

Evación de Obstáculos con un Robot Móvil Terrestre usando Aprendizaje por Refuerzo

Manual para Simulación

M.A. Rojas Andrade, L. R. García Vázquez, E. Y. Aguilera Camacho, M.A. Escobar Carmona, A.S. Torres Villegas, J. P. Ramírez Paredes¹
Departamento de Ingeniería Electrónica, Campus Irapuato-Salamanca
jpi.ramirez@ugto.mx¹

Introducción

En este documento se detalla cómo reproducir y probar los resultados del proyecto de investigación desarrollado durante el XXVIII Verano de la Ciencia de la Universidad de Guanajuato. Este material acompaña al artículo que será publicado en la revista Jóvenes en la Ciencia, donde el lector podrá encontrar detalles técnicos acerca de la simulación y los algoritmos utilizados para conseguir que el robot móvil Pioneer P3-AT evada obstáculos en un escenario.

Programas necesarios

Para iniciar se deben instalar los programas necesarios para poder llevar a cabo la simulación de manera correcta.

Simulador CoppeliaSim

Para este se requiere entrar en la página oficial www.coppeliarobotics.com en el apartado de descargas, se descargará la versión edu ya que es la que se usó para esta simulación

The screenshot shows the CoppeliaSim website header with navigation links: Features, Videos, Downloads, Resources, Contact. Below the header is the CoppeliaSim logo and the text "from the creators of V-REP". A table compares three versions: player, edu, and pro. The 'edu' version is highlighted with a black border. Below the table are download buttons for each version.

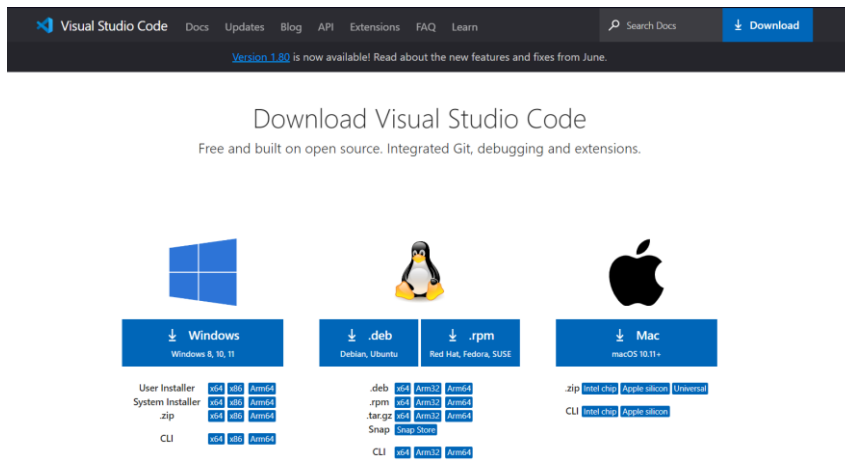
	player	edu	pro
Full simulation functionality	✓	✓	✓
Full editing capabilities		✓	✓
Commercial usage	✓		✓
	Free for everyone. Freely distributable.	May only be used by students, teachers, professors, schools and universities.	No usage restrictions. Contact us for pricing.
	Download CoppeliaSim Player	Download CoppeliaSim Edu	Download CoppeliaSim Pro

Current version: CoppeliaSim V4.5.1 rev4 - Channels

Entorno de desarrollo

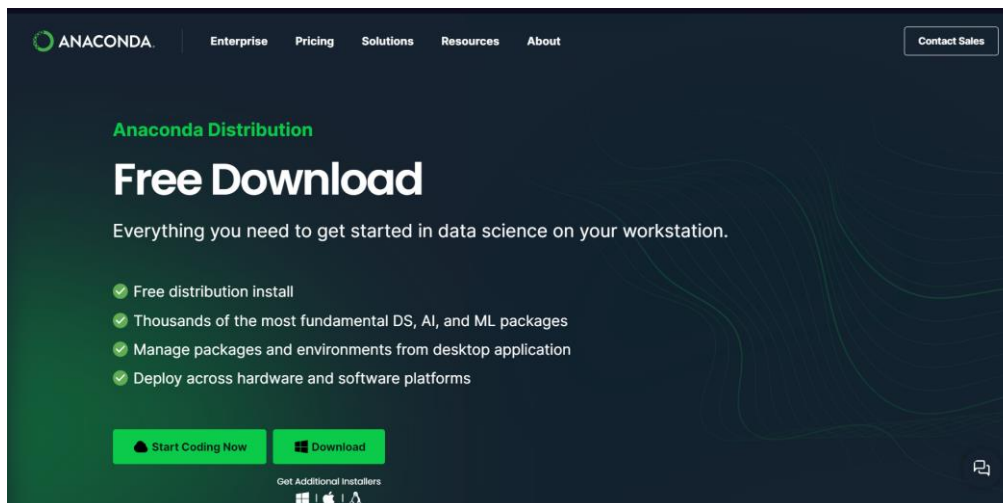
Igualmente se requiere un entorno de desarrollo o editor de código, no es necesario ninguno en específico mientras permita el manejo de Python, en este caso fue usado

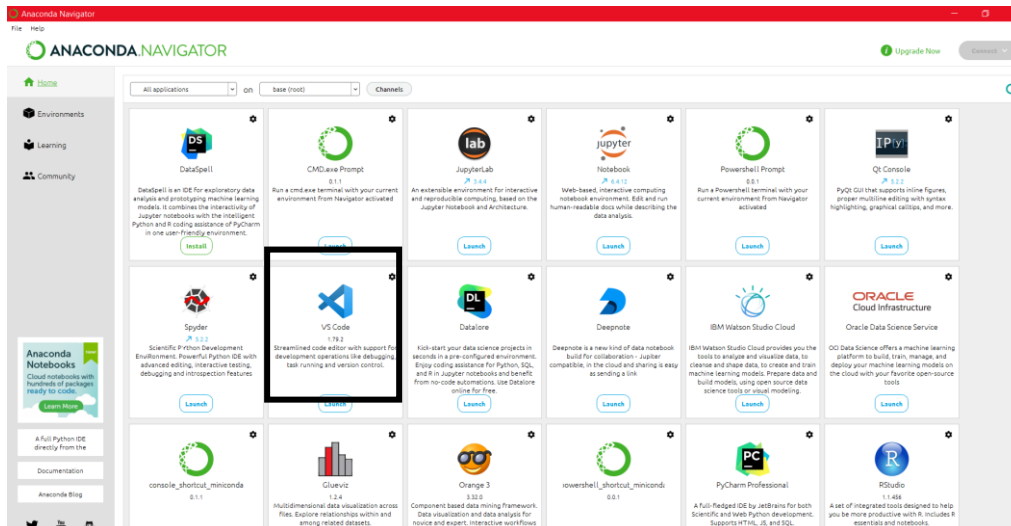
Visual Studio Code. Su descarga la igual que Coppelia, es entrando en la página oficial <https://code.visualstudio.com> en el apartado de descargas y se elige la versión que se requiera según el sistema operativo usado, cabe aclarar que para la simulación se usó Windows 10.



Anaconda

Esta distribución lo que hace es que ya nos permite lanzar desde ella Visual Studio Code con Python instalado, su instalación igualmente es desde la página oficial del <https://www.anaconda.com> en el apartado de descarga gratuita.





Bibliotecas

También se usarán algunas librerías que no viene por defecto en Python, estas son numpy y matplotlib, para su instalación solo es necesario escribir los siguientes comandos en la terminal de Visual Studio Code

- pip install numpy
- pip install matplotlib

```
PS C:\Users\luisr\OneDrive\Documentos\VeranoUG\Simulacion\archivos sim final> conda activate base
PS C:\Users\luisr\OneDrive\Documentos\VeranoUG\Simulacion\archivos sim final>
PS C:\Users\luisr\OneDrive\Documentos\VeranoUG\Simulacion\archivos sim final> pip install numpy
```

```
PS C:\Users\luisr\OneDrive\Documentos\VeranoUG\Simulacion\archivos sim final> conda activate base
PS C:\Users\luisr\OneDrive\Documentos\VeranoUG\Simulacion\archivos sim final>
PS C:\Users\luisr\OneDrive\Documentos\VeranoUG\Simulacion\archivos sim final> pip install matplotlib
```

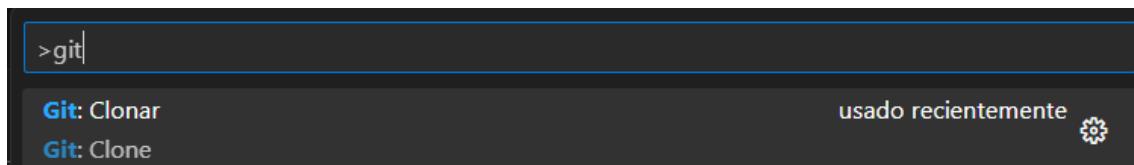
Simulación

En seguida es necesario entrar al Visual Studio Code lanzado desde anaconda, lo siguiente es clonar el repositorio, para esto con una cuenta github, en la misma terminal será necesario iniciar sesión.

```
PS C:\Users\luisr\OneDrive\Documentos\VeranoUG\Simulacion\archivos sim final> git config --global user.name [nombre-usuario]
```

```
PS C:\Users\luisr\OneDrive\Documentos\VeranoUG\Simulacion\archivos sim final> git config --global user.email [email]
```

Se ejecuta el comando Ctrl + Shift + P



Escribiendo en el recuadro que salga “git” y eligiendo la opción “Clonar” saldrá una nueva ventana donde se pegara lo siguiente:

<https://github.com/jpiramirez/VeranoUG2023.git>

<https://github.com/jpiramirez/VeranoUG2023.git>

Dirección URL de repositorio <https://github.com/jpiramirez/VeranoUG2023.git>

Clonar desde GitHub

De aquí sería solo elegir el nombre y lugar donde se desea guardar.

Lo siguiente y final sería abrir el escenario con extensión.ttt descargado del repositorio y ejecutar el código en Visual Studio para iniciar la simulación

```
PS C:\Users\luisr\OneDrive\Documentos\VeranoUG\Simulacion\archi  
ulacion\archivos sim final\VeranoUG2023\obsavoidP3-AT.py"  
Episode 1 Epsilon=0.0999  
Episode 2 Epsilon=0.0998  
Episode 3 Epsilon=0.0997  
Episode 4 Epsilon=0.0996  
Episode 5 Epsilon=0.09949999999999999  
Episode 6 Epsilon=0.09939999999999999
```

