**Algoritmo Colonia de Hormigas**

**Parte biológica.**

Imagen que contiene naranja, pelota, teléfono

Descripción generada automáticamenteLas hormigas son insectos sociales que viven en colonias y se comunican a través de feromonas, cuando están en búsqueda de comida usualmente exploran el área de forma aleatoria, hacia y desde una fuente de alimento depositan en el suelo una sustancia llamada feromona, tendiendo a elegir el camino que seguirán por este olor.

Los caminos mas cercanos acumulan mayor cantidad de feromonas, y los caminos mas lejanos van perdiendo feromona por evaporación, después de algún tiempo toda la colonia converge hacia el uso del mismo camino.

Imagen de la pantalla de un celular con letras

Descripción generada automáticamente con confianza baja

*Información heurística:* Se mide la preferencia heurística que tienen las hormigas para moverse de un nodo a otro.

*Información de los rastros de feromonas artificiales:* En esta se mide la deseabilidad aprendida en el movimiento de un nodo a otro.

Diagrama, Dibujo de ingeniería

Descripción generada automáticamenteLas m hormigas de la colonia se mueven, concurrentemente y de manera asíncrona, a través de los estados adyacentes del problema, este movimiento se realiza siguiendo una regla de transición que esta basada en la información local disponible en los componentes.

Esta información local incluye la información heurística y memorística para guiar la búsqueda. Al moverse por el grafo de construcción, las hormigas construyen incrementalmente soluciones.

Opcionalmente, las hormigas pueden depositar feromonas cada vez que crucen un arco mientras que construyen la solución Una vez que cada hormiga ha generado una solución esta se evalúa y puede depositar una cantidad de feromona que es función de la cantidad de solución.

La evaporación de feromona se lleva a cabo en el entorno y se usa como un mecanismo que evita el estancamiento en la búsqueda y permite que la hormiga busque y explore nuevas regiones en el espacio.

Algoritmo general

1. Inicialización de la feromona
2. Mientras (criterio\_de\_terminacion\_no\_satisfecho)

2.1 Para cada hormiga

2.1.1 Generar solución (usar regla de transición)

[aporte local de feromona]

Fin para

[Refinamiento de soluciones con búsqueda local]

2.2 Elegir la mejor hormiga de la iteración y actualizar la mejor hormiga global (en base a la bondad de la solución)

2.3 Aporte global de feromona

2.4 Evaporación de feromona

Fin mientras

Fin procedimiento

**Componentes del algoritmo y su diversidad.**

**Inicializar**

Tao

Alfa .- Influencia de las feromonas

Beta .- Influencia de la información heurística

Tasa de la feromona p = 0.5

H= numero de hormigas

**Construir soluciones**

Cada hormiga construye una solución al problema, se le asigna una probabilidad dependiendo del rastro de feromonas y la información heurística

Texto

Descripción generada automáticamente

Imagen que contiene exterior, naranja, papalote, colorido

Descripción generada automáticamenteProbabilidad de que la hormiga estando en un lugar i vaya a un lugar j.

EJEMPLO

Feromonas\* información heurística / sumatoria de esta información, pero de todos los lugares posibles a los que puede ir desde la carpa

Búsqueda local. - La implementación es dependiente del problema que se esta resolviendo ya que es algún caso la búsqueda es demasiado costosa, actualizar feromonas después de que una hormiga construye una solución y finaliza la búsqueda local, se actualiza el rastro de feromonas a partir de las soluciones que construyeron las hormigas.

**Actualizar feromonas**

En primer lugar, se realiza la evaporación de feromonas en todos los elementos del rastro, simulando el proceso que ocurre en la naturaleza



Tras esto el siguiente paso es el depósito de feromonas

Texto, Esquemático

Descripción generada automáticamente con confianza media

Texto

Descripción generada automáticamenteEl aumento es en función de la calidad, así que en las mejores soluciones se depositan mayores feromonas.

**Criterios de finalización**

Los criterios de paro que generalmente son usados consisten en fijar:

* Un limite de tiempo de proceso.
* Un número máximo de iteraciones del algoritmo.
* Un número máximo de iteraciones sin mejora de la solución actual o
* Un valor umbral a partir del cual se considera que la solución actual es de suficiente calidad.

**Formas de actualizar el rastro de feromonas**

*Sistema etilitista.- En la mejor solución encontrada hasta el momento (s mejor) deposita una cantidad adicional (delta tao mejor) de feromona en cada iteración, controlada por el parámetro que define el peso dado a dicha solución.*

*Sistema raking.- Cada hormiga deposita feromonas proporcional a la calidad de solución encontrada.*

*Sistema de colonias.- Se deposita la misma cantidad de feromonas sin importar la bondad de la solución.*

Imagen que contiene Escala de tiempo

Descripción generada automáticamenteImagen que contiene Gráfico

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamenteInterfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamenteInterfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamenteInterfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamenteInterfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamenteInterfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamenteUna captura de pantalla de una computadora

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamenteInterfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamenteInterfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamenteInterfaz de usuario gráfica, Aplicación

Descripción generada automáticamenteInterfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Pantalla de computadora

Descripción generada automáticamentePantalla de computadora

Descripción generada automáticamente

Una captura de pantalla de una computadora

Descripción generada automáticamentePantalla de computadora

Descripción generada automáticamente

Pantalla de computadora

Descripción generada automáticamente

Una captura de pantalla de una computadora

Descripción generada automáticamente

**Explicación algoritmo.**

Primero se tienen las librerías necesarias para generar números aleatorios, trabajar cómodamente con caracteres y realzar cálculos matemáticos fácilmente, seguido de las constantes necesarias que caracterizan al algoritmo como lo son alfa, beta, evaporación, el número de iteraciones, la feromona entre otras, después se tiene las características que va a tener la hormiga que es la distancia, la calidad de la solución por así decirlo, la posición y la ruta que va a seguir.

Después se define el numero de la población de hormigas que va a haber, seguido de la matriz de feromonas y la matriz de distancias, cabe mencionar que en este ejemplo también se calcula la media, varianza y desviación estándar por consiguiente se definen todas variables que vamos a necesitar y se definen todas las funciones que se necesitaran para el algoritmo con sus respectivos parámetros, junto con métodos auxiliares como convertir números a caracteres o de impresión de rutas.

En la función principal empezamos con el reloj y el generador de numero aleatorio y de igual manera el numero de iteraciones, mandamos llamar la función de feromonas inicial y de obtener la mejor distancia y mientras la iteración en la que se esta sea menor al máximo que se definió entonces se va a construir una solución, se checara la mejor distancia, se calculará la calidad de la solución, se va a estar evaporado la feromona y se va a estar actualizando esta también y para todo el tamaño de la población de hormigas se va a estar imprimiendo la ruta y finamente se van a estar aumentándolas iteraciones hasta que se lleguen al máximo definido.

Se imprimirá el resultado y el tiempo de ejecución que el reloj marque.

(init\_ant) Primero se inicializa a la colonia con distancia, calidad y posición con 0 y después en la ruta vamos a llenar con -1, seguido de esto se van a posicionar a las hormigas en ciudades aleatorias.

(seed\_initial\_pheromone) Se va a recorrer la matriz y se va a estar asignando la feromona inicial con números aleatorios si i es diferente de j esto va a estar sucediendo y si no en enlace de feromonas va a ser invalido

(get\_greater\_distance) Se va a recorrer la matriz de las ciudades y si la distancia de los enlaces es mayor la mejor distancia que se tiene, la mejor distancia se va a actualizar y esta va a ser igual a la distancia de los enlaces y a esta le vamos a sumar 1.

(check\_visit) Se pasa el parámetro de index de la hormiga y su posición y si en i en la la ruta es igual a posición quiere decir que ya se visito de lo contrario se va a retornar false .

(build\_solution) Se va a recorrer la matriz de ciudades, primero en i horizontalmente se va a inicializar la hormiga e index va a hacer el iterador i que se va a pasar como parámetro, después mientras que posiciones de las hormigas sean menores que la cantidad de ciudades(hasta que pase por todas las ciudades) la posición va a hacer igual a posición de la hormiga actual y se define la probabilidad de transición del tamaño de la cantidad de ciudades, así como la suma de los enlaces, después en j verticalmente si la ciudad ya fue visitada no se entra al análisis, caso contrario , si la feromona de enlace en donde se está posicionado es mayor o igual a cero y la distancia donde se esta es mayor o iguala a cero, la suma de los enlaces va a ser igual a alfa elevado a la feromona de los enlaces de la posición de donde se esta por beta elevado a la mejor distancia menos la distancia de la posición donde se está.

Después calculamos la probabilidad de transición, para cada j verticalmente hasta que sea menor al numero de ciudades , si la ciudad ya fue visitada no se ingresa al análisis , caso contrario si el enlace de feromona de la posición donde nos encontramos es mayor o igual a cero y la distancia es mayor o igual a cero la probabilidad de transición es igual alfa elevado al enlace de feromona multiplicado por beta elevado a la mejor distancia menos la distancia de la posición donde nos encontramos , todo eso entre la suma de los enlaces, si no se cumple la condición entonces la probabilidad de transición va a ser cero y si el análisis no se hace la probabilidad de transición también va a ser cero.

Definimos un numero mejor y menor y una ruleta que va a ser un numero aleatorio y se va a seleccionar el siguiente nodo donde para cada j verticalmente hasta que se menor al umero de ciudades , si la ciudad ya fue visitada no se ingresa al análisis , caso contrario el mayor se le va a sumar la probabilidad de transición de esa posición y si la ruleta es mayor o igual a menor y ruleta es menor o igual a mayor entonces la ruta de las hormigas van a ser igual a j , a la distancia se le sumara la distancia e la posición de la que estamos y a la posición se le sumara 1 y en caso contrario de no cumplirse la condición el menor va a ser igual al mayor.

(check\_best\_distance) Para cada i en la matriz horizontalmente hasta que sea menor que la población de hormigas, si la distancia de esa hormiga es menor a la mejor distancia la mejor distancia va a ser igual a esa distancia que se encontró, y para j en la matriz verticalmente hasta que sea menor que el número de ciudades , la mejor ruta va a ser igual a la ruta de esa hormiga , si no si la distancia de esa hormiga es mayor a la peor distancia que se tiene entonces la peor distancia es igual a esa nueva distancia que se encontró y para cada j verticalmente hasta que se menor al número de ciudades la peor ruta va a ser igual a la ruta de esa hormiga .

(calculate\_fitness) Para cada i horizontalmente en la matriz fitness va a ser igual a la distancia que recorrió la hormiga entre la mejor distancia, si el resultado es menor a la mejor calidad entonces best fitness va a ser igual a esta nueva que se encontró si no si fitness es mayor a la peor calidad que se tiene entonces worse fitness va a ser igual a esta que se encontró y fitness de la hormiga va a ser igual a la que se calculo previamente.

(pheromone\_evaporates) Se va recorrer toda la matriz (i,j) y si el enlace de feromona es diferente de -1 / invalid entonces el enlace de feromona va a ser igual a 1- la taza de evaporación multiplicado por el enlace de feromona.

(update\_pheromone) Para cada i de la matriz horizontalmente hasta que sea menor al numero de hormigas de la población las feromonas a sumar va a ser igual a conts entre fitness y para j verticalmente hasta que sea menor al numero de ciudades menos uno , ciudad va a ser igual a la ruta de la hormiga y la ciudad siguiente va ser igual a esa ruta pero más uno y si el enlace de feromona de la posición ciudad y ciudad siguiente es diferente a -1/invalid entonces el enlace de feromona de ciudad , ciudad siguiente va a ser igual a sumarle la feromona a sumar.

Métodos auxiliares

(get\_random\_number) Consigue un numero random entre 2 números

(calculate\_time) Calcula el tiempo que se tardo en ejecutar el programa

(print\_route) Imprime la ruta de las hormigas y el tamaño(distancia)

(print\_pheromone) Imprime la tasa de feromonas

(print\_result) Imprime el peor resultado con worse fitness, mejor resultado y best fitness con media , varianza y desviación estándar

(number\_to\_String) Convierte un numero a carácter

(calculate\_metrics) Calcula media de la distancia, varianza y desviación estándar