

# Propuesta para reducir espacio de búsqueda en entornos virtuales

Víctor Tomás Tomás Mariano, Felipe de Jesús Núñez Cárdenas,  
Efraín Andrade Hernández

Universidad Autónoma del Estado de Hidalgo: Escuela Superior de Huejutla,  
Huejutla de Reyes, Hidalgo, México

{victor\_tomasm, felipe.huejutla, andrade\_a\_h}@hotmail.com

**Resumen.** Se propone un proceso de etiquetado de un área de búsqueda tipo rejilla, con la cual se identifican celdas en las que se puede navegar de manera segura. Con las etiquetas se identifican las celdas que están libres de colisiones además permite reducir el número de celdas a recorrer durante el proceso de encontrar una trayectoria o ruta solución que conecta a dos celdas distintas dentro del entorno virtual. Los entornos virtuales generados son de distintos tamaños con salida en formato binario lo que facilita hacer una representación gráfica en 2D.

**Palabras clave:** Navegación, entorno virtual, espacio de búsqueda.

## Proposal for Reduction of the Search Space in Virtual Environments

**Abstract.** A process of labeling a search of square type grid is proposed. With labels squares that are free of collisions are identified also reduces the number of squares to travel during the process of finding a solution path or route connects starting point to the goal within the virtual environment. The generated environments are different sizes with output in binary format which facilitates make a graphical representation 2D.

**Keywords:** Navigation, virtual environment, search space.

### 1. Introducción

El problema de búsqueda es un tema en la que en los últimos años ha tenido bastante auge, principalmente en el área de robótica móvil, en este tipo de problemas es necesario definir en primera instancia el tipo de búsqueda que se pretende realizar, básicamente hay dos tipos: búsqueda informada y búsqueda no informada.

La búsqueda no informada o a ciegas, realiza un recorrido del espacio de búsqueda de una forma sistemática, pero sin tener en cuenta ningún tipo de información sobre el dominio del problema que se está resolviendo [5]. Por lo general utiliza un árbol de búsqueda para representar los distintos estados en los que se puede navegar. En ocasiones este árbol es muy grande que es difícil representarlo y el costo computacional es bastante alto.

Por otro lado, la búsqueda informada admite tener un mecanismo que permite dirigir la búsqueda hacia zonas más prometedoras, de modo que se puede encontrar una solución sin necesidad de visitar tantos nodos de un árbol como los que en general requiere una estrategia de búsqueda a ciegas. Este tipo de búsqueda utilizan heurísticas que son criterios, reglas o métodos que ayudan a decidir cuál es la mejor alternativa entre varias posibles para alcanzar un determinado objetivo [5].

Una parte fundamental para decidir el tipo de búsqueda a realizar, depende del entorno en el que se va a modelar el problema, es decir, el espacio de búsqueda.

Entonces, se puede hablar de espacios de búsqueda continuos y discretos. En este trabajo se modela un espacio de búsqueda discreto que consiste en generar un área rectangular de dimensiones de  $m$  por  $n$  en la que hay celdas libres y celdas ocupadas (obstáculos). Así mismo, se propone el uso de un conjunto de máscaras para etiquetar el espacio de búsqueda de manera a priori con el objetivo de reducir el número de celdas a explorar en el entorno durante el recorrido para encontrar una solución, otorgando la posibilidad de que el espacio etiquetado esté libre de colisiones.

El artículo está organizado de la siguiente manera, en la sección dos abarca el problema que implica hacer búsquedas y planificación de trayectorias; en la sección tres de los entornos tipo rejilla; en la sección cuatro del proceso de etiquetado propuesto; en la sección cinco de los resultados obtenidos, conclusiones y del trabajo futuro.

## **2. El problema de hacer búsquedas**

El problema de realizar búsquedas se ha analizado en varios trabajos de investigación [3, 4, 7, 8], principalmente en robótica móvil, y la idea general consiste en que un robot móvil debe ser capaz de encontrar una trayectoria o camino que lo conduzca de un punto de partida a otro punto objetivo, dentro de un entorno virtual o real, este robot es capaz de censar en todo momento el entorno para conocer su posición actual, en base a las mediciones censadas debe ser capaz de tomar decisiones para moverse en la dirección que lo acerque más al objetivo perseguido.

En el proceso de encontrar la ruta, el robot debe evitar obstáculos, y de preferencia se desea que el robot se aleje de esos obstáculos de tal manera que la trayectoria que genera esté libre de colisiones [4, 7].

La búsqueda puede llevarse a cabo en dos tipos de entornos: entornos conocidos y entornos desconocidos. En la búsqueda en entornos conocidos, se tiene información a priori de la posición actual u origen, y también de la posición en la que se encuentra el punto final u objetivo perseguido, esto da la posibilidad de tener orientación hacia dónde debe moverse el robot dentro del entorno [7, 8]. Por lo regular, el entorno a explorar permanece fijo. Lo que se pretende es encontrar la ruta o camino que conduzca al punto final.

Por otro lado, la búsqueda en entornos desconocidos, consiste en que el robot solo conoce la posición actual dentro del entorno, el objetivo perseguido también es alcanzar un punto meta, sin embargo, en estos entornos, el robot desconoce en donde está ubicado tal punto, además el entorno es dinámico o cambiante.

Ambos tipos de búsqueda se pueden modelar en espacios de búsqueda discretos o continuos, según sea el problema a resolver, en el caso discreto se tiene un entorno en la que el área de búsqueda es una rejilla rectangular (*pathfinder grids*) con dimensiones definidas en términos de filas y columnas de  $n$  por  $m$  respectivamente [1].

## 2.1. Planificación de trayectorias

La planificación se define como la búsqueda de una ruta libre de obstáculos desde una posición inicial hasta otra posición final a través del entorno de trabajo del robot móvil. Esta ruta se realiza mediante el uso de la información que se posee del entorno actualmente. El caso más simple consiste en considerar un entorno conocido en su totalidad y estático. Con estas apreciaciones resulta factible la aplicación de un algoritmo de búsqueda en grafos, que emplee cierta función de coste para la obtención de la ruta. De esta forma, la finalidad de la planificación consiste en localizar una ruta libre de obstáculos, definida como una secuencia de ordenada de objetivos, que conduzca al robot desde la posición inicial hasta la final [4,5].

Modelar el entorno de búsqueda como un grafo, conlleva la necesidad de asociar cada nodo o vértice del grafo a una posición  $p[i, j]$  dentro del entorno [9]. La arista que va de  $v_i$  a  $v_j$  que conecta al vértice  $i$  con el vértice  $j$ , ambos nodos son adyacentes, es decir, están conectados, existe un camino que va desde el nodo  $I$  a otro nodo  $J$ , por lo que se dice que es alcanzable y que hay un camino entre ambos nodos.

Esta forma de representar el espacio de búsqueda, consiste en dos etapas, la primera, es construir el grafo asociado al entorno, según sea el caso, el grafo puede ser dirigido, no dirigido o mixto, depende de las restricciones de movilidad en la exploración, este grafo asociado debe ser lo más real o parecido al entorno. En la segunda etapa, se designa un vértice inicial como punto de partida y vértice final como punto que se desea alcanzar, en esta etapa se aplican algoritmos para recorrido de grafos que dan buenos resultados para encontrar el camino entre dos vértices, entre los que se encuentran en la literatura están el algoritmo de búsqueda en profundidad, algoritmo de búsqueda basado en amplitud, y el algoritmo de Dijkstra. De los tres algoritmos, el de Dijkstra garantiza encontrar una ruta y además es la más corta entre dos nodos.

Otra manera de modelar el problema de búsqueda consiste en manejar el entorno como un conjunto de celdas, en la que hay celdas libres, estas celdas son por las que se puede mover el explorador, están conectadas, por lo regular se usa una conectividad cuatro vecinos u ocho vecinos. Con esta representación, se puede elegir una celda libre  $O [i, j]$  inicial como punto origen y otra celda  $F [i, j]$  como celda final, estas celdas están conectadas, es decir, hay una o varias rutas que permiten llevar de la celda origen a la celda final. El problema consiste en encontrar el camino, conjunto de celdas libres que minimicen el costo de ir de un punto  $E$  a otro punto  $F$ . El conjunto total de celdas libres, para el entorno que se analiza en el presente trabajo, tiende a incrementarse, dependiendo de la configuración inicial del entorno que se desea crear. Las celdas ocupadas, por el contrario, representan a los obstáculos, y éstas se deben evitar durante el proceso de exploración. Entre los algoritmos que se pueden aplicar a este tipo de

espacios de búsqueda están el A\*, que ofrece excelentes resultados para entornos tipo rejilla, otro algoritmo, es una variante del primero, el D\*[1].

### 3. Entorno tipo rejilla

Considere un entorno con una vía para transitarlo. Este entorno, está representado como un arreglo bidimensional con valores “1” y “0”, obstáculos y celdas libres respectivamente. Cada celda tiene vecinos adyacentes en cuatro direcciones Norte, Sur, Este y Oeste.

La forma matricial en “1” y “0”, permite obtener entornos con varias vías, en la figura 1.a se muestra un entorno con dos vías. En la figura 1.b se representa al mismo entorno, pero con distinto número de vías, lo que incrementa el tamaño del entorno.

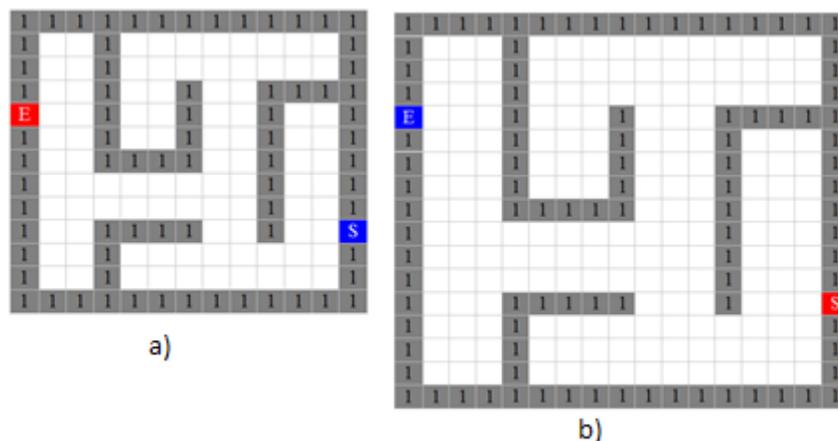


Fig. 1. Entornos ampliados a) con dos vías y b) con tres vías.

Como se puede observar en la figura 2.b, el entorno tiene tres vías en su interior, de manera general, se puede incrementar a  $n$  vías, donde  $n$  es la cantidad de vías o pasillos que se desea tenga el entorno, algo análogo sería como tener autopistas de dos carriles, tres carriles, etc., es más cómodo para el que maneja por un carril de dos vías que el que maneja en el carril de una sola vía, lo que se debe entender es que a mayor número de vías para navegar por el entorno, involucra tener más movilidad y en términos computacionales tener más control del recorrido del explorador con respecto a la posición que mantiene dentro del entorno. Para profundizar en cómo construir este tipo de entornos, se recomienda consultar [2, 3, 10].

### 4. Propuesta de etiquetado

El problema que implica encontrar una solución del entorno ampliado es más difícil, ya que hay varios caminos que se pueden recorrer una vez abandonado la celda inicial marcada como entrada, otro problema es, que el explorador puede quedar atrapado en

una sección muy pequeña del entorno, por lo que de alguna manera se debe “recordar” las celdas ya visitadas o transitadas.

Para etiquetar el entorno, se propone considerar las posiciones [i, j] dentro de la matriz en donde se tiene una conectividad cuatro vecinos, en función de ello, asignar una etiqueta a la celda actual para examinar las diferentes celdas adyacentes a las que se puede mover. En la figura 2 se ha indicado con una “s” las celdas que tienen probabilidad de choque *cero*: el explorador tiene libertad de moverse en cualquiera de las cuatro direcciones posibles; en las orillas, celdas marcadas con una “o”, tiene probabilidad de choque del 25%: el explorador puede moverse en tres de las direcciones posibles; en las esquinas, celdas marcadas con una “e”, aumenta la probabilidad de choque en un 50%; el explorador puede moverse en dos de las opciones posibles.

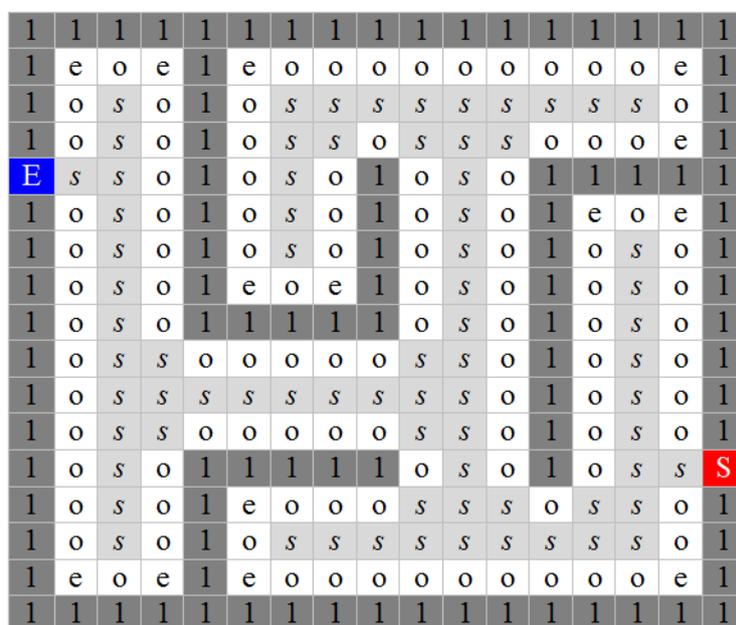


Fig. 2. Entorno etiquetado.

En el espacio de búsqueda de la figura 2, celdas etiquetadas, se observa que se puede reducir el número de celdas libres a explorar una vez abandonado el punto inicial E, es deseable que el recorrido se realice por las celdas etiquetadas como “s”, esto hace que la navegación en el entorno esté libre de colisiones. Otro punto a observar es que las celdas etiquetadas como “s”, están conectadas, es decir, se puede elegir cualquier celda como punto inicial y otra celda como punto final, por lo que se puede ahora encontrar un conjunto de celdas “s” que represente la trayectoria solución entre ambas celdas.

#### 4.1. Etiquetado

El proceso de realizar el etiquetado en el entorno ampliado implica hacer un recorrido del espacio bidimensional de manera a priori, en la que se usan mascarar de tres por

tres con conectividad cuatro vecinos, y con éstas se valida la posición central  $p [i, j]$  de la máscara respectiva y validar las celdas adyacentes a los que se puede llegar. Cada máscara permite etiquetar de manera única a una celda.

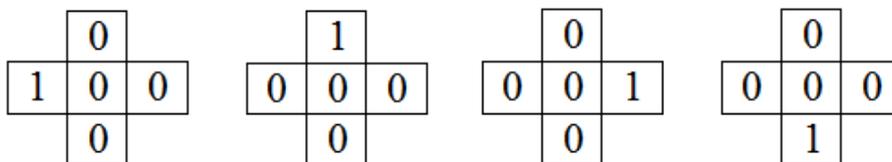


Fig. 3. Máscaras para etiquetar celdas tipo “o”.

En la figura 3, se muestra las máscaras que se utilizan para etiquetar celdas tipo “o”, estas celdas tienen tres celdas libres con una probabilidad de choque del 25%.

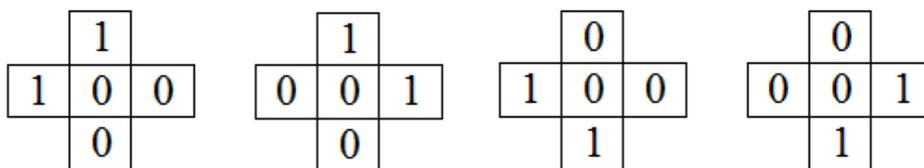


Fig. 4. Máscaras para etiquetar celdas tipo “e”.

En la figura 4, se muestran las máscaras que se utilizan para etiquetar las celdas tipo “e”, estas celdas tienen dos celdas libres con una probabilidad de choque del 50%.

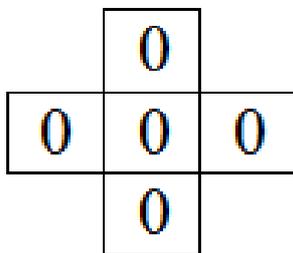


Fig. 5. Máscara para etiquetar celdas tipo “s”.

En la figura 5, se muestra la máscara que se utiliza para etiquetar celdas tipo “s”, estas celdas son las que están libres de obstáculos, tienen cuatro direcciones de movilidad.

#### 4.2. Búsqueda de ruta solución en espacio etiquetado

Para encontrar la trayectoria que conecta a la celda origen E y la celda final S, se ha utilizado el modelo de representación de grafos, en este caso, el total de celdas tipo “s” se le asocia un conjunto de nodos junto con la posición respectiva del entorno.

Esta representación hace suponer que se tienen tantos nodos, como celdas “s” haya en el conjunto de celdas etiquetadas en el entorno, sin embargo, se han discriminado

algunas celdas “s” con la finalidad de construir un grafo menos denso, en la que se aplica el algoritmo de Dijkstra entre dos nodos para encontrar el camino más corto y así visualizar la solución [5, 6].

### 5. Resultados obtenidos

El entorno a explorar depende del número de vías y por lo tanto influye en el tamaño del entorno virtual. El objetivo se cumplió al permitir reducir el número de posibles celdas a explorar al recorrer el entorno teniendo mayor grado de libertad en los movimientos realizados.

Las salidas que arroja el sistema se guardan en archivos TXT para facilitar la lectura y representación gráfica de los entornos.



Fig. 6. Entorno con 7 vías.

En la figura 6 se muestra un archivo binario que representa a un entorno con siete vías en su interior, como se aprecia, no es posible hacer un hurgamiento visual de manera cómoda. Sin embargo, se genera un entorno virtual con salida binaria completamente etiquetado.

En la figura 7 se tiene un archivo binario que representa a un entorno con cinco vías de libertad. De los resultados obtenidos, en el 100% se etiqueta el espacio de búsqueda así como lograr reducir el espacio de búsqueda.



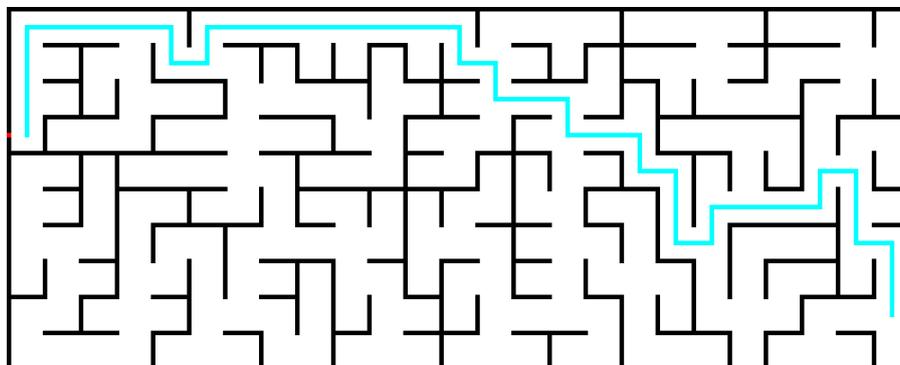


Fig. 8. Entorno con solución indicada.

La ruta solución del espacio de búsqueda está dentro del conjunto de celdas libres etiquetadas como “s”.

## 6. Trabajo futuro

De la propuesta realizada, se observa que se etiquetan de manera única las celdas que están libres de obstáculos, lo que brinda la posibilidad de explorar el entorno virtual conectado. Estas celdas reducen el espacio de exploración permitiendo así obtener un camino libre de colisiones. El espacio, se modela de manera discreta en 2D.

Entre las mejoras que se pueden hacer, es aplicar técnicas de morfología de imágenes binarias, a partir del proceso final de etiquetado, estas celdas se pueden erosionar para reducir aún más el espacio de búsqueda.

La ruta solución se puede mejorar aplicando algún tipo de búsqueda heurística, por ejemplo, el algoritmo A\* o D\*, lo que permitiría obtener una ruta solución más óptima con movimientos en diagonal.

## Referencias

1. Marin, M.J.A., Zamora, I.M.A., Martínez, B.H.: Planificación de trayectorias en un mapa de celdillas difusas. Departamento de ingeniería de información y las comunicaciones, Universidad de Murcia (2001)
2. Martínez, N.C.: Simulación en entornos con robots manipuladores móviles. Escuela Técnica Superior de Ingeniería Informática, Universidad de Valencia (2015)
3. Mera, P.C., Cobano, S.J.A.: Métodos reactivos basados en campos potenciales para UAVs. Departamento Ingeniería de sistemas y Automática, Escuela Superior de Ingeniería, Universidad de Sevilla (2015)
4. Muñoz, M.D.V.: Planeación de trayectorias para robots móviles. Tesis Doctoral, Departamento de Ingeniería de Sistemas y Automática, Universidad de Málaga, (1995)
5. Palma, M.J.T., Marin, M.R.: Inteligencia Artificial: Técnicas, métodos y aplicaciones. MagrawHill (2008)
6. Russel, S., Norvig, P.: Solving Problems by Searching. Chapter 3 and 4, Artificial Intelligences (2010)

*Víctor Tomás Tomás Mariano, Felipe de Jesús Núñez Cárdenas, Efraín Andrade Hernández*

7. Yandun, A., Sotomayor, N.: Planeación de y seguimiento de trayectorias por un robot móvil. Escuela Politécnica Nacional, Quito, Ecuador (2005)
8. Zalamo, G., Gonzalez, A., Delgado, J., Bermejo, J.: Construcción de Mapas de entorno y estimación de la posición de un robot móvil. Departamento de Ingeniería y Sistemas y automático, Universidad de Valladolid (2006)
9. Ahlquist, J., Novak, J.: Game Development Essentials: Game Artificial Intelligence, Capítulo 6 (2008)
10. Tomas, M.V.T., Pozas, C.M., Hernández, C.J.: Propuesta para la generación de laberintos ampliados en 2D. Simposio Iberoamericano Multidisciplinario de Ciencias e Ingenierías (SIMCI), Zempoala, Hidalgo (2011)