

¿Qué es Netlogo?

NetLogo es un entorno de modelado programable multiagente para simular fenómenos naturales y sociales. Es utilizado por cientos de miles de estudiantes, profesores e investigadores en todo el mundo.



Viene con una gran biblioteca de simulaciones existentes que puedes usar y modificar. Las áreas de contenido incluyen ciencias sociales y economía, biología y medicina, física y química, y matemáticas e informática.

Tú también puedes utilizarlo para crear tus propias simulaciones.

1 - Guía de instalación

En el siguiente enlace: <https://ccl.northwestern.edu/netlogo/6.2.0/> podremos encontrar la última versión de Netlogo (6.2.0).





	Mac OS X	Download (235 MB)
	Windows (32-bit)	Download (204 MB)
	Windows (64-bit)	Download (206 MB)
	Linux (32-bit)	Download (225 MB)
	Linux (64-bit)	Download (224 MB)

Figura 1. Página de descarga del software Netlogo

Descargue la versión adecuada dependiendo de su **arquitectura** y **sistema operativo** (Windows, Mac, Linux).

1.1 - Ejemplo de instalación en Windows :

Una vez descargado, nos situamos sobre el fichero (con extensión .msi) y le damos doble click izquierdo para abrir el gestor que nos ayudará a **instalarlo**.

Como la instalación de cualquier otro programa, seguimos los pasos del gestor de instalación hasta que se haya completado.

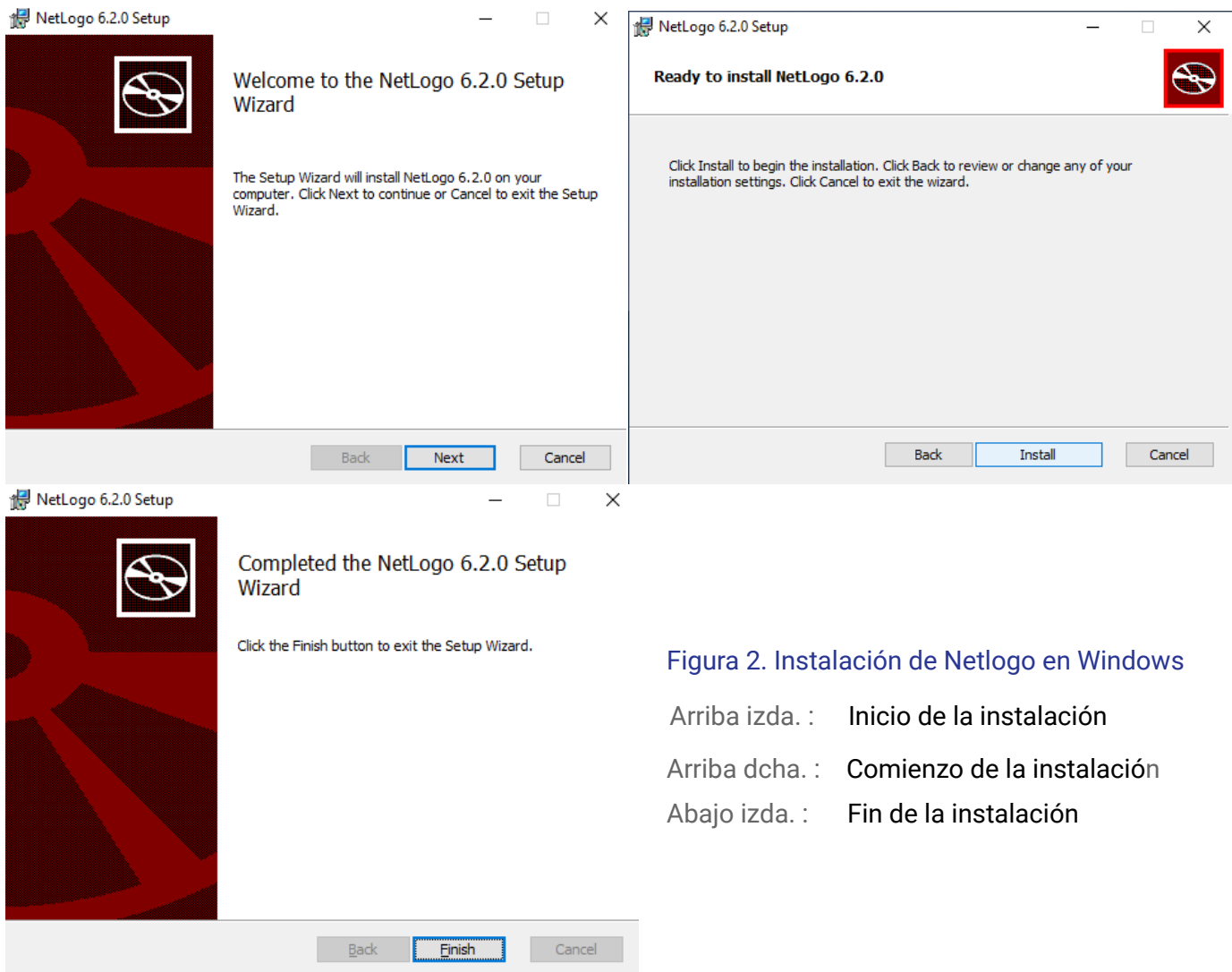


Figura 2. Instalación de Netlogo en Windows

Arriba izda. : Inicio de la instalación

Arriba dcha. : Comienzo de la instalación

Abajo izda. : Fin de la instalación

Finalizada la instalación, escribimos sobre nuestra **barra de búsqueda** del escritorio el término '**NetLogo**' y ya deberían aparecernos (Ver figura 3).

Basta con clicar sobre él para que se **ejecute** la aplicación.

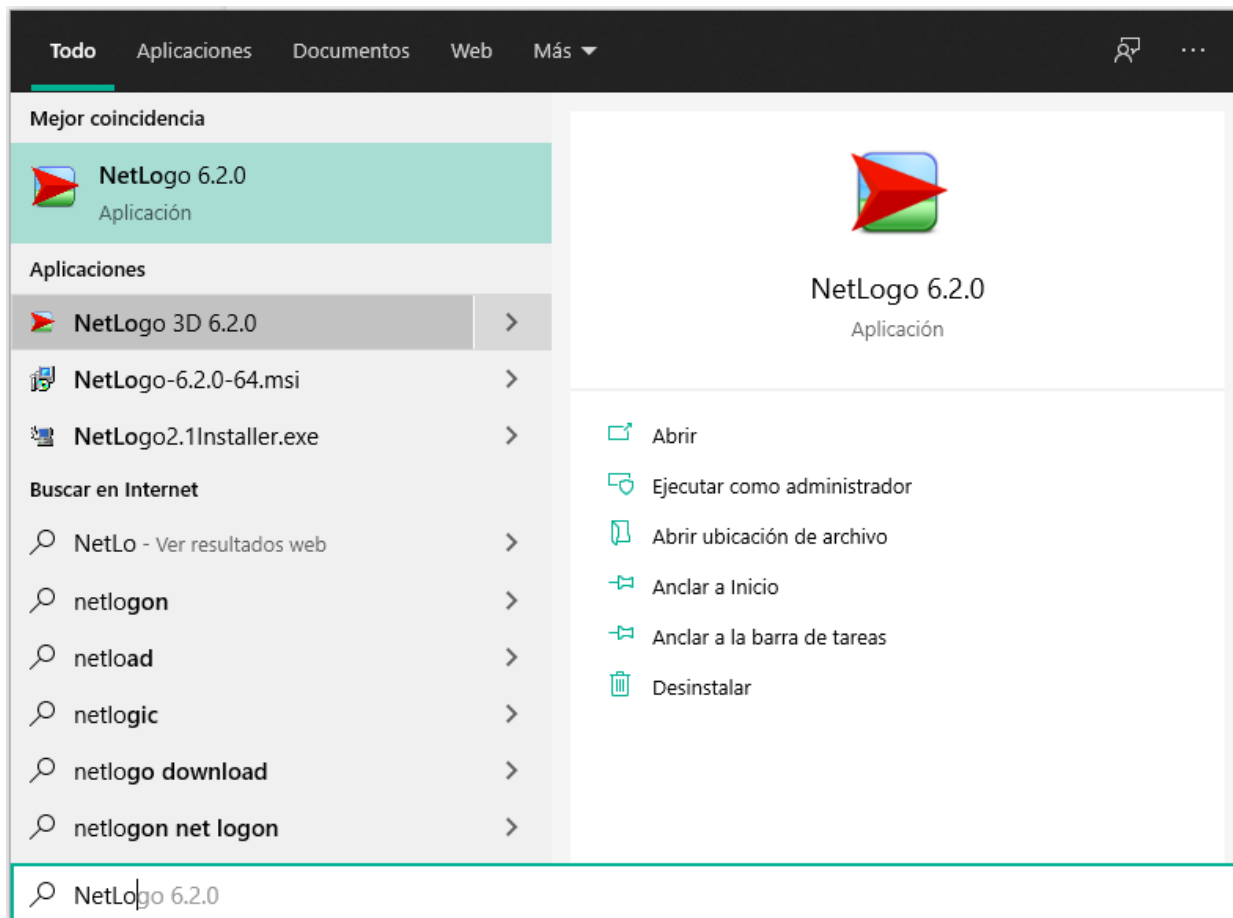


Figura 3. Ejecución de Netlogo en Windows

Importante: Nótese que hay un ejecutable llamado **Netlogo** y otro llamado **Netlogo3D**, éste primero abrirá los ficheros con extensión **.nlogo**, el otro los ficheros con extensión **.nlogo3d**

1.2 - Ejemplo de instalación en Linux: (similar en Mac)

Una vez descargado, nos situamos sobre el fichero (con extensión .tgz en Linux o .dmg en Mac) y clickeamos con el botón derecho para descomprimirlo.

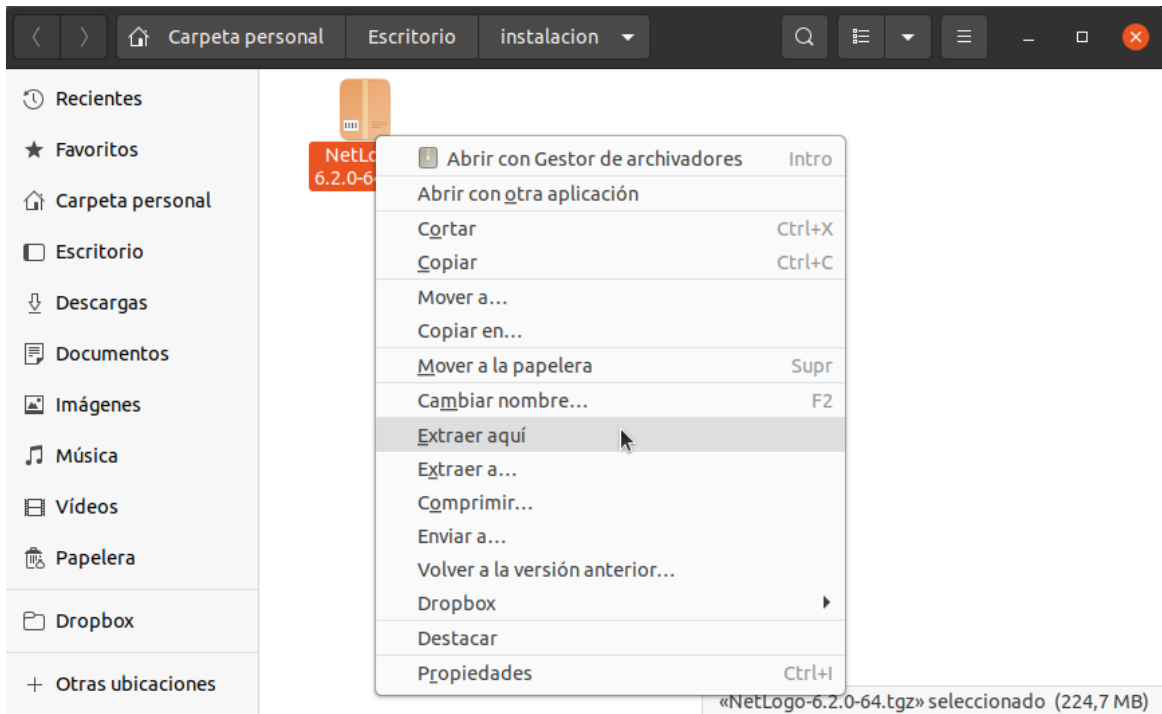


Figura 4. Extracción del programa en Linux

Una vez descomprimido entramos en la carpeta resultante y volvemos a entrar en la única carpeta existente 'Netlogo 6.2.0'. Dentro de ella, encontraremos lo siguiente:

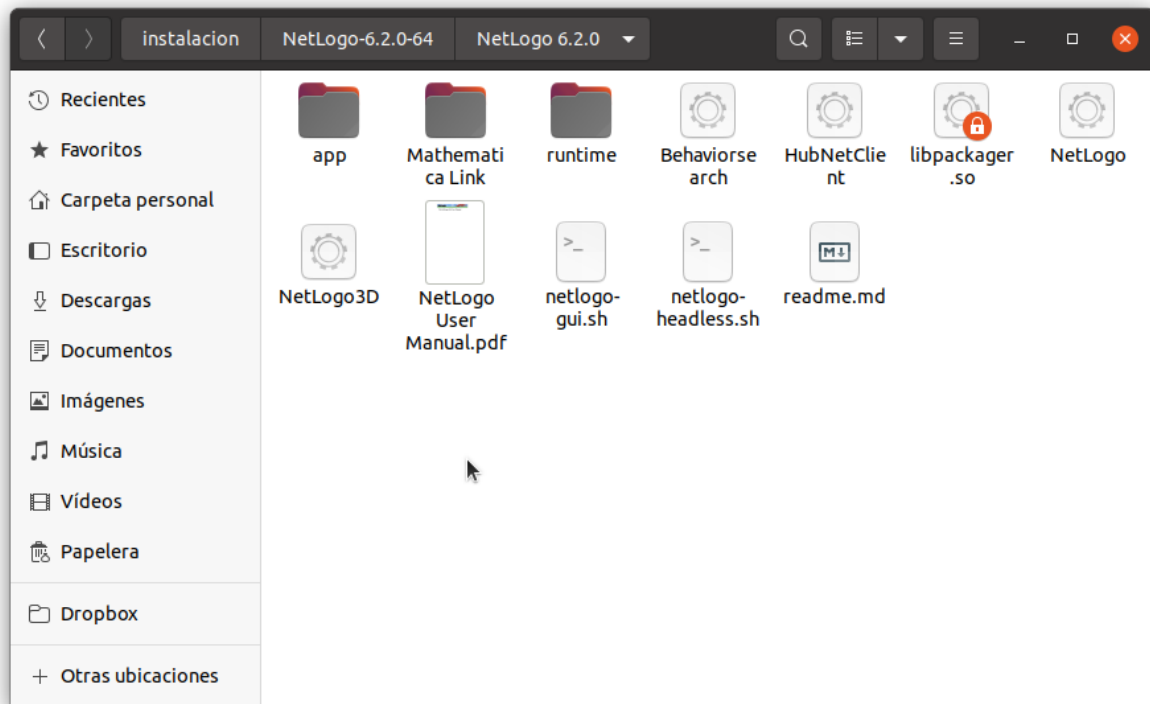


Figura 5. Contenido dentro de la carpeta extraída

Importante: Nótese que hay un ejecutable llamado **Netlogo** y otro llamado **Netlogo3D**, éste primero abrirá los ficheros con extensión **.nlogo**, el otro los ficheros con extensión **.nlogo3d**

Para **ejecutar** cualquiera de ellos, tenemos **2 opciones**:

- Pulsamos doble click para abrirlo, o
- Sobre la carpeta hacemos click derecho y pulsamos 'Abrir en una terminal' y escribimos en la línea de comandos (dependiendo del que queramos abrir):
 - `$> ./NetLogo`
 - `$> ./NetLogo3D`

Seguidamente pulsamos la tecla enter y se nos **abrirá la interfaz** del programa.

2 - Introducción a la **simulación** de modelos

Una vez ejecutado y abierta la interfaz del programa, pulsamos sobre [Archivo > Abrir](#)

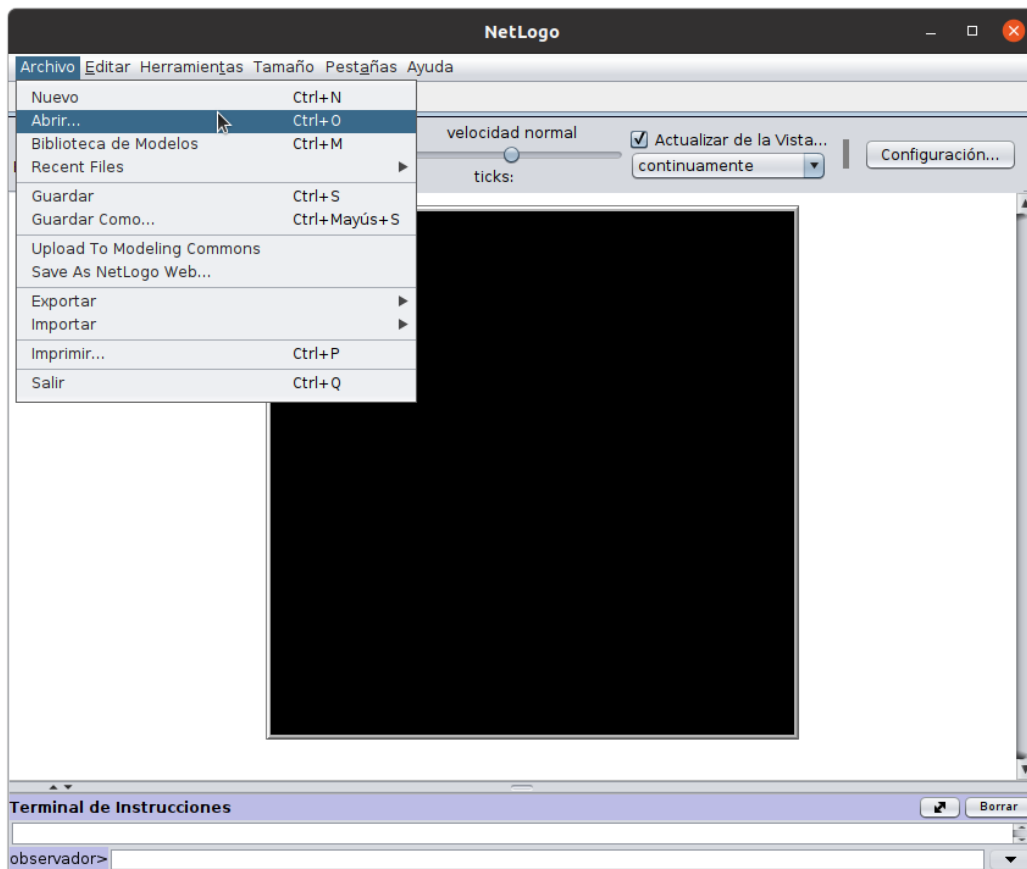


Figura 6. Interfaz de Netlogo al iniciar el programa

Buscamos a través de nuestros directorios la carpeta '**modelos**' que se adjuntará junto a este documento explicativo.

En ella están todos los modelos implementados (*algoritmos geénéticos, de colonia de hormigas y optimización por enjambre de partículas*), con un ejemplo básico y un problema más específico, (cada uno es un fichero independiente) desarrollados para este trabajo. (También se puede probar a abrir algún otro modelo de la 'Biblioteca de Modelos').

Por ejemplo, vamos a abrir el básico para TSP. Lo seleccionamos y pulsamos en [Abrir](#):

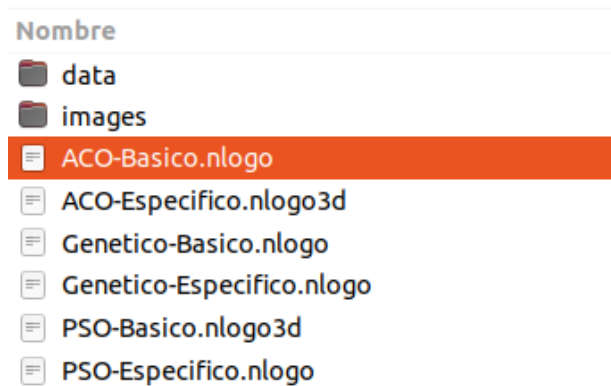


Figura 7. Contenido de la carpeta 'modelos'

Tras la apertura del modelo (ya situados en el apartado [Ejecutar](#)), aparecerá el entorno de ejecución pero sin inicializar. Para ellos **pulsamos** sobre el **botón setup**. Posteriormente sobre el botón **go** para **iniciar** y **detener** la **simulación**. (Ver figura 8).

Se recomienda empezar a utilizar con los modelos básicos y después con los específicos.

El resto de parámetros variará dependiendo del modelo. Se puede obtener una descripción más detallada del funcionamiento de cada uno de ellos y del modelo en general en el apartado [Información](#). (Ver figura 9)

Por último en el apartado [Código](#), se puede ver la implementación de dicho modelo.

Nota: Es importante **no modificar ni eliminar** los **directorios data** e **imágenes**, pues en ellos se encuentra la información necesaria para ejecutar correctamente los modelos.

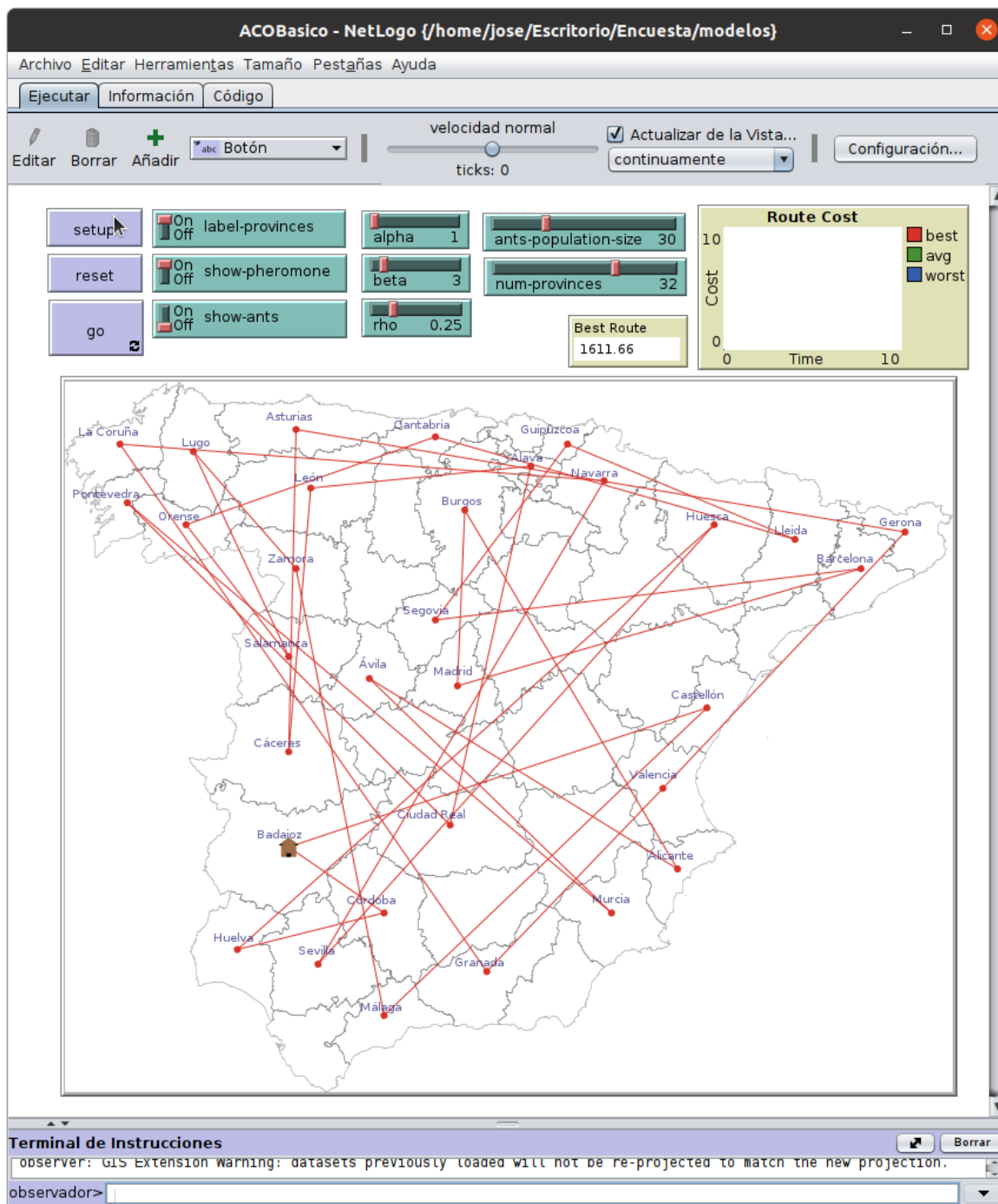


Figura 8. Inicialización del modelo básico ACO

Ejecutar
Información
Código

Buscar...
Editar

¿ QUE ES ?

Ant Colony Optimization es una técnica o método en informática inspirada en el comportamiento de las hormigas en la búsqueda de comida. Esta técnica fue propuesta por M. Dorigo en 1992 [1]. Puede aplicarse para resolver cualquier problema de optimización que pueda ser reducido a encontrar rutas mínimas a través de grafos.

Hay una población de agentes llamados hormigas que construyen iterativamente una posible solución para nuestro problema combinatorio. Usan el entorno para intercambiar información entre sí depositando feromonas.

Suelen comenzar con una solución vacía y luego van agregando componentes para nuestra solución hasta que la solución candidata es completada.

Una vez que se genera la solución, las hormigas dan su feedback sobre las soluciones que construyeron depositando una cantidad de feromona en cada una de las componentes (tramos del camino) que eligieron para construir su solución. Aquellas componentes que son comunes en buenas soluciones acumularán más feromonas y es más probable que las hormigas los utilicen en la próxima iteración

También incluye un mecanismo para simular la evaporación real de feromonas que evita elegir caminos más largos y atascarse en óptimos locales.

Muchos problemas reales, como problema de enrutamiento de vehículos o de redes de Internet, se resuelven utilizando estos algoritmos.

En este modelo, trato de ilustrar el uso de ACO resolviendo el conocido problema TSP: **Traveling Salesman Problem**:

Figura 9. Apartado de Información del modelo

Se anima a que se prueben los modelos y posteriormente a rellenar la siguiente **encuesta de satisfacción**: <https://forms.gle/e5NdYA8Djvm3cebCZ>

Muchas gracias por su tiempo

Autor del trabajo,
José Antonio