

8-15-2024

Research on Path Optimization Algorithm in Dynamic Routing Environment

Xin Xie

College of Electronic Information and Automation, Civil Aviation University of China, Tianjin 300300, China

Xiaobing Hu

College of Safety Science & Engineering, Civil Aviation University of China, Tianjin 300300, China

Hang Zhou

Sino-European Institute of Aviation Engineering, Civil Aviation University of China, Tianjin 300300, China

Follow this and additional works at: <https://dc-china-simulation.researchcommons.org/journal>



Part of the [Artificial Intelligence and Robotics Commons](#), [Computer Engineering Commons](#), [Numerical Analysis and Scientific Computing Commons](#), [Operations Research, Systems Engineering and Industrial Engineering Commons](#), and the [Systems Science Commons](#)

This Paper is brought to you for free and open access by Journal of System Simulation. It has been accepted for inclusion in Journal of System Simulation by an authorized editor of Journal of System Simulation. For more information, please contact xfzxb@126.com.

Research on Path Optimization Algorithm in Dynamic Routing Environment

Abstract

Abstract: In a real dynamic routing environment, static path optimization (SPO) and traditional dynamic path optimization (DPO) tend to encounter issues such as detours, reversals and high computational complexity due to frequent real-time optimization calculation. To address these problems, a novel restart co-evolutionary path optimization (RCEPO) method based on the ripple-spreading algorithm (RSA) is proposed. This method integrates the path optimization process with the dynamic changes of the routing network environment to enhance the effectiveness of path optimization. Moreover, the path reoptimization calculation is performed only when the dynamic changes in the routing environment exceed the predicted range, thereby reducing computational complexity. Experimental results demonstrate that the actual travel path length and the actual travel time of this method are shortened by 17% and 12%, respectively, compared with the traditional DPO method under the dynamic routing network environment. It can effectively solve the path optimization problem under the real dynamic routing network environment. The feasibility and effectiveness of this approach are validated through experiments conducted with a robot dog.

Keywords

path optimization, co-evolutionary, dynamic environment, ripple-spreading algorithm, uncertainty

Recommended Citation

Xie Xin, Hu Xiaobing, Zhou Hang. Research on Path Optimization Algorithm in Dynamic Routing Environment[J]. Journal of System Simulation, 2024, 36(8): 1969-1981.

动态路网环境下的路径优化算法研究

解鑫¹, 胡小兵^{2*}, 周航³

(1. 中国民航大学 电子信息与自动化学院, 天津 300300; 2. 中国民航大学 安全科学与工程学院, 天津 300300;
3. 中国民航大学 中欧航空工程师学院, 天津 300300)

摘要: 为解决真实动态路网环境下, 静态路径优化(static path optimization, SPO)方法和传统动态路径优化(dynamic path optimization, DPO)方法由于频繁实时优化计算, 规划路径过程中容易出现绕路、折返、计算复杂度高等问题, 提出基于涟漪扩散算法(ripple-spreading algorithm, RSA)的重启协同进化路径优化(restart co-evolutionary path optimization, RCEPO)方法。将路径优化过程与路网环境的动态变化过程相结合, 提升了路径优化效果。仅当路网环境的动态变化超出预测范围时才进行路径的重新优化计算, 降低了计算复杂度。实验结果表明: 在动态路网环境下, 该方法的实际行进轨迹长度和行进时间相较于传统 DPO 方法分别缩短了 17% 和 12%。有效解决了真实动态路网环境下路径优化问题。并且通过机器狗实验, 验证了该方法的实用性和有效性。

关键词: 路径优化; 动态环境; 协同进化; 涟漪扩散算法; 不确定性

中图分类号: TP391.1; TP301.6 文献标志码: A 文章编号: 1004-731X(2024)08-1969-13

DOI: 10.16182/j.issn1004731x.joss.23-0779

引用格式: 解鑫, 胡小兵, 周航. 动态路网环境下的路径优化算法研究[J]. 系统仿真学报, 2024, 36(8): 1969-1981.

Reference format: Xie Xin, Hu Xiaobing, Zhou Hang. Research on Path Optimization Algorithm in Dynamic Routing Environment[J]. Journal of System Simulation, 2024, 36(8): 1969-1981.

Research on Path Optimization Algorithm in Dynamic Routing Environment

Xie Xin¹, Hu Xiaobing^{2*}, Zhou Hang³

(1. College of Electronic Information and Automation, Civil Aviation University of China, Tianjin 300300, China;
2. College of Safety Science & Engineering, Civil Aviation University of China, Tianjin 300300, China;
3. Sino-European Institute of Aviation Engineering, Civil Aviation University of China, Tianjin 300300, China)

Abstract: In a real dynamic routing environment, static path optimization (SPO) and traditional dynamic path optimization (DPO) tend to encounter issues such as detours, reversals and high computational complexity due to frequent real-time optimization calculation. To address these problems, a novel restart co-evolutionary path optimization (RCEPO) method based on the ripple-spreading algorithm (RSA) is proposed. This method integrates the path optimization process with the dynamic changes of the routing network environment to enhance the effectiveness of path optimization. Moreover, the path re-optimization calculation is performed only when the dynamic changes in the routing environment exceed the predicted range, thereby reducing computational complexity. Experimental results demonstrate that the actual travel path length and the actual travel time of this method are shortened by 17% and 12%, respectively, compared with the traditional DPO method under the dynamic routing network environment. It can effectively solve the path optimization problem under the real dynamic routing network environment. The feasibility and effectiveness of this approach are validated through

收稿日期: 2023-06-28 修回日期: 2023-08-22

基金项目: 国家自然科学基金(62201577)

第一作者: 解鑫(1999-), 男, 硕士生, 研究方向为人工智能。

通讯作者: 胡小兵(1975-), 男, 教授, 博士, 研究方向为人工智能理论与方法。

experiments conducted with a robot dog.

Keywords: path optimization; co-evolutionary; dynamic environment; ripple-spreading algorithm; uncertainty

0 引言

路径优化是一项研究如何在给定的路网环境中找到最优路径的任务。其目标是通过最小化某个目标函数,例如,路径长度、时间、成本等来寻找从起点到终点的最优路径^[1]。路径优化已广泛应用于各个领域。例如,在物资配送-选址领域,路径优化可以用来确定最佳的物资配送路线和仓库选址。通过考虑交通拥堵、交通规则和配送需求等因素,可以有效地减少配送时间、降低成本,并提高物资供应的效率^[2]。机器人路径规划是另一个重要的应用领域。通过路径优化可以规划机器人或自动化系统的最优移动路径。这需要考虑到环境中的障碍物、机器人的能力以及任务需求,以提高机器人的移动效率和任务执行能力^[3]。路径优化的应用还扩展到交通路径规划^[4]、航空航天、军事作战规划、城市规划等领域。通过应用路径优化技术,可以更好地利用资源、提高效率、降低成本,并对现代社会的发展和运作产生重要影响。

路径优化问题根据路网环境的变化程度可分为静态路径优化(static path optimization, SPO)问题和动态路径优化(dynamic path optimization, DPO)问题。对于SPO问题,已经有了许多成熟的解决方法,如A*算法^[5-6]和Dijkstra算法^[7-8],因其具有最优性和可扩展性被应用于导航、定点货运等多个领域^[9]。然而,现实生活中的路网环境通常是动态变化且具有不确定性的,因此,DPO问题成为更贴近实际的研究内容。对于DPO问题,目前主要有两类解决方法。一类是基于在线重新优化(online re-optimization, OLRO)的DPO方法,可以看作是对SPO问题的拓展,主流算法是对A*算法和Dijkstra算法的改进^[10-11]。主要有D*算法^[12]、LPA*算法^[13]等,在动态路网环境发生变化时进行在线重新

优化。简单来说,就是在动态路网变化的时刻进行在线的静态路径优化重新计算,此时的路网实际上是被视为静态固定的。然而,在进行优化过程中,只能关注当前时刻的最优路径,而忽略了路网环境的可预测性,因此,在时变的动态路网环境下,无法保证规划的路径是最优的,甚至由于路网环境变化的影响,实际行进轨迹可能出现折返或绕路的情况。此外,由于路网环境几乎是时刻变化的,这种方法需要进行大量的在线重新优化计算致使效率较低。另一类是基于时间拓展超图(time-expanded hypergraph, TEHG)的时间相关路径优化(time-DPO, TDPO)方法。这类方法是将DPO问题与时间相关联,以便能够准确地描述动态路网环境的变化^[14-15]。通常这类方法被认为是在时间拓展超图上的SPO方法^[16]。具体而言,假设从起点到终点需要 T_{OD} 个时间单位,那么将这 T_{OD} 个时间单位内的所有路网信息堆叠在一起构成一个静态的时间拓展超图。等同于将DPO问题转化为在这个时间拓展超图上进行的SPO问题^[17]。然而,由于此时间拓展超图的规模远大于原始路网的规模,因此,构建和存储时间拓展超图需要大量的内存消耗。此外,由于时间拓展超图与时间的依赖关系,其构建过程也相对复杂。

根据上述讨论可知,目前,传统DPO方法的解决思路主要是假定当前时刻动态路网环境是静止固定的。当路网环境变化时需要进行频繁重复的在线优化计算,导致效率低下。此外,由于只关注当前时刻的最优路径,实际行进轨迹可能出现折返或绕路等情况,无法保证是理论上的最优路径。为了解决动态路网环境下路径优化问题,本文提出了一种基于涟漪扩散算法(ripple-spreading algorithm, RSA)的重启协同进化路径优化(restart co-evolutionary path optimization, RCEPO)方法,旨在

适应真实的动态路网环境。首先, 对路网环境中的动态障碍区域进行运动轨迹预测, 这为路径优化提供了关于动态障碍区域位置和行为信息。其次, 将路径优化过程与可预测路网环境的动态变化过程协同进化, 可以在预测范围内选择最优路径, 有效避免了折返或绕路的情况。最后, 仅当路网环境的动态变化超出预测范围时才进行路径的重新优化计算, 避免了频繁的重复在线优化计算, 提高计算效率。

1 问题描述

1.1 RCEPO 问题描述

传统的DPO方法在动态路网环境中需要进行实时的在线重新优化。如图1所示, 其只考虑当前时刻的路网环境, 并在当前时刻对路网进行在线重新优化。因此, 只能保证当前状态下的最优路径, 最后导致实际行进轨迹出现图中折返或绕路情况, 与最优路径相差甚远。具体而言, 传统的DPO方法无法考虑路网环境的可预测性和未来变化趋势, 它只能根据当前时刻的路网环境信息进行路径优化, 而忽略了路网环境的动态变化。因此, 所规划的路径可能会受到路网环境变化的影响, 导致移动物体按照规划路径行进至终点时间过长。

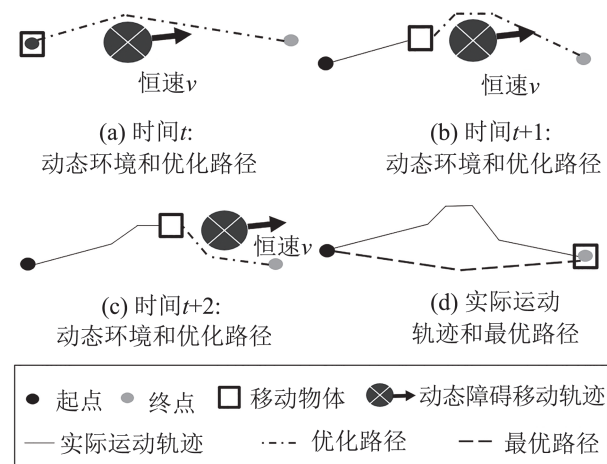


图1 传统DPO方法求解过程

Fig. 1 Solution process of traditional DPO method

根据上述情况, 为了解决动态路网环境下的路径优化问题, 需要考虑更多的因素。例如, 需要将优化过程与可预测路网环境的动态变化过程相结合、当路网环境的动态变化超出预测范围时的重新优化计算。本文提出的基于RSA的RCEPO方法将RSA算法作为实现RCEPO方法的核心算法, 其产生的涟漪以时间为单位进行扩散, 与路网环境的动态变化过程相协调。这种以时间为分析单位的优化过程很好地结合了路径优化过程与可预测路网环境的动态变化, 体现了协同进化优化路径的过程, 因此, RCEPO方法能够有效避免折返或绕路行为的发生。由于路网环境具有一定的可预测性, 因此, 不需要频繁进行实时的在线优化计算。但是在实际生活中动态路网环境可能由于某些不确定性因素很难100%预测准确, 例如, 障碍物(人、动物等)的突然改变方向或速度、火灾由于风向的影响改变蔓延方向等。因此, 仅当路网环境的动态变化超出预测范围时重新计算优化路径即可, 大大提高了计算效率。

1.2 RCEPO方法数学模型

与传统DPO方法不同, 本文关注了路网环境动态变化的可预测性, 其中, $C_{k|0}$ 表示当前时间 $t=0$ 时预测的未来时间 k 的代价矩阵, $A_{k|0}$ 表示当前时间 $t=0$ 时预测的未来时间 k 的邻接矩阵, $C_{k|0}$ 和 $A_{k|0}$ 在预测时间 $k|0$ 上的变化由预测的路网环境动态变化的动力学公式决定。因此, 当路网环境的动态变化是100%预测准确时, 路网环境动态变化的动力学公式 f_D 一般可以表示为

$$f_D(A_{k|0}, C_{k|0}) = [A_{k+1|0}, C_{k+1|0}], k \geq 0 \quad (1)$$

式中: $t+k|t$ 表示当前时间 t 预测未来时间 $t+k$; $k|0$ 表示时间 $t=0$ 时预测未来时间 k ; $C_{0|0}=C_0$ 、 $A_{0|0}=A_0$ 分别表示在 $t=0$ 时刻路网环境的代价矩阵和连通性; $f_D(\cdot)$ 为路网环境动态变化的动力学函数。

据式(1)可以看出, 当路网环境的动态变化是100%预测准确时, 则在初始 $t=0$ 时刻测得 C_0 和 A_0 , 便可通过式(1)得到所有优化过程时刻的路网

变化。因此，仅通过一次离线优化即可在预测准确的动态路网环境下找到最优路径。然而，路网环境的动态变化很难100%准确的在预测范围内，很容易受到某些细微不确定性因素影响，导致真实路网环境的动态变化与预测路网环境的动态变化出现小范围的误差。此误差可以表示为 $\varphi(k)$ ，则此时式(1)路网环境动态变化的动力学公式可以修改为

$$[A_{k+1|0}, C_{k+1|0}] = f_D(A_{k|0}, C_{k|0}) + \varphi(k), k \geq 0 \quad (2)$$

假设式(2)中的 f_D 是100%确定的，对于误差 $\varphi(k)$ 的估计，可以根据 $t=0$ 时刻的 $C_{k|0}$ 和 $A_{k|0}$ 估计出 $k|0$ 时刻的 $\varphi(k)$ 。因此，误差值可以表示为

$$\varphi(k|0) = \varphi(A_{k|0}, C_{k|0}), k \geq 0 \quad (3)$$

通过对误差 $\varphi(k)$ 的估计，可以定义一个新的100%准确预测的路网环境动态变化的动力学公式：

$$[A_{k+1|0}, C_{k+1|0}] = \tilde{f}_D(A_{k|0}, C_{k|0}), k \geq 0 \quad (4)$$

此时对于出现误差的情况，可以用式(4)代替式(1)，为 f_D 提供100%准确包络。换句话说，就是引入了一个100%确定的 \tilde{f}_D 作为包络，以此来表示不确定性因素带来的最大误差影响。然而由于路网环境动态变化的动力学公式引入了不确定性的误差，使得实际行进轨迹的最优性不再得到保证。但若能找到一个 $\varphi(k)$ 使 \tilde{f}_D 足够小，那么求得的实际行进轨迹也会更近似于最优的实际行进轨迹。

路网环境的动态变化会在预测范围内产生误差。若路网环境的动态变化受到很大程度不确定性因素影响，例如，障碍物(人、动物等)的突然改变方向或速度、火灾由于风向的影响改变蔓延方向等，导致真实路网环境的动态变化超出了预测路网环境的动态变化范围，并且产生的误差 $\varphi(k)$ 无法通过 $C_{k|0}$ 和 $A_{k|0}$ 进行包络预测估计，则此时需要对路网环境的动态变化重新进行预测，并以此时刻移动物体到达的位置作为新的起点重新进行路径的优化。

具体过程如图2所示，首先，通过初始时刻 $t=0$ 测得 C_0 和 A_0 按照式(1)的动力学公式得到初始时刻 $t=0$ 到优化完成时刻的所有路网的变化，以此计算出当前预测范围下的最优路径。其次，在移动物体沿着规划路径行进过程中，假设在 z 时刻路网环境的动态变化超出预测范围，则需要以 z 时刻作为新的初始时刻，重新预测真实路网环境的动态变化并测得 C_z 和 A_z ，按照真实路网环境动态变化的动力学式(5)得到此新的初始时刻 z 到优化完成时刻的所有路网的变化，以 z 时刻移动物体到达的位置作为新的起点。最后，计算出新的预测范围下的最优路径。两段路径相加即为移动物体的实际优化路径。若当 z 时刻路网环境的动态变化超出预测范围时，仍不重启协同进化路径优化(no RCEPO, NRCEPO)方法进行路径的重新规划，便会造成移动物体出现长时间的等待行为，甚至出现与障碍区域相撞的不安全行为。

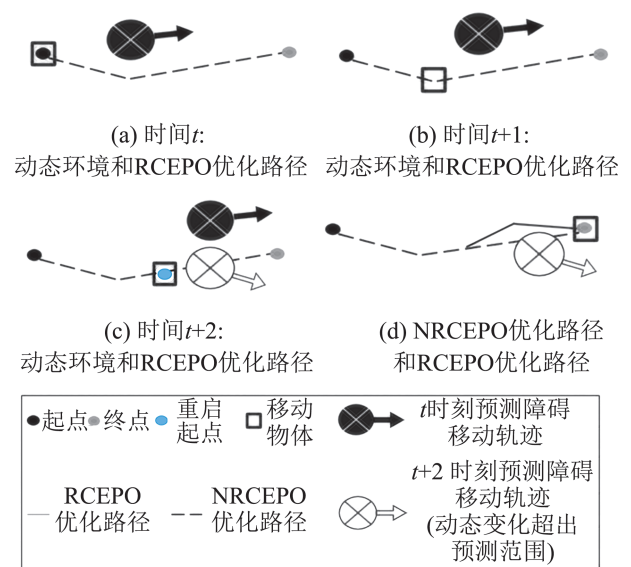


图2 RCEPO方法求解过程

Fig. 2 Solution process of RCEPO method

$$[A_{z+k+1|z}, C_{z+k+1|z}] = f'_D(A_{z+k|z}, C_{z+k|z}), k \geq 0 \quad (5)$$

虽然此情况下得到的路径可能并非理论最优路径，但必须考虑到大部分路网环境的动态变化是不能100%预测准确的。因此，对于超出预测范

围的真实路网环境的动态变化情况, 此优化路径更符合实际的动态路网环境, 更容易被应用到实际应用中。

2 算法及路径优化设计

2.1 RSA算法的基本思想

向自然学习是计算智能的重要基础, 许多成功的计算方法实际上是受到某些自然系统或现象的启发。RSA正是一种受自然界涟漪扩散现象启发而提出的路径优化算法。在自然界中, 涟漪以某一中心点为圆心, 并以固定速度向四周扩散, 因此最靠近中心点(产生涟漪的点)的节点最先被到达。文献[18-19]解决了SPO问题, 即在静态路网环境中求解最优路径, 并验证了其最优性。

RSA是一种自下而上、分散式、多智能体的仿真模型。在解决SPO问题时, 其路径优化过程可以描述为路网环境中各个节点之间的涟漪接力赛, 这个过程始于起始节点, 它产生第一个涟漪, 并以预设的恒定速度向外扩散。当涟漪扩散到一个未被访问过的节点时, 该节点将激活新的涟漪, 并继续进行涟漪间的接力过程。当一个涟漪最先抵达目标位置后, 这场涟漪接力赛随即结束。通过记录该涟漪经过的路径, 即可获得最优路径。在RSA中, 涟漪之间的接力过程是关键, 它使算法能够以时间为分析单位进行优化, 与传统以链接为分析单位的算法, 如A*、Dijkstra有所区别。

RSA对于解决SPO问题已有显著优越性。根据上述可知, 涟漪接力赛是一个以时间为分析单位的过程, 因此, 只要将随时间变化的动态路网环境嵌入到涟漪接力赛中, 即可实现优化过程与路网环境的动态变化过程协同进化。以本文提出的RCEPO方法, 其定义为在真实的动态路网环境中, 将原始规模路网环境的动态变化与以时间为分析单位的优化过程协同进化, 仅当路网环境的动态变化超出预测范围时重新计算优化路径即可。因此, 可以将以时间为分析单位的RSA作为实现

RCEPO方法的核心算法。通过对RSA的改进, 将路网环境的动态变化嵌入到涟漪扩散的优化过程中。因此在每个时刻可以同时改变路网环境和涟漪的扩散过程, 体现出路网环境动态变化与优化步骤的协同进化, 以此解决动态路网环境下的路径优化问题。

2.2 RCEPO方法基本原理

RCEPO方法将路网环境的动态变化过程嵌入到以时间为分析单位的涟漪扩散优化过程中, 简单地说, 涟漪扩散优化的同时, 动态路网也会随之改变, 呈现出协同进化的特征。RCEPO方法描述: ①设定涟漪的扩散速度, 并选择初始节点触发第一个涟漪。涟漪以设定的速度向四周扩散, 被涟漪到达的节点会触发产生新涟漪。如果与某一节点相邻的全部节点都已经被激活, 则该节点产生的涟漪消失并不再产生新的涟漪。②在涟漪扩散的同时, 由于动态路网环境中的障碍区域也在随之变化, 导致路网的通达性也随之变化。当障碍区域致使某个链接变为不可通达时, 则与其相连的节点所产生的涟漪在此链接处将变为等待状态。当此链接变为可通达时, 与其相连节点所产生的涟漪才会继续在此链接处继续扩散。若在某时刻路网环境的动态变化超出预测范围时, 则重新启动RCEPO方法计算实际优化路径。③当某一涟漪第一个到达目标节点时, 此时通过回溯涟漪所走过的路径, 即为动态路网环境下的实际优化路径。

为了实现RCEPO方法以解决动态路网环境下路径优化问题, 需要对现有的RSA算法进行改进, 对路网环境中的每个节点加入等待状态。当路网环境中障碍区域致使某个链接或节点变为不可通达时, 与其相连的节点所产生的涟漪, 在此链接处将变为等待状态。而该节点产生的涟漪仍可沿着, 与节点相连的可通达链接继续扩散。当链接或节点变为可通达时, 与其相连节点所产生的涟漪会继续在此链接处扩散。因为对于在不可

通达节点处等待,可能会比绕离不可通达节点花费的时间更短。

基于RSA算法的RCEPO方法实现流程如图3所示,设定路网环境为 $G(V,E)$, V 表示路网环境中节点的集合, $V=\{v_i|i=1,2,\dots,N\}$, E 表示路网环境中连接各个节点的链接集合, $E=\{e(v_i,v_j)|i\neq j,v_i\in V\}$, v_i,v_j 分别为路段 e 的起点和终点。设路网中有 N_0 个节点,则路网可以用 N_0 的邻接矩阵 A 表示, $A(i,j)=1$ 表示节点之间可以通达,否则 $A(i,j)=0$ 。对于 $A(i,i)=0$ 的节点自我链接的情况不考虑。 $A_{k|0}$ 表示在 $t=0$ 时刻预测 $t=k$ 时刻的路网状态。设定起点为节点1,终点为节点 N_0 ,并且每个节点只会产生一个涟漪。 $S_r(i)=0,1,2,3$ 表示节点 i 为未激活、等待、激活和消亡状态; $R_r(i)$ 为涟漪 i 的半径; $P_{ao}(i)=j$ 为节点 j 是路径 P 中的第 i 个节点; $F_r(i)=j>0$ 为涟漪 j 激活节点 i 。 v_s 为涟漪扩散速度,文献[18]已证明对于涟漪扩散速度 v_s 的设定必须满足

$$0 < v_s \leq L_{\min} \quad (6)$$

式中: L_{\min} 为路网环境中链接的最小长度。将 v_s 设为在一个时间单元内行进的距离,适用于动态路网环境(时间单元足够小,可以获得路网环境中的所有主要的变化)。 P_{ao} 用来记录准确预测范围内的最优路径, $L(z)=n$ 表示移动物体在 z 时刻行进到节点 n ,即代表移动物体的位置。

图3左侧为移动物体从起始位置向目标位置行进流程图。

3 理论分析

3.1 最优性

定理1:根据RSA的基本思想,基于路网环境的动态变化是100%准确预测的,依据涟漪之间的扩散接力赛,最先到达目的地的涟漪所走过的路径即为准确预测范围内的最优路径。

证明:自然界中涟漪以恒定的速度向四周扩散,因此距离中心点最近的节点总是被最先到达。

根据文献[18]中的论证,RSA已经被证明在解决静态路网环境路径优化问题时具有最优性。在基于RSA的RCEPO方法实现过程中,虽然障碍区域会随时间改变位置,但在同一时间内,节点或链接的通达性对于所有涟漪的扩散是一致的。并且当涟漪扩散到障碍区域时,可以选择等待或绕开障碍区域。因此文献[18]给出的最优性证明在动态路网环境下同样适用。尽管当路网环境的动态变化超出预测范围时,需要根据真实障碍区域的移动轨迹重新预测路网环境的动态变化,再次运行RCEPO方法计算预测范围内的路径,但仍可保证在预测范围内的路径为最优路径。

3.2 计算复杂度

定理2:假设路网环境为 $G(V,E)$,拥有 N_0 个节点, N_L 条链接,平均每个节点拥有 $N_{AL}=N_L/N_0$ 条链接。涟漪从一个节点出发通过一条链接平均花费 N_T 个单位时间,则对于路网环境的动态变化准确预测的条件下,RSA实现RCEPO的最大计算复杂度约为 $O(N_L \times N_T)$ 。而对于真实路网环境的动态变化超出预测范围 n 次则需要重新进行RCEPO n 次。其最大计算复杂度约为 $O(N_L \times N_T \times n)$ 。

证明:对于定理2的证明,协同进化的路网环境所消耗的时间,由于具体应用场景的不同而未进行考虑。相反,假设 $[C_{k|0}, A_{k|0}]$ 对的更新可以立即完成(例如, $[C_{k|0}, A_{k|0}]$ 对可以从另一台性能特定的计算机导入,这台计算机特别用于模拟动态路网环境的变化)。RSA的基本优化步骤是按涟漪扩散速度增加涟漪半径,然后将新半径与链接的长度进行比较。即一个加法计算和一个比较计算。对于在障碍区域前的等待行为不视为计算步骤。由于每个节点只能产生不超过一个涟漪,因此可以推断出在找到最短路径之前需要执行大约 $N_{AL} \times N_T + (N_0 - 2) \times (N_{AL} - 1) \times N_T$ 的基本计算步骤。因此,RSA实现RCEPO的计算复杂度可评估为 $O(N_0 \times N_T \times n)$ 。由于 $N_{AL}=N_L/N_0$,计算复杂度实际上约为 $O(N_L \times N_T)$ 。对于真实路网环境的动态变化

超出预测范围 n 次则需要重新进行 RCEPO 操作 n 次。对于障碍区域大多数时间的行进轨迹是在预测范围内行进的, 因此预测路网环境的动态变化由于不确定性而超出预测范围的情况是有限次数

的。对于大规模网络重启 RCEPO 方法的 n 值相对较小, 甚至可以忽略不计的。其最大计算复杂度约为 $O(N_L \times N_T \times n)$ 。

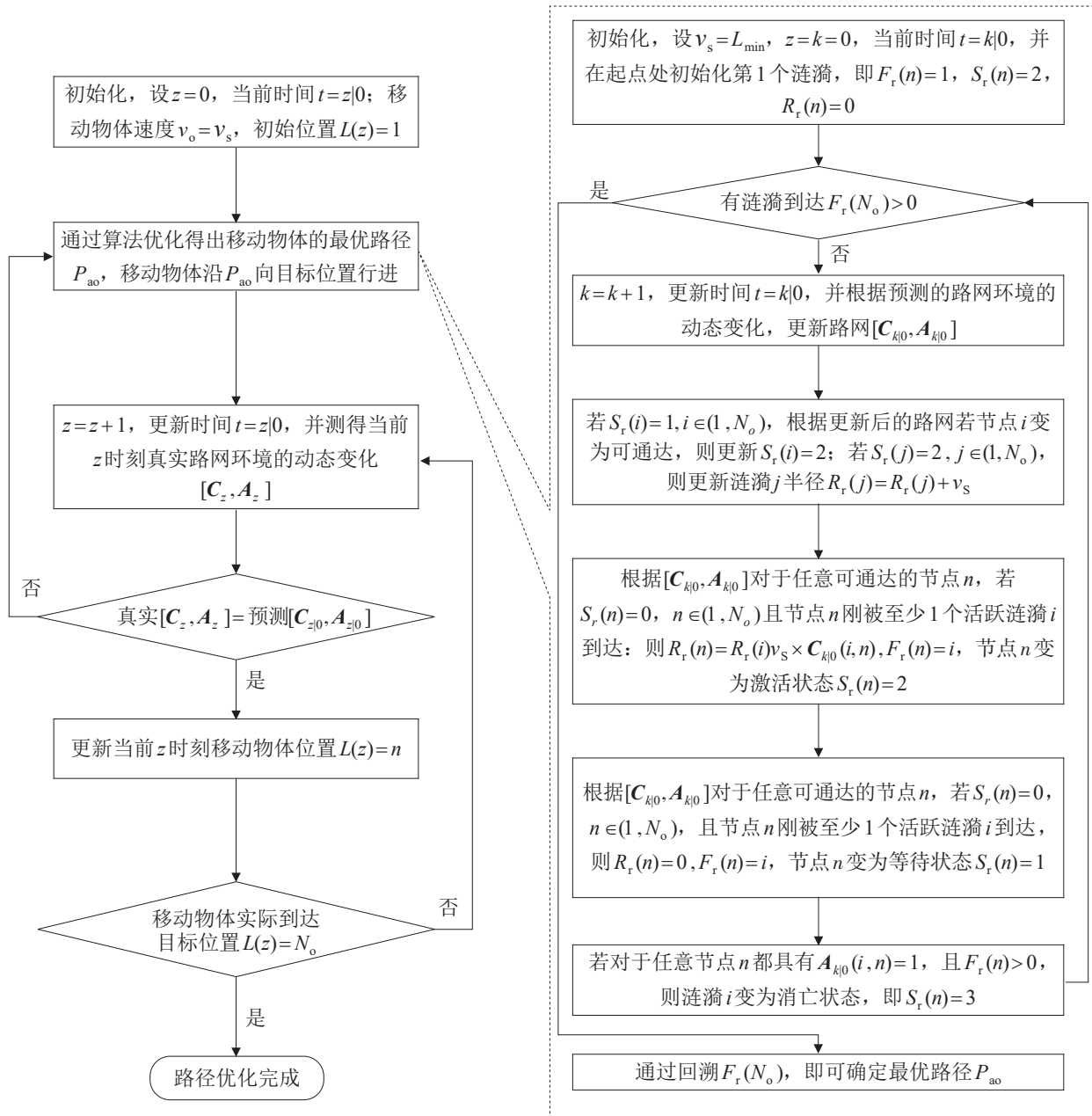


图 3 基于 RSA 的 RCEPO 方法流程
Fig. 3 RCEPO method flow based on RSA

类比 RCEPO 方法复杂度的计算, 基于 OLRO 的 DPO 方法的最大计算复杂度约为 $O(N_0 \times \ln N_0)$ 。这些方法大多都是面向链接为分析单位的。只是

假设与时间相关的链接成本预测已经预先确定。但其并没有考虑到将障碍区域的移动转换为与时间相关的链接成本预测的计算负担。然而在

RCEPO 方法中, 由于采用了以时间为分析单元的优化方法, 将障碍区域的移动转化为当前模拟时间瞬间的路网环境变化。这使其计算复杂度大大降低。对于内存需求方面, 假设使用邻接矩阵来记录/准备路网环境变化的必要信息进行计算, 那么基于 TEHG 的 TDPO 方法所使用的超图的大小约为 $N_0 \times N_0 \times N_{\text{TOD}}$, 其中 N_{TOD} 为从起点到终点需要的所有时间单位。尽管大多数超图中的信息在计算过程中不会被使用, 但是无法预先确定哪些信息不会被使用, 所以在计算开始时, 必须准备一个大小为 $N_0 \times N_0 \times N_{\text{TOD}}$ 的超图, 否则难以保证最优性。不同的是, 基于 RSA 的 RCEPO 方法的实现中, 所使用的网络总是大小为 $N_0 \times N_0$ (邻接矩阵)。因此对于大规模网络中随着优化步骤的急剧增多, 则基于 TEHG 的 TDPO 方法计算复杂度要远高于基于 RSA 的 RCEPO 方法。

4 实验结果

4.1 实验设置

本文通过对 4 类实验设置进行比较分析, 突出了基于 RSA 的 RCEPO 方法在真实不确定动态路网环境中的优越性。比较分析主要基于计算时间 (CT), 以 s 为单位测量, 是所有在线优化时间的总和; 实际运行轨迹长度 (PL); 实际行进时间 (TT), 以网络相关的模拟时间单位进行测量, 表示涟漪以预设的恒定速度通过实际路径的时间; 实际行进时间标准差 (σ_{TT}), 通过标准差反映出移动物体从起点向终点移动过程中的等待时间。标准差越小则说明移动物体的等待时间越短, 路径优化效果越好。反之, 标准差越大, 则等待时间越长, 路径优化效果越差。

第一类: 传统的 DPO 方法大多是在线重新优化的, 与 RCEPO 方法一样, 都是基于原始规模的路网。在该实验中, 选择了路径优化算法 A^* 算法^[20] 作为 DPO 方法的在线优化算法, 简记为 DPO。

第二类: 路网环境的动态变化是 100% 预测准确的情况下, 只进行一次 RCEPO 计算便可得到最优路径, 简记为 CEPO。

第三类: 路网环境的动态变化不是 100% 预测准确的情况下, 即真实路网环境的动态变化与预测路网环境的动态变化发生了较大幅度的不一致, 超出了预测范围。但是此时不重新进行 RCEPO 方法的计算。简记为 NRCEPO。

第四类: 路网环境的动态变化不是 100% 预测准确的情况下, 即真实路网环境的动态变化与预测路网环境的动态变化发生了较大幅度的不一致, 超出了预测范围。此时启动 RCEPO 重新计算得到实际优化路径, 简记为 RCEPO。

为了体现实验结果的随机准确性, 本文采用随机生成的路网作为实验环境。随机生成路网是一个 $[-1\ 000, 1\ 000, -1\ 000, 1\ 000]$ 的矩形区域, 将均匀分布在矩形区域内的 N_0 个节点随机移动位置, 保证节点之间没有重叠。然后随机将各个相邻节点进行连接, 以确保每个节点平均有 4 条链接。实验中将原点节点始终设为网络的左下节点, 目的节点始终设为网络的右上节点。为了保证 RSA 在最优性的情况下具有良好的计算效率, 将涟漪传播速度设置为最小链接长度, 因此模拟的时间单位为涟漪通过网络中最短链接的时间。

为了显示实验结果的一般适用性, 本文选取了 4 个不同尺度的网络, 即 N_0 分别为 100、225、400、900。对于动态路网环境, 设置了 4 种不同的场景, 每个场景中都有障碍区域的移动和变化, 具体设置如图 4 所示。

场景 I: 预测障碍区域从矩形区域的右下角沿对角线移动到左上角, 但由于不确定性行进轨迹变为垂直向上移动并且速度发生改变。

场景 II: 在矩形区域内, 预测障碍区域从右向左水平移动, 但由于不确定性行进轨迹发生偏转后, 障碍区域在实际行进到某一时刻时垂直向下

移动。

场景III: 预测障碍区域从左向右水平移动, 但障碍区域在实际运行到某一时刻时由于不确定性突然消失。

场景IV: 将场景I、场景II和场景III组合, 此时矩形区域内有 3 个障碍区域同时移动。

障碍区域的半径大小是根据矩形区域的大小随机设定的。对于障碍区域速度的设定, 当移动

物体从起点移动到终点时, 障碍区域以某一恒定速度能够从矩形区域的一侧移动到另一侧。

4.2 仿真结果及数据分析

实验对上述 4 种动态路网环境场景分别与 4 个不同尺度(节点个数 N_o) 的网络, 随机生成 1 000 个随机网络测试对比上述 4 种实验设置。实验数据结果如表 1~4 所示。其中选取了场景I和IV的部分实验结果图进行对比分析, 如图 5~6 所示。

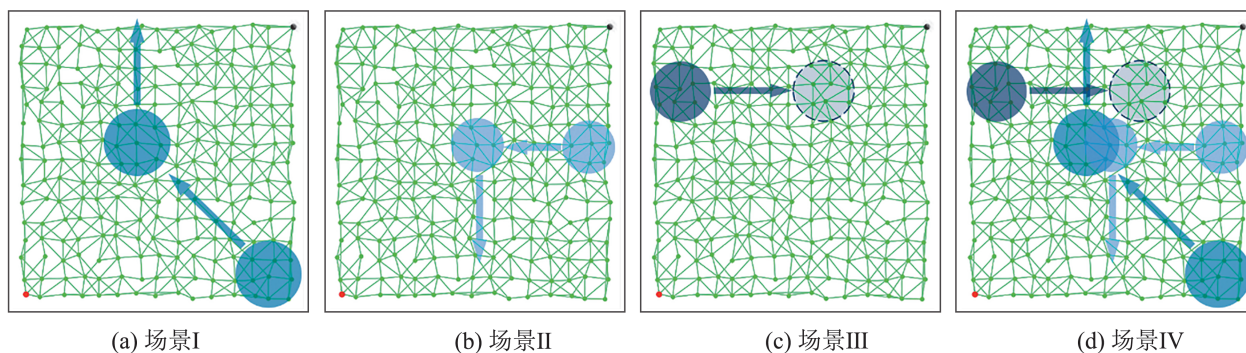


图 4 实验场景设置

Fig. 4 Experimental scenario settings

表 1 场景 1 对比实验平均结果

Table 1 Scenario 1 compare the average results of the experiment

N_o	指标	DPO	CEPO	NRCEPO	RCEPO
100	CT/s	1.955	0.018	0.017	0.031
	PL/m	156.32	135.20	137.24	137.55
	TT/s	30.21	25.66	28.29	26.20
	σ TT	1.90	1.55	1.81	1.57
225	CT/s	2.78	0.06	0.06	0.15
	PL/m	261.19	220.43	221.43	223.57
	TT/s	42.86	35.40	40.58	37.55
	σ TT	1.99	1.58	1.82	1.59
400	CT/s	12.19	0.15	0.14	0.43
	PL/m	350.46	278.66	282.99	279.60
	TT/s	56.25	46.95	50.13	48.13
	σ TT	2.19	1.66	1.98	1.71
900	CT/s	122.30	0.48	0.51	1.59
	PL/m	485.13	400.15	412.14	411.23
	TT/s	78.25	72.64	76.45	73.22
	σ TT	2.50	1.82	2.21	1.88

表 2 场景 2 对比实验平均结果

Table 2 Scenario 2 compare the average results of the experiment

N_o	指标	DPO	CEPO	NRCEPO	RCEPO
100	CT/s	2.010	0.015	0.015	0.032
	PL/m	149.65	135.25	139.32	136.28
	TT/s	29.65	25.01	27.32	26.05
	σ TT	1.90	1.52	1.79	1.55
225	CT/s	2.52	0.05	0.06	0.13
	PL/m	266.53	219.43	225.43	222.57
	TT/s	41.95	34.31	39.28	36.98
	σ TT	2.01	1.58	1.80	1.62
400	CT/s	13.52	0.16	0.16	0.39
	PL/m	362.26	278.62	285.68	280.60
	TT/s	57.62	46.02	51.23	48.44
	σ TT	2.15	1.65	2.00	1.70
900	CT/s	119.10	0.52	0.55	1.72
	PL/m	490.10	411.12	420.15	415.12
	TT/s	80.01	73.15	78.88	75.22
	σ TT	2.42	1.82	2.30	1.86

表3 场景3对比实验平均结果

Table 3 Scenario 3 compare the average results of the experiment

N_o	指标	DPO	CEPO	NRCEPO	RCEPO
100	CT/s	1.898	0.014	0.016	0.029
	PL/m	142.26	125.26	126.26	128.68
	TT/s	28.33	23.21	26.22	25.19
	σ TT	1.81	1.49	1.77	1.52
225	CT/s	2.01	0.05	0.05	0.11
	PL/m	232.91	216.43	220.43	219.59
	TT/s	39.61	33.15	36.58	34.42
	σ TT	1.91	1.60	1.82	1.65
400	CT/s	10.95	0.12	0.15	0.35
	PL/m	332.15	256.20	274.33	258.36
	TT/s	54.26	42.31	46.08	43.22
	σ TT	2.05	1.60	1.88	1.64
900	CT/s	110.32	0.47	0.45	1.53
	PL/m	477.15	380.22	400.15	390.15
	TT/s	76.35	70.36	75.74	71.36
	σ TT	2.36	1.77	2.20	1.79

表4 场景4对比实验平均结果

Table 4 Scenario 4 compare the average results of the experiment

N_o	指标	DPO	CEPO	NRCEPO	RCEPO
100	CT/s	2.515	0.021	0.029	0.044
	PL/m	172.16	146.15	150.87	146.95
	TT/s	39.26	28.99	35.15	30.11
	σ TT	1.97	1.58	1.88	1.62
225	CT/s	4.01	0.10	0.15	0.23
	PL/m	301.91	250.43	257.43	255.59
	TT/s	45.61	39.15	42.58	40.42
	σ TT	2.11	1.65	1.99	1.72
400	CT/s	19.11	0.39	0.36	0.68
	PL/m	420.27	301.25	325.78	304.52
	TT/s	69.77	50.12	58.14	52.91
	σ TT	2.45	1.79	2.15	1.88
900	CT/s	180.15	0.55	0.61	2.61
	PL/m	561.15	448.15	480.68	450.14
	TT/s	90.19	77.15	81.66	79.77
	σ TT	2.81	1.77	2.30	1.89

DPO: 实际优化路径: 258.07 m; 行进时间: 42.12 s CEPO: 实际优化路径: 220.61 m; 行进时间: 36.20 s NRCEPO: 实际优化路径: 211.94 m; 行进时间: 39.98 s RCEPO: 实际优化路径: 218.58 m; 行进时间: 37.03 s

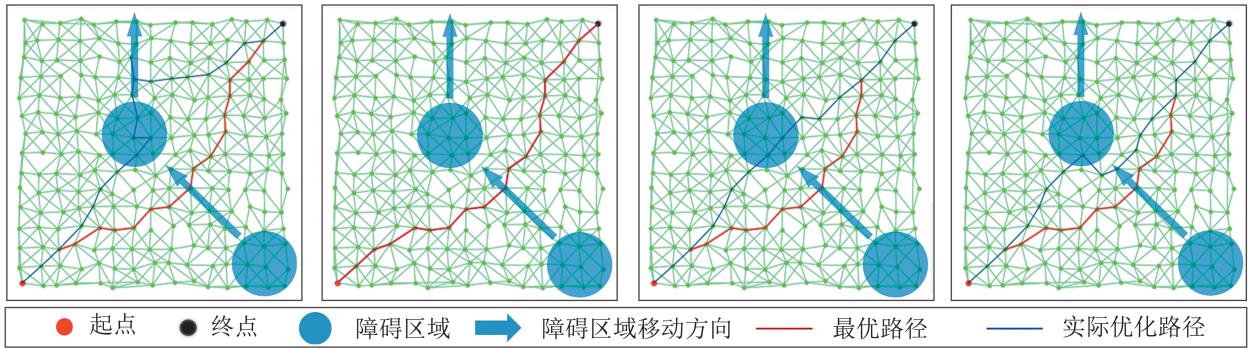


图5 场景I路径优化对比图

Fig. 5 Comparison of path optimization in scenario I

DPO: 实际优化路径: 293.55 m; 行进时间: 46.09 s CEPO: 实际优化路径: 241.90 m; 行进时间: 38.10 s NRCEPO: 实际优化路径: 245.40 m; 行进时间: 43.40 s RCEPO: 实际优化路径: 244.58 m; 行进时间: 39.41 s

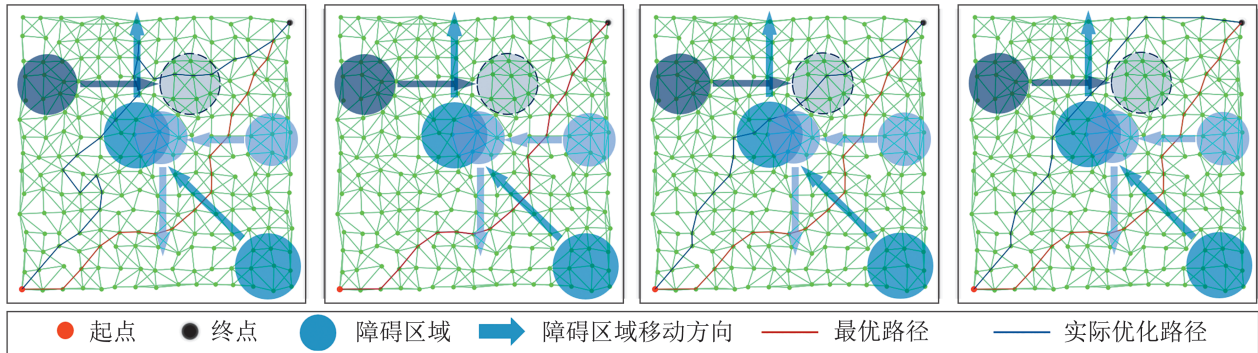


图6 场景IV路径优化对比图

Fig. 6 Comparison of path optimization in scenario IV

从图5~6可以看出, DPO方法更关注的是当前时间内求得的最短路径, 当移动物体遇到障碍区域时, 会反复求其最短路径并基于当前最短路径对行进轨迹进行频繁重新规划, 导致其实际行进轨迹长度和行进时间较大。CEPO求得的路径是理论最优路径, 由于路径优化步骤与100%预测准确的路网环境的动态变化协同进化, 使实际的行进轨迹与理论最优路径一致, 在合适的时间恰好避开移动的障碍区域。实际行进轨迹长度和行进时间比第一类实验小得多。NRCEPO和RCEPO可以得出, 当预测值与真实值发生很大程度的不一致时, NRCEPO和RCEPO的方法, 实际行进轨迹长度或长或短。但是实际行进时间RCEPO比NRCEPO方法短。因为当移动物体在实际移动时, 由于障碍区域的预测行进轨迹和实际行进轨迹不一致, NRCEPO规划出的路径可能会使移动物体在行进过程中遇到障碍区域从而出现长时间的等待, 而RCEPO规划的路径没有等待时间或等待时间很短。结果表明, 在实际应用中, 当预测路网环境的动态变化与真实路网环境的动态变化在某一时刻出现很大程度不一致时, RCEPO方法无论是规划行进轨迹还是实际行进时间都体现出明显的优势。

表1~4给出了4个不同场景和4个不同节点数量各组合下, 对于每种组合进行1000组随机实验的实验结果, 每组实验使用随机生成的网络。从表1中得出以下结论: 相较DPO方法, RCEPO在CT、PL、TT和 σ_{TT} 方面表现出明显的优势。实际运行轨迹长度和实际行进时间分别比传统DPO方法缩短了约17%和12%。同样实际行进时间的标准差也比传统DPO方法减少了21%。由于RCEPO方法中加入了等待行为, 这使实际行进轨迹和时间更短。加入等待行为意味着行进速度为0, 将明显降低从起点到终点的平均行进速度。而DPO方法中没有任何等待行为。因此, 当实际运行轨迹长度除以实际运行时间时, 可以得到RCEPO方法

实际运行速度小于DPO方法。

相较于NRCEPO方法, RCEPO方法在实际运行轨迹上可能长于或短于NRCEPO方法, 这是因为移动物体在遇到障碍物时可能会采取等待行为。即使实际行进轨迹短, 但是时间的等待行为也会导致最终实际行进时间增加。对于实际行进时间和实际行进时间标准差而言, RCEPO方法比NRCEPO方法分别缩短了约5%和14%。

4.3 机器人路径规划仿真结果

为了验证本文方法的实用性, 将其应用到机器人路径规划领域。在Ubuntu18.04系统内使用机器人操作系统(ROS)进行机器人路径优化的仿真对比, 并使用宇树科技Go1机器狗进行了实物实验测试。使用DPO方法与RCEPO方法分别对机器人的行进路线进行规划, 规划结果如图7所示。可以看出在动态路网环境中, RCEPO方法在实际运行轨迹长度和实际运行时间上都要优于DPO方法。

4.4 实验结论

根据进行的仿真实验及不同方法之间的对比结果, 可以得出结论: RCEPO方法在动态路网环境中具有更强的实用性和有效性。不仅解决了DPO方法出现的绕路或折返问题, 还解决了当路网环境的动态变化超出预测范围时, 路径的重新优化计算问题。RCEPO方法在实际行进轨迹上不会出现折返或绕路等路径过长的情况。在实际行进时间方面, 也保证了在真实动态路网环境中最短。综上所述, RCEPO方法在解决真实动态路网环境下路径优化问题时, 在实际行进轨迹和实际行进时间等方面具有更好的实用性、有效性和灵活性。相较于传统的DPO方法、CEPO方法和NRCEPO方法, RCEPO方法能够更好地应对动态路网环境的挑战, 并提供更优秀的路径优化解决方案。

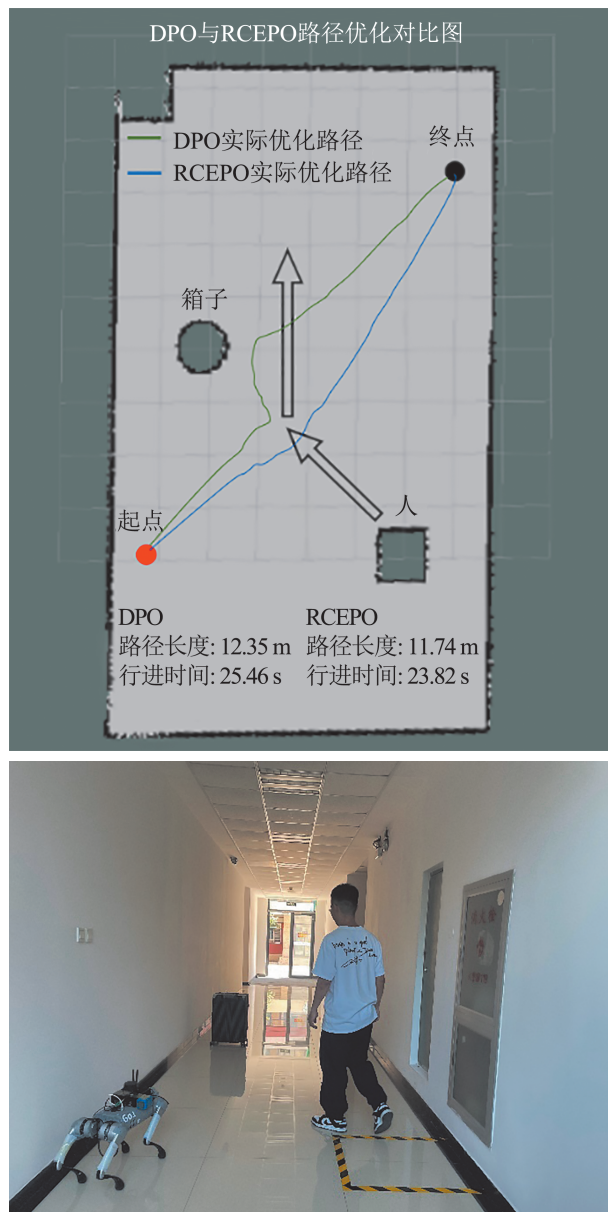


图7 DPO与RCEPO路径优化结果对比

Fig. 7 Comparison of DPO and RCEPO path optimization results

5 结论

针对实际应用中动态路网环境下路径规划问题, 现有的OLRO、TEHG方法主要是面向链接分析路径优化问题。无法很好地结合路网环境的动态变化与路径优化过程, 因此, 在实用性、有效性和灵活性方面存在一定的局限性。本文提出的基于RSA的RCEPO方法成功地将路网环境的动态

变化过程与路径优化过程相结合。RSA是基于原始规模的路网环境进行涟漪接力赛的, 涟漪接力赛是一个面向时间的过程。因此, 涟漪接力赛和路网环境的动态变化随着时间协同进化, 当某个涟漪最先抵达目标位置后, 此次涟漪接力赛的过程随即结束, 通过记录该涟漪经过的路径, 即可得到最优路径。当路网环境是高度动态变化时, 即路网环境的动态变化超出预测范围时, 也可以在此刻重新确定起点并规划路径。实验结果表明: 符合实际场景应用的RCEPO方法在实际行进轨迹长度和实际行进时间分别比传统DPO方法缩短了约17%和12%。因此, 基于RSA算法的RCEPO方法在实用性、有效性和灵活性方面都表现出更突出的优势。

在未来研究中, 将该方法应用到具体的应用领域中。例如, 将实体机器人用于变电站的巡检任务、光伏发电场光伏板下的除草任务等, 以此验证该方法的实际可行性, 进一步评估RCEPO方法在实际应用中的表现, 完善和改进基于RSA的RCEPO方法, 并将其推广应用于更多实际应用中。

参考文献:

- [1] Sapundzhi F I, Popstoilov M S. Optimization Algorithms for Finding the Shortest Paths[J]. Bulgarian Chemical Communications, 2018, 50(SB): 115-120.
- [2] 张渝彬, 徐飞. 城市生鲜冷链物流配送中心选址与路径优化问题研究现状分析[J]. 中国储运, 2022(10): 97-98.
- [3] Noraziah Adzhar, Yuhani Yusof, Muhammad Azrin Ahmad. A Review on Autonomous Mobile Robot Path Planning Algorithms[J]. Advances in Science, Technology and Engineering Systems Journal, 2020, 5(3): 236-240.
- [4] 陈晓明, 李引珍, 沈强, 等. 基于双层复杂网络的城市交通网络协同优化方法[J]. 计算机应用, 2019, 39(10): 3079-3087.
Chen Xiaoming, Li Yinzhen, Shen Qiang, et al. Urban Traffic Networks Collaborative Optimization Method Based on Two-layered Complex Networks[J]. Journal of Computer Applications, 2019, 39(10): 3079-3087.
- [5] Teleweck P E, Chandrasekaran B. Path Planning Algorithms and Their Use in Robotic Navigation

- Systems[J]. *Journal of Physics: Conference Series*, 2019, 1207(1): 012018.
- [6] Ju Chunyu, Luo Qinghua, Yan Xiaozhen. Path Planning Using an Improved A-star Algorithm[C]//2020 11th International Conference on Prognostics and System Health Management (PHM-2020 Jinan). Piscataway: IEEE, 2020: 23-26.
- [7] Javaid M A. Understanding Dijkstra's Algorithm[J]. *SSRN Electronic Journal*, 2013.
- [8] 李全勇, 李波, 张瑞, 等. 基于改进Dijkstra算法的AGV路径规划研究[J]. *机械工程与自动化*, 2021(1): 23-25, 28.
Li Quanyong, Li Bo, Zhang Rui, et al. Research on AGV Path Planning Based on Improved Dijkstra Algorithm[J]. *Mechanical Engineering & Automation*, 2021(1): 23-25, 28.
- [9] Abhishek Goyal, Prateek Mogha, Rishabh Luthra, et al. Path Finding: A* or Dijkstra's?[J]. *International Journal in IT and Engineering*, 2014, 2(1): 1-15.
- [10] Sungkwan Kim, Hojun Jin, Minah Seo, et al. Optimal Path Planning of Automated Guided Vehicle Using Dijkstra Algorithm Under Dynamic Conditions[C]//2019 7th International Conference on Robot Intelligence Technology and Applications (RiTA). Piscataway: IEEE, 2019: 231-236.
- [11] 刘建美, 马寿峰, 马帅奇. 基于改进的Dijkstra算法的动态最短路计算方法[J]. *系统工程理论与实践*, 2011, 31(6): 1153-1157.
Liu Jianmei, Ma Shoufeng, Ma Shuaiqi. Computation Method of the Dynamic Shortest Path Based on Improved-Dijkstra Algorithm[J]. *Systems Engineering-Theory & Practice*, 2011, 31(6): 1153-1157.
- [12] Seifedine Kadry, Gennady Alferov, Viktor Fedorov, et al. Path Optimization for D-star Algorithm Modification[J]. *AIP Conference Proceedings*, 2022, 2425(1): 080002.
- [13] Alice Segato, Marco Di Marzo, Sara Zucchelli, et al. Inverse Reinforcement Learning Intra-operative Path Planning for Steerable Needle[J]. *IEEE Transactions on Biomedical Engineering*, 2022, 69(6): 1995-2005.
- [14] Chen Biyu, William H K Lam, Agachai Sumalee, et al. Reliable Shortest Path Problems in Stochastic Time-dependent Networks[J]. *Journal of Intelligent Transportation Systems*, 2014, 18(2): 177-189.
- [15] Daniel Yamín, Andrés L Medaglia, Prakash A A. Exact Bidirectional Algorithm for the Least Expected Travel-time Path Problem on Stochastic and Time-dependent Networks[J]. *Computers & Operations Research*, 2022, 141: 105671.
- [16] Arun Prakash A. Algorithms for Most Reliable Routes on Stochastic and Time-dependent Networks[J]. *Transportation Research Part B: Methodological*, 2020, 138: 202-220.
- [17] David López, Angélica Lozano. Shortest Hyperpaths in a Multimodal Hypergraph with Real-time Information on some Transit Lines[J]. *Transportation Research Part A: Policy and Practice*, 2020, 137: 541-559.
- [18] Hu Xiaobing, Wang Ming, Leeson M S, et al. Deterministic Agent-based Path Optimization by Mimicking the Spreading of Ripples[J]. *Evolutionary Computation*, 2016, 24(2): 319-346.
- [19] Hu Xiaobing, Zhang Mingkong, Zhang Qi, et al. Co-evolutionary Path Optimization by Ripple-spreading Algorithm[J]. *Transportation Research Part B: Methodological*, 2017, 106: 411-432.
- [20] Zhang Jing, Wu Jun, Shen Xiao, et al. Autonomous Land Vehicle Path Planning Algorithm Based on Improved Heuristic Function of A-star[J]. *International Journal of Advanced Robotic Systems*, 2021, 18(5): 17298814211042730.