过路线，总路程为 $S = 471.0372$ 个单位；总时间 $T = 96.0178$ 秒，经过二条直线段和一以条以 (80 210) 为圆心半径 $r = 10$ 的弧线段，途中具体情况如表 1：

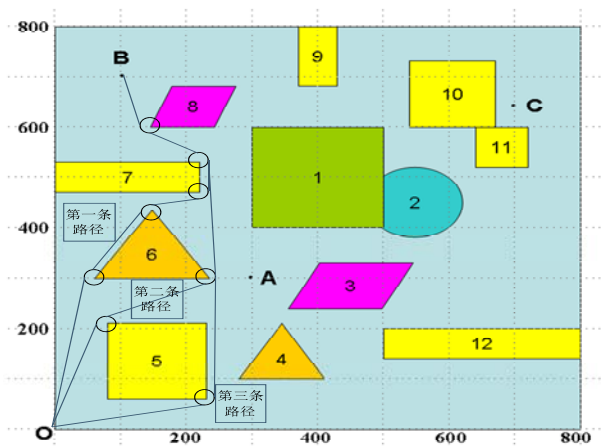
表 1 0-A 最短路径途中情况

	起点坐标点	终点坐标点	转弯弧圆心	距离	时间
直线 1	0 0	70.50595 213.1405	—	224.4994	44.9
弧线 1	70.50595 213.1405	76.60645 219.4066	80 210	9.051	3.7402
直线 2	76.60645 219.4066	300 300	—	237.4868	47.4974
总和	—	—	—	471.0372	96.1376

【2】0-B 的最短路

如图 10 所示，机器人从 0 到 B 点有三条可能最短路线，在路径途中经过多次转弯到达目标点，因此先用模型二把路径分为若干个基本线圆组合，然后用模型一对每个线圆组合求解，可得出答案。

图 10 0-B 可能最佳路线



经 MATLAB 计算可得到三条路线的路程分别为 853.7127，865.4391，901.299 个单位；最佳总路程为路线一，总路程为 $S = 853.7127$ 个单位；总时间 $T = 179.0851$ 秒，经过六条直线段和五条分别以 (60 300)，(150 435)，(220 470)，(220 530)，(150 600) 为圆心半径 $r = 10$ 的弧线段，具体起始点如表 2：

表 2 O-B 最短路径途中情况

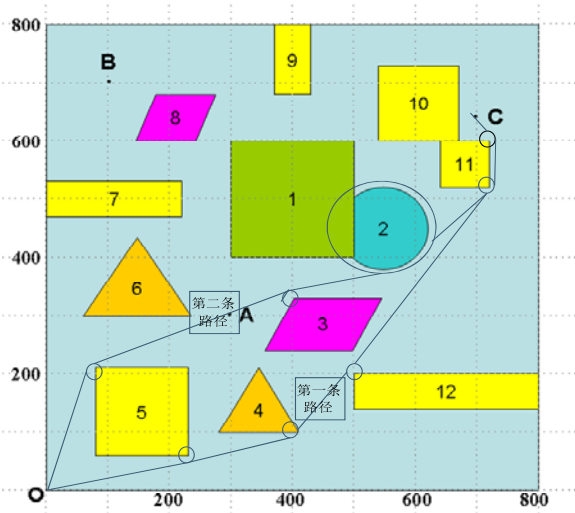
	起点坐标点		终点坐标点		转弯弧圆心	距离	时间
直线 1	0	0	50.136	301.649	—	305.78	61.156
弧线 1	50.136	301.649	51.6564	305.512	60 300	4.188	1.6752
直线 2	51.6564	305.512	157.5493	428.4419	—	162.25	32.45
弧线 2	157.5493	428.4419	159.9754	435.7009	150 435	7.854	3.1416
直线 3	159.9754	435.7009	229.2334	466.1602	—	75.66	15.132
弧线 3	229.2334	466.1602	230	470	220 470	13.6136	5.44544
直线 4	230	470	230	530	—	60	12
弧线 4	230	530	225.1602	538.5658	220 530	9.9484	3.97936
直线 5	225.1602	538.5658	144.4749	591.6149	—	96.95	19.39
弧线 5	144.4749	591.6149	140.6933	596.3414	150 600	6.1087	2.44348
直线 6	140.6933	596.3414	100	700	—	111.36	22.272
总和	—		—		—	853.7127	179.085

【2】O-C 的最短路径

如图 11 所示，机器人从 O 到 C 点有三条可能最短路线，在路径途中经过多次转弯到达目标点，因此先用模型二把路径分为若干个基本线圆组合，然后用模型一对每个线圆组合求解，可得出答案。

特别要说明的是在 O-C 的第二条路径中，通过前文的证明，经过圆形障碍物的转弯半径应该为 $R + \rho = 70$ 。

图 11 O-C 可能最佳路线



经 MATLAB 计算得到距离分别为 1087.6，1117.418 个单位，最佳总路径为路线一，总距离为 $S = 1087.6$ 个单位；总时间 $T = 221.9$ 秒，经过五条直线段和四条分别以

(410 100), (500 200), (720 540), (720 600)为圆心半径 $r=10$ 的弧线段, 途中具体情况如表 3:

表 3 0-C 最短路径途中情况

	起点坐标点	终点坐标点	转弯弧圆心	距离	时间
直线 1	0 0	412.0796 90.2186	—	421.8 ₄	84.368
弧线 1	412.0796 90.2186	418.2907 94.4085	410 100	7.679 ₄	3.0718
直线 2	418.2907 94.4085	491.6 205.4275	—	133.0 ₄	26.608
弧线 2	491.6 205.4275	492 206	500 200	0.698 ₁	0.2792
直线 3	492 206	727.8719 513.8328	—	387.8 ₁	77.562
弧线 3	727.8719 513.8328	730 520	720 540	6.457 ₇	2.5751
直线 4	730 520	730 600	—	80	16
弧线 4	730 600	727.7715 606.2932	720 600	6.806 ₈	2.7227
直线 5	727.7715 606.2932	700 640	—	43.59	8.718
总和	—	—	—	1087.6	221.904 ₈

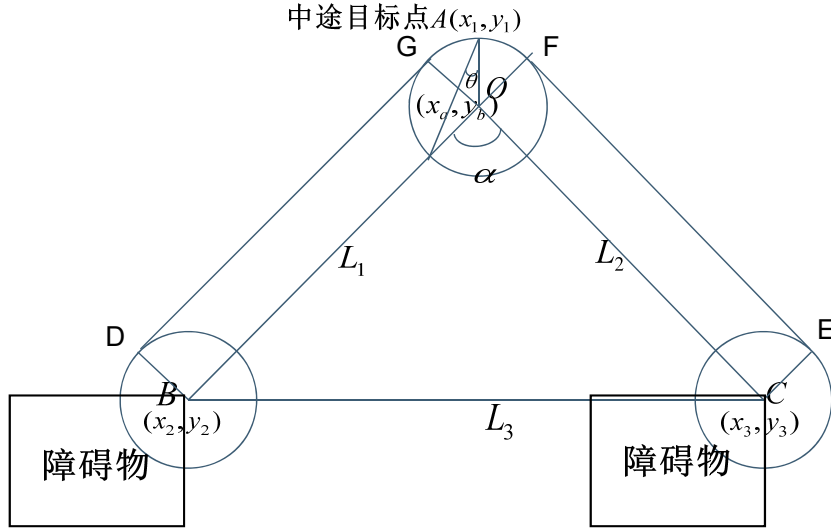
5.1.2 经过多个目标点 (0-A-B-C-0)

本问题的特点是要经过中间若干个目标点后在回到起点, 不能简单的处理为求解每个两点之间的最短路径之和, 因为机器人不能直线转向, 所以在经过目标点时, 应该考虑提前转向, 且中间目标点应该在转弯弧上。因此对于中间目标点处转弯圆弧圆心的确定为本题的一个难点。

◆经过中间目标点的路径 (模型三)

如图12所示已知点 $B(x_2, y_2)$, $C(x_3, y_3)$ 分别为两个障碍物的顶点, 且机器人要绕过B点障碍物经过中间目标点 $A(x_1, y_1)$ 到另一个障碍物顶点转弯。其中 $A(x_1, y_1)$ 为在圆心为 $O(x_a, y_b)$ 半径为 R 的圆上。由于点 $B(x_2, y_2)$, $C(x_3, y_3)$ 分别引同侧相切于圆心为 $O(x_a, y_b)$ 的圆的切线, 因此我们需要用点 $B(x_2, y_2)$ 和点 $C(x_3, y_3)$ 来确定圆心 $O(x_a, y_b)$ 的位置。

图12 机器人经过中途目标点图



由于需要确定的圆心 $O(x_a, y_b)$ 是以 R 为半径点 $A(x_1, y_1)$ 为圆上一点，所以需要进行搜索求解求出要确定的圆心。这样就将目标转化为距离和弧长最短的目标模型。

$$\text{Min } z = (\pi - \alpha) \times r + l_1 + l_2$$

根据点 $A(x_1, y_1)$ 是圆上一点的关系就可以 $A(x_1, y_1)$ 点为圆心 R 为半径搜索得出圆心 $O(x_a, y_b)$ ，那么得出关系为：

$$x_a = x_1 + r \times \sin \theta$$

$$y_a = y_1 - r \times \cos \theta$$

将上述关系转化为圆的标准方程可以得到：

$$(x_a - x_1)^2 + (y_a - y_1)^2 = r^2$$

根据两点直接的距离公式可以得出以下条件：

$$(x_a - x_2)^2 + (y_a - y_2)^2 = l_1^2$$

$$(x_a - x_3)^2 + (y_a - y_3)^2 = l_2^2$$

$$(x_2 - x_3)^2 + (y_2 - y_3)^2 = l_3^2$$

由于 DG 与 EF 是两个圆的同侧公切线，两圆的半径相等而 DG 与 OB 和 OC 与 EF 是矩形的对边，所以 $DG = OB = L_1$ ， $EF = OC = L_2$ 。

在 $\triangle BOC$ 中由余弦定理可以得出：

$$\cos \alpha = \frac{l_1^2 + l_2^2 - l_3^2}{2 \times l_1 \times l_2}$$

最后我们建立的模型为：

$$\text{Min } z = (\pi - \alpha) \times R + L_1 + L_2$$

$$\begin{cases} (x_a - x_1)^2 + (y_a - y_1)^2 = r^2 \\ (x_a - x_2)^2 + (y_a - y_2)^2 = l_1^2 \\ (x_a - x_3)^2 + (y_a - y_3)^2 = l_2^2 \\ (x_2 - x_3)^2 + (y_2 - y_3)^2 = l_3^2 \\ \cos \alpha = \frac{l_1^2 + l_2^2 - l_3^2}{2 \times l_1 \times l_2} \end{cases}$$

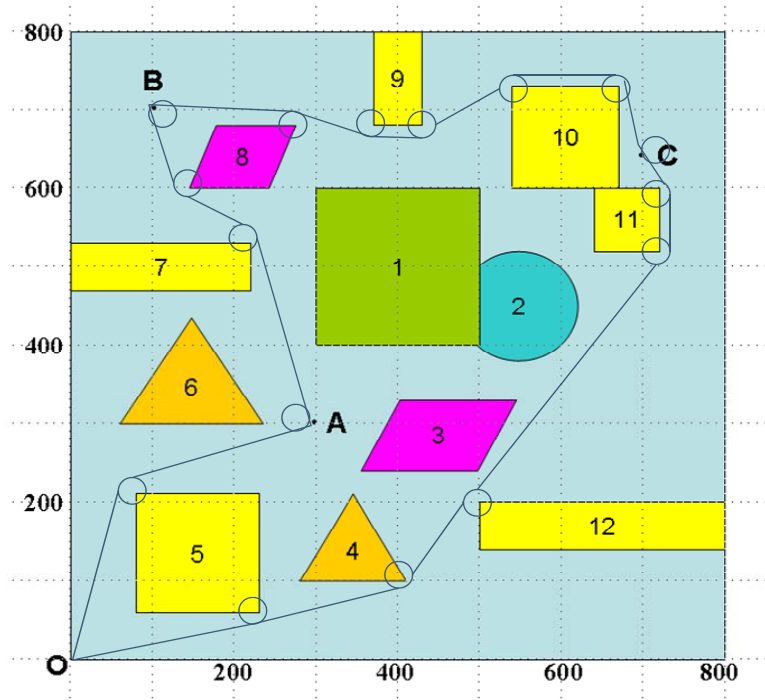
根据上面模型我们可以求解出 A 和 B 圆心坐标，由于 C 点的圆心与其他附近的圆是交错相切，根据交错相切时切线与圆心的连线的交点是两圆心的中间点（前文已经证明）可以得出与同侧相切时对应的点 $B(c,d)$ 和 $C(e,f)$ 的坐标，所以我们只需套用上面同侧相切的模型求解 C 点的圆心坐标。 C 点的圆心与 C 的连线刚好与 $B(c,d)$ 到圆的切线垂直，即点 C 是 $B(c,d)$ 与圆相交的切点。

通过本模型可以求出两个障碍物中间为中途目标点的线圆组合，其他情况的线圆组合可用模型一和模型二进行求解。

◆模型计算

如图 13 所示，通过前文证明以及单目标点的计算模型，可以得到从 O 点出发，经过 A 、 B 、 C 三点在回到 O 点的最短路线只有一条。首先利用模型三算出 A 、 B 、 C 三点处转弯弧度的圆心坐标，然后用模型二把路径转化为若干个基本线圆组合。利用模型一计算。

图 13 $O \rightarrow A \rightarrow B \rightarrow C \rightarrow O$ 最短路径



利用 MATLAB 编程计算得到，机器人从 $O \rightarrow A \rightarrow B \rightarrow C \rightarrow O$ 最佳总路程为

$S = 2812.52$ 个单位；总时间 $T = 585.6712$ 秒，一共经过十六条直线段和十五条分别以 $(80\ 210)$ $(290.8855\ 304.1141)$ ， $(220\ 530)$ ， $(150\ 600)$ ， $(108.2296\ 694.3191)$ ， $(270\ 680)$ ， $(370\ 680)$ ， $(420\ 680)$ ， $(540\ 730)$ ， $(670\ 730)$ ， $(709.7933\ 642.0227)$ $(700\ 640)$ ， $(720\ 600)$ ， $(500\ 200)$ ， $(410\ 100)$ 为圆心半径 $r = 10$ 的弧线段，具体起始点如附表 1：

表 4 经过多目标点最短路径途中部分情况

	起点坐标点	终点坐标点	转弯弧圆心	距离	时间
直线 1	0 0	70.5059 213.1405	—	224.4994	44.8999
弧线 1	70.5059 213.1405	76.6064 219.4066	80 210	9.25	3.7
直线 2	76.6064 219.4066	294.1547 294.6636	—	229.98	45.996
弧线 2	294.1547 294.6636	281.3443 301.7553	290.8855 304.1141	15.3589	6.1436
...
直线 14	727.9377 513.9178	492.0623 206.0822	—	387.81	77.562
弧长 14	492.0623 206.0822	491.6552 205.5103	500 200	0.6981	0.2792
直线 15	491.6552 205.5103	412.1387 90.2314	—	133.04	26.608
弧线 15	412.1387 90.2314	418.3348 94.4085	410 100	7.6794	3.0718
直线 16	418.3348 94.4085	0 0	—	421.84	84.368
总和				2812.52	585.6712

5.2 最短时间路径模型（问题二）

◆模型分析

本问题是研究机器人从 0 点出发绕过障碍物 5 到达 A 的最短时间，根据问题一所求最短路径，可以知道最短时间路线也是从障碍物 5 的左上通过，机器人所有的时间由直线段和弧线段两部分时间组成。由于最短时间和最短距离不是同一条路线，因此在求最短时间时先要确定最短时间的路径。机器人靠近障碍物时距离障碍物的最近距离不能少于 10 个单位。所以根据机器人在行走路线中要避免遇到的障碍物和影响其行动的范围来确定机器人行走的范围。

机器人转弯时最大转弯速度为：

$$v = v(\rho) = \frac{v_0}{1 + e^{10 - 0.1\rho^2}}$$

机器人经过圆弧时间为：

$$t = \frac{\alpha \rho}{v_\rho}$$

由上式可以看出当转弯半径变小时，机器人速度也变小，而机器人通过的圆弧长度也变小。 t 的变化不好确定。

当机器人转弯半径增大时，转弯速度也增大，而机器人通过的圆弧长度也随着增大。 t 的变化不好确定。

因此对于转弯时间 t 随转弯半径 ρ 的变化关系，没有直接的线性关系。而对于转弯半径 ρ 的变化在题目中是有限制的， ρ 过大会发生碰撞。 ρ 过小（小于 10）会发生侧翻。

所以我们可以根据 ρ 的变化范围，建立最小时间的非线性规划模型，对 t 最小时的 ρ 进行搜索求解。

◆模型准备：

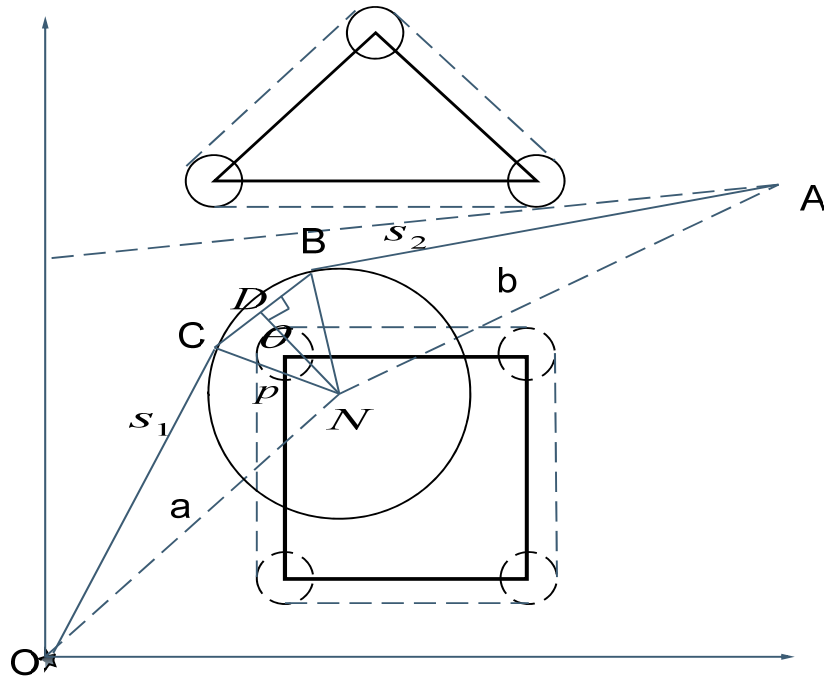
如图 14 所示，已知 $O(x_1, y_1)$ 是起始点， $A(x_2, y_2)$ 是终点， $P(x_3, y_3)$ 是障碍物 5 的左上顶点。假设机器人在 $C(x_c, y_c)$ 处以 $N(x, y)$ 为圆心， r 为半径转弯，经弧 \widehat{BC} ，在 $B(x_b, y_b)$ 点转弯，可以得到最短时间路径。连结 BC ，做 $ND \perp$ 于 BC 。 ON 长度为 a ， AN 长度为 b ，切线 OC 的长度为 s_1 ，切线 AB 的长度为 s_2 。

令 $BD = d$ ，由于 B 点和 C 点为切点，所以

$$\angle DNB = \frac{1}{2}\theta$$

$$BD = \frac{1}{2}BC$$

图 14 最短时间示意图



由两点之间的距离公式得出：

$$a = \sqrt{(x - x_1)^2 + (y - y_1)^2}$$

$$b = \sqrt{(x - x_2)^2 + (y - y_2)^2}$$

因为 B 点和 C 点是切点，所以：

$$\begin{aligned}\sqrt{(x_c - x)^2 + (y_c - y)^2} &= r \\ \sqrt{(x_b - x)^2 + (y_b - y)^2} &= r\end{aligned}$$

根据勾股定理可以得到：

$$\begin{aligned}s_1 &= \sqrt{a^2 - r^2} \\ s_2 &= \sqrt{b^2 - r^2}\end{aligned}$$

根据距离公式：

$$BC = \sqrt{(x_b - x_c)^2 + (y_b + y_c)^2}$$

在 $Rt\triangle NBD$ 中

$$\sin\left(\frac{\theta}{2}\right) = \frac{\sqrt{(x_b - x_c)^2 + (y_b + y_c)^2}}{2 \times r}$$

所以

$$l = 2r \times \arcsin\left(\frac{\sqrt{(x_b - x_c)^2 + (y_b + y_c)^2}}{2r}\right)$$

◆目标分析

根据机器人行走的直线距离和速度以及弧线长度和对应的弧线速度建立最短时间的目标：

$$Min = \frac{s_1 + s_2}{v_o} + \frac{l}{v_p}$$

◆约束分析

根据机器人与障碍物的距离至少要超过 10 个单位可以得到半径 r 的约束条件，即必须使圆弧离障碍物 10 个以上的单位：

$$r - \sqrt{(x - x_3)^2 + (y - y_3)^2} \geq 10$$

对于两个切点的范围：

$$\begin{aligned}x_c &< 80 \\ y_b &> 210\end{aligned}$$

根据问题一证明的结论圆心在障碍物以内的距离最短可以得出约束条件为：

$$\begin{aligned}80 &\leq x \leq 230 \\ 0 &< y \leq 210\end{aligned}$$

因为机器人不能直线转弯，转弯路径由与直线路径相切的一段圆弧组成，因此对于 C、B 两点来说，必须保证是切点：

C、B 两点在圆上，与圆心的距离为半径 r ：

$$\begin{aligned}\sqrt{(x_c - x)^2 + (y_c - y)^2} &= r \\ \sqrt{(x_b - x)^2 + (y_b - y)^2} &= r\end{aligned}$$

$\triangle NOC$ 和 $\triangle ABN$ 必须是直角三角形，B 点和 C 点为两直角边交点。

$$s_1 = \sqrt{a^2 - r^2}$$

$$s_2 = \sqrt{b^2 - r^2}$$

◆模型的建立与求解

综合以上条件，建立时间最短的优化模型：

$$\begin{aligned} \text{Min} = & \frac{s_1 + s_2}{v_o} + \frac{l}{v_p} \\ \left\{ \begin{array}{l} r - \sqrt{(x - x_3)^2 + (y - y_3)^2} \geq 10 \\ r = \sqrt{(x_b - x)^2 + (y_b - y)^2} \\ r = \sqrt{(x_c - x)^2 + (y_c - y)^2} \\ s_2 = \sqrt{(x - x_2)^2 + (y - y_2)^2} - r^2 \\ s_1 = \sqrt{(x - x_1)^2 + (y - y_1)^2} - r^2 \\ 80 < x < 230 \\ y_b > 210 \\ x_c < 80 \\ 0 < y < 210 \end{array} \right. \end{aligned}$$

用 Lingor 软件编程求得最短时间为 94.22825 秒，转弯半径为 $r = 12.9886$ ，具体结果见表 5：

表 5 0-A 最短时间路径途中情况

	起点坐标点	终点坐标点	转弯弧圆心	距离	时间
直线 1	0 0	69.8045 211.9779	—	223.1755	44.6351
弧线 1	69.8045 211.9779	77.74918 220.1387	82.1414 207.9153	11.7899	2.360433
直线 2	77.74918 220.1387	0 0	—	236.1636	47.23272
总和	—			471.129	94.22825

◆结果分析

把结果同问题一比较发现最短路径未必所用时间最少，关键在于转弯半径和转弯圆心的选择。因此机器人在转弯过程中的转弯半径和转弯圆心的确定，对路线的优劣有着不可避免的影响。

6 模型推广与评价

◆模型改进

本题障碍物并不是太多，当障碍物多的时候利用本题目所用的穷举法是不现实的。

我们在做题目的时候有如下一些想法，先划去障碍物的危险区，再求出所有的切线，包括出发点和目标点到所有圆弧的切线以及所有圆弧与圆弧之间的切线，然后把这些相交联的线看成是最小生成树，和最短路。

假设机器人从起点 R 到到目标点 M_0 ，路径一定是由圆弧和线段组成，设有 m 条线

段，n 条圆弧。那么目标函数可以表示为：

$$\begin{aligned} \text{Min} &= \sum_{i=1}^m d_i + \sum_{j=1}^n l_j \\ &\begin{cases} r \geq 1 \\ k \geq 1 \end{cases} \end{aligned}$$

利用 matlab 或者 lingo 等计算机软件，用最编程求最优解。此程序可以起到起点到目标点之间的路径进行优化求解

◆模型评价

优点：

本模型简单易懂，便于实际检验及应用，可以应用到机动车驾驶证考试、城市汽车运输等一系列实际生活问题，主要优点：

- 1、运用多个方案对路径进行优化，在相对优化之中取得最优解。
- 2、模型优化后用解析几何进行求解，精确度较高。

缺点：

- 1、利用解析几何进行求解，精确度较高的前提计算时间比较慢，所以此模型利用起来效率比较低。
- 2、当障碍物足够多的时候本模型并不适合。

参考文献

- [1] 吴家龙 《弹性力学》 北京：高等教育出版社，2001
- [2] 姜启源 谢金星，数学建模，北京：高等教育出版社，2003 年；
- [3] 李 涛 贺勇军，应用数学篇，北京：电子工业出版社，2000 年。

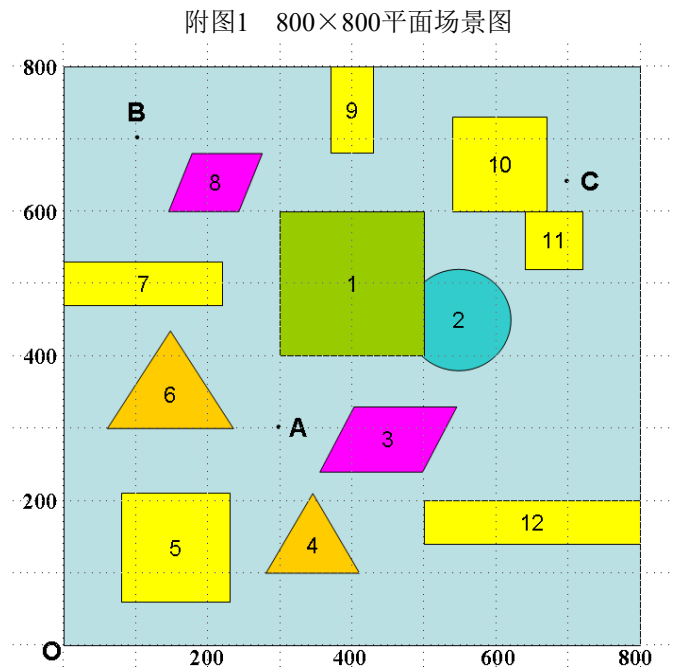
7附录

附表 1：经过多目标点最短路径途中具体情况

	起点坐标点	终点坐标点	转弯弧圆心	距离	时间
直线 1	0 0	70.5059 213.1405	—	224.4994	44.8999
弧线 1	70.5059 213.1405	76.6064 219.4066	80 210	9.25	3.7
直线 2	76.6064 219.4066	294.1547 294.6636	—	229.98	45.996
弧线 2	294.1547 294.6636	281.3443 301.7553	290.8855 304.1141	15.3589	6.1436
直线 3	281.3443 301.7553	229.8206 531.8855	—	236.83	47.366
弧长 3	229.8206 531.8855	225.4967 537.6459	220 530	6.8068	2.7228
直线 4	225.4967 537.6459	144.5033 591.6462	—	96.95	19.39
弧长 4	144.5033 591.6462	140.8565 595.9507	150 600	5.7596	2.3038
直线 5	140.8565 595.9507	99.08612 690.2698	—	103.16	20.632
弧线 5	99.08612 690.2698	109.113 704.2802	108.296 694.3191	20.7694	8.3078
直线 6	109.113 704.2802	270.8817 689.9611	—	162.4	32.48
弧长 6	270.8817 689.9611	272 689.7980	270 680	1.0472	0.4188
直线 7	272 689.7980	368 670.282	—	97.98	19.596
弧线 7	368 670.282	370 670	370 680	2.0944	0.8378
直线 8	370 670	430 670	—	60	12
弧长 8	430 670	435.5878 671.7068	420 680	5.9341	2.3736
直线 9	435.5878 671.7068	534.4115 738.2932	—	119.16	23.832
弧线 9	534.4115 738.2932	540 740	540 730	5.9341	2.3736
直线 10	540 740	670 740	—	130	26
弧长 10	670	679.9126	670	14.3846	5.7538

	740	731.3196	730		
直线 11	679.9126 731.3196	690.9183 648.6458	—	83.403	16.6806
弧线 11	690.9183 648.6458	693.5095 643.1538	709.7933 642.0227	6.17	2.468
直线 12	693.5095 643.1538	727.3214 606.8116	—	129.6305	25.9261
弧长 12	727.3214 606.8116	730 600	720 600	7.4928	2.997
直线 13	730 600	730 520	—	80	16
弧线 13	730 520	727.9377 513.9178	720 520	6.4577	2.583
直线 14	727.9377 513.9178	492.0623 206.0822	—	387.81	77.562
弧长 14	492.0623 206.0822	491.6552 205.5103	500 200	0.6981	0.2792
直线 15	491.6552 205.5103	412.1387 90.2314	—	133.04	26.608
弧线 15	412.1387 90.2314	418.3348 94.4085	410 100	7.6794	3.0718
直线 16	418.3348 94.4085	0 0	—	421.84	84.368
总和	—			2812.52	585.6712

附图



程序一：基本线圆组合计算程序（仅以 O-A 为例，其它类似）

```
clc
o=[0,0];
c=[80,210];
m=[300,300];
r=10;
oc=sqrt((o(1)-c(1))^2+(o(2)-c(2))^2);
om=sqrt((o(1)-m(1))^2+(o(2)-m(2))^2);
cm=sqrt((c(1)-m(1))^2+(c(2)-m(2))^2);
a1=acos((oc^2+cm^2-om^2)/(2*oc*cm));
a2=acos(r/oc);
a3=acos(r/cm);
a4=2*pi-a1-a2-a3;
d1=sqrt(oc^2-r^2);
d2=sqrt(cm^2-r^2);
d3=r*a4;
dd=d1+d2+d3
```

程序二：求 A、B 点所在的圆心坐标（以 A 为例）；

%%求经过（A）点弧的圆心坐标

```
clc
clear
%%坐标
A=[300,300]; %所求圆弧上的坐标
B=[80,210]; %一边圆心的坐标
C=[220,530]; %另一边圆心的坐标
R=10; %圆的半径
sita=[-pi:0.0001:pi]; %搜索圆心坐标的范围
for i=1:length(sita)
O=[A(1)+R*sin(sita(i)),A(2)-R*cos(sita(i))];
l1=sqrt((O(1)-B(1))^2+(O(2)-B(2))^2);
l2=sqrt((O(1)-C(1))^2+(O(2)-C(2))^2);
l3=sqrt((B(1)-C(1))^2+(B(2)-C(2))^2);
afa=acos((l1^2+l2^2-l3^2)/(2*l1*l2));
L(i)=l1+l2+R*(pi-afa); %直线与弧长的路程
end
sita(find(L==min(L))) %最短路程
min(L)
[A(1)+R*sin(sita(find(L==min(L)))),A(2)-R*cos(sita(find(L==min(L))))] %
圆心坐标
```

程序三：求 C 点所在圆的圆心坐标；

```
clc
clear
%%坐标
```

```

A=[700,640];
C=[720,600];
B=[670,730];
R=10;
sita=[(51/90)*pi/2:0.001:(80/90)*pi/2];
for i=1:length(sita)
O=[A(1)+R*sin(sita(i)),A(2)+R*cos(sita(i))];
l1=sqrt((O(1)-B(1))^2+(O(2)-B(2))^2);
l2=sqrt((O(1)-C(1))^2+(O(2)-C(2))^2);
l3=sqrt((B(1)-C(1))^2+(B(2)-C(2))^2);
afa=acos((l1^2+l2^2-l3^2)/(2*l1*l2))
afa1=acos(20/l1)
afa2=acos(20/l2)
L(i)=sqrt(l1^2-400)+sqrt(l2^2-400)+R*(afa-afa1-afa2);
end
sita(find(L==min(L)))
min(L)
[A(1)+R*sin(sita(find(L==min(L)))),A(2)+R*cos(sita(find(L==min(L)))))]

```

程序四：问题二求最短时间路径 lingo 程序

```

min=@sqrt((x-300)^2+(y-300)^2-r^2)/5+@sqrt(x^2+y^2-r^2)/5+2*r*@asin(d/(2*r))
)/5*(1+@exp(10-0.1*r^2));
d=@sqrt((xc-xb)^2+(yc-yb)^2);
@sqrt((xb)^2+(yb)^2+r^2)=@sqrt(x^2+y^2);
@sqrt((xc-300)^2+(yc-300)^2+r^2)=@sqrt((x-300)^2+(y-300)^2);
init:
x=80;
y=210;
endinit
xb<=80;
yc>210;
r=@sqrt((xb-x)^2+(yb-y)^2);
r=@sqrt((xc-x)^2+(yc-y)^2);
r>@sqrt((80-x)^2+(210-y)^2)+10;
x>80;
x<200;
y<210;

```