

Universidad Tecnológica Centroamericana

UNITEC



Mine Sweeper

Estudiante:

Joel Alexander Pineda Pastrana

11211362

Análisis de Algoritmos, Sección: 313

Catedrático: Jorge García

Tegucigalpa, M.D.C.,

25 de septiembre de 2014

Contenido

Introducción	3
Marco Teórico.....	4
Implementación.....	5
Resultados	7
Conclusiones.....	7
Anexos	8

Introducción

Mine Sweeper o Busca-Minas fue uno de los juegos más famosos en los inicios de la computación especialmente en los sistemas operativos de Microsoft. Fue desde entonces que se ha tratado de encontrar una solución a este juego, que en realidad se han hecho avances para resolverlo. Este problema pertenece a la familia de los NP-Complejos. Y para resolver uno de ellos dice la teoría que se podrían resolver todos los NP-Complejos. Múltiples caminos se han seguido para ello. En esta presentación se hablara de Álgebra Lineal para resolver matrices que aseguran ciertos resultados parciales para encontrar todas las minas del juego.

El juego consiste en que se proporciona una tabla de minas de n columnas por m filas, las cuales contienen casillas. Cada casilla está oculta al inicio, que puede contener Minas, números que indican cuántas minas hay a su alrededor, o puede estar vacía. El jugador va descubriendo casilla por casilla de tal manera que marque donde están todas las minas con banderas o describir todas las demás casillas que no son minas.



El objetivo de este proyecto es el de analizar este juego y diseñar e implementar un algoritmo que encuentre la solución a este juego de un tamaño de tablero cualquiera. Por simple que parezca, no se ha encontrado un algoritmo exacto, pero se han encontrado métodos para resolver de manera parcial este juego.

Marco Teórico

Mine Sweeper o Buscaminas no ha sido solamente un juego para pasar el tiempo, sino que se ha convertido en uno de los temas más pronunciados en la Teoría de la complejidad computacional, introducido por Richard Kaye que menciona que este problema de buscaminas pertenece a la conocida familia de los NP-Completo.

Los problemas NP-Completo tienen dos características importantes, una de ellas es que son computacionalmente máquinas de Turing No deterministas en tiempo Polinomial y la otra característica es que cualquier problema que pertenece a NP-Completo, puede reducirse al problema NP-Completo en cuestión, es decir, si resuelve uno de ellos, resuelve automáticamente todos los problemas NP-Completo que se reducen a este problema.

De acuerdo con Kaye el Problema del Buscaminas es:

“Dado una cuadrícula particularmente marcada con Números y/o Minas y algunos cuadros en blanco, es determinar si hay pares de minas de tal manera que allí existan la cantidad de minas que se mira en los cuadros adyacentes”.

Obviamente hay instancias de buscaminas que pueden responder si o no para la mayoría de los problemas estudiados. En general los problemas de este tipo, no solamente utilizan la lógica para resolverlos, sino que predicen un siguiente movimiento. Pero como se menciona en todos los papers acerca de esto, es muy difícil encontrar un algoritmo implementado que pueda resolver esto.

Este problema es NP-Completo por una sencilla razón, es que puede ser verificado en tiempo Polinomial. Al encontrar una mina nos damos cuenta que el jugador perdió.

Solamente con recorrer la tabla nos damos cuenta donde están las minas. Pero la dificultad está en resolver dicho problema.

Hay varias estrategias para resolver de manera parcial este problema, muchos son de personas que se han dedicado a ellos, por el ejemplo CSPStrategy o utilizar Algebra lineal para resolver matrices, etc. Son métodos que lo hacen más rápido que probar todas las posibilidades pero aún no lo suficiente. Los enlaces para algunos de ellos están dentro de la bibliografía para que se puedan ver con mayor detalle.

Implementación

La implementación de este Solucionador del Buscaminas está basado en algebra lineal, para que de una forma matemática se puedan generar casos y sean resueltos.



Por ejemplo este caso de tener estas minas ocultas pero sabemos los números adyacentes a estas, podemos obtener una solución vía ecuaciones y resolverlas o una matriz de valores y utilizar el método de Gauss para resolverlo. Las ecuaciones resultantes serian de esta manera:

$$\begin{aligned}x_1 + x_2 &= 1 \\x_1 + x_2 + x_3 &= 2 \\x_2 + x_3 + x_4 &= 2 \\x_3 + x_4 + x_5 &= 2 \\x_4 + x_5 &= 1\end{aligned}$$

De una forma matricial estas ecuaciones pueden verse como:

$$\begin{aligned}1x_1 + 1x_2 + 0x_3 + 0x_4 + 0x_5 &= 1 \\1x_1 + 1x_2 + 1x_3 + 0x_4 + 0x_5 &= 2 \\0x_1 + 1x_2 + 1x_3 + 1x_4 + 0x_5 &= 2 \\0x_1 + 0x_2 + 1x_3 + 1x_4 + 1x_5 &= 2 \\0x_1 + 0x_2 + 0x_3 + 1x_4 + 1x_5 &= 1\end{aligned} \quad \vec{x} = \left[\begin{array}{ccccc|c} 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 0 & 2 \\ 0 & 1 & 1 & 1 & 0 & 2 \\ 0 & 0 & 1 & 1 & 1 & 2 \\ 0 & 0 & 0 & 1 & 1 & 1 \end{array} \right]$$

Resolver una matriz se vuelve más fácil si se tienen las herramientas correctas, en mi caso, me apoye de una librería llamada Jama, que es una librería de Algebra lineal y matrices para Java, que permite resolver matrices cuadradas por el método de Gauss y además de muchas funcionalidades.

<http://math.nist.gov/javanumerics/jama/>

La matriz resuelta queda de la siguiente manera:

$$\vec{x} = \left[\begin{array}{ccccc|c} 1 & 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 1 & 0 & 2 \\ 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{array} \right]$$

Se puede observar que tenemos una solución para la tercera línea, la cual dice que es 1,(1 es mina, 0 no es mina), entonces para este punto nos dimos cuenta que en casilla no hay una mina, y así podemos ir formando muchos casos dependiendo de la forma en que se presenten.

El código de este proyecto se encuentra en el siguiente link:

<https://github.com/Dark1024/Minesweeper>

Resultados

Los resultados de este proyecto no fueron los esperados, primero pues porque resolver este problema del Mine Sweeper no se ha creado un algoritmo que funcione 100%, esto se volvió tedioso por el hecho de el algoritmo fallaba en muchos casos. Se puede observar en la imagen 4 de los anexos que si existía una mina en una de las esquinas se para la ejecución porque el jugador había perdido.

No se encontró un algoritmo completo que resolviera este problema y mucho más difícil fue encontrar una implementación para dichos algoritmos. Por lo tanto no se pudo comparar ninguno de ellos.

Por otro lado, el proyecto no fue concluido debido a malas implementaciones y falta de herramientas para resolverlo completamente.

Conclusiones

Sin duda que resolver el problema de Mine Sweeper que es uno de los problemas de la computación que son complejos, fue un gran reto que no se pudo completar. Este problema con su complejidad de NP-Completo, resulta ser un problema del millón de dólares. Claro por supuesto no se pedía resolver completamente este problema con un algoritmo pero si tratar de resolverlo.

De cualquier forma adjunto los enlaces de las personas que si lograron implementar estos, de una forma especial y con mucho trabajo:

<http://quantum-p.livejournal.com/19616.html>

<http://robertmassaioli.wordpress.com/2013/01/12/solving-minesweeper-with-matrices/>

Anexos

Imagen 1

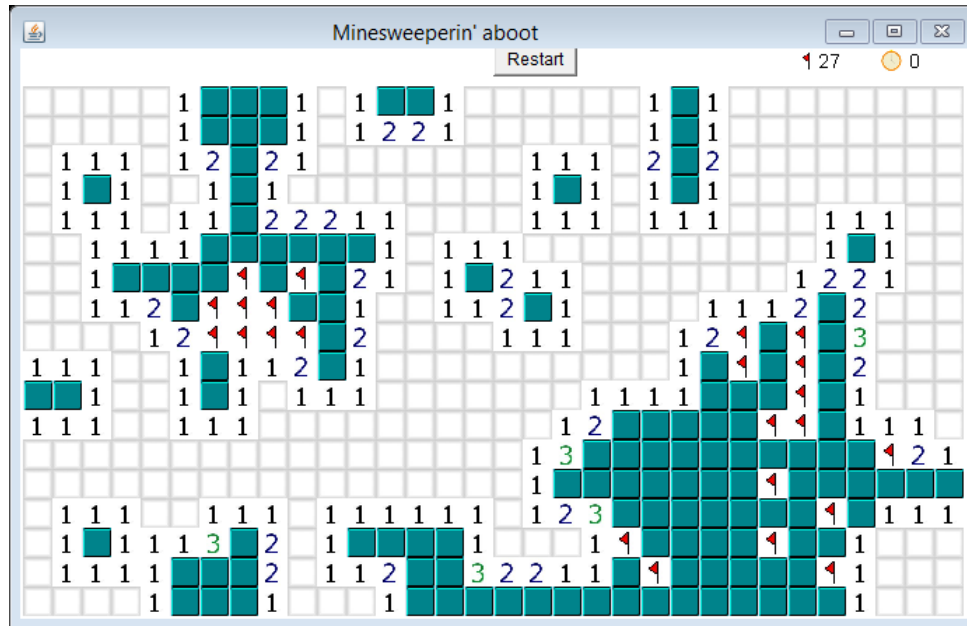


Imagen 2

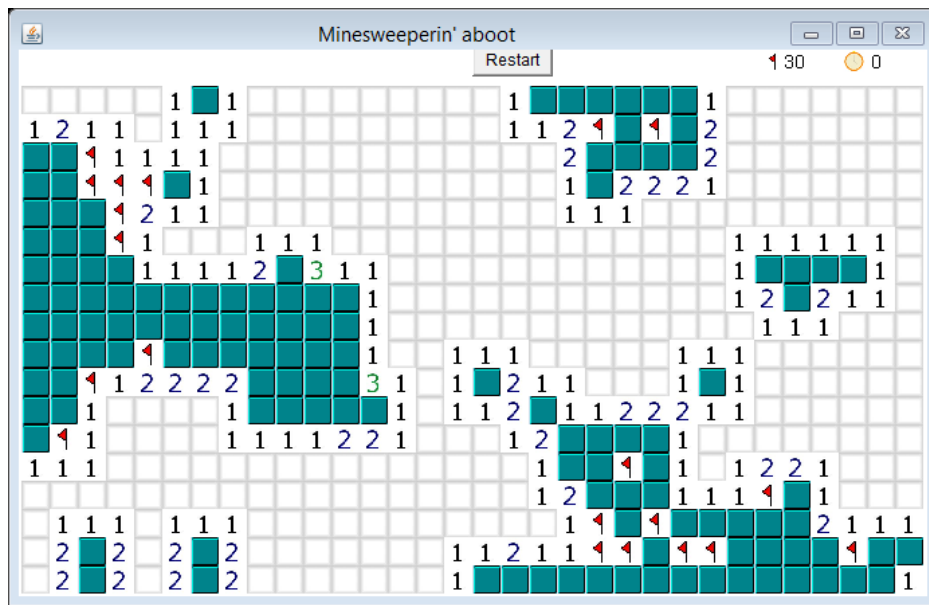


Imagen 3

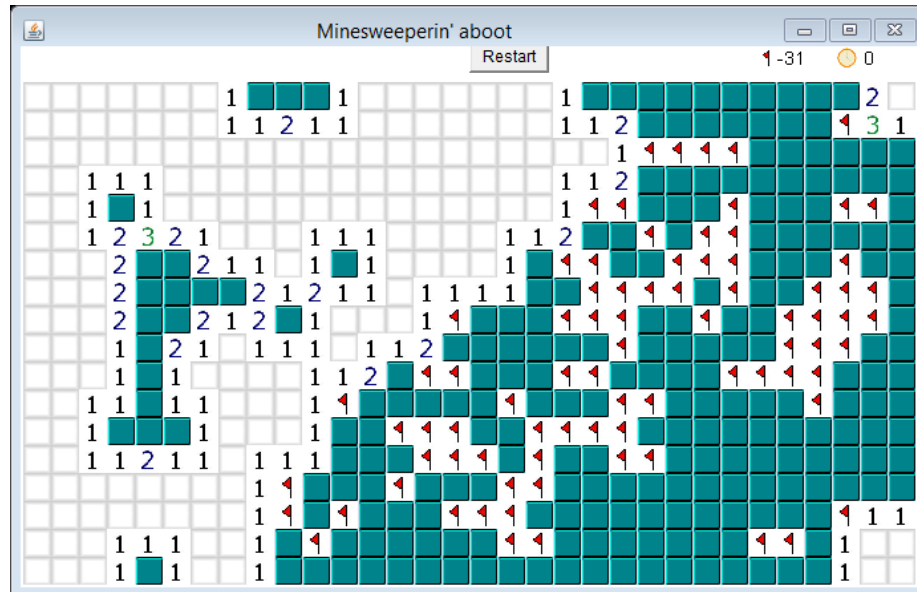


Imagen 4

