



# Tecnológico de Monterrey

## **Evidencia 2. Avances y presentación del reto**

Sergio David Pimentel Pérez A01737331

Javier Cuatepotzo Hernandez A01737789

Anhuar Maldonado Aguilar A01737767

Juan Daniel Salmerón Mora A01737637

27 de Noviembre del 2024

Modelación de sistemas multiagentes con gráficas computacionales

(Gpo 502)

Luciano García Bañuelos

Iván Olmos Pineda

# Avances y presentación del reto

## Repositorio

[https://github.com/searching-ser/bin\\_packing](https://github.com/searching-ser/bin_packing)

## Discusión de la biblioteca 3d-bin-packing

La biblioteca 3d-bin-packing es una herramienta poderosa para resolver problemas de optimización tridimensional, especialmente en el acomodo de objetos rectangulares dentro de contenedores 3D. Su objetivo principal es maximizar el uso del espacio disponible o reducir la cantidad de contenedores necesarios, algo esencial en áreas como logística, almacenamiento y transporte.

En nuestro caso, trabajamos con cajas de diferentes tamaños y formas, no solo cubos. Cada caja representa un paquete con productos variados, lo que hace necesario encontrar la mejor manera de organizarlas en el contenedor. Para lograr esto, la biblioteca evalúa hasta cinco posibles rotaciones para cada caja. Cuando un producto llega, el sistema analiza cuál es la orientación más eficiente para que encaje mejor en el espacio disponible.

Esta flexibilidad en el manejo de rotaciones asegura un uso más eficiente del contenedor, optimizando el espacio y facilitando una distribución ordenada.

## Decisiones de diseño

### *1. Tamaño Variable de las Cajas:*

Las cajas que se colocarán dentro del contenedor tienen diferentes dimensiones (ancho, alto y profundidad), lo que añade complejidad al proceso de optimización del espacio. Esta variabilidad requiere algoritmos de colocación flexibles y eficientes que puedan adaptarse a distintos tamaños y formas.

### *2. Representación del Piso en OpenGL:*

El contenedor tridimensional se representará en un entorno gráfico usando OpenGL. El piso del contenedor servirá como referencia base para colocar las cajas, visualizando la disposición de los objetos en tiempo real.

### *3. Contenedor Único (Bin) en 3D Bin-Packing:*

Toda la simulación se centrará en llenar un único contenedor tridimensional. El objetivo es organizar las cajas de manera óptima dentro de este espacio limitado, maximizando el uso del volumen disponible.

#### *4. Implementación de Pathfinding con Julia:*

Se utilizarán algoritmos de pathfinding desarrollados en Julia para calcular la ruta óptima que debe seguir cada caja desde la entrada del contenedor hasta su posición final. Estos algoritmos evitarán colisiones con otras cajas o las paredes del contenedor.

### Arquitectura de la solución

El proceso comienza con la inicialización del modelo en el archivo ``packing.jl``, donde se generan las cajas con las dimensiones especificadas por el socio formador. Estas dimensiones se exportan a un archivo de texto (`` .txt``), el cual es leído posteriormente por un script en Python. Utilizando la biblioteca ``3dbinpacking``, Python calcula y devuelve a Julia las posiciones y rotaciones óptimas de cada caja dentro del contenedor. Con esta información, a cada caja se le asigna su rotación y posición específicas para garantizar un acomodo ordenado.

Para que el robot pueda localizar cada caja y transportarla al contenedor, se emplea el algoritmo A\* (A estrella), que calcula la ruta más eficiente hacia la siguiente caja en el orden predefinido. Esto permite que las cajas sean acomodadas correctamente dentro del contenedor, siguiendo las posiciones y rotaciones establecidas.

Finalmente, toda la información sobre las cajas y los robots, como posiciones y rotaciones, se transmite a OpenGL para su visualización. Esta comunicación se realiza a través de la Web API utilizando el puerto 8000, asegurando que los elementos se dibujen correctamente en la interfaz. Cuando el robot deposita una caja en el contenedor, OpenGL se encarga de actualizar su posición y rotación en tiempo real, mostrando el acomodo final de manera precisa.

### Heurísticas seguidas para coordinar los montacargas

La heurística utilizada para coordinar los montacargas inicia con la información proporcionada por el algoritmo de 3D bin packing. En este proceso, se genera una cola (queue) que define el orden en que las cajas deben ser introducidas en el contenedor (bin). Con base en esta cola, cada montacargas recibe la asignación de una caja específica que debe buscar y transportar, garantizando así que las cajas ingresen al contenedor en el orden correcto.

Para localizar las cajas en el espacio, se implementa el algoritmo A (A estrella)\*, que trabaja sobre una malla (grid) representativa del entorno. Este algoritmo permite identificar la posición objetivo y las casillas transitables para el montacargas. En escenarios donde hay múltiples robots operando, si uno encuentra una caja antes que otro y el orden de entrada no coincide, ambos robots esperan en el punto de entrega hasta que todas las cajas necesarias estén listas para ser colocadas en el contenedor en el orden adecuado.

Este enfoque asegura que las cajas sean introducidas en el contenedor de manera eficiente y en el orden requerido, optimizando el proceso de almacenamiento y garantizando la organización correcta al momento de acomodarlas.

## Reflexión personal

Javier Cuatepotzo Hernández

Estar en la clase de sistemas multiagentes es una experiencia que va más allá de lo técnico, me invitó a comprender cómo las interacciones entre agentes individuales pueden generar patrones complejos en un entorno. Durante el proceso de aprendizaje combinamos teoría y práctica en proyectos que me ayudaron a comprender cada uno de los temas. Al experimentar con herramientas y los modelos que los maestros nos brindaban, no sólo desarrollé habilidades técnicas, sino también una mayor capacidad para analizar y diseñar sistemas dinámicos.

Juan Daniel Salmerón Mora

La clase de sistemas multiagentes me permitió explorar muchos conceptos que en su mayoría eran parecidos a la clase del semestre anterior. A través de proyectos prácticos, como simulaciones y modelado de escenarios reales, pude aplicar conceptos teóricos y adquirir habilidades para analizar y diseñar sistemas dinámicos complejos. También el proceso de

aprendizaje de cómo generar mis propias funciones de movimiento de los agentes fue un proceso que personalmente se me dificultó pero con las prácticas hechas en clase me ayudó a comprender.

Anhuar Maldonado Aguilar

Estar en la clase de sistemas multiagentes fue una experiencia transformadora que me ayudó a comprender cómo resolver problemas con el uso de agentes con decisiones independientes. Los ejercicios prácticos y los modelos proporcionados por los profesores fueron fundamentales para reforzar mi aprendizaje, permitiéndome aplicar estas ideas a la solución del reto de la situación problema la cual puede tener una aplicación real.

Sergio David Pimentel Pérez

La clase de sistemas multiagentes fue un espacio para reflexionar sobre cómo la tecnología puede imitar comportamientos colectivos donde podemos tener diversos agentes que cumplan con tareas individuales que lleven a un bien colectivo. Este enfoque me llevó a valorar la importancia de diseñar reglas simples que generan soluciones eficientes y escalables en sistemas complejos. Personalmente no me imaginaba el concepto de multiagentes como lo puedo definir ahora. Veo muchos problemas que pueden ser resueltos por medio de la aplicación de multiagentes.