



(12) 发明专利

(10) 授权公告号 CN 116484798 B

(45) 授权公告日 2024.05.03

(21) 申请号 202310465049.3

CN 106017486 A, 2016.10.12

(22) 申请日 2023.04.27

CN 113324550 A, 2021.08.31

(65) 同一申请的已公布的文献号

CN 113341991 A, 2021.09.03

申请公布号 CN 116484798 A

CN 114721370 A, 2022.07.08

(43) 申请公布日 2023.07.25

CN 115268441 A, 2022.11.01

(73) 专利权人 苏州容汇科技有限公司

CN 115454068 A, 2022.12.09

地址 215000 江苏省苏州市吴江区东太湖

US 2005261004 A1, 2005.11.24

生态旅游度假区(太湖新城)东太湖大

WO 2022056770 A1, 2022.03.24

道11588号财智汇商务大厦C幢605

WO 2023045029 A1, 2023.03.30

(72) 发明人 邱柯妮 李元康 崔天昊 秦少平

辛鹏, 马希青. 优化改进A* 和动态窗口法的
机器人路径规划.《组合机床与自动化加工技
术》.2022, (第4期), 全文.

(74) 专利代理机构 北京卓胜佰达知识产权代理

常慧; 王乐; 牛青妍; 王阳; 牛长流. 路径预测
的A*算法优化. 数码世界. 2020, (第02期), 全
文.

有限公司 16026

专利代理师 刘冬梅

(51) Int. Cl.

G06F 30/394 (2020.01)

G06F 30/398 (2020.01)

郭强. 基于改进A*算法的仿生机器鱼全局
路径规划. 西华大学学报(自然科学版). 2011,
(第03期), 全文.

(56) 对比文件

CN 103488816 A, 2014.01.01

审查员 李瑞梅

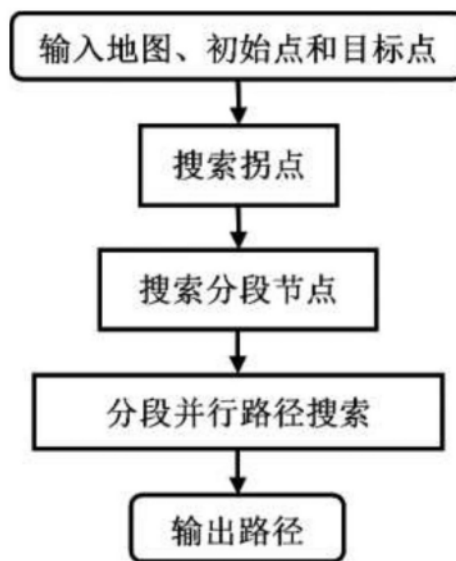
权利要求书1页 说明书4页 附图4页

(54) 发明名称

一种基于分段并行搜索的PCB布线加速优化方法

(57) 摘要

本发明提供一种基于分段并行搜索的PCB布线加速优化方法, 步骤为: (1) 搜索拐点: 输入地图信息, 利用初始点和目标点直线上的障碍物找到符合限定条件的拐点; (2) 搜索分段节点: 通过拐点找到合适的分段节点; (3) 分段并行的A*搜索算法布线: 根据所述分段节点划分布线区域, 在各个布线区域内执行分段并行的A*搜索算法布线。本发明的方法, 将一个较大区域内的搜索问题分解成多个小区域内的并行搜索问题并求解, 并且针对多个区域内障碍物的特征采用适应性的启发引导函数指导布线。



1. 一种基于分段并行搜索的PCB布线加速优化方法,其特征在于,步骤为:

(1) 搜索拐点:输入地图信息,利用初始点和目标点直线上的障碍物找到符合限定条件的拐点;

(2) 搜索分段节点:通过拐点找到合适的分段节点;

(3) 分段并行的A*搜索算法布线:根据所述分段节点划分布线区域,在各个布线区域内执行分段并行的A*搜索算法布线;

所述搜索拐点具体为:通过遍历初始点 $S(x_1, y_1)$ 和目标点 $T(x_2, y_2)$ 线段之间的元素得到第一个遇到障碍物的节点 x ,从节点 x 开始搜索,首先搜索节点垂直方向相邻两个节点,如果两个节点都为障碍物,那么将节点 x 向垂直方向平移,继续搜索该节点上下相邻节点是否都为障碍物;重复此搜索过程,直至将黄色节点 x 垂直移动到障碍物边缘,满足该节点的上下相邻两个节点有且仅有一个障碍物,接下来搜索该节点左右相邻节点,与垂直方向的搜索过程一样,直至该节点左右相邻节点有且仅有一个障碍物,最后根据搜索方向推算出最后需要的点的位置;如果既定目标为 N 段并行,则由此在该线段上选取 $N-1$ 个障碍物,并找到绕过这些障碍物的 $N-1$ 个拐点;

所述(2)中的搜索分段节点具体为:根据给定的 N 段并行规划,形成 $N-1$ 个拐点集;结合起点、终点和拐点集中的关键点,依据其中相邻点构成的矩形构建 N 个布线地图子区域;

为使各段的搜索时间尽可能短并且均衡,定义布线地图复杂度 ε_i 和布线距离比 λ_i :

$\varepsilon_i = N_{S_i-障碍物格数} / N_{S_i-总格数}$,表示布线区域内元器件障碍物与区域面积的比值;

$\lambda_i = |x-x_0|+|y-y_0| / (|x-x_0|+|y-y_0|)$,表示某布线区域内节点 $n(x, y)$ 和初始拐点 $S_i(x_0, y_0)$ 的曼哈顿距离与初始节点 $S_i(x_0, y_0)$ 和目标拐点 $T(x_x, y_y)$ 的曼哈顿距离的比值;

建立代价函数 $P(n)$ 来指导分段节点的决策:

$P(n) = \varepsilon_i + \lambda_i$;搜索分段节点过程中逐一计算节点对应的代价函数 $P(n)$,直至找到代价最小的点作为分段节点;

其中, $N_{S_i-总格数}$ 表示在区域 S_i 中所有方格的数量, $N_{S_i-障碍物格数}$ 表示在区域 S_i 中障碍物方格的数量;

所述分段并行的A*搜索算法布线具体为:找到布线子区域内的分段节点集合 $\{n1, n2, n3, \dots, n(N-1)\}$ 后,将布线搜索过程分成 N 段,用A*搜索算法并行执行,搜索时间为 $\max\{T1, T2, T3, \dots, TN\}$,并且根据布线场景选择合适启发函数指导布线;

所述拐点是布线路径的搜索区域内满足其上下相邻的节点有且仅有一个为障碍物,并且它左右相邻的节点有且仅有一个为障碍物的布线网格点。

一种基于分段并行搜索的PCB布线加速优化方法

技术领域

[0001] 本发明属于电子设计自动化(EDA)领域,具体涉及一种PCB布线加速优化方法。

背景技术

[0002] 印制电路板(Printed Circuit Board,PCB)广泛应用于消费电子、医疗设备、汽车工业、国防、航空航天等与国计、民生密切相关的领域。电子设备中的PCB板就是通过物理导线将各电子元器件按照设计意图进行电气互连,由此实现指定的功能。PCB布线是电子产品设计中至关重要的一环,其目标可以归结为两个层次:(1)首要目标是根据电路的连接关系描述,在满足设计、工艺规则的要求和满足电学性能的条件下,在限定区域(面积、形状、层次等)内100%地完成所需的电气互连;(2)在完成布线的前提下,PCB布线的模板还在于进一步优化布线结果,使所需的PCB面积最小化、PCB完成的功能和性能更佳。

[0003] 随着集成电路工艺的迅速发展及复杂应用不断增多,电子设备的体积越来越小、功能日益复杂,PCB布线难度也持续加大。此时,纯粹人工布线的效率已经远远不能满足设计场景的需求,而基于EDA(Electronic design automation)工具实现的自动布线功能能够大大提升设计效率。尽管现有的PCB EDA工具功能完备,但随着电子元器件管脚排布密度越来越高、PCB尺寸越来越小,PCB自动布线的EDA工具正面临着日益严峻的挑战,主要表现为PCB自动布线的难度越来越大,布线效率越来越不能满足设计周期的需求。

[0004] 实现高效的PCB自动布线需要依赖成熟可靠的布线算法。最早的布线算法为1961年提出的Lees算法,其基本思想是在整个搜索过程中总是向四周同时扩散多个搜索方向,其缺点是搜索过程缓慢、对内存空间的需求很大。之后,人们对Lees算法进行了不断的改进,其中,A*搜索算法是之后最经典也是目前应用最多的算法之一。该算法通过启发式搜索,基于评估函数找到一条最优路径。A*搜索算法比Lees算法速度更快捷高效,但是计算量依然很大。而且,A*搜索算法并没有遍历所有可行解,不同的启发式函数会造成不同的布线结果,糟糕的情况下会造成布线拥堵。

发明内容

[0005] 本发明的目的是至少解决上述问题,并提供一种基于分段并行搜索的PCB布线加速优化方法,将一个较大区域内的搜索问题分解成多个小区域内的并行搜索问题并求解,并且针对多个区域内障碍物的特征采用适应性的启发引导函数指导布线。

[0006] 为实现上述目的,本发明采用如下技术方案:

[0007] (1)拐点的定义和搜索方法:

[0008] 拐点是指布线路径的搜索区域内满足其上下相邻的节点有且仅有一个为障碍物,并且它左右相邻的节点有且仅有一个为障碍物的布线网格点,拐点的意义在于为起始点和目标点之间提供了有效绕过障碍物的线段连接。

[0009] 通过遍历初始点 $S(x_1, y_1)$ 和目标点 $T(x_2, y_2)$ 线段之间的元素可以得到第一个遇到障碍物的节点 x 。我们从节点 x 开始搜索。首先搜索节点垂直方向相邻两个节点。如果两个节

点都为障碍物,那么将节点x向垂直方向平移,继续搜索该节点上下相邻节点是否都为障碍物。重复此搜索过程,直至将黄色节点x垂直移动到障碍物边缘,满足该节点的上下相邻两个节点有且仅有一个障碍物。接下来搜索该节点左右相邻节点,与垂直方向的搜索过程一样,直至该节点左右相邻节点有且仅有一个障碍物,最后根据搜索方向推算出我们最后需要的点的位置(如图2所示)。

[0010] 通过该搜索方法可以从障碍物着眼找到初始点和目标点线段上的拐点。如果既定目标为N段并行,则由此在该线段上选取N-1个障碍物,并找到绕过这些障碍物的N-1个拐点。

[0011] (2)分段节点搜索方法:

[0012] 本发明定义了两个术语:

[0013] ①布线地图复杂度 $\varepsilon \in [0, 1)$,用来描述地图中元器件障碍物的复杂程度,辅助找到合适的分段节点。具体来说,布线地图复杂度表示布线区域内元器件障碍物与区域面积的比值, $\varepsilon_i = N_{S_i\text{-障碍物格数}} / N_{S_i\text{-总格数}} \cdot N_{S_i\text{-总格数}}$ 表示在区域 S_i 中所有方格的数量, $N_{S_i\text{-障碍物格数}}$ 表示在区域 S_i 中障碍物方格的数量。如图3所示,绿色的点为分段节点,分段节点与初始点围成的矩形区域 S_i 。

[0014] ②布线距离比 λ_i ,能在一定程度上反映路径搜索时间的重要指标。具体来说,布线距离比是区域内节点n(x,y)和初始拐点 $S_i(x_0, y_0)$ 的曼哈顿距离与初始节点 $S_i(x_0, y_0)$ 和目标拐点 $T(x_x, y_y)$ 的曼哈顿距离的比值,即 $\lambda_i = |x-x_0|+|y-y_0| / |x_x-x_0|+|y_y-y_0|$ 。

[0015] 综合上述两个关键的考量因素:布线地图复杂度和布线距离比,建立代价函数P(n)来指导分段节点的决策, $P(n) = \varepsilon_i + \lambda_i$ 。

[0016] 由 $\{S, y_1, y_2, y_3, \dots, T\}$ 构建布线地图子区域,分别以拐点集 $\{y_1, y_2, y_3, \dots\}$ 中的节点为父节点搜索其周围相邻节点,逐一计算每个子区域对应的代价函数值,直至找到 $\max\{\text{子区域代价函数值集合}\}$ 最小的各子区域节点集合 $\{n_1, n_2, n_3, \dots, n(N-1)\}$,即为分段节点。

[0017] (3)分段并行搜索:

[0018] 通过上述方法找到布线子区域内的分段节点集合 $\{n_1, n_2, n_3, \dots, n(N-1)\}$ 后,将布线搜索过程分成N段,用A*搜索算法并行执行,搜索时间为 $\max\{T_1, T_2, T_3, \dots, T_N\}$ 。并且根据布线场景选择合适启发函数指导布线。

[0019] 本发明的技术效果:

[0020] 1、通过拐点构建分段节点,并提出建立代价函数P(n)来指导分段节点的决策,使得各段的搜索时间尽可能短并且均衡;

[0021] 2、实现了线长与Lees算法和A*搜索算法基本相同的情况下,减少搜索空间、加速搜索速度和优化布线效果,并由此减少搜索成本、提高布线效率。

附图说明

[0022] 附图大体上通过举例而不是限制的方式示出各种实施例,并且与说明书以及权利要求书一起用于对所发明的实施例进行说明。在适当的时候,在所有附图中使用相同的附图标记指代同一或相似的部分。这样的实施例是例证性的,而并非旨在作为本装置或方法的穷尽或排他实施例。

- [0023] 图1示出了本发明的分段并行搜索算法示意图；
- [0024] 图2示出了本发明的拐点搜索示例图；
- [0025] 图3示出了本发明的二分阶段并行搜索示例图；
- [0026] 图4示出了本发明的二分阶段节点搜索示例图；
- [0027] 图5示出了本发明的搜索分段节点遇到障碍物时的示例图；
- [0028] 图6示出了本发明的A*搜索(a)(c)和优化后的分段并行搜索(b)(d)的结果对比图。

具体实施方式

[0029] 需要说明的是,在不冲突的情况下,本申请中的实施例及实施例中的特征可以相互组合。下面将参考附图并结合实施例来详细说明本申请。

[0030] 图2展示了一种针对起始点和目标点之间存在一个障碍物的拐点搜索过程(本申请不限于起始点和目标点只存在一个障碍物的情况),通过遍历初始点 $S(x_1, y_1)$ 和目标点 $T(x_2, y_2)$ 线段之间的元素可以得到第一个遇到障碍物的节点 x 。从节点 x 开始搜索。首先搜索节点垂直方向相邻两个节点。如果两个节点都为障碍物,那么将节点 x 向垂直方向平移,继续搜索该节点上下相邻节点是否都为障碍物。重复此搜索过程,直至将节点 x 垂直移动到障碍物边缘,满足该节点的上下相邻两个节点有且仅有一个障碍物。接下来搜索该节点左右相邻节点,与垂直方向的搜索过程一样,直至该节点左右相邻节点有且仅有一个障碍物,最后根据搜索方向推算出我们最后需要的点的位置。

[0031] 通过图2演示的方法可以从障碍物着眼找到初始点和目标点线段上的拐点。如果既定目标为 N 段并行,则由此在该线段上选取 $N-1$ 个障碍物,并找到绕过这些障碍物的 $N-1$ 个拐点。其搜索过程可描述如下:

[0032] (1) 输入地图 $mazeMap$,初始点 S ,目标节点 T 。

[0033] (2) 从初始点开始,遍历 $mazeMap$ 中初始点和目标点线段上元素,直到遇到障碍物节点 x 。

[0034] (3) 搜索节点 x 垂直相邻节点,判断是否为障碍物。

[0035] (4) 如果垂直相邻节点都为障碍物,则将节点垂直平移,然后重复(3)否则进行(5)。

[0036] (5) 如果垂直相邻节点都为障碍物,则将节点水平平移,然后重复(4)否则输出节点 x 继续水平、垂直平移一个单位的节点,即拐点 y 。

[0037] 搜索到拐点后,根据拐点的位置搜索分段节点的方法:

[0038] 为了更加合理评判分段节点的位置,本发明定义了两个术语:

[0039] ① 布线地图复杂度 $\epsilon \in [0, 1)$ 的概念,表示布线区域内元器件障碍物与区域面积的比值。

[0040] $\epsilon_i = N_{S_i-障碍物格数} / N_{S_i-总格数}$

[0041] S_i 表示分段节点和初始节点横纵坐标所包含的矩形的区域。 $N_{S_i-总格数}$ 表示在区域 S_i 中所有方格的数量, $N_{S_i-障碍物格数}$ 表示在区域 S_i 中障碍物方格的数量。如图3所示,分段节点与初始点围成的矩形区域 S_i 。 $N_{S_i-总格数}=36, N_{S_i-障碍物格数}=2, \epsilon_i=1/18$ 。

[0042] ②布线距离比 λ_i 的概念,表示某布线区域内节点 $n(x, y)$ 和初始拐点 $S_i(x_0, y_0)$ 的曼哈顿距离与初始节点 $S_i(x_0, y_0)$ 和目标拐点 $T(x_x, y_y)$ 的曼哈顿距离的比值。

[0043] $\lambda_i = |x-x_0|+|y-y_0|/|x_x-x_0|+|y_y-y_0|$

[0044] 综合上述两个关键的考量因素:布线地图复杂度和布线距离比,建立代价函数 $P(n)$ 来指导分段节点的决策。

[0045] $P(n) = \varepsilon_i + \lambda_i$

[0046] 图4为本发明提供的一种二分节点搜索示例图(本申请不限于二分节点的情况),从拐点开始,沿着初始点S和目标点T直线的方向搜索。如图4,首先遍历目标节点,并逐一计算这些节点对应的代价函数值,直至找到 $\max\{P_s(n), P_T(n)\}$ 为最小的节点n,即为其二分节点。

[0047] 分段节点搜索方法具体描述如下:

[0048] (1) 输入布线地图 $mazeMap$, 拐点集 $\{y_1, y_2, y_3, \dots\}$, 初始点S, 目标点T

[0049] (2) 由 $\{S, y_1, y_2, y_3, \dots, T\}$ 构建布线地图子区域;

[0050] (3) 分别以拐点集 $\{y_1, y_2, y_3, \dots\}$ 中的节点为父节点搜索其周围相邻节点,逐一计算每个子区域对应的代价函数值;

[0051] (4) 直至找到 $\max\{\text{子区域代价函数合计}\}$;

[0052] (5) 最小的各子区域节点集合 $\{n_1, n_2, n_3, \dots, n(N-1)\}$, 即为分段节点。

[0053] 在搜索的过程中如果遇到障碍物,如图5所示,搜索拐点周围的节点m恰好经过障碍物,那么我们会根据节点m和算法1找到该障碍物上的拐点。然后根据这个新的拐点,利用算法2找合适的分段节点。

[0054] 图6是A*搜索算法和本发明提供的一种优化后的分段并行搜索算法的结果对比图,分段并行搜索算法所使用的第一段搜索的启发函数是切比雪夫距离,第二段搜索的启发函数是曼哈顿距离,根据实验结果可以看到使用这两个启发函数使布线通道腾出更多空间用于其他网络布线。

[0055] 尽管本发明的实施方案已公开如上,但其并不仅仅限于说明书和实施方式中所列运用,它完全可以被适用于各种适合本发明的领域,对于熟悉本领域的人员而言,可容易地实现另外的修改,因此在不背离权利要求及等同范围所限定的一般概念下,本发明并不限于特定的细节和这里所示出与描述的图例。

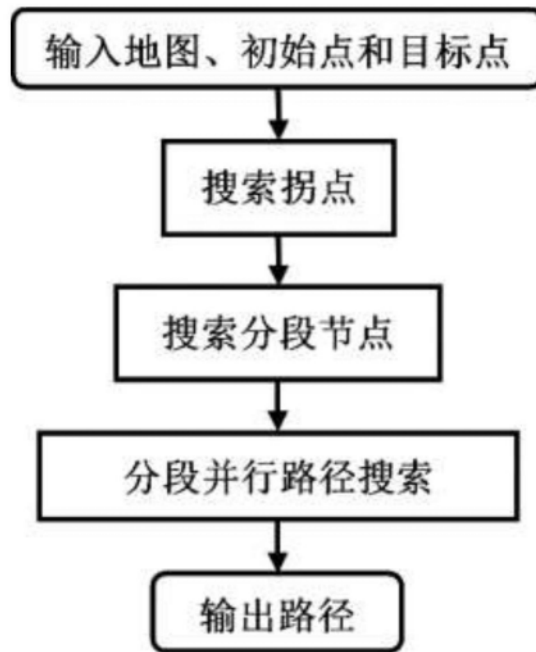


图1

■ 障碍物 X 当前搜索点 y 拐点 S 初始点 T 目标点

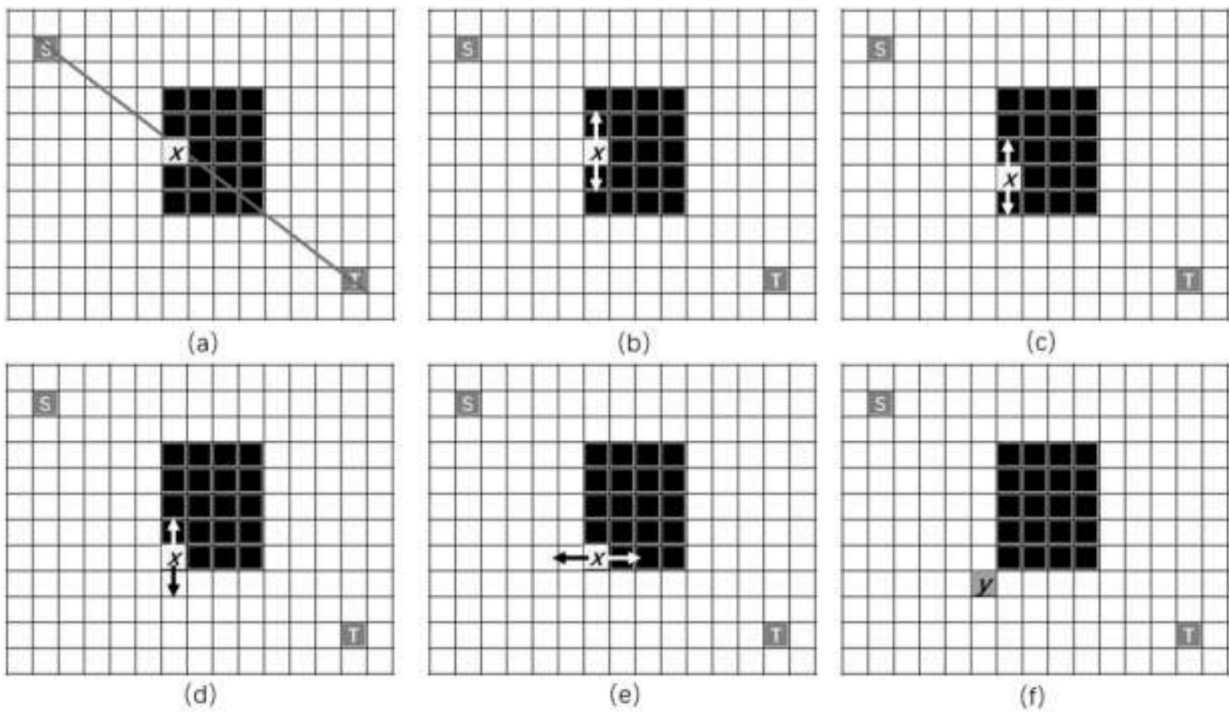


图2

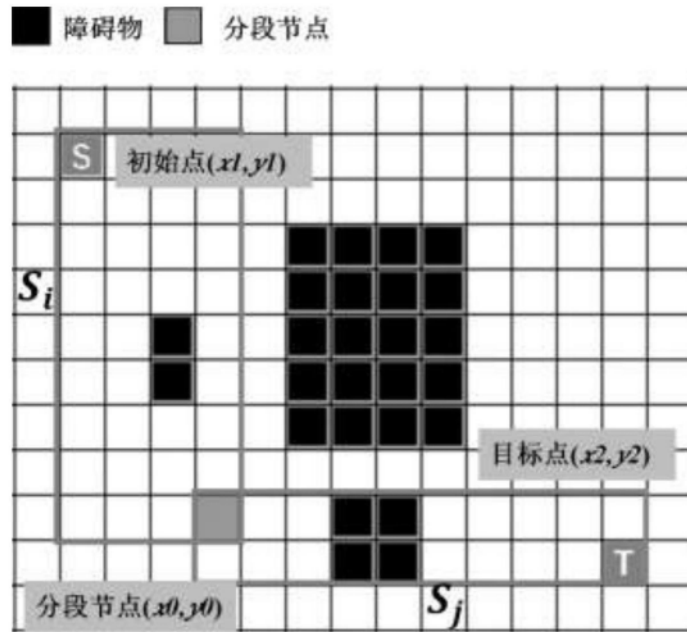


图3

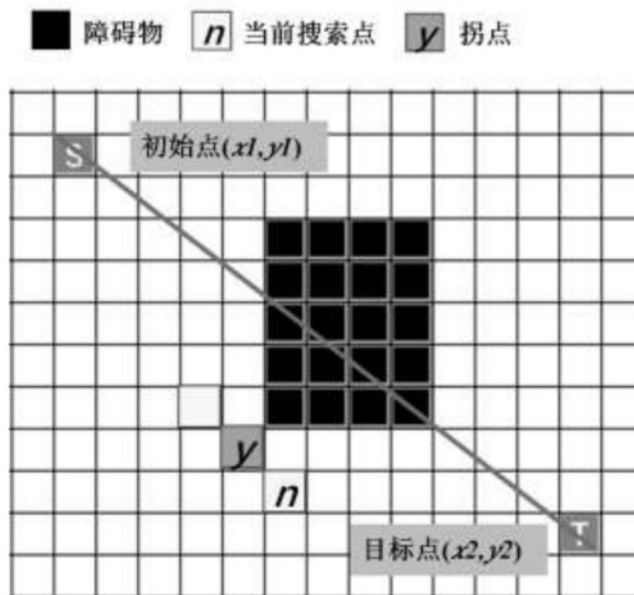


图4

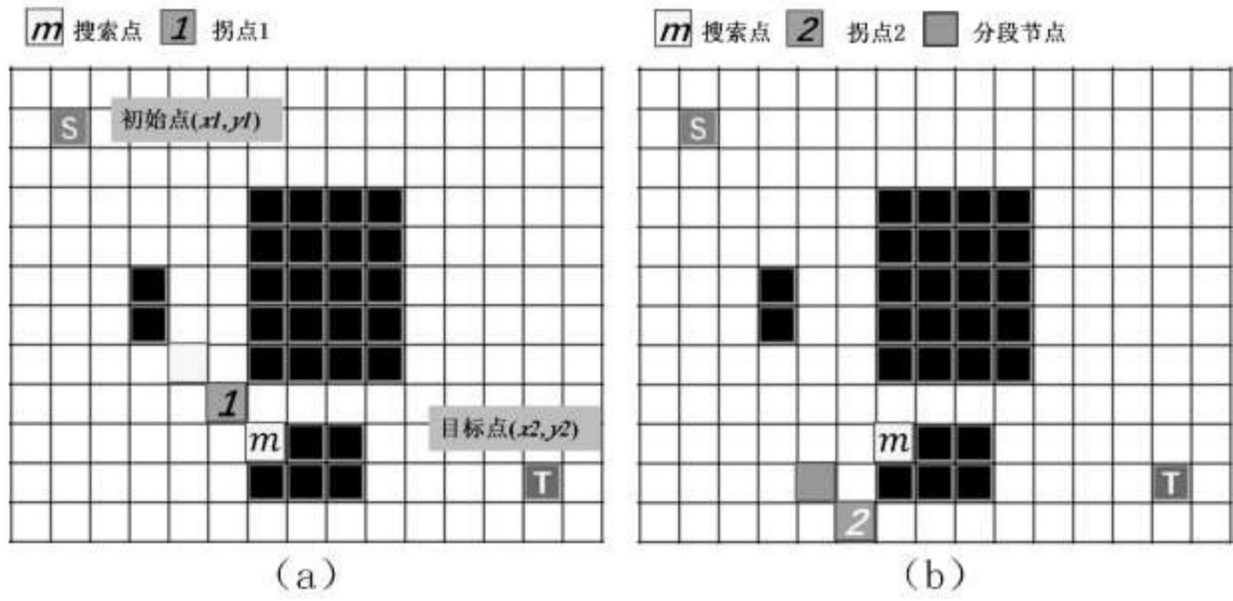


图5

