

宋哲昊¹ 唐芃¹

1. 东南大学建筑学院; songzhehao1996@126.com

Song Zhehao¹ Tang Peng¹

1. School of Architecture, Southeast University

基于 A* 寻径算法的三维路径系统生成工具研究^①

Research on the Tool of Generating 3D Path System Based on A* Path Finding Algorithm

摘要: 在城市设计中,道路的规划一直是一个重要的问题。最短路径算法是计算机图形学领域寻找两节点之间最短路径的一种算法,代表算法有 Dijkstra 算法和 A* 算法等。目前已经有许多将最短路径算法应用于城市设计的研究,但现有的研究普遍局限在二维层面上。而随着城市的垂直发展与功能的进一步复合,建筑师开始面临着在三维空间中规划路径的问题。研究通过 Java 语言构建了基于 A* 寻径算法的三维路径生成工具。首先程序读取城市空间数据,并在其中构建三维寻径网络。随后对 A* 算法进行改写,使其能够在三维寻径网络中运行。然后基于 A* 三维寻径算法实现在三维寻径网络中依据建设成本、景观品质、规范要求的多目标优化的路线寻径,并使生成的三维道路满足上下坡度、建设规范等要求,从而形成一套完整的可实际应用的三维路径系统生成工具。研究表明,随着计算机技术在城市与建筑设计中的应用,在城市尺度上通过算法进行三维路径系统的生成是可以实现的。同时,研究对寻径算法中移动代价这一概念进行了拓展,使其能够充分考虑城市路径设计中影响路径系统布局的多个限定条件,从而实现面向实际问题的多目标综合优化方案。

关键词: 路径系统; 三维寻径算法; A* 算法; 多目标优化; 移动代价

Abstract: In urban design, road planning has always been an important issue. Shortest path algorithm is an algorithm to find the shortest path between two nodes in computer graphics. The representative algorithms are Dijkstra algorithm, A* algorithm. At present, there are many researches on the application of shortest path algorithm in urban design, but the existing researches are generally limited to the two-dimensional level. With the vertical development of the city and the further integration of functions, architects begin to face the problem of planning path in three-dimensional space. In this paper, a three-dimensional path generation tool based on the A* routing algorithm is constructed by Java programming language. Firstly, the program reads the urban spatial data, and constructs a three-dimensional routing network. Then the A* algorithm is rewritten to run in 3D routing network. Next, based on A* three-dimensional routing algorithm, the multi-objective optimization of routing is realized in the three-dimensional routing network according to the construction cost, landscape quality and specification requirements, and the generated three-dimensional road meets the re-

^① 2021 年江苏省研究生培养创新工程研究生实践创新计划,项目代码: SJCX21_0024。

quirements of proper slope and construction specification. Thus a complete set of practical 3D path system generation tool is formed. The research shows that with the application of computer technology in urban and architectural design, it is feasible to generate a three-dimensional path system on the scale of cities. Also in this study, the concept of moving cost in path finding algorithm is extended, which can fully consider the constraints of urban path system's design, so as to realize the multi-objective comprehensive optimization design facing practical problems.

Keywords: Path System; Three Dimensional Path Finding Algorithm; A* Algorithm; Multi-Objective Optimization; Moving Cost

1 引言

1.1 研究背景

在城市中，路径作为连接空间的元素，对我们理解城市空间体系有着重要的作用^[1]。早在 20 世纪，以柯布西耶为代表的国际现代建筑协会已经将交通作为与居住、工作、游憩并列的四大城市功能之一。在城市设计中，道路的规划不仅仅影响着地块的交通，亦影响着区域内的功能定位、产业布局、建筑形态与景观设计，因此对道路设计方法的深入探索尤为重要。随着数字技术的不断发展，以人机互动的数字技术方法工具变革为核心特征的第四代城市设计快速发展^[2]。而路径的设计往往目标明确，且有许多可以量化评判的标准，因此非常适合通过参数化手段开发寻找最优路径的设计方法。

最短路径算法是计算机图形学领域在方格网范围内寻找两节点之间最短路径的一种算法，代表算法有 Di jkstra 算法和 A* 算法等。最短路径算法可以辅助设计师在城市设计的问题中找到城市节点之

间的最优路径，进而为城市车行与步行系统的设计提供参考。

1.2 算法选择

最短路径算法最为普遍的是基于方格网进行寻径的算法。将空间抽象为方格网并基于方格网的寻径算法按照原理可分为广度优先搜索，Di jkstra 算法，最佳优先搜索和 A* 算法等。

广度优先搜索即从起点出发，由近到远遍历网络中的所有点（图 1），直到找到终点，是最基本的寻径算法，适用于网络中各点性质完全相同的均质寻径网络。

Di jkstra 算法是一种求两固定点之间最短距离的核心算法^[3]，其原理为从起点出发不断寻找剩余的点中从起点开始“移动代价”最小的点，直至找到终点，比较适用于各点易达度不同的寻径网络（图 2）。在城市设计问题中，路径会自动绕开障碍物（如建筑、水系）从而找到最短路径，但由于在网络中寻找点的依据与终点无关，所以需要遍历较长时间，效率较低。

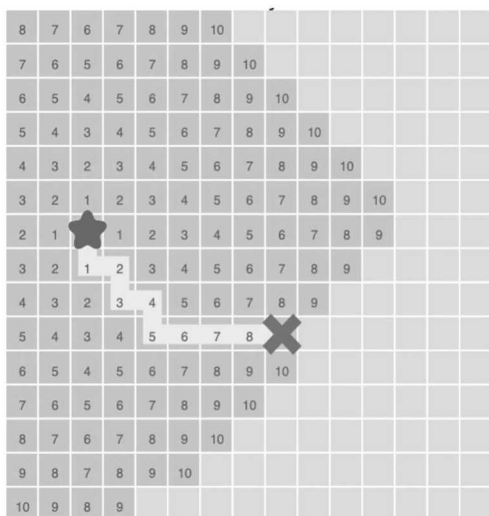


图 1 平面上的广度优先搜索

(图片来源: <https://paul.pub/a-star-algorithm/>.)

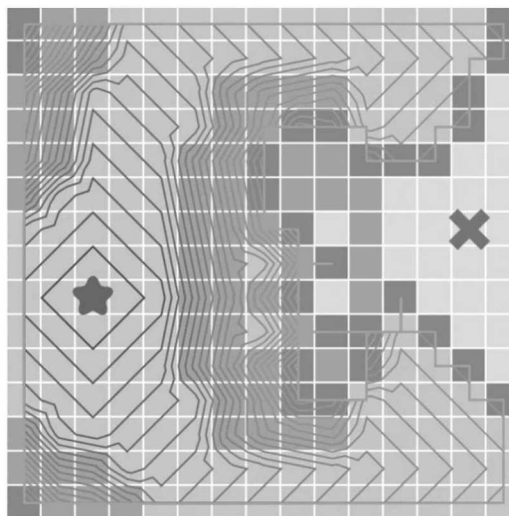


图 2 平面上的 Di jkstra 算法

(图片来源: <https://paul.pub/a-star-algorithm/>.)

最佳优先搜索即在网络中不断搜索当前位置距离终点最近的点，具有明确的指向性（图 3）。但由于不考虑移动代价，最佳优先算法每步的计算是

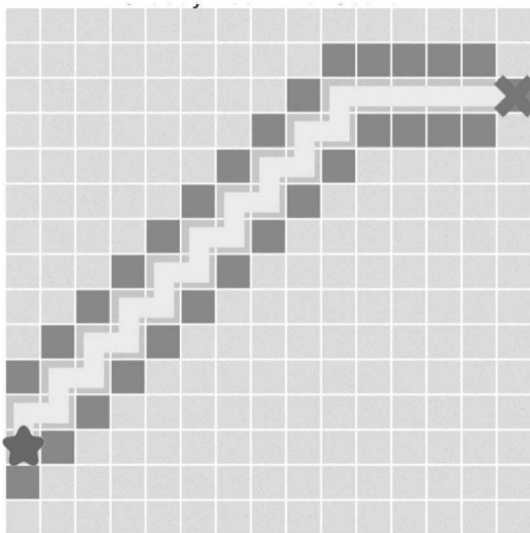


图 3 平面上的最佳优先搜索

（图片来源：<https://paul.pub/a-star-algorithm/>。）

独立的。因此如果起点和终点之间存在障碍物，则最佳优先算法找到的很可能不是最短路径（图 4）。

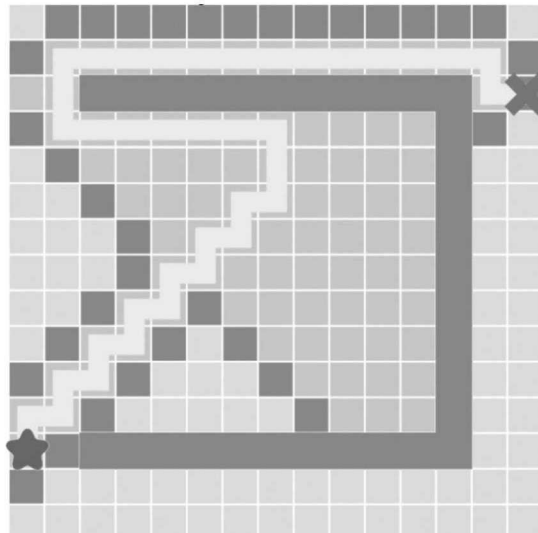


图 4 最佳优先搜索找到的不一定是最短路径

（图片来源：<https://paul.pub/a-star-algorithm/>。）

A* 算法可以看作在 Dijkstra 算法的基础上引入当前节点与终点的估计函数^[4]，其搜索下一个点的表达式为 $f(n) = g(n) + h(n)$ 。其中 $g(n)$ 为从起点开始的“移动代价”， $h(n)$ 则被称为启发函数，即当前点与终点的位置关系，启发函数直接影响了 A* 算法的行为，决定了 A* 算法在寻径过程中运算效率与寻径结果准确性的平衡。A* 算法兼具 Dijkstra 算法的最短特性与最佳优先搜索的高效性，因此比较适合本次研究。

1.3 既有研究与问题导向

目前已经有许多将最短路径算法应用于城市设计的研究，主要包括景观规划中的道路选择、选址分析中的最短路径计算、防火防洪疏散的路径模拟等^[5]。但由于计算机算力和现有寻径网络的限制，已有的研究普遍局限在二维层面上，即使是面对三维问题，也往往会将三维问题映射到二维平面上解决。随着城市的垂直发展与功能的进一步复合，路径系统的定义早已突破地面上的步行与车行道路，城市需要三维空间中的路径系统来解决交通问题，需要一套能够对三维路径系统进行解析与生成优化的程序化方法。因此本研究立足于在 Java 程序环境中构建三维寻径网络，并将 A* 寻径算法拓展到

三维层面上。同时对算法进行改进，使其能够针对多个领域的限制条件进行多目标综合优化寻径。最后基于此算法形成一套程序平台中的设计辅助工具，使其能够辅助城市设计过程中三维路径的设计。

2 三维寻径网络的搭建

2.1 城市数据的读取

对场地已有信息的读取是构建三维寻径网络的前提。程序可自动读取场地内的建筑、广场、绿地、水系的平面信息及标注的建筑高度信息（图 5），并依据给定的网络中各点在 X, Y, Z 轴方向上的间距在场地范围内生成寻径网络。网络中各点均会被赋予一个 Mark 值，代表该点是否适合作为交通路径上的点的评分。在赋值过程中，建筑内部的点的 Mark 值被赋值为无限大（即不可通过），空间中的其他点则会因为其所处的平面位置与高度不同而造成在该点铺设道路或架设天桥的建设成本不同（例如是否位于水面上），因此各点的 Mark 值可以看作场地内的每个点建设成本的评分。

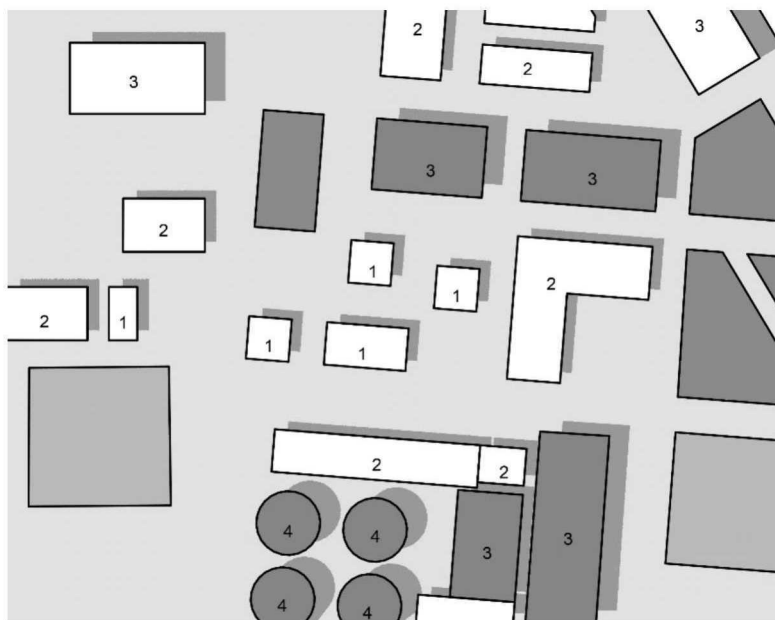


图5 DXF导入场地信息

(图片来源:作者自绘)

2.2 地块性质对寻径网络的限定

三维寻径算法主要用于城市中商业街区和景观公园的三维交通系统的设计,但三维路径在这两类街区中的设置方式并不完全相同。在商业街区中,建筑较为密集,三维路径往往是围绕在建筑周围的空中步行天桥,起到连接各核心节点的步行交通的作用。而景观公园中的三维路径则主要为步行天桥,发挥空中观景的功能,可以在远离建筑物较为开阔的空间中漫游。

在构建三维寻径网络的过程中对这一问题进行了考虑,在商业街区的寻径网络中,点与周围建筑物的距离会对各点 Mark 值产生影响,使得三维寻径网络中的各点越远离建筑物则其 Mark 值越高,从而让适宜布置三维路径的区域集中在建筑周围。

2.3 景观影响因素的引入

在实际设计过程中,三维路径的布置应考虑景观的优劣,因此将对景观的量化评价引入三维寻径网络的生成过程中。在程序运行界面上,建筑师可基于实际的设计目标,手动在三维空间中添加景观节点,这些景观节点即用地范围内景观价值较高的位置。景观节点添加完毕后,程序会对三维寻径网络中的每一个点进行遍历,并计算能与之建立直接视线联系的景观节点数。这一数值越高,代表着该点的景观品质越好,而这一数值亦会影响该点的 Mark 值。在遍历过程结束之后,各点的 Mark 值可

以看作场地内的每个点建设成本、用地限制与景观品质的综合评分。

3 基于 A* 算法的三维路径系统生成工具

3.1 A* 寻径算法原理

A* 算法的基本逻辑即从起点出发,不断将当前点周围还未被程序遍历的点加入一个集合中,这一集合被命名为 Open-set。Open-set 中 $f(n)$ 值最小的点会被作为下一个当前点,且该点会被从 Open-set 中删除并加入另一个集合 Close-set 中。程序多次迭代,将越来越多的点加入 Close-set 中,直至找到终点。

程序每次遍历到一个点时,只要其还未处于 Close-set 中,都会被赋予或更新一个父节点信息。父节点即该点周围 Close-set 中的点相对于起点总移动代价最小的点。因此当找到终点后,从终点不断回溯其父节点,即为移动代价最低的最优路径。

在选取下一个当前点的过程中,其依据为表达式 $f(n) = g(n) + h(n)$,即选取当前点临近节点中 $f(n)$ 值最小的点作为下一个遍历的点。在这一问题中, $g(n)$ 即从起点开始的移动代价,由各点的 Mark 值累加决定,使下一个当前点的选取受到点本身的建设成本、用地限制与景观品质这一系列客观条件的综合影响。而 $h(n)$ 即启发函数,可

直接采用当前点与终点在三维空间中的欧几里得距离，使下一个当前点的选取受到其与终点距离的影响，使点的选取在空间中具有倾向性，以减少遍历次数，提高程序运行效率。

3.2 二维寻径生成地面路径

三维路径系统由地面上的二维路径及空中的三维路径共同组成。为保证设计的合理性，二维路径需要

一定的净空高度，会对三维路径产生影响，因此应首先在地面上完成二维的路径寻径。在程序界面中，设计师的操作可以被程序读取并将数据反馈到寻径网络中。设计师可以在程序界面手动给定路径的起点和终点，并依据设计需要在场地内选取地面道路必须经过的中间节点，随后在寻径网络中进行基于 A* 算法的二维寻径，得到场地内的地面道路（图 6）。

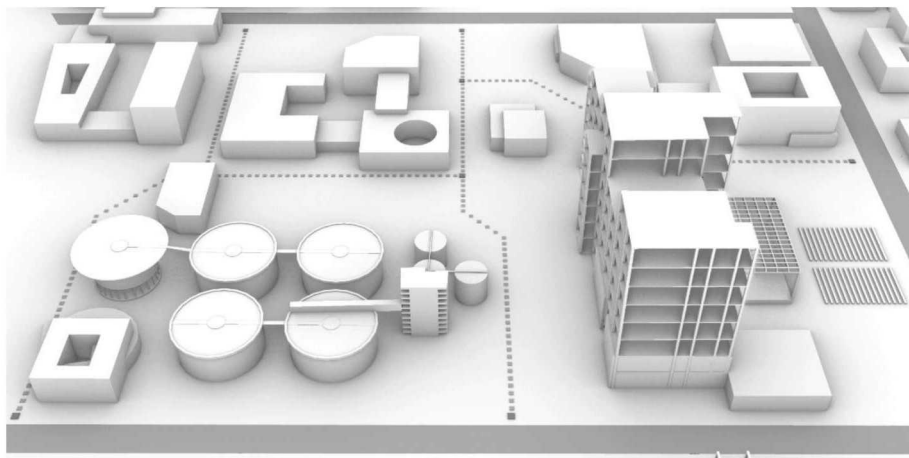


图 6 二维路径节点与寻径

（图片来源：作者编写的程序导出）

寻径算法得到的路径是没有宽度的，且是各自独立的，不考虑周围其他的路径。但实际情况是地面上的道路尤其是车行道需保证一定宽度和安全的上空高度。因此在程序中二维路径寻径完成后，会依据设计师根据设计需要预先设定的道路宽度和规范高度，将寻径网络中所有位于道路宽度范围内且高度低于规范高度的点的 Mark 值调整为无限大，即后续的三维路径不可通过。此时空间中各点的 Mark 值可以看作场地内的每个点建设成本、用地限

制、景观品质、规范要求的综合评分。

3.3 三维寻径生成空中路径

空间中各点的 Mark 值依据规范要求更新后，在三维寻径网络中完成景观天桥的寻径。设计师手动在程序界面给定路径的起点、终点及中间节点，这一步操作仍在二维平面上完成，但同时必须输入节点的高度信息。随后在三维寻径网络内进行基于 A* 算法的三维寻径，得到场地内的三维路径（图 7）。

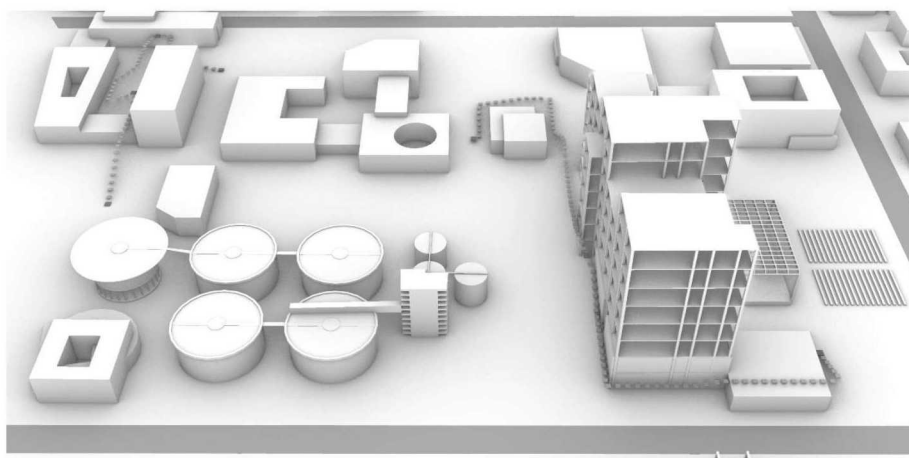


图 7 三维路径节点与寻径

（图片来源：作者编写的程序导出）

3.4 三维路径系统生成工具的完善

在有高度信息的三维空间中，景观天桥不仅仅要满足路径最优，还要实现高度变化的合理性，在天桥的高度变化过程中应有合理的上下坡，不应出现直上直下或交错上升的情况。通过对寻径算法的寻径依据进行调整，在每一个计算当前点的过程中，选取可被加入 Open-set 的点之前，会对其与已经遍历过的点的位置关系进行判定，避免不合理情况，保证寻径结果在建设上的合理性。

在路径生成的过程中，需要不断根据程序生成的结果进行调整；且城市设计中场地内的道路不会仅有一条，需要构建的是互相连接的道路网。因此通过对程序运行逻辑的完善，使用者在同一程序界面内可不断增减寻径节点并完成多次寻径以及道路信息的存储。为保证道路网整体的合理性，每次寻径前会对已有的路径在三维空间中的位置进行判定，并对寻径网络中各点的属性进行更新，从而避免天桥在三维空间中不合理的交叉层叠。

对程序的调整与完善，使得程序生成的路径结果在面对实际设计问题时更趋于合理，且能够生成完整的路径系统。最终的程序不仅仅是在算法层面上的创新性探索，而是一套可为任意城市地块的路径设计提供优化结果的路径生成工具。

4 总结与展望

随着计算机运算能力的不断提高，在三维网络中实现 A* 算法寻径并在城市的尺度上生成三维路径系统是可以实现的。本次研究一方面在将 A* 寻径算法拓展到三维层面上，另一方面将往常的寻径算法中移动代价这一概念进行了拓展，使其变为多个优化目标综合影响的 Mark 值。从而在以 Mark 值这一个条件为依据的寻径过程中，实现了以建设成本、用地限制、景观品质、规范要求等多个条件为目标

的综合优化生成。

从根本上讲，算法工具的开发是为了更好地指导城市设计。但设计问题往往是复杂的，程序基于量化的数据得到的最优路径可能并不能完全满足现实的设计条件。在这种情况下，寻径算法工具所提供的更多的是设计的参考，算法可快速提供一个在可量化的影响因素限制下的最优解，而设计师基于这一结果，可回溯调整程序参数或进行下一步的深化设计，从而实现基于程序快速生成与反馈的高效设计流程。

参考文献

- [1] 李昊. 基于多种寻径算法的路径生成研究与应用 [C]. 全国高等学校建筑学专业教育指导委员会建筑数字技术教学工作委员会. 数字技术·建筑全生命周期——2018 年全国建筑院系建筑数字技术教学与研究学术研讨会论文集 [C]. 全国高等学校建筑学专业教育指导委员会建筑数字技术教学工作委员会: 全国高校建筑学学科专业指导委员会建筑数字技术教学工作委员会, 2018: 6.
- [2] 王建国. 基于人机互动的数字化城市设计——城市设计第四代范型刍议 [J]. 国际城市规划, 2018, 33 (1): 1-6.
- [3] 郭建科, 张仁平, 邹孙楷, 等. Dijkstra 改进算法及其在地理信息系统中的应用 [J]. 计算机系统应用, 2007 (01): 59-62.
- [4] 熊伟, 张仁平, 刘奇韬, 等. A* 算法及其在地理信息系统中的应用 [J]. 计算机系统应用, 2007 (4): 14-17.
- [5] 胡震宇, 唐芄. 高密度历史文化街区更新改造中的道路生成方法研究与应用——以西安碑林历史街区为例 [J]. 当代建筑, 2020 (2): 134-137.