

Inteligencia Artificial, 2023-i, Tarea 2

Martínez Buenrostro Jorge Rafael.

Los números aleatorios y la ingeniería.

Reporte propio

Para que la simulación sea posible es necesario contar con procedimientos capaces de producir números aleatorios; un número aleatorio es aquel que puede ser generado con igual de probabilidad y en forma independiente de cualquier resultado previo.

METODOS DE GENERACIÓN

En primer lugar están los métodos manuales como: el de la ruleta y las tablas de números aleatorios. Tocker ha sugerido tres métodos para producir números aleatorios cuando se usan computadoras digitales: **provisión externa**; generación interna a partir de un proceso físico al azar y la generación interna de sucesiones de dígitos.

El primer método implica la grabación de tablas aleatorias, una objeción es la lentitud del proceso de entrada. El segundo método hace uso de un aditamento especial de la computadora capaz de registrar los resultados de algún proceso aleatorio; dentro de estos procesos aleatorios se incluyen el decaimiento de los materiales radioactivos y el ruido térmico de un circuito de válvulas electrónicas. El defecto es que los resultados no se pueden reproducir. El tercer método implica la generación de números pseudo aleatorios por medio de una ecuación recursiva algorítmica.

PROPIEDADES

Un generador debe: generar un número en el menor tiempo posible, no debe requerir grandes cantidades de memoria, debe ser capaz de producir un conjunto de números aleatorios diferentes; si produce el mismo número debe poder hacer correcciones y seguir adelante. Debe presentar un largo periodo, es decir, que la cantidad de números que se generan antes de que reaparezca la misma secuencia debe ser grande. Estos números deben cumplir lo siguiente:

- Deben lucir como obtenidos de una distribución uniforme $U(0,1)$, su media debe ser $1/2$, la suma promedio de los cuadrados debe ser $1/3$ y la suma promedio de los cubos debe ser $1/4$.
- Los números aleatorios deben ser independientes.

TIPOS DE GENERADORES

Los podemos clasificar en: *generadores basados en métodos de congruencia*; generadores compuestos y generadores de Tausworthe. Los generadores basados en métodos de congruencia se pueden explicar como: $\mathbf{U} = \mathbf{V} \pmod{\mathbf{T}}$. Donde: \mathbf{U} y \mathbf{V} son números enteros. Si la expresión $(U - V)/T$ es un entero, entonces \mathbf{U} es congruente a \mathbf{V} con módulo en \mathbf{T} .

GENERADORES CONGRUENTES LINEALES Definidos por la relación $Z_i = (aZ_{i-1} + C) \pmod{m}$. Donde \mathbf{a} es un multiplicador, \mathbf{C} el incremento y \mathbf{Z}_0 la semilla o valor inicial son valores no negativos y deben satisfacer $0 < m, C < m, Z_0 < m, a < m$. La longitud de un ciclo se denomina periodo (p). Si $p = m$, el generador es de periodo completo; para tener periodo completo es necesario seleccionar \mathbf{a} , \mathbf{m} y \mathbf{C} considerando lo siguiente:

- C debe ser impar y primo en relación con m . Para un computador binario $c \pmod{8} = 5$, y para un computador decimal $c \pmod{200} = 21$
- a debe ser un entero impar, no divisible por 3 ó 5. $a = 1 \pmod{p}$ si p es un factor primo de m . $a = 1 \pmod{4}$ si 4 es un factor de m .
- m debe ser: 2^b para un computador binario y 10^d para uno decimal. \mathbf{b} es el número de bits y \mathbf{d} el número de dígitos decimales en una palabra.

GENERADORES COMPUESTOS Están basados en la combinación de otros generadores separados con la esperanza que el generador final presente mejor comportamiento estadístico. Inicialmente un vector $\mathbf{V} = (V_1 V_2 \dots V_k)$ es llenado con los primeros \mathbf{k} U_1 obtenidos del primer generador congruente lineal; luego el segundo generador es utilizado para generar un entero aleatorio distribuido uniformemente en los enteros de 1 hasta \mathbf{k} y \mathbf{V}_1 es entregado como el primer número aleatorio de la serie; el primer generador entonces reemplaza la posición del entero generado por el segundo en \mathbf{V} con su siguiente \mathbf{U}_i y el segundo generador selecciona la siguiente posición del \mathbf{V} actualizado.

GENERADORES DE TAUSWORTHE Este método opera de la siguiente manera:

- Define una secuencia $\mathbf{b}_1, \mathbf{b}_2 \dots$ de dígitos binarios.
- Se hace $\mathbf{b}_i = (C_1 b_i + C_2 b_i \dots + C_R b_i) \pmod{2}$, donde $C_1 C_2 \dots C_R$ son cero o uno.
- Se agrupan \mathbf{k} bits para formar un entero binario de longitud \mathbf{k} con valor entre 0 y 2^k , el cual es dividido por 2^k , para dar el número $U(0,1)$.

COMPROBACIÓN DE UN GENERADOR DE NÚMEROS ALEATORIOS

Entre las pruebas más comúnmente utilizadas están las siguientes:

- **Pruebas sobre la uniformidad de la distribución.** Se interesan por el grado de acuerdo con que existen entre la distribución de una muestra de números aleatorios generados y la distribución uniforme, ambas pruebas están basadas en la hipótesis nula.
- **Pruebas de corridas.** Implica el conteo del número actual de ocurrencias de corridas de diferente longitud y comparar estos valores a los esperados usando Chi-cuadrado.
- **Pruebas de auto correlación.** Examinan la tendencia de los números de ir seguidos por otros números.
- **Pruebas de huecos.** Cuenta el número de dígitos que aparecen entre repeticiones de un dígito en particular.
- **Prueba de poker.** Analiza la frecuencia con la que se repiten los dígitos en números aleatorios individuales.
- **Pruebas de series.** Comprueba el grado de aleatoriedad entre los números sucesivos en una serie se utiliza la estadística Chi-cuadrado.
- **Pruebas de estadísticas en orden.** Verifica el valor máximo o mínimo de n números consecutivos o el rango de n valores consecutivos.

Reporte mejorado por ChatGPT

Para llevar a cabo una simulación es necesario contar con procedimientos capaces de generar números aleatorios. Un número aleatorio es aquel que puede ser generado con igual probabilidad e independientemente de cualquier resultado previo.

Existen métodos manuales para generar números aleatorios, como el de la ruleta y las tablas de números aleatorios. Sin embargo, cuando se utilizan computadoras digitales, existen tres métodos sugeridos por Tocker para generar números aleatorios: provisión externa, generación interna a partir de un proceso físico al azar y la generación interna de sucesiones de dígitos.

El primer método implica la grabación de tablas de números aleatorios, pero presenta la objeción de ser un proceso lento de entrada. El segundo método hace uso de un aditamento especial de la computadora capaz de registrar los resultados de algún proceso aleatorio, como el decaimiento de los materiales radioactivos y el ruido térmico de un circuito de válvulas electrónicas. Sin embargo, los resultados no se pueden reproducir. El tercer método implica la generación de números pseudoaleatorios por medio de una ecuación recursiva algorítmica.

Para que un generador sea eficiente, debe cumplir con ciertas propiedades. Debe generar un número en el menor tiempo posible, no debe requerir grandes cantidades de memoria y debe ser capaz de producir un conjunto de números aleatorios diferentes. Además, si produce el mismo número, debe poder hacer correcciones y seguir adelante. Debe presentar un largo periodo, es decir, que la cantidad de números que se generan antes de que reaparezca la misma secuencia debe ser grande. Estos números deben cumplir con las siguientes características: deben lucir como obtenidos de una distribución uniforme $U(0,1)$, su media debe ser $1/2$, la suma promedio de los cuadrados debe ser $1/3$ y la suma promedio de los cubos debe ser $1/4$. Los números aleatorios también deben ser independientes.

Los generadores pueden clasificarse en tres tipos: generadores basados en métodos de congruencia, generadores compuestos y generadores de Tausworthe. Los generadores basados en métodos de congruencia se definen por la relación $Z_i = (aZ_{i-1} + C) \pmod{m}$. La longitud de un ciclo se denomina periodo (p). Si $p = m$, el generador es de periodo completo. Para tener un periodo completo, es necesario seleccionar a , m y C de acuerdo con ciertas condiciones. Los generadores compuestos están basados en la combinación de otros generadores para obtener una mejor distribución estadística. Los generadores de Tausworthe generan números aleatorios dividiendo una secuencia binaria en subsecuencias más pequeñas y aplicando una fórmula algorítmica para obtener los números aleatorios.

Reporte creado por ChatGPT

Para este resumen intenté pegar el texto directamente, subir el texto a Google Docs y PasteBin para poder compartirlo con ChatGTP pero cada una de las veces hacía un resumen distinto sin relación al tema del texto original. A continuación hay uno de esos resúmenes.

Este documento es una guía para la creación y gestión de una pequeña empresa, enfocada en los aspectos fundamentales que los emprendedores deben considerar al iniciar su negocio. El documento se divide en cinco secciones principales:

- **Planificación:** Se describen los pasos necesarios para definir la visión, misión, objetivos y estrategias de la empresa. También se explican las herramientas de planificación, como el análisis DAFO y el modelo Canvas.
- **Financiamiento:** Se detallan las diferentes opciones de financiamiento para una pequeña empresa, como el capital propio, los préstamos bancarios, las subvenciones y el financiamiento colectivo. Se describen los requisitos necesarios para acceder a cada opción y se ofrecen consejos para elegir la mejor opción.
- **Marketing:** Se explica cómo crear una estrategia de marketing efectiva, incluyendo la investigación de mercado, la segmentación de clientes, la selección de canales de comunicación y la creación de una marca.
- **Operaciones:** Se analizan las diferentes opciones para la organización y gestión de los recursos humanos, así como la selección de proveedores y la gestión de inventarios.
- **Crecimiento:** Se ofrece información sobre cómo expandir el negocio y mantener su rentabilidad a largo plazo, mediante la creación de nuevas líneas de productos, la diversificación de los canales de venta y la exploración de nuevos mercados.

En cada sección, se presentan consejos prácticos y herramientas que los emprendedores pueden utilizar para desarrollar sus planes de negocio y gestionar su empresa de manera efectiva.

Análisis de algoritmos de búsqueda en espacio de estados.

Reporte propio

Grafos y dígrafos.

Los grafos son estructuras discretas que constan de vértices y aristas que conectan estos vértices. Un grafo $G = \{V, E\}$ está formado por un conjunto de vértices, V , y un conjunto de aristas E . Cada arista es un par (u, v) donde $u, v \in V$. Algunas veces las aristas tienen una tercera componente, denominada peso o costo. Las aristas de un grafo pueden ser dirigidas o no dirigidas. Una arista (u, v) es dirigida de u a v si el par ordenado (u, v) es ordenado y u precede a v . Mientras que es no dirigida si es par (u, v) no es ordenado.

Los grafos suelen visualizarse trazando los vértices en forma de óvalos o rectángulos y las aristas como segmentos o curvas que unen pares de vértices. Si todas las aristas de un grafo son no dirigidas, es un grafo no dirigido; en el caso que todas sus aristas sean dirigidas es un grafo dirigido. Si el grafo tiene aristas dirigidas y no dirigidas se le llama grafo mezclado.

- **Adyacencia.** Dos vértices son adyacentes si están conectados por una arista. Los vértices adyacentes de un vértice se les llaman vecinos.
- **Caminos.** Es una secuencia de vértices w_1, w_2, \dots, w_n tal que $(w_i, w_{i+1}) \in E$ para $1 \leq i \leq n$. Donde n es el número de aristas en el camino, es decir, la longitud. Un grafo es conectado si hay al menos un camino desde cualquier vértice a otro vértice cualquiera. Si no existe un camino entre todos los vértices, es un grafo no conectado.

Un laberinto puede ser representado por un grafo, donde los pasillos del laberinto son las aristas y sus intersecciones los vértices del grafo. Esta representación nos brinda la posibilidad de analizar sólo la estructura del grafo; para aplicar la teoría de grafos, es necesario conocer la estructura del grafo e incluso tomar en cuenta ciertas propiedades para poder decidir que algoritmos son los más adecuados para su implementación. Saber estas propiedades nos permite aplicar el recorrido de grados para encontrar la conexión entre dos nodos. Entre los algoritmos que se pueden aplicar están: *primera búsqueda en amplitud (BFS)*, *primera búsqueda en profundidad (DFS)*, *Dijkstra* y *S-Star*.

Algoritmos de búsqueda.

Es posible encontrar la ruta con el menor costo posible entre dos puntos dados. Debido a que en un laberinto se puede modelar también como un arreglo

bidimensional de $N \times M$, en la que hay celdas libres y celdas pared; un laberinto es un área de dos dimensiones en forma de rejilla de cualquier tamaño.

- **Algoritmo de búsqueda en amplitud.** Etiqueta todas las celdas, buscando la celda del final en todos sus vecinos adyacentes. Si no llega a la celda "T", la búsqueda continua hacia habitaciones adyacentes encontradas a partir de la habitación inicial; hasta que la celda "T" sea localizada. A continuación se muestran los pasos que sigue.

1. Etiquetar la celda de inicio como 0.
2. $i=0$.
3. Para cada celda etiquetada con i , etiquete todas las celdas adyacentes no etiquetadas con $i+1$. Si no hay celdas adyacentes, parar.
4. Si alguna de las celdas recién etiquetadas es la celda objetivo, terminar; una "ruta" solución ha sido encontrada.
5. $i=i+1$.
6. Ir al paso 3.

- **Algoritmo de búsqueda en profundidad.** Etiqueta todas las habitaciones como no visitadas, usa una pila para recordar donde debe ir cuando no alcanza un extremo muerto. Este algoritmo encuentra un camino si existe, y se puede aplicar a laberintos de conexión simple o múltiple (tiene circuitos internos), pero no garantiza dar la ruta más corta. Para empezar se elige una habitación inicial "S" y final "T" para poder seguir las siguientes reglas:

Regla 1. Si es posible, visitar una habitación (si hay más de una se elige una al azar) adyacente no visitado, marcarlo como visitado, si la celda es la objetivo, terminar.

Regla 2. Si no se puede aplicar la regla 1, entonces, si es posible, regresar a la última bifurcación.

Regla 3. Si no se puede aplicar las reglas anteriores, el proceso ha terminado

- **Algoritmo de Nayfeth.** Un laberinto se puede representar como un arreglo bidimensional con valores "1" y "0" representando paredes y pasillos. Cada celda tiene vecinos adyacentes en cuatro direcciones: *Norte, Sur, Este y Oeste*.

Para empezar se elige una celda inicial E y final S, ambas tienen "0", estas deben ser distintas. El algoritmo bloquea todos los puntos muertos en el laberinto y cada celda libre es accesible solamente en una dirección; las celdas libres se hacen nuevas celdas pared. Este procedimiento se

repite hasta que el "espacio de búsqueda" queda sin cambios, es decir que las únicas celdas libres son la solución del laberinto. Este algoritmo no funciona adecuadamente para laberintos de conexión múltiple. Se basa en las siguientes reglas:

Regla 1. Las celdas pared no cambian.

Regla 2. Las celdas libres, se vuelven pared si tienen tres o más vecinos pared.

Regla 3. Las celdas libres permanecen igual si tiene menos de tres celdas pared.

- **Algoritmo A-Star.** Utiliza una búsqueda heurística para encontrar la ruta óptima entre dos puntos. Maneja tres funciones: F, G y H. Donde:

La función **G** es el costo del mejor camino desde la celda inicial a la celda **n** hasta el momento durante la búsqueda.

La función **H** es el costo del camino más corto desde la celda **n** a la celda objetivo más cercana **n**.

$F = G + H$, es decir, F es el costo del camino más corto desde la celda inicial a la celda objetivo.

El costo de la función **H**, se calcula con una distancia de Manhattan, que consiste en sumar la cantidad de bloques en horizontal y vertical que restan para llegar a la meta y multiplicar por el costo que tiene asignado este tipo de movimientos.

Reporte mejorado por ChatGPT

Grafos y dígrafos son estructuras discretas compuestas por vértices y aristas que conectan estos vértices. Un grafo $G = V, E$ se compone de un conjunto de vértices, V , y un conjunto de aristas, E . Cada arista es un par (u, v) donde $u, v \in V$. A veces, las aristas tienen una tercera componente llamada peso o costo. Las aristas pueden ser dirigidas o no dirigidas. Una arista (u, v) es dirigida de u a v si el par ordenado (u, v) es ordenado y u precede a v . Si es un par no ordenado, se considera una arista no dirigida. Si un grafo tiene aristas dirigidas y no dirigidas, se llama grafo mixto.

Los grafos se representan a menudo visualmente, trazando los vértices como óvalos o rectángulos y las aristas como segmentos o curvas que unen pares de vértices.

Dos vértices se consideran adyacentes si están conectados por una arista. A los vértices adyacentes de un vértice se les llama vecinos. Un camino es una secuencia de vértices w_1, w_2, \dots, w_n tal que $(w_i, w_{i+1}) \in E$ para $1 \leq i \leq n$, donde n es el número de aristas en el camino, es decir, la longitud. Un grafo se considera conectado si hay al menos un camino desde cualquier vértice a cualquier otro vértice. Si no existe un camino entre todos los vértices, se trata de un grafo no conectado.

Los laberintos se pueden representar mediante grafos, donde los pasillos del laberinto son las aristas y sus intersecciones son los vértices del grafo. Esta representación permite analizar solo la estructura del grafo y aplicar la teoría de grafos. Es importante conocer las propiedades del grafo para decidir qué algoritmos son los más adecuados para su implementación y poder aplicar el recorrido de grados para encontrar la conexión entre dos nodos. Algunos de los algoritmos que se pueden utilizar son la búsqueda en amplitud (BFS), la búsqueda en profundidad (DFS), Dijkstra y S-Star.

Es posible encontrar la ruta con el menor costo posible entre dos puntos dados en un laberinto, el cual se puede representar como un arreglo bidimensional de $N \times M$, en el que hay celdas libres y celdas pared. En la búsqueda en amplitud, se etiquetan todas las celdas, buscando la celda del final en todos sus vecinos adyacentes. Si no se encuentra la celda "T", la búsqueda continúa hacia habitaciones adyacentes encontradas a partir de la habitación inicial hasta que se localiza la celda "T". La búsqueda en profundidad etiqueta todas las habitaciones como no visitadas y utiliza una pila para recordar dónde debe ir cuando no alcanza un extremo muerto. Este algoritmo encuentra un camino si existe, pero no garantiza dar la ruta más corta. En el algoritmo de Nayfeth, un laberinto se representa como un grafo ponderado y se usa el algoritmo de Dijkstra para encontrar la ruta más corta.

Reporte creado por ChatGPT

El artículo se enfoca en la resolución de problemas en inteligencia artificial, que requieren una secuencia de acciones o decisiones que minimicen un costo o maximicen un beneficio. Para ello, se utiliza el espacio de búsqueda y se asume que el explorador se mueve en un entorno accesible y que el costo o beneficio de las acciones se pueden percibir con exactitud.

Se describe la relación entre grafos y espacios de búsqueda y se presentan algunos conceptos como vértices, aristas y adyacencia. Se explica que un camino en un grafo es una secuencia de vértices conectados por aristas y que su longitud puede ser con o sin pesos. También se menciona que un grafo es conectado si existe al menos un camino entre cualquier par de vértices y que, de lo contrario, es un grafo no conectado.

Además, se describe la relación entre grafos y laberintos, donde los vértices representan los puntos de decisión y las aristas las conexiones entre estos puntos. Por último, se describe el funcionamiento de los algoritmos de búsqueda aplicados en problemas tipo rejilla.

Comparación de resultados

La ventaja de los resúmenes que hice sobre los que genera ChatGPT es que tengo una idea de lo que tratan los textos, ya que con al menos dos léidas puedo encontrar puntos que puedo usar en los problemas que han salido durante el curso. Un gran punto en contra para el resumen que hice es que considero que no es un buen resumen, no tengo una gran experiencia haciendo resúmenes por lo que es probable que la calidad no sea buena; considero que la mejor alternativa es mi resumen mejorado por ChatGPT, conservo el conocimiento de haber leído el texto y se mejora la sintaxis y gramática.