



# Tecnológico de Monterrey

**Descenso al fondo de un cráter en Marte**

Ana Paula Ponce Dávila A01620477

Andrés Olvera Varela A01639918

Martín Alejandro Hermosillo García A01634552

Campus GDL

Diseño de agentes inteligentes

Grupo 101

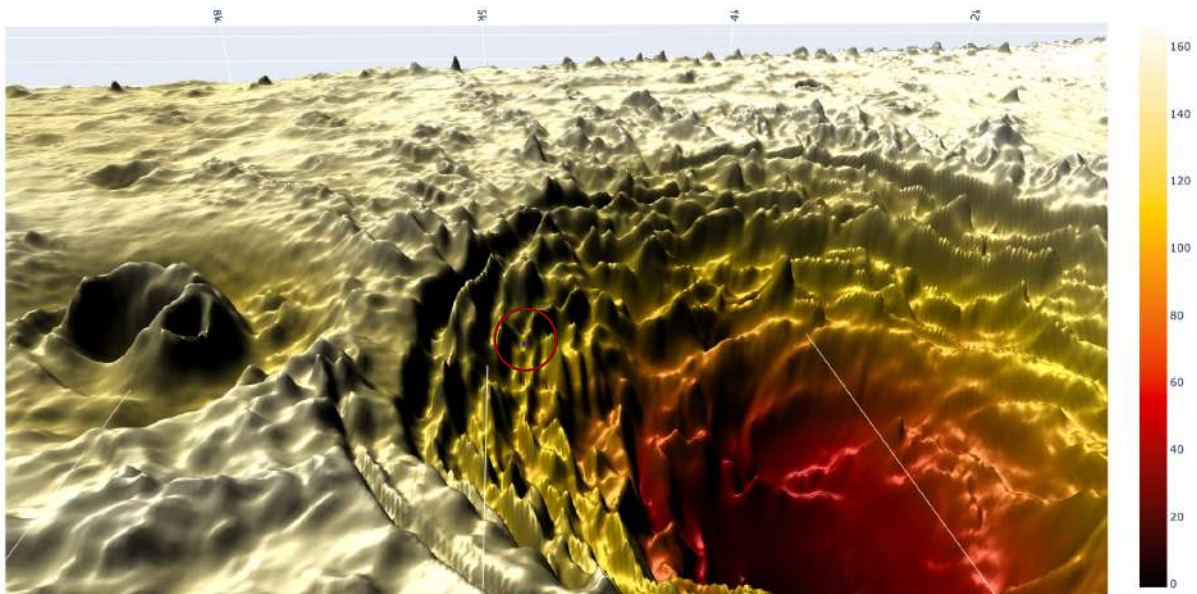
ITESM

Jueves 17 de marzo del 2022

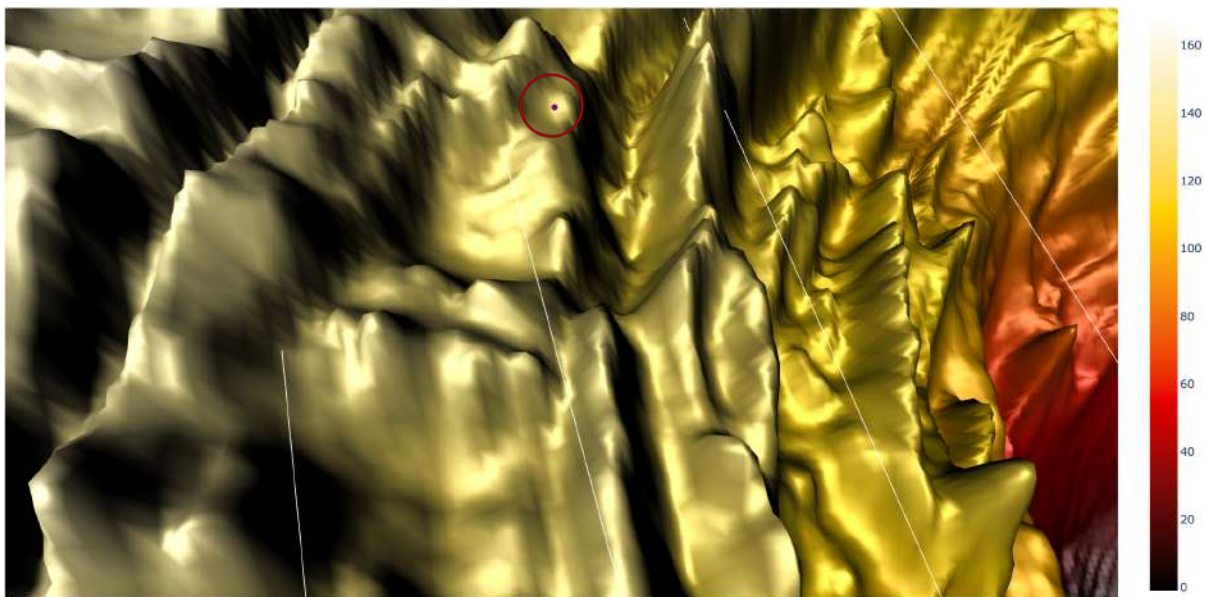
## Descenso al fondo de un cráter en Marte

Búsqueda codiciosa

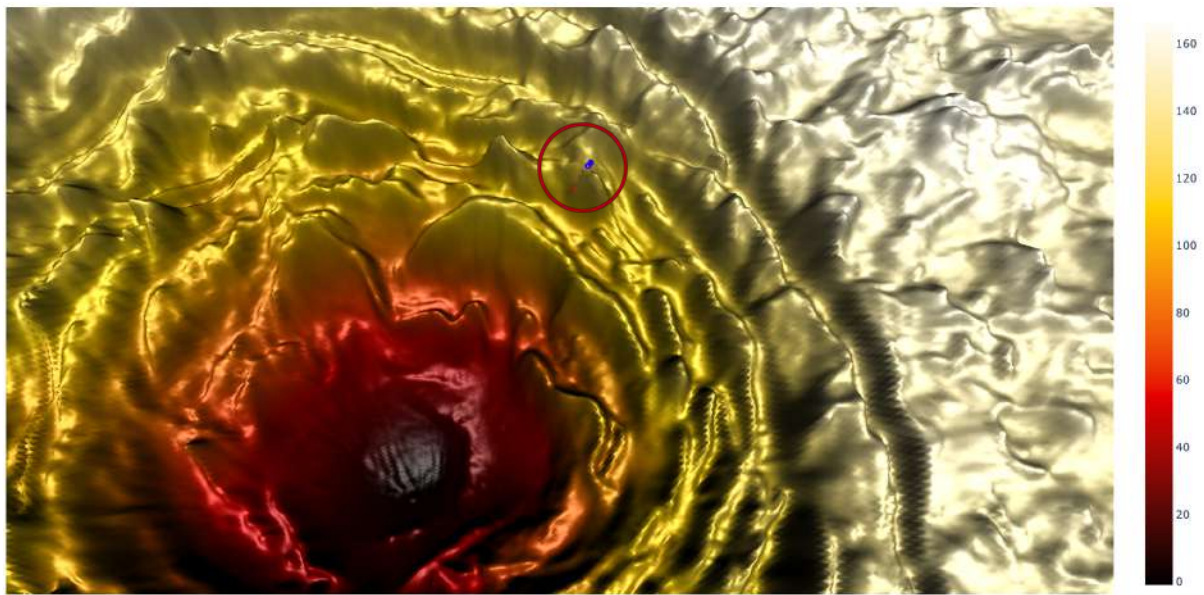
Pruebas:



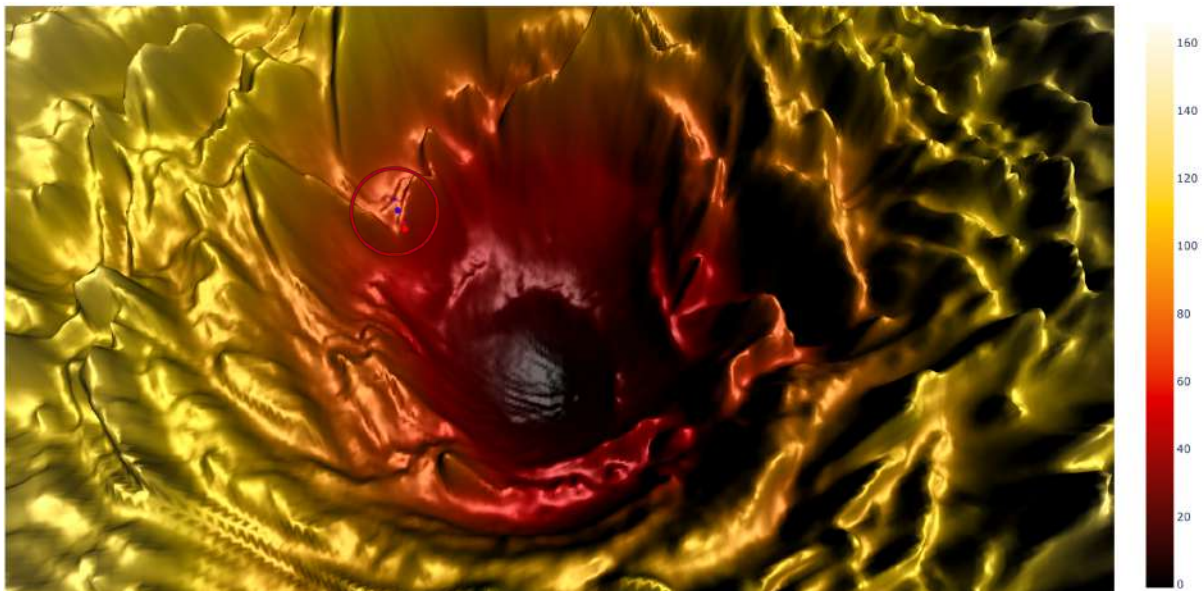
**Fig 1.** Ruta del rover partiendo de la coordenada X= 3350, Y= 5800



**Fig 2.** Ruta del rover partiendo de la coordenada X= 4078, Y= 4440

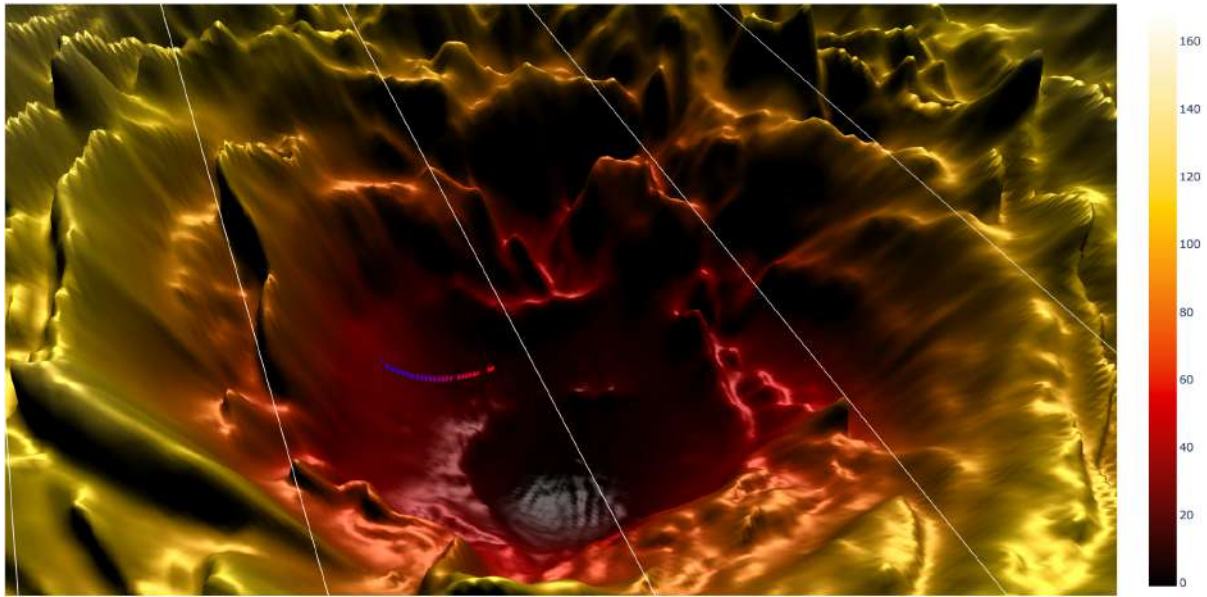


**Fig 3.** Ruta del rover partiendo de la coordenada  $X = 2049$ ,  $Y = 4912$

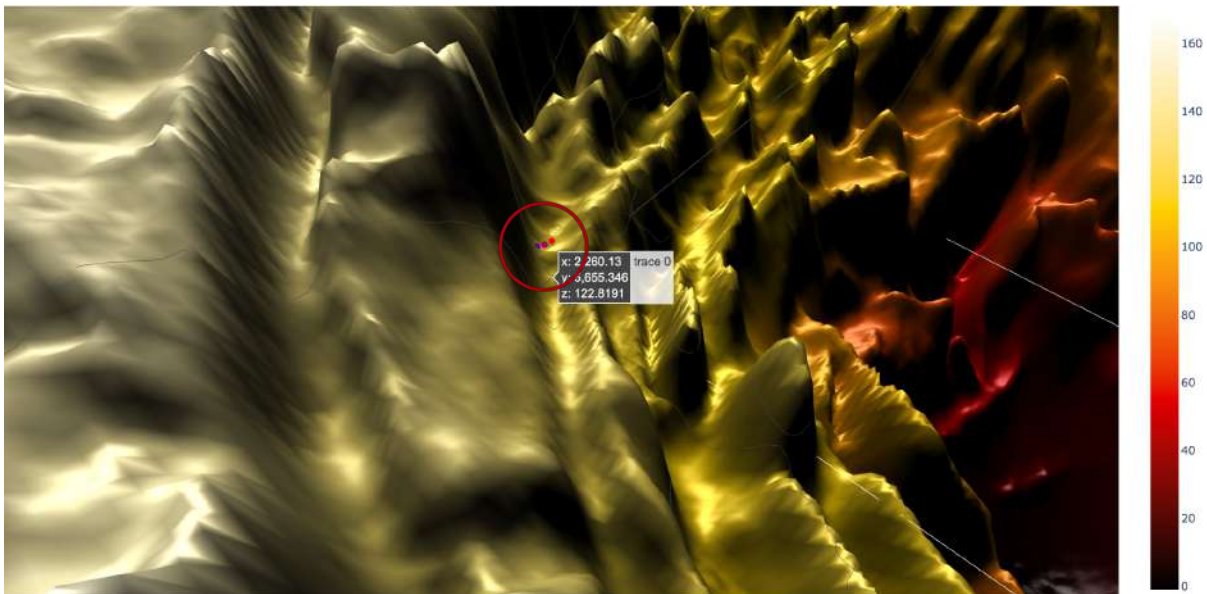


**Fig 4.** Ruta del rover partiendo de la coordenada  $X = 2572$ ,  $Y = 4671$





**Fig. 5** Ruta del rover partiendo de la coordenada X= 2622, Y= 5223



**Fig. 6** Ruta del rover partiendo de la coordenada X= 2260, Y= 5685

### **¿Hasta dónde es capaz el explorador de llegar?**

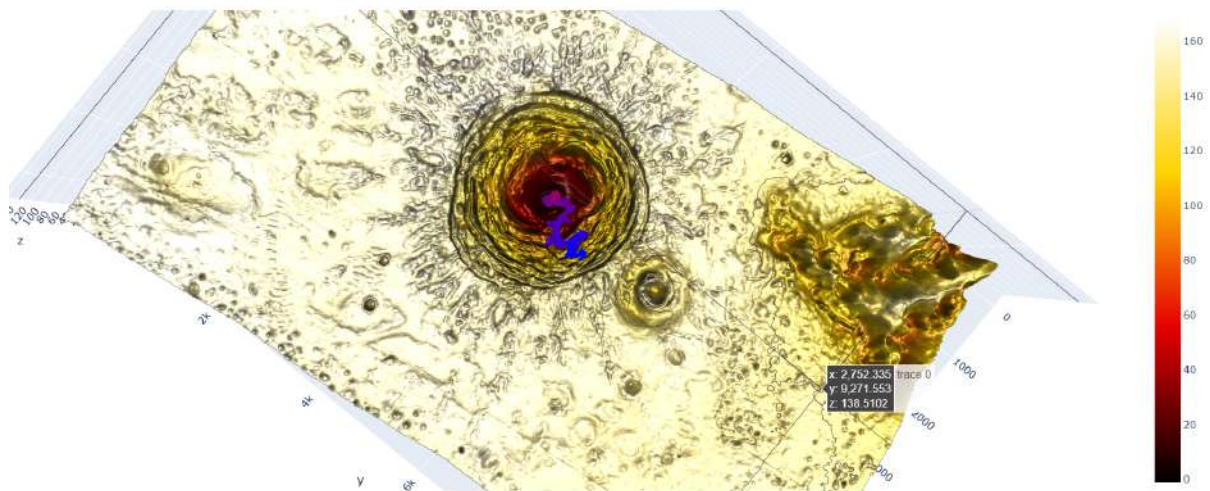
Al usar búsqueda local codiciosa el explorador no llega muy lejos. Debido al método solo puede ir hacia coordenadas en las cuales la profundidad es menor. Pero a veces para bajar, primero se debe subir y este método no nos permite ese lujo y en consecuencia se estanca en los mínimos.

### **¿Qué tan bueno es el algoritmo para llegar al fondo del cráter en los casos probados?**

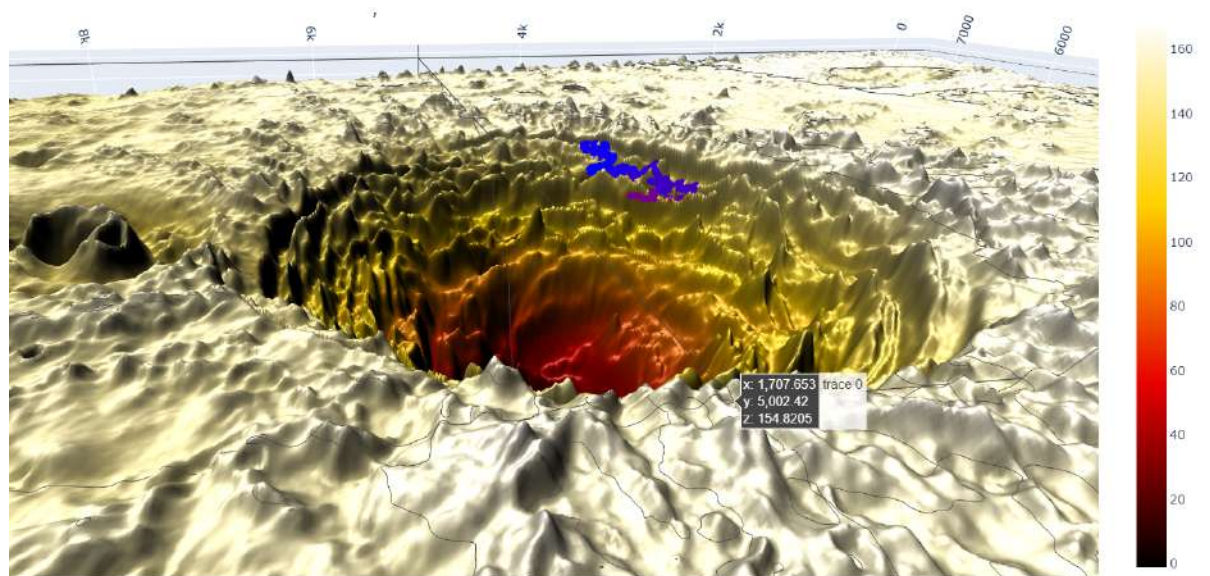
En algunos casos probados el algoritmo si fue capaz de hacer que el explorador bajara una distancia considerable, pero realmente no llega al fondo. Sin embargo, para la mayoría de los casos este algoritmo será completamente inefectivo debido al estancamiento en mínimos locales.

## Búsqueda con recocido simulado

### Pruebas:

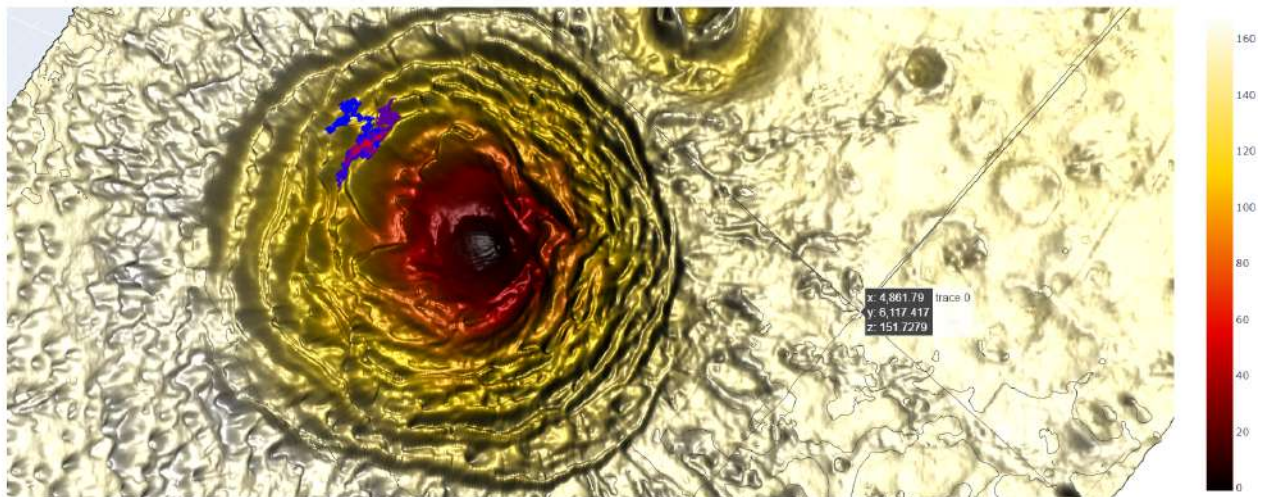


**Fig 7.** Ruta del rover partiendo de la coordenada X= 3350, Y= 5800

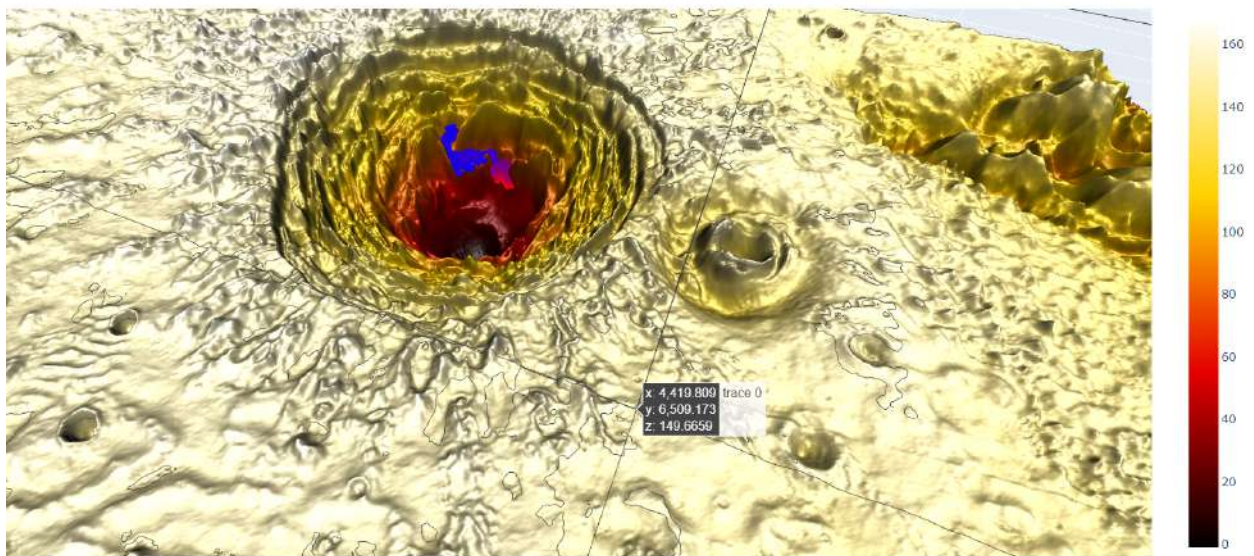


**Fig 8.** Ruta del rover partiendo de la coordenada X= 4078, Y= 4440



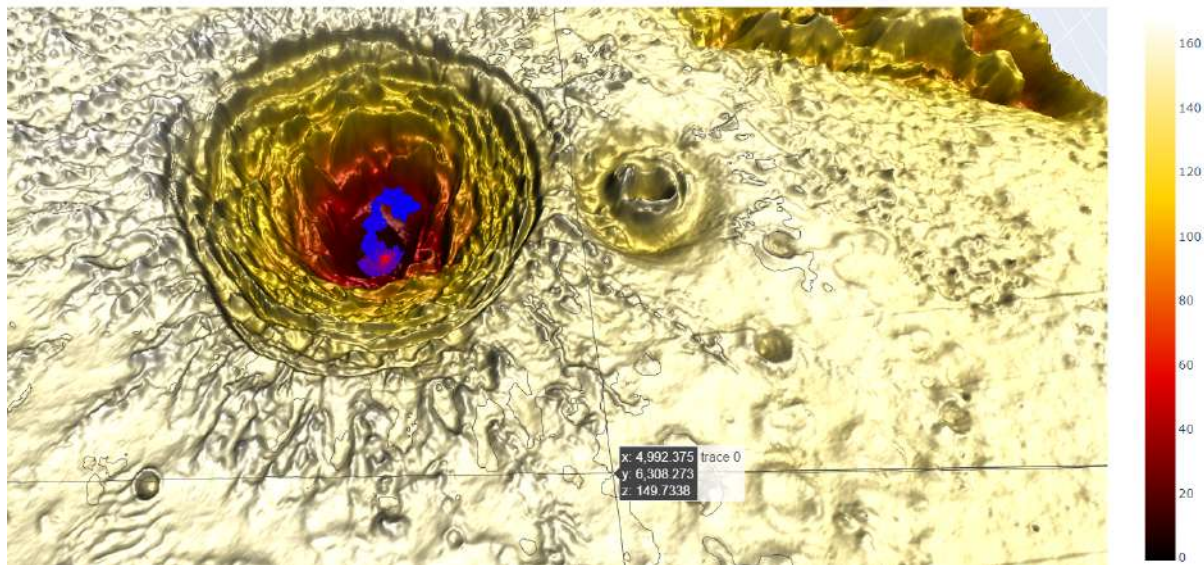


**Fig 9.** Ruta del rover partiendo de la coordenada X= 2049, Y= 4912

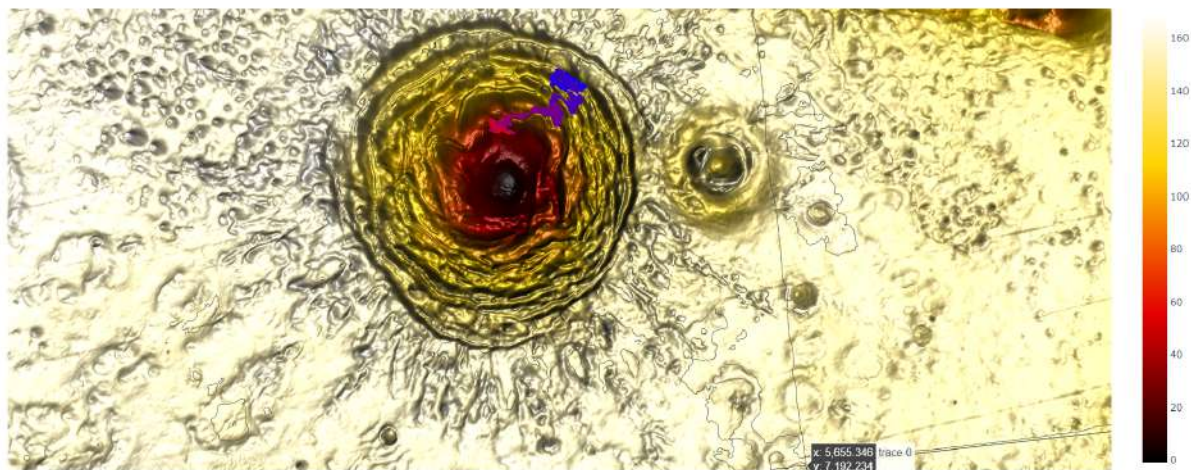


**Fig 10.** Ruta del rover partiendo de la coordenada X = 2572, Y= 4671





**Fig 11.** Ruta del rover partiendo de la coordenada X= 2622, Y= 5223



**Fig 12.** Ruta del rover partiendo de la coordenada X= 2260, Y= 5685

### ¿Qué algoritmo logra llegar más profundo en el cráter?

Definitivamente el algoritmo de recocido simulado es el que logra llegar a mayor profundidad dentro del cráter, ya que gracias al planteamiento de su algoritmo, (el comienzo con la caminata aleatoria), se le permite al rover explorar por el área a pesar de no bajar el costo en principio, para que después en la parte de búsqueda voraz, sepa por donde le

conviene moverse y no atascarse en mínimos locales, algo muy frecuente en la implementación del algoritmo de búsqueda voraz.

### **¿Recomendarían a los ingenieros del robot utilizar alguno de estos algoritmos?**

Sí, obviamente dependiendo del objetivo que se tenga en la misión, pero creemos que este tipo de algoritmos de búsqueda local son muy buenos cuando lo único que se busca es llegar a una posición o estado objetivo, en este caso el fondo del cráter, y que no importa necesariamente el optimizar el camino recorrido para llegar hasta dicha posición. Es decir, cuando queremos asegurar el llegar lo más cerca posible a nuestro estado objetivo, es sumamente conveniente utilizar algoritmos de búsqueda local.

### **Conclusiones**

*Andrés Olvera Varela*

Esta parte del reto ha sido la más retadora y desafiante de lograr. Requeríó del uso de todo nuestro conocimiento adquirido en estas últimas semanas para poder implementar un código que lograra solucionarlo. Después de haber trabajado con clases en python siento que debo estudiar más y entenderle mejor a este tipo de programación ya que en comparación con mis compañeros de clase siento que estuve demasiado atrasado en varios aspectos, desde constructores de clase hasta la utilización de los métodos de mi objeto creado, sé que es algo en lo que tengo que mejorar ya que es una muy buena habilidad a tener y más dentro de mi carrera que es ciencia de datos. Una vez empezamos a programar, el buscar la implementación de las funciones correctas fue un reto desafiante, el tema de pensar en los conceptos de costo, qué reciben y devuelven las funciones fue un tema algo abstracto, pero poco a poco empezamos a encontrar la solución.

Uno de los mejores aspectos una vez habíamos logrado la implementación correcta del código fue ver cómo los distintos tipos de búsqueda afectan la trayectoria del rover. Si partimos de la teoría y escuchar como un método que es capaz de regresar a algún estado anterior para encontrar mejores caminos, como recocido simulado, nos lleva a pensar que este es el mejor método en todos los casos, pero te vas dando cuenta de la importancia y relevancia de los demás métodos de búsqueda para cada situación en específico, y como no siempre el método que en teoría suena mejor suele serlo en la práctica.

En general me llevo un buen sabor de boca, como dije anteriormente me noté un tanto débil dentro del ámbito de la programación orientada a objetos, pero el hecho de ver esto como una debilidad no me sirve, al contrario, el poder reconocer mis debilidades para poder mejorarlas siento que es un muy buen aspecto dentro de la vida estudiantil.



La realización de este código fue más complicada de lo que creí en un principio, requirió de mucho tiempo de razonamiento y de pensamiento crítico y lógico para llegar a la solución. Primero que nada, el trabajar con clases y programación orientada a objetos, ya es un reto en sí pues considero que en mi caso es la segunda vez que trabajo con este tipo de programación, lo que puede llegar a ser retador, pero sin duda, una muy buena manera de aprender. Segundo, el plantear las funciones de la clase destinadas al seguimiento de los algoritmos de búsqueda local fue complicado, es decir razonar como debían de verse, que debían recibir y devolver cada una de las funciones de obtención de vecinos, costo, condiciones de aceptación o rechazo de los nuevos vecinos y las acciones posibles a realizar. Esto fue complicado pues debíamos anidar tanto el conocimiento que teníamos en búsqueda local, como el previo código que teníamos del recorrido en Marte, para así poder recorrer y guardar correctamente la ruta que se seguía, pues como habíamos mencionado, para los algoritmos de búsqueda local no es relevante el camino recorrido, si no la llegada al objetivo, por lo que plantear el como se iba a guardar el camino recorrido también fue algo interesante.

Fue también interesante el comparar ambos algoritmos en este programa de búsqueda para comprender como es que muchas veces pensamos que el algoritmo de recorrido simulado o en particular su parte de caminata aleatoria es innecesaria, impráctica o tardada y que el algoritmo de búsqueda codiciosa es más eficiente. Sin embargo, en esta situación por ejemplo era de suma importancia que primero le permitiéramos a nuestro algoritmo la exploración de muchos de los vecinos cercanos y sus costos, para que llegando el momento de aplicar la búsqueda voraz, el algoritmo ya conociera varios puntos y reconociera por donde es que le conviene más irse, además de que el recorrido simulado es muy bueno para prevenir el estancamiento en mínimos o máximos locales.

Me divertí mucho además jugando con las coordenadas de inicio y el cambio de algunos parámetros para ver como afectaban en el recorrido del Rover al ejecutar el programa. Esta materia, como ya lo había mencionado, me despertó el interés en estos campos de la inteligencia artificial y la optimización, pues eran campos que no consideraba anteriormente como atractivos y vaya que he cambiado de opinión actualmente.

Me gustó mucho el proyecto pues creo que a pesar de que sí fue un proyecto pesado y retador, considero que aprendí mucho de él y mejoré mucho mis habilidades respecto al trabajo en equipo, la comunicación efectiva de mis dudas y razonamientos y obviamente mis habilidades de programación y pensamiento lógico. Creo que al tratarse de un proyecto muy real y aterrizado a las exploraciones espaciales que si se llevan a cabo actualmente, te llena de motivación y alegría cuando ves que lo que aprendiste en clase y tus momentos de esfuerzo son totalmente aplicables y relevantes a gran escala.

Este proyecto ha sido uno de los más entretenidos que he implementado durante toda mi carrera e inclusive me atrevería a decir que en lo que llevo de estudiante en el Tecnológico de Monterrey. Fue un proyecto que me resultó bastante práctico ya que sentía que si era una aplicación útil de los conceptos vistos durante el periodo. Esta última entrega en lógica la verdad me resultó más sencillo de lo que esperaba. Creo que esto habla bien de la manera en la que el profesor dio la clase, ya que debido a esto la comprensión de los conceptos fue que a la hora de pensar en como se iba a dar resolución al problema no hubo complicaciones mayores. En donde si hubo complicaciones fue en la manera en la cual se podía y tenía que plasmar esta corriente de pensamiento a través de lenguajes de programación. El haber intentado hacer este proyecto me dio una clara señal de que las bases de programación que he adquirido durante la carrera no son suficientes para las clases de proyectos que debo hacer y también para los proyectos personales que quisiera hacer. Agradezco que este proyecto haya revelado dichas fallas, para que me ponga las pilas y también estudie por mi cuenta, porque si es claro que lo visto en la escuela no será suficiente para ser competitivo en el mundo laboral.

Ya hablando más a fondo de lo visto en el proyecto debo admitir que me resultó una excelente manera de enseñar las enormes aplicaciones de los algoritmos de búsqueda vistos en clase. El cráter me pareció que fue una excelente analogía para como se podría modelar una función horrorosa. Poder hacer optimización en dichas funciones sin algún algoritmo de búsqueda que utilice algún tipo de metaheurística sería una verdadera pesadilla y posiblemente sería una tarea imposible. Inclusive los métodos que utilizan restricciones serían poco efectivos contra cosas tan espantosas. Pero poder ver que si hay maneras de resolver este tipo de problemas de optimización con herramientas que ya hemos visto y comprendido es un verdadero alivio. Al momento de tomar la materia de optimización, si me encuentro algún problema con alguna función que se asemeje a algo tan espantoso, me acordaré de inmediato de este proyecto y ni siquiera pensaré dos veces en por lo menos intentar resolverlo de esta manera.

En general diría que fue un proyecto altamente divertido. Cansado, pero divertido. Estar jugando y probando con distintos algoritmos de búsqueda local y los parámetros en el problema y ver como el buen rover de exploración se quedaba atorado a las 10 iteraciones era bastante fascinante. Y ni siquiera mencionaré el gusto que da ver como si te da una gráfica que si mapee el camino del rover después de estar trabajando en el durante unas buenas horas. En retrospectiva, me resulta un poco deplorable de mi parte haberme tardado tanto en el proyecto. Pero reitero que esto fue por la programación y no la lógica. Me voy muy satisfecho con los aprendizajes adquiridos durante el periodo y me voy emocionado de lo que veré en el futuro de inteligencias artificiales y optimización.