
PROYECTO 2 – SISTEMA DE RIEGO AUTOMATIZADO

202405935 – Erwin Alejandro Girón Menéndez

Resumen

El proyecto consiste en un sistema integral de riego inteligente para invernaderos que utiliza drones autónomos para optimizar el proceso de irrigación de plantas, desarrollado como una aplicación web completa con Flask que permite cargar configuraciones de invernaderos mediante archivos XML, simular rutas de riego eficientes, y generar planes optimizados de irrigación. El sistema implementa estructuras de datos personalizadas (listas enlazadas y diccionarios propios) sin utilizar las estructuras nativas de Python, incorpora algoritmos de optimización para calcular automáticamente los tiempos mínimos de ejecución, las cantidades exactas de agua y fertilizante necesarias, y las secuencias de instrucciones detalladas para cada dron en función de las características específicas de cada planta y las restricciones operativas de los drones. La interfaz web proporciona visualización interactiva de los datos de entrada, simulación en tiempo real con tablas de instrucciones por tiempo que muestran las acciones de cada dron segundo a segundo, generación automática de reportes XML con los planes optimizados, y visualización gráfica del estado de las estructuras de datos abstractas utilizando Graphviz, creando así una solución completa que combina teoría computacional avanzada con aplicaciones prácticas en agricultura automatizada, demostrando la implementación efectiva de algoritmos de optimización y estructuras de datos personalizadas en un contexto real de gestión agrícola inteligente.

Palabras clave

Graphviz, Listas Enlazadas, Nodos, TDA, planta

Abstract

The project consists of an integrated smart irrigation system for greenhouses that uses autonomous drones to optimize plant irrigation. Developed as a comprehensive web application using Flask, it allows users to upload greenhouse configurations via XML files, simulate efficient irrigation routes, and generate optimized irrigation plans. The system implements custom data structures (linked lists and dictionaries) without relying on Python's built-in structures. It incorporates optimization algorithms to automatically calculate the minimum execution times, the precise amounts of water and fertilizer required, and detailed instruction sequences for each drone, based on the specific characteristics of each plant and the operational constraints of the drones. The web interface provides interactive visualization of input data, real-time simulation with time-based instruction tables showing each drone's actions second by second, automatic generation of XML reports with the optimized plans, and graphical visualization of the state of the abstract data structures using Graphviz. This creates a complete solution that combines advanced computational theory with practical applications in automated agriculture, demonstrating the effective implementation of optimization algorithms and custom data structures in a real-world smart farming management context.

Keywords

Graphviz, Linked List, Nodos, TDA, planta

Introducción

La agricultura moderna enfrenta desafíos significativos relacionados con la optimización de recursos hídricos y la automatización de procesos de irrigación, especialmente en entornos controlados como invernaderos donde la precisión en el riego es fundamental para maximizar la productividad y minimizar el desperdicio de recursos. En este contexto, el desarrollo de sistemas inteligentes que integren tecnologías de automatización con algoritmos de optimización representa una solución innovadora para abordar estas problemáticas. El presente proyecto desarrolla un sistema integral de riego automatizado que utiliza drones para la irrigación de plantas en invernaderos, implementando estructuras de datos personalizadas y algoritmos de optimización para generar planes de riego eficientes. El sistema no solo simula y optimiza rutas de navegación para drones, sino que también calcula automáticamente los recursos necesarios (agua, fertilizante y tiempo) mientras proporciona una interfaz web interactiva para la gestión y visualización de todo el proceso de planificación y ejecución .

Arquitectura del Sistema

El sistema se estructura en una arquitectura modular de cinco capas principales que garantizan la separación de responsabilidades y la escalabilidad del proyecto. La **capa de presentación** implementa una interfaz web desarrollada con Flask que permite la interacción del usuario con el sistema, incluyendo la carga de archivos XML de configuración, visualización de datos de invernaderos y plantas, y la generación de reportes. La **capa de lógica de negocio** contiene los algoritmos de simulación y optimización que procesan la información de entrada para generar planes de riego eficientes, calculando rutas óptimas para los drones y determinando los recursos necesarios. La **capa de estructuras de datos** implementa tipos de datos abstractos personalizados, específicamente listas enlazadas simples y

diccionarios personalizados, desarrollados desde cero sin utilizar las estructuras nativas de Python. La **capa de persistencia** maneja la lectura y escritura de archivos XML tanto para la entrada de configuraciones como para la salida de planes optimizados. Finalmente, la **capa de visualización** genera representaciones gráficas de las estructuras de datos utilizando Graphviz para proporcionar retroalimentación visual del estado interno del sistema.

Componentes Principales y Funcionalidades

El **motor de simulación** constituye el núcleo del sistema, implementando algoritmos que modelan el comportamiento de drones navegando por hileras de plantas en invernaderos. Este componente utiliza técnicas de optimización para determinar secuencias de movimientos que minimizan el tiempo total de ejecución mientras garantizan que todas las plantas reciban la cantidad adecuada de agua y fertilizante según sus requerimientos específicos. El **sistema de gestión de datos** emplea las estructuras personalizadas para almacenar y manipular información sobre invernaderos, plantas, drones y planes de riego, asegurando que todas las operaciones se realicen sin depender de las estructuras nativas de Python. El **generador de reportes** produce archivos XML estructurados que contienen los planes optimizados, incluyendo secuencias detalladas de instrucciones para cada dron, cálculos precisos de recursos, y tiempos de ejecución, facilitando la implementación práctica de los planes generados. La **interfaz de usuario** proporciona herramientas intuitivas para cargar configuraciones, visualizar el progreso de la simulación en tiempo real mediante tablas de instrucciones por tiempo, y descargar los resultados en formatos estándar.

Metodología de implementación del proyecto

La implementación sigue una metodología de desarrollo incremental que prioriza la funcionalidad core antes de agregar características avanzadas. Los ****algoritmos de optimización**** emplean técnicas de búsqueda y planificación para determinar rutas eficientes, considerando restricciones como la capacidad de carga de los drones, los requerimientos específicos de cada planta, y las limitaciones de movimiento dentro del invernadero. El ****sistema de cálculo de recursos**** utiliza modelos matemáticos para determinar las cantidades exactas de agua y fertilizante necesarias, basándose en las características de las plantas y los parámetros de eficiencia de los drones. La ****visualización de estructuras de datos**** implementa algoritmos de graficación que representan el estado interno de las listas enlazadas y diccionarios personalizados, proporcionando una herramienta educativa valiosa para comprender el funcionamiento de estas estructuras. El ****procesamiento de archivos XML**** utiliza parsers personalizados que validan la estructura de entrada y generan salidas conformes a esquemas predefinidos, asegurando la interoperabilidad con otros sistemas de gestión agrícola.

Conclusiones

1. **Integración Exitosa de Teoría y Práctica Computacional**

El proyecto demuestra de manera efectiva cómo los conceptos teóricos de estructuras de datos y algoritmos de optimización pueden aplicarse a problemas reales del sector agrícola. La implementación de listas enlazadas y diccionarios personalizados, junto con algoritmos de búsqueda y optimización, proporciona una base sólida que conecta la educación académica con aplicaciones prácticas en la automatización agrícola moderna.

2. **Arquitectura Modular y Escalable**

La estructura del sistema en capas bien definidas (presentación, lógica de negocio, estructuras de datos, persistencia y visualización) facilita el mantenimiento, la extensión y la reutilización del código. Esta arquitectura modular permite agregar nuevas funcionalidades como diferentes tipos de drones, sensores adicionales, o algoritmos de optimización más avanzados sin afectar el funcionamiento de otros componentes del sistema.

3. **Optimización Efectiva de Recursos Agrícolas**

El sistema logra optimizar significativamente el uso de recursos críticos en la agricultura, calculando con precisión las cantidades exactas de agua y fertilizante necesarias, minimizando los tiempos de operación, y generando rutas eficientes para los drones. Esta optimización no solo reduce costos operativos sino que también contribuye a prácticas agrícolas más sostenibles y responsables con el medio ambiente.

4. **Interfaz Usuario-Sistema Intuitiva y Funcional**

La implementación de una interfaz web completa que permite la carga de configuraciones XML, visualización en tiempo real de simulaciones, y generación automática de reportes, demuestra la importancia de crear sistemas accesibles para usuarios no técnicos. La capacidad de visualizar el estado de las estructuras de datos mediante gráficos Graphviz añade un valor educativo significativo al proyecto.

5. **Potencial de Aplicación Industrial y Escalabilidad**

El sistema desarrollado presenta características que lo hacen viable para implementación en entornos industriales reales, con capacidad para manejar múltiples invernaderos, diferentes tipos de cultivos, y planes de riego complejos. La generación de archivos

XML estándar facilita la integración con otros sistemas de gestión agrícola, mientras que la arquitectura permite escalar desde operaciones pequeñas hasta grandes complejos agrícolas automatizados.

