

引文格式: 官祥锦, 陈娟, 张为民. 基于改进 A* 算法的多 AGV 路径规划研究[J]. 航空制造技术, 2023, 66(5): 76-85, 90.

GUAN Xiangjin, CHEN Juan, ZHANG Weimin. Research on multi-AGV path planning based on improved A* algorithm[J]. Aeronautical Manufacturing Technology, 2023, 66(5): 76-85, 90.

基于改进 A* 算法的多 AGV 路径规划研究

官祥锦¹, 陈娟¹, 张为民²

(1. 北京化工大学, 北京 100029;

2. 中国航空制造技术研究院, 北京 100024)

[摘要] 自动导引车 AGV (Automated guided vehicle, AGV) 的路径规划问题是工业生产和物流领域的关键问题, 其中多 AGV 无碰撞路径规划是研究的难点。本文以实际工业生产现场为背景, 提出一种改进的 A* 算法, 该算法利用切比雪夫距离对传统 A* 算法的启发函数进行加权, 显著地减少了 A* 算法的搜索时间和搜索节点数, 提高了传统 A* 算法的路径搜索效率, 并将改进的 A* 算法和时间窗模型结合, 通过时间窗模型提前预判多 AGV 路径上节点占用情况, 根据生产任务需求和 AGV 离终点的远近程度, 动态调整 AGV 的优先级, 有效地解决了多 AGV 同时行驶时产生的死锁, 碰撞冲突问题。试验结果表明, 该算法在多 AGV 进行动态路径规划时, 路径的搜索效率得到显著的提高, 路径冲突问题也得到有效的解决。

关键词: AGV 路径规划; 改进 A* 算法; 多 AGV 调度; 时间窗模型; 多 AGV 冲突

Research on Multi-AGV Path Planning Based on Improved A* Algorithm

GUAN Xiangjin¹, CHEN Juan¹, ZHANG Weimin²

(1. Beijing University of Chemical Technology, Beijing 100029, China;

2. AVIC Manufacturing Technology Institute, Beijing 100024, China)

[ABSTRACT] The path planning of automated guided vehicles (AGV) is an important research topic in the field of industrial production and logistics. Collision-free path planning of multiple AGVs is a difficult problem in research. In this paper, based on the actual industrial production site, the traditional A* algorithm is improved by using the Chebyshev distance, which significantly reduces the search time and the number of search nodes of the A* algorithm, and improves the path search efficiency at the same time. The improved A* algorithm combined with the time window algorithm is used to solve this problem. Pre-judge the node occupancy on multiple AGV paths through the time window model, and dynamically adjust the AGV priority according to the production task requirements and the distance of the AGV from the end point. This algorithm effectively solves the deadlock and collision problems caused by multiple AGVs driving at the same time. The experimental results show that when the algorithm is used for dynamic path planning of multiple AGVs, the path search efficiency is significantly improved and the path conflict problem is effectively resolved.

Keywords: AGV path planning; Improved A* algorithm; Multiple AGVs scheduling; Time window model; Multi-AGV conflict

DOI: 10.16080/j.issn1671-833x.2023.05.076

AGV 在工业生产和物流领域扮演着越来越重要的作用, 能够有效地改善工业生产和生活的运输系统结构, 将人力从繁琐重复的体力劳动中解放出来, 不仅大大地降低了生产成本, 也提高了生产效率^[1]。而多

AGV 的路径规划和无碰撞调度是 AGV 应用面临的两大问题^[2-3], 其在环境复杂的工业现场和生产车间尤为突出。根据是否已知全局路径信息, 可将 AGV 路径规划分为全局路径规划和局部路径规划。全局路径规划

是指 AGV 在行驶前已经掌握了全局的路径信息,包括 AGV 的位置信息、所处环境的障碍物信息以及可行驶的路径信息,然后根据掌握的全局信息规划出一条从起点到终点的最优路径;局部路径规划是指 AGV 在行驶前对环境信息一无所知或知道部分环境信息,需要在行驶的过程中实时动态地了解自身的位置和环境信息,然后通过掌握的环境信息规划出一条从起点到终点的最优路径。目前应用于全局路径规划的算法主要有:栅格法(其中最具代表性的是 A* 算法^[4]、D* 算法和 Dijkstra 算法^[5-7])、快速搜索随机树法^[8](Rapidly-exploring random tree, RRT)、Voronoi 图法等。应用于局部路径规划的算法主要有:人工势场法^[9]、动态窗口法(Dynamic window approach, DWA),以及一些智能算法(其中主要有模糊算法、遗传算法^[10]、神经网络、粒子群算法^[11]、蚁群算法等)^[12-15]。由于计算的复杂度以及现实生产环境的限制,且大多工业生产和物流环境是全局已知的,所以在实际的 AGV 路径规划中,局部路径规划算法和智能路径规划算法应用的较少,大多还停留在仿真研究阶段,并没有应用到实际的生产生活中。

A* 算法由于其算法简单,在中小型地图中具有较好的搜索性能,是当前 AGV 进行路径规划时应用的最广泛的算法。但传统的 A* 算法存在需要搜索的节点数多、路径转角多、生成的路径平滑度低等问题^[16],这些问题在多 AGV 同时进行路径规划时尤为突出,特别是当多辆 AGV 存在路径冲突或者已规划好的路径上出现障碍物,需要对其进行重规划时,搜索的节点数将大大增加,降低了重规划的效率。针对传统 A* 算法存在的问题, Song 等^[17]做了很多研究,但大都是根据特定的应用环境进行的改进与研究。余星宝等^[18]采用传统 A* 算法找出最短路径上转弯处的特征点,再利用四阶贝塞尔曲线对转弯处的特征点进行平滑处理,得到 1 条无碰撞、平滑度高、具有一定曲率,且相对距离更短的路径,解决了传统 A* 算法生成的路径转角多、平滑度低、易发生碰撞的问题。但该方法对于只能沿着已铺设路径引导线行走的 AGV 不实用。胡蔚旻等^[19]以传统 A* 算法为基础,引入 3 轴 -2 象限、AGV 路径转向数来删除路径上的无效备选点,从而使 AGV 的行驶路径得到有效的平滑。但该算法在多辆 AGV 发生路径冲突时,不能对其路径进行动态调整,所以不适用于对多辆 AGV 进行路径规划的情况。刘生伟等^[20]在传统 A* 算法启发函数的基础上,引入了父节点的启发函数值,对当前节点和父节点启发函数之和进行加权处理,并通过判断转折点的方式,删除多余的冗余节点,使得规划出的路径更加平滑。该算法和传统的 A* 算法及指数衰减加权改

进 A* 算法相比,在搜索时间、路径长度以及路径包含的节点数上都减少。然而本文所使用的八邻域搜索方式,对于已铺设引导线的 AGV 来说并不实用。余文凯等^[21]将环境信息和 AGV 位置信息融入到评价函数中,并使用障碍率和直通率给 A* 算法的评价函数进行加权处理。通过基于 K-Means 聚类算法对栅格地图进行分区预处理后,再使用改进的 A* 算法搜索最短路径。对搜索到的最短路径,使用改进的 Floyd 算法对路径进行双向平滑度优化处理,并添加防碰撞安全距离系数,有效地增加了路径的平滑度和安全性。但该算法仅考虑单辆 AGV 的情况,并且对规划出的路径进行平滑处理的方式不适用于已铺设路径引导线的 AGV。卞永明等^[22]根据 AGV 从起点到当前点发生的转弯次数,对 A* 算法的启发函数进行奖励和惩罚,可以显著地降低转弯次数,但该算法没有考虑一条直线上的多余冗余节点。吴飞龙等^[23]将 AGV 的位置信息和环境信息融入到了 A* 算法的启发函数中,并使用环境的混乱程度和障碍率对启发函数进行加权处理,删除路径中多余的转折点和冗余节点,使 AGV 规划出的路径更加平滑。陆佳依等^[24]采用分裂和筛选的方案对传统 A* 算法进行优化,通过在未搜索节点的启发函数中增加转弯权值,在路径规划过程中考虑转弯带来的消耗,缩短了路径的总长度,减少了 AGV 的转弯次数。该方法明显提高了 A* 算法的复杂度,降低了搜索效率。

现有的 AGV 路径规划算法,考虑的情况大都比较理想化,没有对 AGV 的行驶方向做约束,导致在使用 A* 算法对 AGV 行驶路径进行规划时,既可以四方搜索,也可以八方搜索,从而出现规划的路径只要存在空白区域就可随便转弯的情况。这对于重载 AGV 以及对生产安全要求极为严格的工业生产现场来说是不现实的,在对 AGV 进行路径规划的过程中,如果不对 AGV 的行驶方向做约束,直接按照规划出的路径行驶,将非常容易和障碍物发生碰撞,发生安全事故。并且当涉及多辆 AGV 同时运行时,出于工业生产安全的考虑,对 AGV 的路径规划要求将变得更加严苛。此时,安全是路径规划优先考虑的问题,所以在进行多 AGV 路径规划时,需尽量将多 AGV 发生冲突碰撞的概率降到最低,其次才是 AGV 路径最优的问题,这种情况下,单纯地使用 A* 算法将不再满足路径规划要求。当涉及多辆 AGV 时,AGV 在运行的过程中,彼此互为障碍物,并且障碍物在实时移动,此时的 AGV 路径规划问题将从 1 个静态路径规划问题转变为 1 个动态路径规划问题。邢普学等^[25]通过增加 AGV 共享路径的惩罚值对 A* 算法进行改进,结合改进的 A* 算法设置了碰撞解决规则,

能够解决 AGV 的碰撞问题,但是该规则在 AGV 发生路径冲突时,只考虑对其进行重规划,重归化后的路径可能会导致新的碰撞出现及多 AGV 死锁的发生。泰应鹏等^[26]通过结合 A* 算法和时间窗模型,对多 AGV 进行了动态路径规划,通过提前计算 AGV 经过每个节点的时间,对时间窗进行动态的排布和更新,并为多辆 AGV 动态地分配优先级,实现了无冲突,无重复的系统调度。但是在实际中,准确计算每辆 AGV 到达每个节点的时间是困难的,如果发生较大的时间计算偏差,容易导致 AGV 发生碰撞问题。廉胤东等^[27]将当前节点到相邻候选节点的期望运行时间加入到传统 A* 算法的启发函数中,对传统 A* 算法进行了改进,在改进 A* 算法的基础上,对 AGV 下一个运行位置进行预判,实现 AGV 的冲突避让,提高了 AGV 路网资源的利用率。但该算法也存在计算量大,对 AGV 系统要求较高等问题。高新浩等^[28]基于改进时间窗的两阶段动态路径规划算法,构建环境拓扑地图,使用 A* 算法实现 AGV 离线路径规划。通过动态环境拓扑地图,使用 BP 神经网络实现 AGV 路径的动态规划,降低了多 AGV 在行驶过程中发生冲突和碰撞的可能性。但是该算法会随着 AGV 的频繁调度和重规划,以及环境地图的扩大而增加计算复杂度,降低 AGV 路径搜索效率。

基于以上问题,本文以铺设 AGV 导引线的工业生产现场为背景,提出了一种改进的 A* 算法,该算法使用切比雪夫距离对当前节点的启发函数进行加权处理,在相同的地图环境下,相比于 Dijkstra 算法,传统 A* 算法以及已有的使用切比雪夫距离对当前节点和父节点的启发函数进行加权的方法,其搜索节点数和搜索时间得到了有效的降低,显著地提高了 A* 算法在进行重规划时的搜索效率。在改进 A* 算法的基础上,结合时间窗模型,通过对每辆 AGV 的行驶节点进行提前预判,动态调整 AGV 的优先级,有效地解决了多 AGV 路径规划时的冲突问题,在 AGV 遇到其他车辆需要进行避让或重规划时,通过比较避让预计增加的时间和重规划预计增加的预估时间,并选择预计增加时间少的方案解决多 AGV 路径规划时的死锁以及路径冲突问题。采用该算法既可以让 AGV 得到了相对最优路径,也有效地解决了多 AGV 在运行过程中的时间、空间冲突问题。该算法增加的算法复杂度小,通过试验证明,此方法在对多 AGV 进行路径规划时,搜索速度快、实时性较好。

1 改进 A* 算法

A* 算法是从 Dijkstra 算法发展而来的,是一种启发式搜索算法。它利用启发信息来指导路径的搜索,以实

现减少搜索范围和降低搜索问题的复杂度。同时,也是静态路网中搜索最短路径时最有效的直接搜索方法。在 A* 算法中,估计值(即启发函数 $h(n)$ 值)与实际距离值越接近, A* 算法的搜索速度越快,搜索效率也就越高。由于 A* 算法在中小型地图中搜索的高效性,在实际中得到了广泛的应用。但是传统的 A* 算法由于其搜索路径时需要遍历的节点数多,且规划出的路径存在过多的冗余节点和转折点,所以很多学者对传统的 A* 算法进行了改进。传统的 A* 算法为

$$f(n) = g(n) + h(n) \quad (1)$$

式中, n 为当前节点; $f(n)$ 为从起始节点经过当前节点 n 到达目标节点的最小代价估计; $g(n)$ 为起始节点到当前节点的最小代价估计; $h(n)$ 为当前节点 n 到目标节点的最小代价估计,也称为启发函数。

启发函数 $h(n)$ 在 A* 算法中起着至关重要的作用。这里设当前点到目标点的实际代价为 Q 。当 $h(n)=0$ 时, A* 算法将变成 Dijkstra 算法, Dijkstra 算法能够保证找到一条最短路径;当 $h(n)<Q$ 时, $h(n)$ 的值越小, A* 算法需要扩展的节点数越多,算法的运行速度也就变得越慢,但是也能够保证找到一条最短路径,并且运算速度比 Dijkstra 算法快;当 $h(n)=Q$ 时, A* 算法只寻找最佳路径而不扩展任何节点,这种情况下也能够保证找到一条最短路径,并且运算速度非常快;当 $h(n)>Q$ 时, A* 算法在寻找最佳路径的过程中会扩展别的任何一个节点,这种情况下不能够保证找到一条最短路径。根据以上原则,为了能够得到一条从起点到终点的最短路径,要保证 $h(n) \leq Q$, 并且 $h(n)$ 越逼近 Q 值,其搜索速度会越快。这也是对 A* 算法进行改进的重要思路。

A* 算法中 $h(n)$ 常用曼哈顿距离、欧几里德距离、欧几里德距离的平方表示。欧几里德距离的表示方式为

$$h_o = \sqrt{(x_e - x_n)^2 + (y_e - y_n)^2} \quad (2)$$

曼哈顿距离的表示方式为

$$h_m = |x_e - x_n| + |y_e - y_n| \quad (3)$$

切比雪夫距离的表示方式为

$$h_c = \max(|x_e - x_n|, |y_e - y_n|) \quad (4)$$

式(2)~(4)中, (x_e, y_e) 为终点坐标; (x_n, y_n) 为当前点 n 的坐标。

由于传统 A* 算法在搜索路径时需要遍历的节点数和生成路径中的冗余节点较多,且在地图较大时尤为突出,严重影响了 A* 算法的搜索效率。本文遵循启发函数 $h(n)$ 越逼近当前点到目标点的实际代价值 Q , 其搜索速度会越快的原则,对 A* 算法进行了改进。

从式(2)~(4)可得,对于 $\forall n$ 和 $\forall e$ 有

$$\sqrt{(x_e - x_n)^2 + (y_e - y_n)^2} \leq |x_e - x_n| + |y_e - y_n| \quad (5)$$

也即, $h_o \leq h_m$

$$\max(|x_e - x_n|, |y_e - y_n|) \leq |x_e - x_n| + |y_e - y_n| \quad (6)$$

也即, $h_c \leq h_m$

$$h_o = \sqrt{(x_e - x_n)^2 + (y_e - y_n)^2} \geq |x_e - x_n| \quad (7)$$

$$h_o = \sqrt{(x_e - x_n)^2 + (y_e - y_n)^2} \geq |y_e - y_n| \quad (8)$$

$$\begin{cases} h_c = |x_e - x_n|, & |x_e - x_n| \geq |y_e - y_n| \\ h_c = |y_e - y_n|, & |x_e - x_n| < |y_e - y_n| \end{cases} \quad (9)$$

也即, $h_c = |x_e - x_n| \leq h_o$

从式(5)、(6)和(9)可以得到

$$h_c \leq h_o \leq h_m \quad (10)$$

由于本文所涉及的工业环境已经铺设了 AGV 路径导引线, AGV 不能沿着任意角度移动,只能沿着网格的方向移动。欧几里德距离 h_o 表示的是对角线距离,对于本文已铺设引导线的 AGV 不适用。本文使用取值更接近实际代价 Q 的曼哈顿距离 h_m 作为启发函数,并使用切比雪夫距离 h_c 对启发函数 h_m 进行加权,改进后的 A* 算法为

$$f(n) = g(n) + h_c \times h_m \quad (11)$$

改进后的 A* 算法可以减少在路径搜索时需要遍历的节点数,提高搜索效率。

A* 算法需要维护一个 Open 表和一个 Close 表, Open 表中记录了所有已经遍历过但是没有被考察的节点; Close 表中记录了所有已经考察过的节点,初始化时 Open 表中记录了起始节点, Close 表为空。执行 A* 算法时,从起始节点开始,不断地向外扩充,并维护和更新 Open 表和 Close 表,直到终点加入 Close 表为止。改进后的 A* 算法流程如图 1 所示。

本文以实际的工业生产制造车间为背景,利用栅格法和拓扑图法对工业生产制造车间的平面图进行电子地图的构建,图 2 为实际工业生产制造车间平面图,圆形数字标识的点为 AGV 的运输站点,其他数字标识的点为 AGV 路径上的节点。图 3 为与图 2 相对应的使用栅格法构建的电子地图。

图 3 中的栅格地图大小为 20×42 ,横坐标范围 $[0, 41]$ 、纵坐标范围 $[0, 19]$ 。栅格地图中的每个数字点代表着生产车间的一个站点,其中站点 1 和 17 是 AGV 的车库和充电桩所在位置。由于使用栅格地图规划出的 AGV 路径存在冗余的节点,因此本文对使用改进 A* 算法在栅格地图下得到的路径做进一步处理,删除生成路径上不必要的冗余节点,使最终规划出的路径需要保留的节点数更少。

为了验证本文提出的改进 A* 算法的有效性,分别对 Dijkstra 算法、传统 A* 算法、Tang 等^[16]提出的改进

的 A* 算法和本文提出的改进的 A* 算法进行仿真试验对比。以图 3 的栅格地图为试验地图,每次试验分别以栅格地图中的每个站点为起点或终点,经排列组合每次共验证 378 条路径。共做 10 次试验,最后取 10 次的平均值。试验结果及其对比如表 1 和表 2 所示。

从试验结果可以得出,与 Dijkstra 算法、传统 A* 算法和 Tang 等^[16]提出的改进的 A* 算法相比,本文提出的改进 A* 算法的平均搜索时间和平均遍历节点数得到了显著的减少,搜索效率得到了显著提高。与 Dijkstra 算法相比,平均搜索时间和平均遍历节点数分别减少了 74.95% 和 67.87%; 与传统 A* 算法相比,平均搜索时间和平均遍历节点数分别减少了 59.91% 和 20.81%; 与 Tang 等^[16]提出的改进的 A* 算法相比,平均搜索时间和平均遍历节点数分别减少了 17.57% 和 8.00%。使用改进的 A* 算法规划从节点 1 (坐标(2,1)) 到节点 14 (坐标(22,11)) 的路径如图 4 所示。

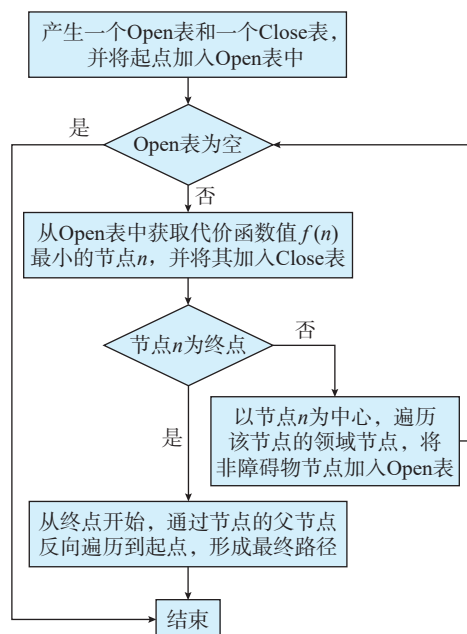


图 1 改进 A* 算法流程图

Fig.1 Flow chart of improved A* algorithm

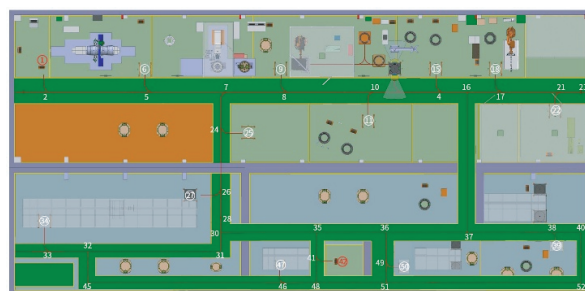


图 2 工业生产车间平面图

Fig.2 Floor plan of industrial production workshop

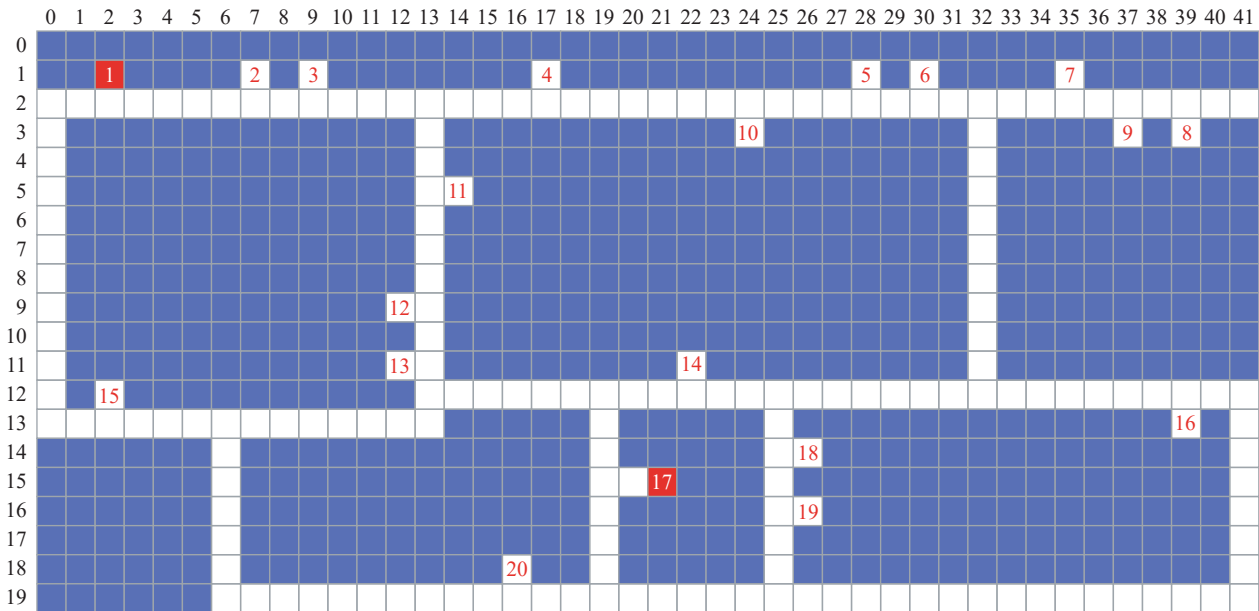


图3 工业生产车间栅格地图

Fig.3 Raster map of industrial production workshop

表1 4种路径规划算法遍历节点数和平均搜索时间对比

Table 1 Comparison of the number of traversed nodes and the average search time of four path planning algorithms

	平均搜索时间 /s				遍历节点数			
	Dijkstra	传统 A*	改进 A*[16]	本文改进 A*	Dijkstra	传统 A*	改进 A*[16]	本文改进 A*
	29.78	18.61	9.05	7.46	242.81	98.52	84.80	78.02

表2 本文改进 A* 算法相较于其他 3 种算法遍历节点数和平均搜索时间减少百分比

Table 2 Compared with the other three algorithms, the improved A* algorithm in this paper reduces the number of nodes traversed and the average search time by a percentage %

相比平均减少搜索时间 /%			相比平均减少遍历节点数		
Dijkstra	传统 A*	改进 A*[16]	Dijkstra	传统 A*	改进 A*[16]
74.95	59.91	17.57	67.87	20.81	8.00

图4中x轴代表栅格点的横坐标; y轴代表栅格点的纵坐标,其中曲线上的每个点代表规划出的路径需要经过的每个节点。从图4中可以看出,使用改进的A*算法规划出的路径需要保存的节点为32个,本文在规划出的原始路径基础上做了进一步处理,删除路径中冗余的节点后,路径需要保留的节点数仅剩5个,路径也变得更加光滑。

2 多AGV无碰撞规划

工业生产车间常存在多个站点以及多个生产任务,需要多辆AGV同时进行无碰撞协调运行,所以需要

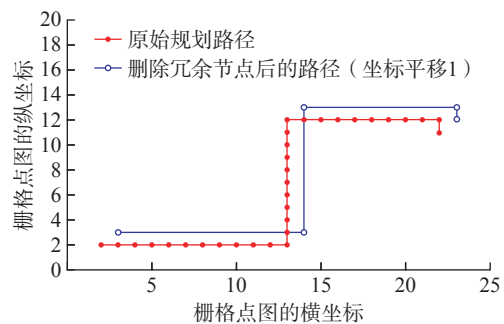


图4 使用改进 A* 算法规划的从节点 1 到节点 14 的路径

Fig.4 Using the improved A* algorithm to plan the path from node 1 to node 14

多辆AGV的行驶路径进行合理规划,在保证不发生碰撞的基础上,使每辆AGV的行驶路径最优。由于多辆AGV在行驶的过程中互为障碍物,所以需要考虑多辆AGV在行驶过程中的动态规划问题,保证每段路上在同一时间只能有1辆AGV通过。只有合理的规划和协调好AGV行驶路径的空间和时间资源利用问题,才能保证多辆AGV有条不紊地运行。

2.1 多AGV冲突类型

当多辆AGV同时运行时,AGV之间主要存在路径不冲突、路径冲突但时间不冲突、相向冲突、节点冲突4种情况。

(1) 路径不冲突。当多辆AGV之间规划的路径不存在重合节点和交叉节点,也即路径不存在冲突时,每辆AGV可各自独立运行,不需要考虑碰撞问题。

(2) 路径冲突时间不冲突。当多辆AGV之间存在路径冲突,但是时间不冲突时,因为在同一节点或同一

路段上行驶的时间没有交叉和重叠,每辆 AGV 可独自运行,不需要考虑碰撞问题

(3) 相向冲突。如图 5 所示,当两辆 AGV 同一时间在同一条路段上相向行驶时,会产生相向冲突。此时需要采用相应的碰撞预测方法和策略,解决多辆 AGV 同一时间在同一路段上的相向冲突问题。

(4) 节点冲突。如图 6 所示,当多辆 AGV 同一时间在交叉路口相遇时,会产生节点冲突问题,此时也需要采用相应的碰撞预测方法和策略解决该问题。

2.2 基于时间窗的冲突解决方案

在 AGV 路径规划中,时间窗是指 AGV 从进入某路段到离开该路段所占用的时间段。可以将某路段的时间窗划分为空闲时间段和占用时间段。空闲时间段为该路段空闲的时间段,在空闲时间段内任何一辆 AGV 都可在该路段上行驶;占用时间段为当前 AGV 在该路段行驶占用的时间段,在占用时间段内其他车辆不能驶入该路段。时间窗模型的示意图如图 7 所示。

这里设 AGV 路径上的每个节点都用相应的坐标 (x,y) 表示, $(x_1,y_1) - (x_2,y_2)$ 表示路径上 2 个节点 (x_1,y_1) 和 (x_2,y_2) 之间的路段。图 8 为两辆 AGV 在路段 $(2,9) - (2,13)$ 上发生相向冲突的情况。图 9 为两辆 AGV 在节点 $(2,13)$ 发生节点冲突的情况。

所有 AGV 在路段 k 上的时间窗模型可以表示为

$$T(k) = \{t_{1k}, t_{2k} \dots t_{nk}\}, i \in [1, n] \quad (12)$$

式中, $T(k)$ 为所有 AGV 在路段 k 上的时间窗集合; i 表示第 i 辆 AGV; n 表示 AGV 的数量。

根据式(12)可以得到所有 AGV 在所有路段上的时间窗集合,为

$$T = \begin{pmatrix} T_1 \\ T_2 \\ \vdots \\ T_k \\ \vdots \\ T_m \end{pmatrix} = \begin{bmatrix} t_{11} & t_{21} & \dots & t_{i1} & \dots & t_{n1} \\ t_{12} & t_{22} & \dots & t_{i2} & \dots & t_{n2} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ t_{13} & t_{23} & \dots & t_{i3} & \dots & t_{n3} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \dots \\ t_{1m} & t_{2m} & \dots & t_{im} & \dots & t_{nm} \end{bmatrix} \quad (13)$$

式中, T_m 表示所有 AGV 在路段 m 上的时间窗集合; m 表示 AGV 规划出的路径的路段数; n 表示 AGV 的数量。式(13)中的每一列代表每一辆 AGV 在所有路段上的时间窗集合。

多辆 AGV 在路段 k 上发生相向冲突的时间窗模型为

$$t'_{ik} \leq t_{qk} \leq t''_{ik}, i \in [1, n], q \in [1, n], q \neq i \quad (14)$$

式中, t'_{ik} 为第 i 辆 AGV 进入路段 k 时的时间; t''_{ik} 为第 i 辆 AGV 离开路段 k 时的时间; t_{qk} 为第 q 辆 AGV 在路段 k 上的时间窗。即当两辆 AGV 在同一路段 k 上的时间窗发生重叠时, AGV 将发生相向冲突。

当多辆 AGV 同时在一节点的相邻路段上行驶,并且同时到达该节点时,将发生节点冲突,节点冲突的时间窗模型为

$$|t''_{ik} - t''_{qp}| < \delta, \delta \propto 0, i \in [1, n], q \in [1, n], q \neq i \quad (15)$$

式中, t''_{ik} 为第 i 辆 AGV 离开路段 k 时的时间; t''_{qp} 为第 q 辆 AGV 离开路段 p 时的时间。路段 k 和路段 q 有相同的节点,并且 $k, q \in [1, m]$ 。即多辆 AGV 同时离开不同的路段,到达同一个节点时将发生节点冲突。

从图 8 和 9 的相向冲突及节点冲突的时间窗中可



图 5 AGV 路径发生相向冲突

Fig.5 AGV conflicts in same direction

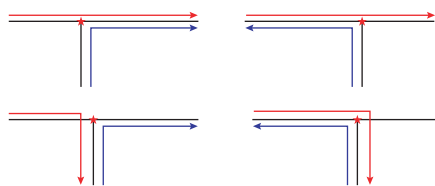


图 6 AGV 路径发生节点冲突

Fig.6 AGV conflicts in same node

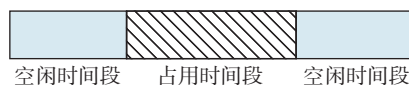


图 7 时间窗模型

Fig.7 Time window model

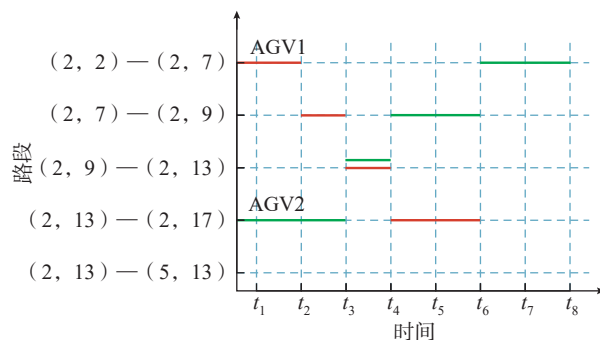


图 8 两辆 AGV 发生相向冲突时间窗

Fig.8 Two AGVs conflict time window in the same direction

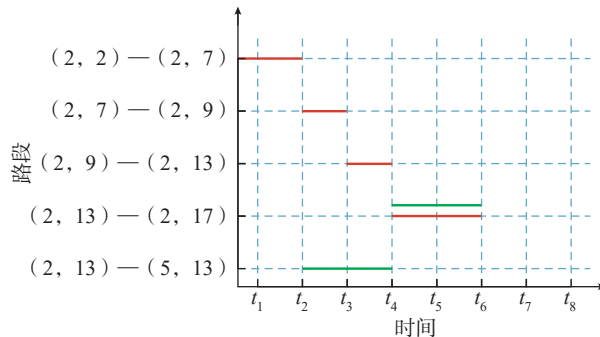


图 9 两辆 AGV 发生节点冲突时间窗

Fig.9 Two AGVs conflict time window in the same node

以看出,将时间窗右移是解决多 AGV 路径冲突的主要方法。本文在每辆 AGV 行驶过程中,提前对 AGV 路径上的 3 个节点进行时间窗的更新和排布,当检测到时间窗有冲突时,根据生产任务需求以及当前 AGV 离终点的远近程度动态调整相应 AGV 的优先级,让优先级的 AGV 进行路径重规划或避让,即让其在冲突路段上的时间窗右移,从而避免 AGV 产生路径上的冲突与碰撞。在重规划或避让策略的选择上,可以通过式 (16) 进行选择。

$$\begin{cases} \text{采用重规划策略, } \Delta t_r \leq \Delta t_e \\ \text{采用避让策略, } \Delta t_r \geq \Delta t_e \end{cases} \quad (16)$$

式中, Δt_r 表示 AGV 进行重规划预计需要增加的行驶时间; Δt_e 表示 AGV 进行避让等待预计需要增加的行驶时间。

多 AGV 无碰撞规划流程如图 10 所示。

图 10 中节点标志 g 表示该节点既没有 AGV 占用也不是 AGV 规划路径中的节点。节点标志 y 表示该节点是 AGV 路径规划中需要经过的节点,但是现在 AGV 还没有经过该节点。节点标志 r 表示该节点是 AGV 路径规划中需要经过的节点,并且当前 AGV 离该节点的距离在 3 个节点以内。

3 试验结果与分析

为了验证本文提出的改进的 A^* 算法以及多 AGV 无碰撞规划策略的有效性,使用 Visual Studio 2017+Qt5 进行编程,将改进的算法及策略应用到实际工业生产 AGV 上。AGV 长 2 m、宽 90 cm、高 60 cm、载重 1500 kg,其中 AGV 自带车载工控机用于 AGV 的控制,现场的 AGV 通道为单行道,即同一时间只允许有 1 辆 AGV 通过。出于工业现场安全考虑,AGV 在运输过程以 0.5 m/s 的速度匀速行驶。AGV 原型图如图 11 所示。

在图 12 中,AGV1 从 1 号站点到 17 号站点;AGV2 从 8 号站点到 19 号站点。图 13 为相对应的时间窗模型。使用改进的 A^* 算法规划出的路径不存在重合和交叉的节点,所以两辆 AGV 无需考虑路径冲突问题,在图 13 中对应的时间窗上也不存在重叠情况。

在图 14 中,AGV1 从 t_1 时刻开始,从 17 号站点到 10 号站点;AGV2 从 t_4 时刻开始,从 8 号站点到 20 号站点。图 15 为相对应的时间窗模型。使用本文提出的改进的 A^* 算法规划出的路径存在重合和交叉的节点,从图 15 时间窗模型中可以得出,AGV1 和 AGV2 在路段 $(2,39) - (2,32)$ 和 $(12,32) - (2,32)$ 交叉位置将发生节点冲突,所以根据 AGV 的优先级情况,AGV2 将会在 $(2,37)$ 节点选择等待避让,待 AGV1 经过 $(2,32)$ 节点后,AGV2 再按照原规划路径行驶到

20 号站点。

在图 16 中,AGV1 从 t_1 时刻开始,从 1 号站点到 16 号站点;AGV2 从 t_1 时刻开始,从 9 号站点到 2 号站点。图 17 为相对应的时间窗模型。使用本文提出的改进的 A^* 算法规划出的路径存在重合和交叉的节点,规划出的 AGV1 原始路径为 $(1,2) \rightarrow (2,2) \rightarrow (2,32) \rightarrow (12,32) \rightarrow (12,39) \rightarrow (13,39)$ 。从图 17 时间窗模型中可

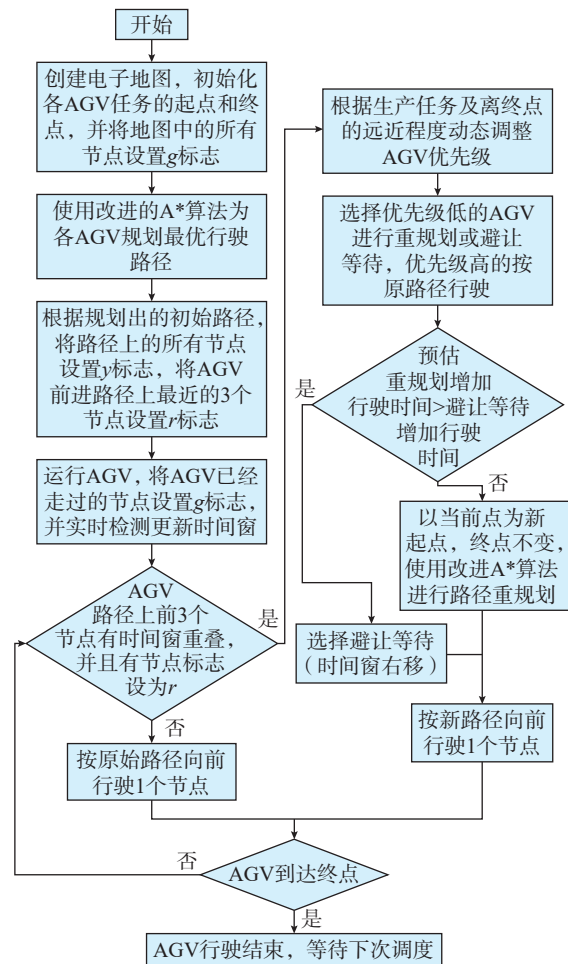


图 10 多 AGV 无碰撞规划流程图
Fig.10 Flow chart of multi-AGV collision-free planning

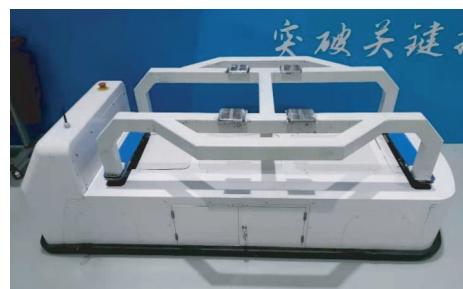


图 11 AGV 原型
Fig.11 AGV prototype

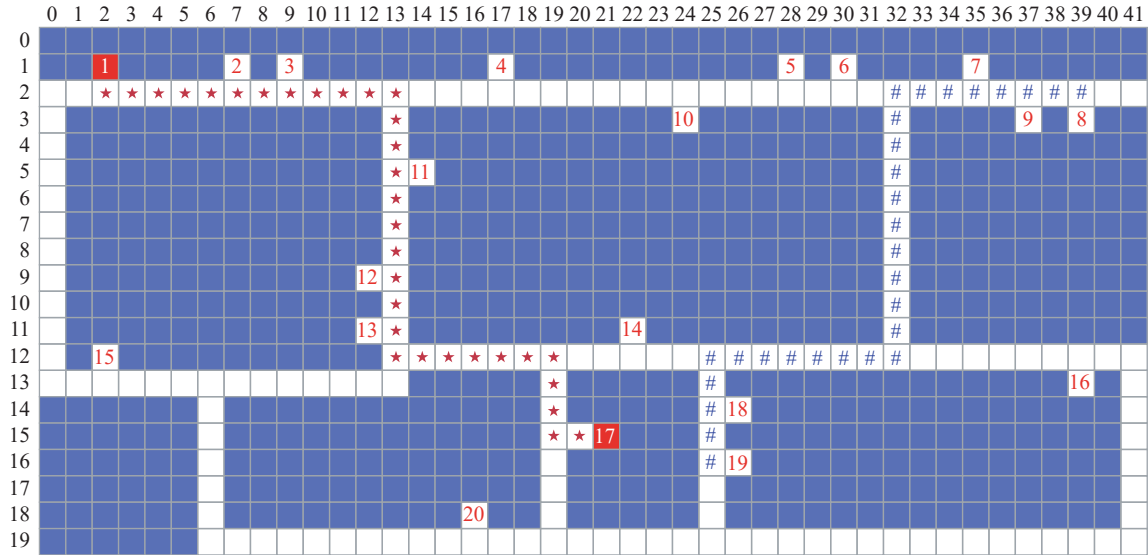


图 12 两辆 AGV 路径无冲突
Fig.12 No conflict between two AGVs

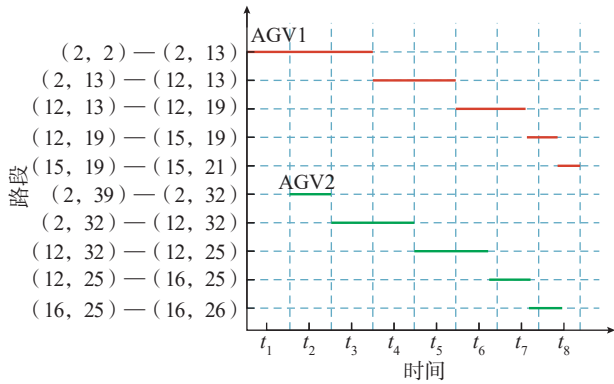


图 13 两辆 AGV 路径无冲突时间窗
Fig.13 Two AGV conflict-free time window

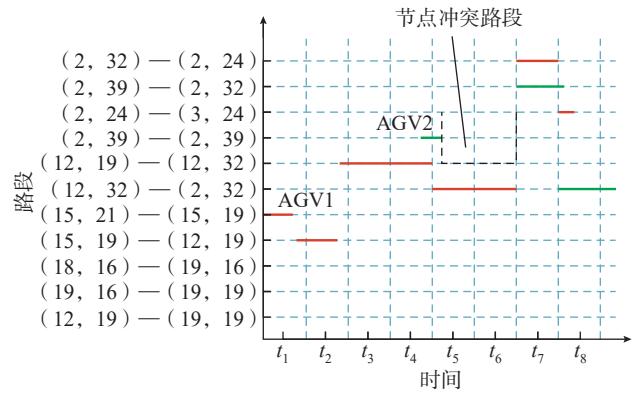


图 15 两辆 AGV 在节点(2,32)发生节点冲突时间窗(避让等待)
Fig.15 Time window when two AGVs have node conflict at coordinates (2,32) (Waiting)

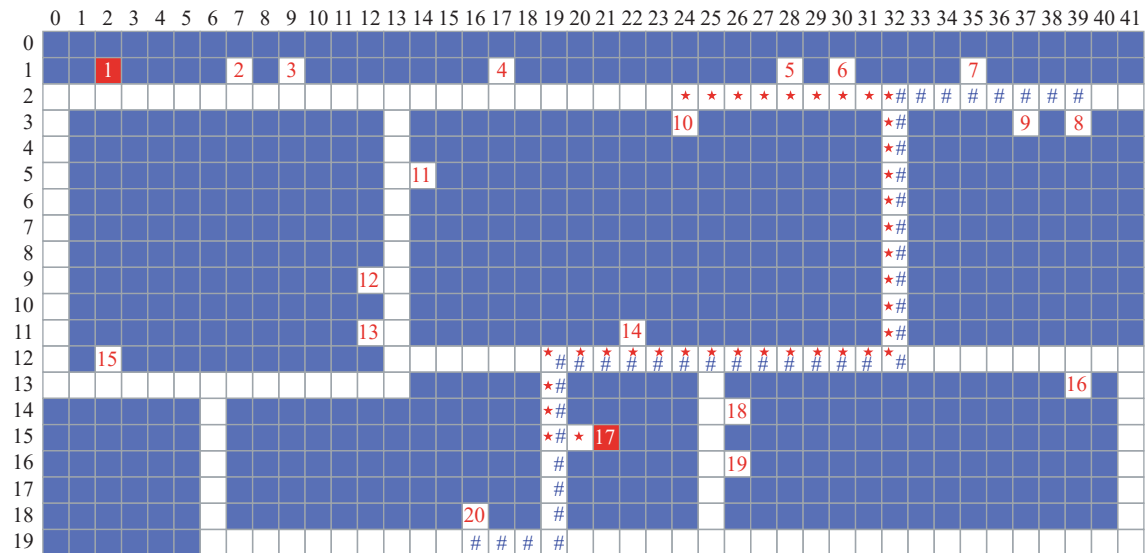


图 14 两辆 AGV 在节点(2,32)发生节点冲突
Fig.14 Two AGVs have a node conflict at coordinates (2,32)

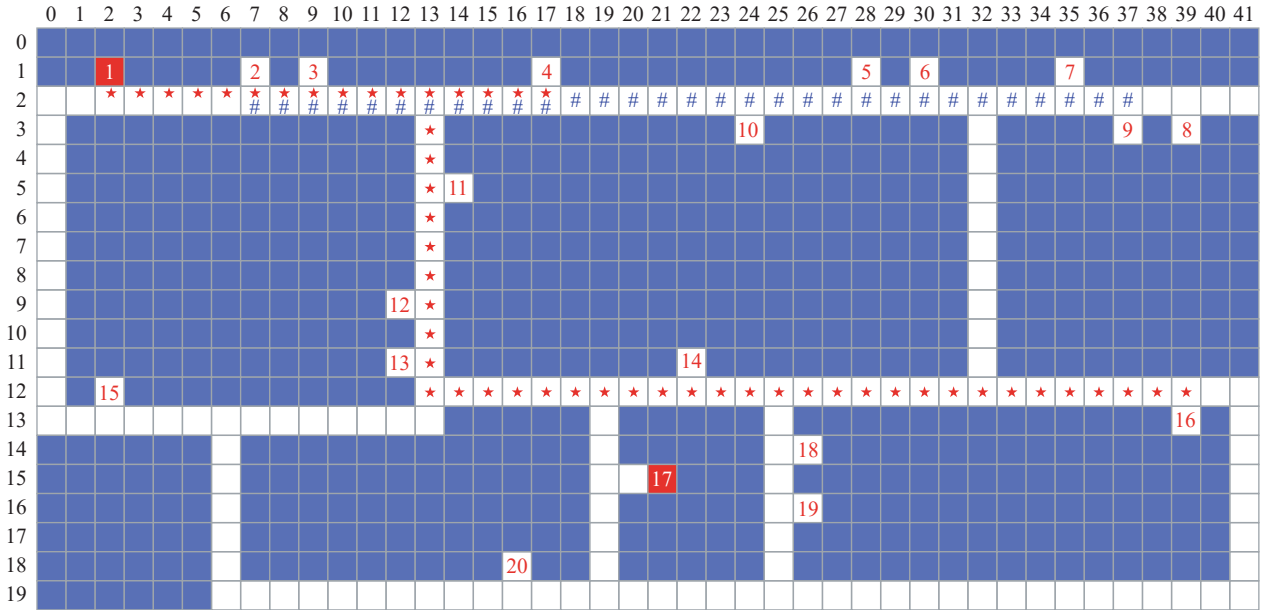


图 16 两辆 AGV 在节点(2,17)发生相向路径冲突
Fig.16 Two AGVs have a same direction conflict at coordinates (2,17)

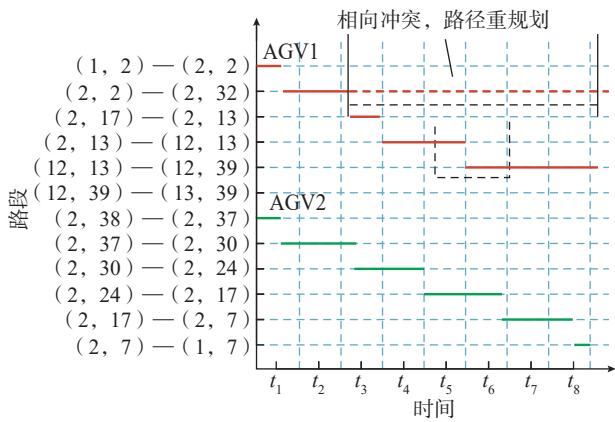


图 17 两辆 AGV 在节点(2,17)发生相向冲突时间窗(重规划)
Fig.17 Time window when two AGVs have a same direction conflict at coordinates (2,17)

以得出, AGV1 在节点(2,17)位置检测到前向路径上的节点被占用,根据 AGV 的无碰撞规划策略,AGV1 选择以节点(2,17)为新节点,终点不变,重新进行路径规划,AGV1 重规划后的路径为:(2,17)→(2,13)→(12,13)→(12,39)→(13,39)。AGV2 的路径保持不变,从而避免两辆 AGV 发生路径冲突碰撞。

本文以 Visual Studio 2017+Qt5 为开发环境,在图 11 所示的 AGV 上进行了试验验证。表 3 和 4 分别对应图 12、14 和 16 中,两辆 AGV 没有路径冲突、发生相向冲突(避让等待)和发生相向冲突(重规划)时进行路径规划消耗的时间,并在 4 种路径规划算法上进行了验证和对比分析。

从以上试验可以看出,本文提出的改进 A* 算法与多 AGV 路径规划策略结合,在多 AGV 无路径冲突,以

表 3 AGV1 路径规划消耗时间(不包括 AGV 运行时间)
Table 3 AGV1 path planning consumption time (not including AGV driving time)

AGV 行驶情况	Dijkstra	传统 A*	改进 A*[16]	本文改进 A*
两辆 AGV 没有路径冲突	2.52	2.12	1.28	0.99
两辆 AGV 发生相向冲突(避让等待)	3.74	3.24	2.56	2.32
两辆 AGV 发生相向冲突(重规划)	2.43	1.93	1.22	0.81

表 4 AGV2 路径规划消耗时间(不包括 AGV 运行时间)
Table 4 AGV2 path planning consumption time (not including AGV driving time)

AGV 行驶情况	Dijkstra	传统 A*	改进 A*[16]	本文改进 A*
两辆 AGV 没有路径冲突	1.91	1.62	1.20	0.87
两辆 AGV 发生相向冲突(避让等待)	1.89	1.51	1.24	0.94
两辆 AGV 发生相向冲突(重规划)	4.46	4.08	3.65	3.26

及存在节点或路径相向冲突时,能够得到较好的无碰撞规划结果,能够有效地解决实际工业生产过程中多AGV同时行驶时的路径冲突问题。当某辆AGV出现故障,长时间占用某条路径时,调度系统会检测到故障并产生报警,通知工作人员进行处理,同时调度系统会将电子地图中的该栅格点置为障碍物点,其他AGV进行路径规划时将不会再考虑该条路径。从表3和4中可以看出,相较于Dijkstra算法、传统A*算法和Tang等^[16]提出的改进A*算法,使用本文改进的A*算法,无论是在两辆AGV没有路径冲突或者发生路径冲突时,都能够很快地对每辆AGV进行路径规划,能够较好地满足实时性要求。

4 结论

本文以实际工业生产现场为背景,研究了多AGV路径规划算法和无碰撞策略。通过使用切比雪夫距离对启发函数进行加权,对传统的A*算法进行了改进,同时,通过时间窗模型和动态调整AGV的优先级,解决了多AGV的路径冲突问题,具有较好应用价值。试验表明,本文提出的改进的A*算法显著地降低了AGV在进行路径规划时的路径搜索时间,提高了搜索效率,实时性也得到了进一步的提高。

参考文献

- [1] FU B, CHEN L, ZHOU Y T, et al. An improved A* algorithm for the industrial robot path planning with high success rate and short length[J]. *Robotics and Autonomous Systems*, 2018, 106: 26–37.
- [2] ZHANG Y, WANG F L, FU F K, et al. Multi-AGV path planning for indoor factory by using prioritized planning and improved ant algorithm[J]. *Journal of Engineering and Technological Sciences*, 2018, 50(4): 534–547.
- [3] ZHONG M S, YANG Y S, DESSOUKY Y, et al. Multi-AGV scheduling for conflict-free path planning in automated container terminals[J]. *Computers & Industrial Engineering*, 2020, 142: 106371.
- [4] ZHOU L J, CUI W, FU H Y. AGV path planning combining A* and ant colony algorithm[J]. *Journal of Physics: Conference Series*, 2021, 1948(1): 012006.
- [5] ZHENG Y C, WANG J, GUO D, et al. Study of multi-objective path planning method for vehicles[J]. *Environmental Science and Pollution Research*, 2020, 27(3): 3257–3270.
- [6] LUO M, HOU X R, YANG J. Surface optimal path planning using an extended dijkstra algorithm[J]. *IEEE Access*, 2020, 8: 147827–147838.
- [7] SUN Y H, FANG M, SU Y X. AGV path planning based on improved dijkstra algorithm[J]. *Journal of Physics: Conference Series*, 2021, 1746: 12052.
- [8] YUAN Z H, YANG Z M, LV L L, et al. A Bi-level path planning algorithm for multi-AGV routing problem[J]. *Electronics*, 2020, 9(9): 1351.
- [9] CHEN Z Y, XU B. AGV path planning based on improved artificial potential field method[C]//2021 IEEE International Conference. Shenyang: IEEE, 2021.
- [10] PRADHAN B, NANDI A, HUI N B, et al. A novel hybrid neural network-based multirobot path planning with motion coordination[J]. *IEEE Transactions on Vehicular Technology*, 2020, 69(2): 1319–1327.
- [11] DU L Z, KE S F, WANG Z, et al. Research on multi-load AGV path planning of weaving workshop based on time priority[J]. *Mathematical Biosciences and Engineering: MBE*, 2019, 16(4): 2277–2292.
- [12] 霍凤财, 迟金, 黄梓健, 等. 移动机器人路径规划算法综述[J]. *吉林大学学报(信息科学版)*, 2018, 36(6): 639–647.
- [13] HUO Fengcai, CHI Jin, HUANG Zijian, et al. Review of path planning for mobile robots[J]. *Journal of Jilin University (Information Science Edition)*, 2018, 36(6): 639–647.
- [14] 宋晓茹, 任怡悦, 高嵩, 等. 移动机器人路径规划综述[J]. *计算机测量与控制*, 2019, 27(4): 1–5, 17.
- [15] SONG Xiaoru, REN Yiyue, GAO Song, et al. Survey on technology of mobile robot path planning[J]. *Computer Measurement & Control*, 2019, 27(4): 1–5, 17.
- [16] DE RYCK M, VERSTHEYHE M, DEBROUWERE F. Automated guided vehicle systems, state-of-the-art control algorithms and techniques[J]. *Journal of Manufacturing Systems*, 2020, 54: 152–173.
- [17] CHEN Z Y, FENG Z X, FAN Z Q. Improved AGV path planning algorithm based on grid map model[C]//2021 IEEE 5th Advanced Information Technology, Electronic and Automation Control Conference. Chongqing: IEEE, 2021.
- [18] TANG G, TANG C Q, CLARAMUNT C, et al. Geometric A-star algorithm: An improved a-star algorithm for AGV path planning in a port environment[J]. *IEEE Access*, 2021, 9: 59196–59210.
- [19] SONG R, LIU Y C, BUCKNALL R. Smoothed A* algorithm for practical unmanned surface vehicle path planning[J]. *Applied Ocean Research*, 2019, 83: 9–20.
- [20] 余星宝, 杨慧斌, 周玉凤, 等. 改进A*的4阶贝塞尔曲线路径规划[J]. *轻工机械*, 2020, 38(6): 64–67.
- [21] YU Xingbao, YANG Huibin, ZHOU Yufeng, et al. Improved path planning of A* quartic Bezier curve[J]. *Light Industry Machinery*, 2020, 38(6): 64–67.
- [22] 胡蔚旻, 靳文舟. 改进平滑A*算法的多AGV路径规划[J]. *计算机工程与应用*, 2020, 56(16): 204–210.
- [23] HU Weimin, JIN Wenzhou. Multi-AGV path planning based on improved smooth A* algorithm[J]. *Computer Engineering and Applications*, 2020, 56(16): 204–210.
- [24] 刘生伟, 马钺, 孟树峰, 等. 改进A*算法的AGV路径规划[J]. *计算机应用*, 2019, 39(Z2): 41–44.
- [25] LIU Shengwei, MA Yue, MENG Shufeng, et al. Improved A* algorithm for path planning of AGV[J]. *Journal of Computer Applications*, 2019, 39(Z2): 41–44.
- [26] 余文凯, 章政, 付雪画, 等. 基于地图预处理及改进A*算法的路径规划[J]. *高技术通讯*, 2020, 30(4): 383–390.
- [27] YU Wenkai, ZHANG Zheng, FU Xuehua, et al. Path planning based on map partition preprocessing and improved A* algorithm[J]. *Chinese High Technology Letters*, 2020, 30(4): 383–390.

(下转第90页)

逐渐粗化,940 ℃和 980 ℃成形后的平均晶粒尺寸分别约为 9 μm 和 11 μm。

3 结论

(1) 在 920~960 ℃、应变速率 $1 \times 10^{-3} \sim 5 \times 10^{-5} \text{ s}^{-1}$ 条件下,TA32 合金板材具有良好的超塑性,采用单向超塑拉伸时,温度 920~940 ℃和高应变速率 $1 \times 10^{-3} \text{ s}^{-1}$ 时为最佳变形条件。

(2) 随着变形温度的增加,锥形件胀破压力逐渐减小,在 960 ℃和 980 ℃时最小,为 0.6 MPa;在 940 ℃和 960 ℃时,锥形件的高度较高,分别为 90 mm 和 92 mm;经过不同变形量的变形后的微观组织变化并不显著,该结果为结构件的工艺设计提供参考。

参考文献

[1] 李志强,郭和平.超塑成形/扩散连接技术的应用进展和发展趋势[J].航空制造技术,2010,53(8):32-35.
LI Zhiqiang, GUO Heping. Application progress and development tendency of superplastic forming/diffusion bonding technology[J]. Aeronautical Manufacturing Technology, 2010, 53(8): 32-35.

[2] HEFTI L D. Innovations in the superplastic forming and diffusion bonded process[J]. Journal of Materials Engineering and Performance, 2008, 17(2): 178-182.

[3] 王清江,刘建荣,杨锐.高温钛合金的现状与前景[J].航空材料学报,2014,34(4):1-26.
WANG Qingjiang, LIU Jianrong, YANG Rui. High temperature titanium alloys: Status and perspective[J]. Journal of Aeronautical Materials, 2014, 34(4): 1-26.

[4] FAN R L, WU Y, CHEN M H, et al. Relationship among microstructure, mechanical properties and texture of TA32 titanium alloy sheets during hot tensile deformation[J]. Transactions of Nonferrous Metals Society of China, 2020, 30(4): 928-943.

[5] 吴荣华,陈明和,谢兰生,等. TA32 钛合金高温应力松弛行为及其对应的本构方程[J].稀有金属材料与工程,2019,48(10):3142-3148.
WU Ronghua, CHEN Minghe, XIE Lansheng, et al. Stress relaxation behavior and corresponding constitutive relation of TA32 titanium alloy at high temperature[J]. Rare Metal Materials and Engineering, 2019, 48(10): 3142-3148.

[6] 陈灿,陈明和,谢兰生,等. TA32 新型钛合金高温流变行为及本构模型研究[J].稀有金属材料与工程,2019,48(3):827-834.
CHEN Can, CHEN Minghe, XIE Lansheng, et al. Flow behavior of TA32 titanium alloy at high temperature and its constitutive model[J]. Rare Metal Materials and Engineering, 2019, 48(3): 827-834.

[7] 龚宗辉,谢兰生,陈明和,等.新型 TA32 钛合金板的高温拉伸变形行为[J].机械工程材料,2019,43(1):69-74.
GONG Zonghui, XIE Lansheng, CHEN Minghe, et al. High temperature tensile deformation behavior of new titanium alloy sheet TA32[J]. Materials for Mechanical Engineering, 2019, 43(1): 69-74.

[8] 刘章光,李培杰,尹西岳,等.变形参数对 TA32 合金的超塑性变形行为及微观组织演化的影响[J].稀有金属材料与工程,

2018, 47(11): 3473-3481.

LIU Zhangguang, LI Peijie, YIN Xiyue, et al. Effects of deformation parameters on the superplastic behavior and microstructure evolution of TA32 alloy[J]. Rare Metal Materials and Engineering, 2018, 47(11): 3473-3481.

[9] 程超,陈志勇,刘建荣,等. TA32 钛合金板材的超塑性性能研究[J].钛工业进展,2020,37(2):21-25.

CHENG Chao, CHEN Zhiyong, LIU Jianrong, et al. Research on superplastic behavior of TA32 titanium alloy plate[J]. Titanium Industry Progress, 2020, 37(2): 21-25.

通讯作者:周丽娜,工程师,硕士,研究方向为金属成形技术。

(责编 古京)

(上接第 85 页)

[22] 卞永明,马道阳,高飞,等.基于改进 A-Star 算法的 AGV 全局路径规划[J].机电一体化,2019,25(6):9-15.

BIAN Yongming, MA Xiaoyang, GAO Fei, et al. AGV global path planning based on the improved A-Star algorithm[J]. Mechatronics, 2019, 25(6): 9-15.

[23] 吴飞龙,郭世永.基于改进 A* 算法的 AGV 路径规划[J].黑龙江工业学院学报(综合版),2020,20(9):90-93.

WU Feilong, GUO Shiyong. AGV's path planning based on improved A* algorithms[J]. Journal of Heilongjiang University of Technology (Comprehensive Edition), 2020, 20(9): 90-93.

[24] 陆佳依,金晓怡,朱天宝,等.基于改进 A* 算法的 AGV 路径研究[J].机械设计与研究,2020,36(1):49-52.

LU Jiayi, JIN Xiaoyi, ZHU Tianbao, et al. Research on AGV path based on improved A* algorithm[J]. Machine Design & Research, 2020, 36(1): 49-52.

[25] 邢普学,李强,魏巍,等.改进 A* 算法的 AGV 路径规划在智慧仓储中的应用[J].信息技术,2019,43(5):130-133.

XING Puxue, LI Qiang, WEI Wei, et al. Application of AGV path planning based on improved A* algorithm in intelligent warehousing[J]. Information Technology, 2019, 43(5): 130-133.

[26] 泰应鹏,邢科新,林叶贵,等.多 AGV 路径规划方法研究[J].计算机科学,2017,44(S2):84-87.

TAI Yingpeng, XING Kexin, LIN Yegui, et al. Research of path planning in multi-AGV system[J]. Computer Science, 2017, 44(S2): 84-87.

[27] 廉胤东,谢巍.基于视觉引导多 AGV 系统的改进 A* 路径规划算法研究[J].控制与决策,2020,36(8):1881-1890.

LIAN Yindong, XIE Wei. Research on improved A* path planning algorithm based on vision-guided multi-AGV system[J]. Control and Decision, 2020, 36(8): 1881-1890.

[28] 高新浩,陈晓华,王占山,等.多 AGV 动态交通调度算法设计[J].山东工业技术,2019(9):149-151.

GAO Xinhao, CHEN Xiaohua, WANG Zhanshan, et al. Design of multi-AGV dynamic traffic dispatching algorithm[J]. Shandong Industrial Technology, 2019(9): 149-151.

通讯作者:陈娟,教授,博士生导师,博士,研究方向为检测与智能检测、人工智能及其应用。

(责编 晓月)