

Optimización con inteligencia artificial del despacho de riego solar en cultivos de pistacho

Ing. Tomás Corteggiano

Carrera de Especialización en Inteligencia Artificial

Director: Dr. Ing. Carlos Larisson

Jurados:

Jurado 1 (pertencia)
Jurado 2 (pertencia)
Jurado 3 (pertencia)

Optimización con inteligencia artificial del despacho de riego solar en cultivos de pistacho

Ing. Tomás Corteggiano

Carrera de Especialización en Inteligencia Artificial

Director: Dr. Ing. Carlos Larisson

Jurados:

Jurado 1 (pertencia)
Jurado 2 (pertencia)
Jurado 3 (pertencia)

Ciudad de Mendoza, noviembre de 2025

Ciudad de Mendoza, noviembre de 2025

Resumen

La presente memoria describe el desarrollo de un sistema avanzado de optimización del riego, cuyo fin es obtener un cronograma de despacho de riego óptimo para una plantación de pistachos, diseñado para una empresa del sector agrícola. Con el propósito de superar las limitaciones de los modelos físicos puros, se aplicaron conocimientos de modelado físico con técnicas de Inteligencia Artificial (IA) para crear un gemelo digital híbrido de alta precisión, el cual corrige los errores residuales del simulador tradicional. Además, se utilizaron algoritmos de búsqueda avanzados (Algoritmo Genético) para resolver eficientemente el complejo problema de programación del despacho, identificando como un Problema de Programación Mixta-Entera No Lineal (MINLP). El sistema se diseñó sobre una función de costos unificada que permite al agrónomo balancear objetivos claves como la uniformidad del riego, la eficiencia del bombeo y la sostenibilidad del acuífero.

Resumen

La presente memoria describe el desarrollo de un sistema avanzado para la optimización del riego en una plantación de pistachos, con el objetivo de generar un cronograma de despacho de riego óptimo para una empresa agrícola. El sistema busca balancear objetivos clave como la uniformidad del riego, la eficiencia del bombeo y la sostenibilidad del acuífero mediante una función de costos unificada.

Para superar las limitaciones de los modelos físicos, se creó un gemelo digital híbrido de alta precisión combinando modelado físico con técnicas de Inteligencia Artificial (IA) para corregir errores residuales. Además, se utilizó un Algoritmo Genético, un algoritmo de búsqueda avanzado, para resolver eficientemente el complejo Problema de Programación Mixta-Entera No Lineal (MINLP) asociado al despacho de riego.

Índice general

Resumen	I
1. Introducción	1
1.1. Contexto y problemática del riego solar	1
1.2. Estado del arte: modelado y optimización	2
1.2.1. El modelado: consistencia vs. precisión	2
1.2.2. Optimización: optimalidad vs. factibilidad	3
1.3. Motivación	4
1.4. Objetivos y alcance	5
2. Introducción específica	7
2.1. Teoría del modelado híbrido: corrección de residuales	7
2.2. Formulación del problema de optimización	7
2.3. Teoría de optimización: algoritmos genéticos	7
2.4. Manejo de restricciones	7
3. Diseño e implementación	9
3.1. Arquitectura General del Software	9
3.2. Implementación del gemelo digital híbrido	9
3.3. Implementación del optimizador de despacho (GA)	9
3.4. Implementación de funciones de costo y penalizaciones	9
4. Ensayos y resultados	11
4.1. Entorno y banco de pruebas	11
4.2. Pruebas del gemelo digital híbrido	11
4.3. Caso de estudio: simulación de despacho	11
5. Conclusiones	13
5.1. Resultados obtenidos	13
5.2. Trabajo futuro	13

Índice general

Resumen	I
1. Introducción	1
1.1. Contexto y problemática del riego solar	1
1.2. Estado del arte: modelado y optimización	2
1.2.1. El modelado: consistencia vs. precisión	2
1.2.2. Optimización: optimalidad vs. factibilidad	3
1.3. Motivación	4
1.4. Objetivos y alcance	5
2. Introducción específica	7
2.1. Teoría del modelado híbrido: corrección de residuales	7
2.1.1. Componente físico ($f_{físico}$)	8
2.1.2. Componente corrector (f_{AI})	8
2.2. Formulación del problema de optimización	9
2.2.1. Formulación matemática formal	9
Función objetivo ponderada	9
Restricciones del problema	10
2.3. Teoría de optimización: algoritmos genéticos	10
2.3.1. Anatomía de un algoritmo genético	10
Codificación del cromosoma (representación de la solución)	10
Operadores de selección	11
Operadores de cruce (<i>crossover</i> / recombinación)	12
Operadores de mutación (diversidad)	12
2.4. Manejo de restricciones: funciones de penalización	12
3. Diseño e implementación	15
3.1. Arquitectura General del Software	15
3.2. Implementación del gemelo digital híbrido	15
3.3. Implementación del optimizador de despacho (GA)	15
3.4. Implementación de funciones de costo y penalizaciones	15
4. Ensayos y resultados	17
4.1. Entorno y banco de pruebas	17
4.2. Pruebas del gemelo digital híbrido	17
4.3. Caso de estudio: simulación de despacho	17
5. Conclusiones	19
5.1. Resultados obtenidos	19
5.2. Trabajo futuro	19
Bibliografía	21

Índice de figuras

1.1. Diagrama conceptual de los paradigmas de modelado: caja blanca (física), caja negra (IA pura) y caja gris (híbrido).	3
1.2. Diagrama en bloques de la arquitectura completa del sistema de riego.	6

Índice de figuras

1.1. Diagrama conceptual de los paradigmas de modelado: caja blanca (física), caja negra (IA pura) y caja gris (híbrido).	3
---	---

Índice de tablas

Índice de tablas

2.1.	Nomenclatura del Modelo de Optimización.	9
2.2.	Análisis comparativo de operadores de selección en AG.	11
2.3.	Análisis comparativo de operadores de cruce relevantes.	12

Capítulo 1

Introducción

En este capítulo se presenta una introducción general **al proyecto**. Se describe la problemática fundamental del riego solar **que motivó este trabajo** y se analiza el estado del arte en las áreas **clave** del modelado de sistemas y su optimización. Finalmente, se expone la **motivación** de la solución propuesta y se delimitan los objetivos y el alcance del **software que fue desarrollado**.

1.1. Contexto y problemática del riego solar

La implementación de sistemas de riego 100 % solares es una solución **tecnológica** de amplio uso en el sector agrícola, particularmente en plantaciones que operan de forma autónoma (es decir, en modo *off-grid*), donde la red eléctrica de **distribución** pública no está disponible o su conexión resulta **económicamente** inviable. Estas instalaciones se componen típicamente de una **Capa Física** (la bomba sumergible y **las válvulas de campo**) y una **Capa de Control y Adquisición** (el variador de frecuencia (VFD) y el PLC que **comandan** dicho hardware). En la operación estándar, esta arquitectura de control es robusta para ejecutar **comandos**, pero carece de la capa superior de optimización que define el plan de operación.

Esta ausencia de una capa de gestión inteligente traslada por completo la carga de optimización al operario. El despacho de agua se convierte en una consecuencia pasiva de la **meteorología —siguiendo** la curva de irradiación solar— o, en el mejor de los casos, sigue un plan estático (basado en temporizadores) definido manualmente por el ingeniero agrónomo. Este enfoque manual representa una carga significativa de horas de ingeniería y es inherentemente subóptimo, ya que no puede reaccionar a la variabilidad diaria de las condiciones hídricas o energéticas.

La necesidad de esta optimización inteligente se vuelve crítica en contextos agrícolas como el del cultivo de pistacho, **donde** los ciclos de inversión son largos y la planta requiere varios años para entrar en **producción**. En regiones donde el agua es un recurso escaso o sujetas a sequías estacionales, una gestión **deficiente** del riego no solo reduce la eficiencia, sino que compromete la viabilidad y **sostenibilidad** a largo plazo de la plantación.

Por lo tanto, la problemática central que este trabajo aborda es esta falta de inteligencia operativa. La proliferación de la telemetría y los sensores de campo (IoT) en la agricultura moderna han generado un vasto potencial de optimización que, **en la mayoría de los sistemas de riego, permanece sin explotar. Los controladores**

Capítulo 1

Introducción

En este capítulo se presenta una introducción general **del trabajo**. Se describe la problemática fundamental del riego solar y se analiza el estado del arte en las áreas **claves** del modelado de sistemas y su optimización. Finalmente, se expone la **justificación** de la solución propuesta y se delimitan los objetivos y el alcance del **desarrollo**.

1.1. Contexto y problemática del riego solar

La implementación de sistemas de riego 100 % solares es una solución **tecnológica** de amplio uso en el sector agrícola, particularmente en plantaciones que operan de forma autónoma (es decir, en modo *off-grid*[1]), donde la red eléctrica de **distribución** pública no está disponible o su conexión resulta **económicamente** inviable. Estas instalaciones se componen típicamente de una capa física, que cuenta con una bomba sumergible y válvulas de **campo, y una capa de control y adquisición, que cuenta con un** variador de frecuencia (VFD) y **un** PLC que **comanda** dicho hardware. En la operación estándar, esta arquitectura de control es robusta para ejecutar **comandos**, pero carece de la capa superior de optimización que define el plan de operación.

Esta ausencia de una capa de gestión inteligente traslada por completo la carga de optimización al operario. El despacho de agua se convierte en una consecuencia pasiva de la **meteorología, siguiendo** la curva de irradiación solar, o, en el mejor de los casos, sigue un plan estático (basado en temporizadores) definido manualmente por el ingeniero agrónomo. Este enfoque manual representa una carga significativa de horas de ingeniería y es inherentemente subóptimo, ya que no puede reaccionar a la variabilidad diaria de las condiciones hídricas o energéticas.

La necesidad de esta optimización inteligente se vuelve crítica en contextos agrícolas como el del cultivo de pistacho, **en que** los ciclos de inversión son largos y la planta requiere varios años para entrar en **producción** [2, 3, 4]. En regiones donde el agua es un recurso escaso o sujetas a sequías estacionales, una gestión **deficiente** del riego no solo reduce la eficiencia, sino que compromete la viabilidad y **sostenibilidad** a largo plazo de la plantación.

Por lo tanto, la problemática central que este trabajo aborda es esta falta de inteligencia operativa. La proliferación de la telemetría y los sensores de campo (IoT) en la agricultura moderna han generado un vasto potencial de optimización que,

actuales se limitan a registrar estos datos o a reaccionar a umbrales simples, pero carecen de la capacidad de utilizarlos para una planificación proactiva.

El verdadero desafío que se resuelve en este trabajo es integrar la **Inteligencia Artificial** para dos tareas claves: primero, para transformar este flujo de datos en **conocimiento** predictivo (un modelo de alta fidelidad del sistema); y **segundo**, para actuar sobre dicho conocimiento. Se requiere un motor de **optimización** —en este caso, un **Algoritmo Genético**— que funcione como un buscador capaz de **generar** un cronograma de despacho óptimo a largo plazo. Esta capacidad de búsqueda y planificación proactiva, que balancea múltiples objetivos, es la pieza de gestión que falta en los sistemas actuales, **los cuales** obligan a operar muy por debajo de su potencial de eficiencia.

1.2. Estado del arte: modelado y optimización

El problema central de este **trabajo** —generar un cronograma de despacho de riego óptimo para un sistema solar **autónomo**— no es un desafío singular, sino la intersección de dos conceptos técnicos que el estado del arte ha abordado históricamente por vías separadas:

1. **El desafío del modelado:** se requiere un simulador de muy alta fidelidad, un **"gemelo digital"**, que actúe como la función de evaluación del **optimizador**. Este modelo debe predecir con precisión la respuesta hidráulica (**caudales**, presiones, nivel freático) ante cualquier plan de riego candidato.
2. **El desafío de la optimización:** se necesita un algoritmo de búsqueda capaz de encontrar la combinación óptima de decisiones (qué válvula abrir y cuándo) dentro de un espacio de soluciones de dimensionalidad masiva, respetando un conjunto de restricciones físicas (energía, agua, presión) y un tiempo de cómputo estricto.

A continuación, se analiza el panorama de las soluciones existentes para cada uno de estos desafíos, identificando las limitaciones que motivan el desarrollo de este trabajo.

1.2.1. El modelado: consistencia vs. precisión

El algoritmo de optimización es, por naturaleza, agnóstico a la física del **problema**; **depende** estrictamente de una **función de evaluación** (en este caso, el gemelo digital) para cuantificar la calidad, o *fitness*, de un cronograma de riego **candidato**. El éxito de la optimización, por lo tanto, depende enteramente de la **fidelidad** de este modelo. El enfoque tradicional para construir esta función se basa en modelos de **"caja blanca"** o de primeros principios, como el estándar **industrial EPANET**, que resuelven las ecuaciones fundamentales de la mecánica de fluidos. Su fortaleza indiscutible es la **consistencia física**: al estar **basados** en leyes **universales**, son robustos y pueden generalizar su comportamiento a escenarios de operación **novedosos**. Sin embargo, su debilidad es la **incertidumbre** en sus **parámetros físicos**; el rendimiento del modelo depende de una calibración precisa de variables estáticas (ej. rugosidad de tuberías) que son **difíciles** de medir y cambian con el **tiempo**. Esto genera un **error sistemático** o **"residual"** entre la simulación idealizada y la realidad del campo.

en la mayoría de los sistemas de riego, **permanece sin explotar** [5]. Los **controladores** actuales se limitan a registrar estos datos o a reaccionar a umbrales simples, pero carecen de la capacidad de utilizarlos para una planificación proactiva.

El verdadero desafío que se resuelve en este trabajo es integrar la **inteligencia artificial** para dos tareas claves: primero, para transformar este flujo de datos en **conocimiento** predictivo (un modelo de alta fidelidad del sistema); y **segundo**, para actuar sobre dicho conocimiento. Se requiere un motor de **optimización**, en este caso, un **algoritmo genético**, que funcione como un buscador capaz de generar un cronograma de despacho óptimo a largo plazo. Esta capacidad de búsqueda y planificación proactiva, que balancea múltiples objetivos, es la pieza de gestión que falta en los sistemas actuales, **que** obligan a operar muy por debajo de su potencial de eficiencia.

1.2. Estado del arte: modelado y optimización

El problema central de este **trabajo**, **generar** un cronograma de despacho de riego óptimo para un sistema solar **autónomo**, no es un desafío singular, sino la intersección de dos conceptos técnicos que el estado del arte ha abordado históricamente por vías separadas:

1. El desafío del modelado: se requiere un simulador de muy alta fidelidad, un **gemelo digital**, que actúe como la función de evaluación del **optimizador**. Este modelo debe predecir con precisión la respuesta hidráulica (**caudales**, presiones, nivel freático) ante cualquier plan de riego candidato.
2. El desafío de la optimización: se necesita un algoritmo de búsqueda capaz de encontrar la combinación óptima de decisiones (qué válvula abrir y cuándo) dentro de un espacio de soluciones de dimensionalidad masiva, respetando un conjunto de restricciones físicas (energía, agua, presión) y un tiempo de cómputo estricto.

A continuación, se analiza el panorama de las soluciones existentes para cada uno de estos desafíos, identificando las limitaciones que motivan el desarrollo de este trabajo.

1.2.1. El modelado: consistencia vs. precisión

El algoritmo de optimización es, por naturaleza, agnóstico a la física del **problema**. **Depende** estrictamente de una función de evaluación [6] (en este caso, el gemelo digital) para cuantificar la calidad, o *fitness*, de un cronograma de riego **candidato**. El éxito de la optimización, por lo tanto, depende enteramente de la **fidelidad** de este modelo. El enfoque tradicional para construir esta función se basa en modelos de **caja blanca** o de primeros principios, como el estándar **industrial EPANET** [7], que resuelven las ecuaciones fundamentales de la mecánica de fluidos [8, 9]. Su fortaleza indiscutible es la **consistencia física**: al estar **basados** en leyes **universales**, son robustos y pueden generalizar su comportamiento a escenarios de operación **novedosos** [10]. Sin embargo, su debilidad es la **incertidumbre** en sus **parámetros físicos**. El rendimiento del modelo depende de una calibración precisa de variables estáticas (ej. rugosidad de tuberías) que son **difíciles** de medir y cambian con el **tiempo** [11]. Esto genera un error sistemático o **residual** entre la simulación idealizada y la realidad del campo.

Para solucionar esta brecha de precisión, en el extremo opuesto del espectro se encuentran los modelos puramente empíricos de “caja negra” (IA), que buscan aprender el comportamiento del sistema basándose únicamente en datos **históricos** de telemetría. La ventaja de este enfoque es la capacidad de alcanzar una alta precisión, capturando dinámicas complejas que el modelo físico **ignora**. No obstante, esta solución es **frágil**: requiere volúmenes masivos de datos para cubrir todo el espacio **operativo**, no es robusta para extrapolar a escenarios no vistos y, crucialmente, no garantiza la consistencia **física**.

El panorama de las soluciones de modelado presenta, por lo tanto, un dilema **fundamental**: el ingeniero debe elegir entre la **consistencia física** de la “caja blanca” (sufriendo su imprecisión) o la **precisión** de la “caja negra” (sufriendo su falta de **robustez**). Esta dicotomía, junto con la alternativa conceptual de un modelo híbrido o de “caja gris” que busca unificar **ambas**, se ilustra en la figura 1.1.

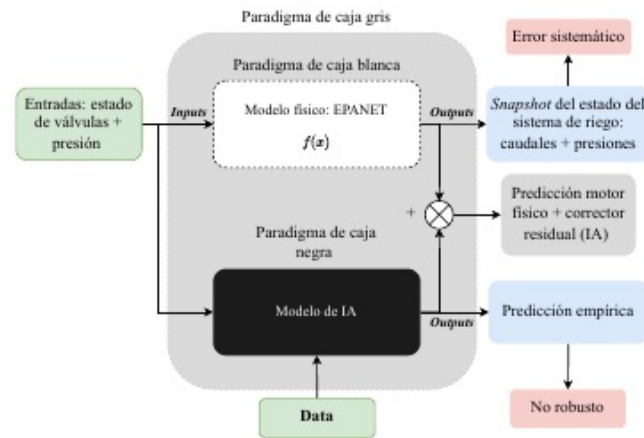


FIGURA 1.1. Diagrama conceptual de los paradigmas de modelado: caja blanca (física), caja negra (IA pura) y caja gris (híbrido).

1.2.2. Optimización: optimalidad vs. factibilidad

El segundo desafío fundamental radica en el propio algoritmo de búsqueda. La elección de la herramienta de optimización no es arbitraria, sino que está **dictada** por la naturaleza matemática inherente al problema del despacho de **riego**. Al **formularlo**, el problema se revela como un **Problema de Programación Mixta-Entera No Lineal (MINLP)**, una clasificación con profundas **implicaciones computacionales**.

La naturaleza **Mixta-Entera (MI)** surge de las variables de decisión: el estado de una válvula o bomba en un intervalo de tiempo es una decisión fundamentalmente binaria (0 para cerrado o 1 para abierto). A esto se suma la **No Linealidad**

Para solucionar esta brecha de precisión, en el extremo opuesto del espectro se encuentran los modelos puramente empíricos de **caja negra** (IA), que buscan aprender el comportamiento del sistema basándose únicamente en datos **históricos** de telemetría. La ventaja de este enfoque es la capacidad de alcanzar una alta precisión, capturando dinámicas complejas que el modelo físico **ignora** [12]. No obstante, esta solución es **frágil**: requiere volúmenes masivos de datos para cubrir todo el espacio **operativo** [13], no es robusta para extrapolar a escenarios no vistos y, crucialmente, no garantiza la consistencia **física** [13].

El panorama de las soluciones de modelado presenta, por lo tanto, un dilema **fundamental**: el ingeniero debe elegir entre la consistencia física de la **caja blanca** (sufriendo su imprecisión) o la precisión de la **caja negra** (sufriendo su falta de **robustez**). Esta dicotomía, junto con la alternativa conceptual de un modelo híbrido o de **caja gris** que busca unificar **ambas** [14, 15], se ilustra en la figura 1.1.

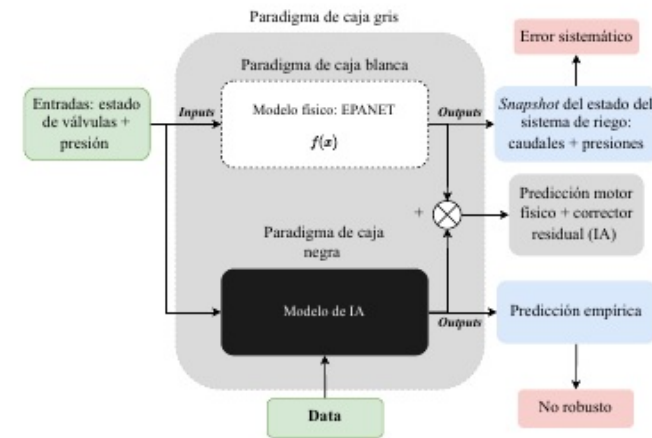


FIGURA 1.1. Diagrama conceptual de los paradigmas de modelado: caja blanca (física), caja negra (IA pura) y caja gris (híbrido).

1.2.2. Optimización: optimalidad vs. factibilidad

El segundo desafío fundamental radica en el propio algoritmo de búsqueda. La elección de la herramienta de optimización no es arbitraria, sino que está **dictada** por la naturaleza matemática inherente al problema del despacho de **riego**. Al **formularlo**, el problema se revela como un **Problema de Programación Mixta-Entera No Lineal (MINLP)**[16], una clasificación con profundas **implicaciones computacionales**.

La naturaleza **Mixta-Entera (MI)** surge de las variables de decisión: el estado de una válvula o bomba en un intervalo de tiempo es una decisión fundamentalmente binaria (0 para cerrado o 1 para abierto). A esto se suma la **No Linealidad**

(NL), que proviene directamente de la función de evaluación. El modelo **hidráulico** que predice el caudal y la **presión** —ya sea de caja blanca o negra— es una función compleja de la física de fluidos, no una simple suma algebraica.

La combinación de decisiones binarias (un espacio de búsqueda combinatorio) y una función de evaluación no lineal clasifica este problema en la categoría de complejidad **NP-hard**. Esto significa que es computacionalmente intratable: el tiempo requerido para encontrar la solución óptima garantizada crece **exponencialmente** a medida que el problema escala (más válvulas o más intervalos). El espacio de soluciones a explorar es demasiado grande.

Esta realidad computacional genera un conflicto directo e irresoluble con las necesidades de un sistema de riego operativo. El estado del arte, si bien ofrece solvers exactos para MINLP que buscan el óptimo global, no puede garantizar el tiempo de convergencia.

Dado este compromiso entre la optimalidad y la factibilidad, la práctica establecida para problemas de *scheduling* (programación) y asignación de recursos NP-hard es el uso de **metaheurísticas**. Algoritmos como los **Algoritmos Genéticos (GA)** proponen un enfoque alternativo: en lugar de buscar una garantía de **optimalidad** global, ofrecen una **garantía de tiempo de ejecución**. Están diseñados para encontrar soluciones de alta calidad (casi óptimas) dentro de un **presupuesto** computacional **fijo**. Para un problema de ingeniería en un entorno **productivo**, esta capacidad de obtener una solución factible en un tiempo acotado es un **requisito** operativo fundamental, **convirtiendo** a esta estrategia en un enfoque viable.

1.3. Motivación

La motivación de este trabajo **final** es proponer una arquitectura de software **unificada** que resuelva los dos dilemas fundamentales identificados en el estado del arte. Este proyecto se fundamenta en la hipótesis de que es posible lograr tanto la precisión en el modelado como la factibilidad en la optimización, sin **sacrificar** la robustez física ni el rendimiento operativo. Ante el desafío del modelado, que obliga a elegir entre la consistencia de la “**caja blanca**” y la precisión de la “**caja negra**”, este trabajo implementa un enfoque de “**caja gris**” o Gemelo Digital Híbrido. Esta arquitectura utiliza el simulador físico (basado en EPANET/WNTR) para obtener una línea de base robusta y físicamente coherente. Adicionalmente, emplea un modelo de **Inteligencia Artificial (IA)** no para modelar el sistema completo, sino con la tarea específica de predecir y corregir el error residual del modelo físico. De esta forma, se busca unificar la consistencia física de los **primeros** principios con la precisión empírica de los modelos de IA.

El desafío de la optimización (MINLP) también requiere un enfoque específico. Si bien los solvers exactos son la herramienta fundamental para garantizar la optimalidad global, su tiempo de convergencia es, por la naturaleza NP-hard del problema, indefinido. Un cálculo que puede tardar horas o días no es una herramienta útil para la toma de decisiones agronómicas diarias. Por esta razón, el proyecto encontró un enfoque alternativo que prioriza la velocidad de **respuesta**. Se seleccionó una metaheurística, el **Algoritmo Genético (GA)**, fundamentada en su probada robustez para problemas de programación **complejos**. Esta **estrategia** permite al sistema encontrar soluciones de alta calidad (casi óptimas) dentro de

(NL), que proviene directamente de la función de evaluación. El modelo **hidráulico** que predice el caudal y la **presión**, ya sea de caja blanca o negra, es una función compleja de la física de fluidos, no una simple suma algebraica.

La combinación de decisiones binarias (un espacio de búsqueda combinatorio) y una función de evaluación no lineal clasifica este problema en la categoría de complejidad **NP-hard** [17]. Esto significa que es computacionalmente intratable: el tiempo requerido para encontrar la solución óptima garantizada crece **exponencialmente** a medida que el problema escala (más válvulas o más intervalos). El espacio de soluciones a explorar es demasiado grande.

Esta realidad computacional genera un conflicto directo e irresoluble con las necesidades de un sistema de riego operativo. El estado del arte, si bien ofrece solvers exactos para MINLP que buscan el óptimo global, no puede garantizar el tiempo de convergencia.

Dado este compromiso entre la optimalidad y la factibilidad, la práctica establecida para problemas de *scheduling* (programación) y asignación de recursos NP-hard es el uso de **metaheurísticas** [18, 19]. Algoritmos como los **algoritmos genéticos (AG)** proponen un enfoque alternativo: en lugar de buscar una garantía de **optimalidad** global, ofrecen una garantía de tiempo de ejecución. Están diseñados para encontrar soluciones de alta calidad (casi óptimas) dentro de un **presupuesto** computacional **fijo** [20]. Para un problema de ingeniería en un entorno **productivo**, esta capacidad de obtener una solución factible en un tiempo acotado es un **requisito** operativo fundamental, **lo que convierte** a esta estrategia en un enfoque viable.

1.3. Motivación

La motivación de este trabajo es proponer una arquitectura de software **unificada** que resuelva los dos dilemas fundamentales identificados en el estado del arte. Este proyecto se fundamenta en la hipótesis de que es posible lograr tanto la precisión en el modelado como la factibilidad en la optimización, sin **sacrificar** la robustez física ni el rendimiento operativo. Ante el desafío del modelado, que obliga a elegir entre la consistencia de la **caja blanca** y la precisión de la **caja negra**, este trabajo implementa un enfoque de **caja gris** o Gemelo Digital Híbrido. Esta arquitectura utiliza el simulador físico (basado en EPANET/WNTR) [21] para obtener una línea de base robusta y físicamente coherente. Adicionalmente, emplea un modelo de **inteligencia artificial (IA)** no para modelar el sistema completo, sino con la tarea específica de predecir y corregir el error residual del **modelo físico** [22, 23]. De esta forma, se busca unificar la consistencia física de los **primeros** principios con la precisión empírica de los modelos de IA.

El desafío de la optimización (MINLP) también requiere un enfoque específico. Si bien los solvers exactos son la herramienta fundamental para garantizar la optimalidad global, su tiempo de convergencia es, por la naturaleza NP-hard del problema, indefinido. Un cálculo que puede tardar horas o días no es una herramienta útil para la toma de decisiones agronómicas diarias. Por esta razón, el proyecto encontró un enfoque alternativo que prioriza la velocidad de **respuesta**. Se seleccionó una metaheurística, el **algoritmo genético**, fundamentada en su probada robustez para problemas de programación **complejos** [18]. Esta **estrategia** permite al sistema encontrar soluciones de alta calidad (casi óptimas) dentro

un límite de tiempo fijo, lo **cual** resulta aceptable y práctico para las **necesidades** de la operación real.

1.4. Objetivos y alcance

Basado en la problemática y la motivación expuestas, el objetivo general de este trabajo es diseñar, implementar y validar el software que constituye la “**Capa de Gestión y Optimización**” para un sistema de riego solar. El proyecto se enfoca en desarrollar los dos componentes técnicos fundamentales que responden directamente a los dilemas identificados en el estado del arte.

El primer objetivo específico es la creación de un Gemelo Digital Híbrido. Esto **implica** implementar un simulador físico (**basado en EPANET/WNTR**) y, de forma crucial, desarrollar un **modelo** de inteligencia artificial cuya función es predecir y corregir el error residual de dicho modelo **físico**. Este componente servirá como la función de evaluación precisa y robusta para el optimizador.

El segundo objetivo específico es el diseño de un optimizador metaheurístico. Se implementará un **Algoritmo Genético (GA)** capaz de navegar el complejo **espacio** de búsqueda MINLP. Este agente utilizará el Gemelo Digital Híbrido **como** su función de aptitud **para** generar cronogramas de despacho casi óptimos, priorizando la factibilidad temporal sobre la optimalidad exacta. El alcance de la implementación incluye también el desarrollo de la función de costos híbrida y unificada, que permite al agrónomo ponderar los objetivos de uniformidad, **eficiencia** y sostenibilidad del acuífero.

Es importante delimitar que el alcance de este trabajo es exclusivamente de software y se centra en el motor de decisión. El proyecto no incluye el diseño, provisión, instalación o mantenimiento de ningún componente de hardware (como bombas, sensores, PLCs o gateways). Asimismo, el desarrollo de la interfaz de usuario final o “**Capa de Presentación**” (dashboard) y el soporte operativo post-entrega quedan fuera del alcance de esta memoria. Esta arquitectura global, y el **foco** específico del proyecto, se ilustran en la figura 1.2.

de un límite de tiempo fijo, lo **que** resulta aceptable y práctico para las **necesidades** de la operación real.

1.4. Objetivos y alcance

Basado en la problemática y la motivación expuestas, el objetivo general de este trabajo es diseñar, implementar y validar el software que constituye la **capa de gestión y optimización** para un sistema de riego solar. El proyecto se enfoca en desarrollar los dos componentes técnicos fundamentales que responden directamente a los dilemas identificados en el estado del arte.

El primer objetivo específico es la creación de un Gemelo Digital Híbrido. Esto **implica** implementar un simulador físico y, de forma crucial, desarrollar un **modelo de** inteligencia artificial cuya función es predecir y corregir el error residual de dicho modelo **físico**. Este componente servirá como la función de evaluación precisa y robusta para el optimizador.

El segundo objetivo específico es el diseño de un optimizador metaheurístico. Se implementará un **AG** capaz de navegar el complejo **espacio** de búsqueda MINLP. Este agente utilizará el Gemelo Digital Híbrido **como** su función de aptitud **para** generar cronogramas de despacho casi óptimos, priorizando la factibilidad temporal sobre la optimalidad exacta. El alcance de la implementación incluye también el desarrollo de la función de costos híbrida y unificada, que permite al agrónomo ponderar los objetivos de uniformidad, **eficiencia** y sostenibilidad del acuífero.

Es importante delimitar que el alcance de este trabajo es exclusivamente de software y se centra en el motor de decisión. El proyecto no incluye el diseño, provisión, instalación o mantenimiento de ningún componente de hardware (como bombas, sensores, PLCs o gateways). Asimismo, el desarrollo de la interfaz de usuario final o **capa de presentación** (dashboard) y el soporte operativo post-entrega quedan fuera del alcance de esta memoria.

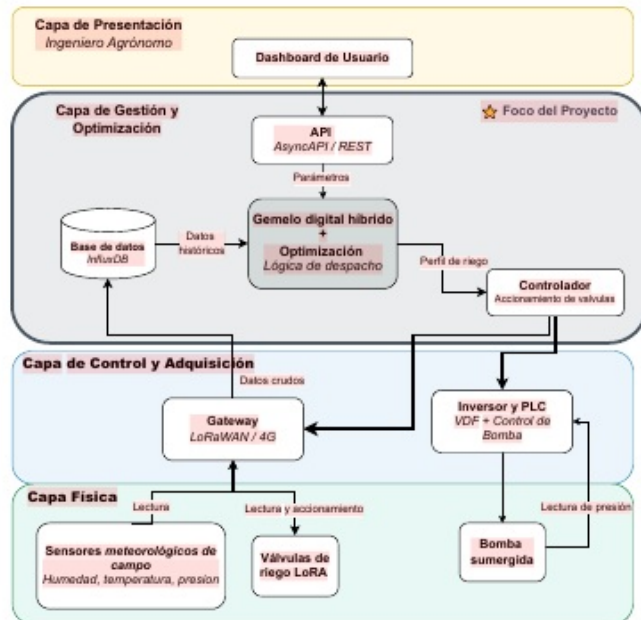


FIGURA 1.2. Diagrama en bloques de la arquitectura completa del sistema de riego.

Capítulo 2

Introducción específica

Todos los capítulos deben comenzar con un breve párrafo introductorio que indique cuál es el contenido que se encontrará al leerlo. La redacción sobre el contenido de la memoria debe hacerse en presente y todo lo referido al proyecto en pasado, siempre de modo impersonal.

- 2.1. Teoría del modelado híbrido: corrección de residuales
- 2.2. Formulación del problema de optimización
- 2.3. Teoría de optimización: algoritmos genéticos
- 2.4. Manejo de restricciones

Capítulo 2

Introducción específica

En este capítulo se presentan los fundamentos teóricos que sustentan la arquitectura de optimización del proyecto. Se detalla el modelado híbrido de caja gris, la formulación matemática del problema como un MINLP, la teoría de los Algoritmos Genéticos como solución metaheurística y el Método de Funciones de Penalización para el manejo de restricciones.

2.1. Teoría del modelado híbrido: corrección de residuales

La simulación de sistemas físicos complejos, como lo es un sistema hidráulico, se aborda generalmente desde dos enfoques principales [24].

El primer enfoque es el modelado de caja blanca (*White-Box*), de naturaleza determinística, basado en leyes físicas de la mecánica de fluidos. [24, 25]. El motor de simulación hidráulico EPANET es un ejemplo representativo de este enfoque [26, 27]. La ventaja principal de los modelos de caja blanca es su interpretabilidad y la garantía de predicciones físicamente consistentes. Sin embargo, los modelos que se basan únicamente en leyes físicas tienen una limitación fundamental: debido a simplificaciones, inevitablemente producen desviaciones significativas, o sesgos, en comparación con la realidad [28, 29]. Estos sesgos surgen de dinámicas no modeladas y de la incertidumbre en los parámetros intrínsecos, como la rugosidad exacta de las tuberías (el coeficiente C en la ecuación de Hazen-Williams) o las pérdidas de carga menores no contempladas en el modelo.

El segundo enfoque es el modelado de caja negra (*Black-Box*). Estos modelos son de naturaleza estocástica, como las redes neuronales (NN) o los perceptrones multicapa (MLP), que mapean entradas a salidas basándose puramente en datos observados [24, 25]. Su fortaleza es la capacidad de modelar relaciones no lineales extremadamente complejas sin necesidad de conocer el modelo físico del sistema [30]. Sin embargo, sus desventajas son críticas para sistemas de ingeniería. Primero, requieren volúmenes de datos de entrenamiento que a menudo son muy grandes [28, 29]. Segundo, y más importante, son propensos a generar predicciones que pueden ser físicamente inconsistentes, como violar la conservación de masa, un riesgo inaceptable para un sistema de control de recursos hídricos.

Como solución, en este proyecto se propone un modelado de caja gris (*Gray-Box*), un paradigma que busca combinar las fortalezas de ambos enfoques[30]. Específicamente, se implementa la arquitectura de corrección de residuales.

En este paradigma, el modelo de inteligencia artificial no intenta aprender la totalidad del sistema desde cero. En su lugar, el modelo físico ($f_{físico}$) proporciona

la predicción principal y la robustez, mientras que el modelo de IA (f_{AI}) tiene una tarea mucho más acotada y factible: aprender y predecir el error residual (el sesgo sistemático) del modelo físico [29].

La formulación matemática de esta arquitectura híbrida es:

$$f_{hybrid} = f_{fisico} + f_{AI}(\mathbf{X}) \quad (2.1)$$

Donde el objetivo de entrenamiento (el *target*) para f_{AI} es el residual R , definido como:

$$R = Y_{real} - Y_{fisico} \quad (2.2)$$

La literatura académica lo describe como una herramienta de predicción eficiente [28, 29], donde el modelo f_{fisico} aporta la consistencia y el f_{AI} aporta la precisión, corrigiendo los sesgos propios del primero.

A continuación se analizan ambas partes de la ecuación 2.1.

2.1.1. Componente físico (f_{fisico})

El componente físico se desarrolla utilizando el motor de simulación EPANET [26, 27]. Este modelo se rige por leyes físicas claves que definen la complejidad del problema:

1. Pérdida de carga por fricción: se utiliza la ecuación de Hazen-Williams, una relación empírica que define la pérdida de carga (H_f) en una tubería.

$$H_f = \frac{10,67 \cdot L}{C^{1,852} \cdot D^{4,87}} \cdot Q^{1,852} \quad (2.3)$$

2. Comportamiento de emisores (goteros): el caudal de salida (q_e) en un gotero se modela usando la ley de potencia, que relaciona el caudal con la presión (p) en el nodo. En el proyecto se utiliza un exponente γ de 0.46, característico del equipamiento instalado, introduciendo otra no linealidad.;

$$q_e = C_d \cdot p^\gamma \quad (2.4)$$

Además, se emplea la simulación en modo Análisis Dirigido por Presión (PDA). A diferencia del Análisis Dirigido por Demanda (DDA), que asume que la demanda siempre se satisface, el PDA ajusta el caudal entregado si la presión en un nodo cae, ofreciendo una simulación mucho más realista para un sistema de riego solar donde la presión de la bomba varía constantemente.

2.1.2. Componente corrector (f_{AI})

El componente de IA, f_{AI} , se implementa como un Perceptrón Multicapa (MLP).

El vector de características \mathbf{X} está compuesto por variables que describen tanto el estado operativo del sistema como sus acciones de control. Entre las primeras se incluyen magnitudes medidas en campo, como la presión a la salida de la bomba y el nivel freático del acuífero; entre las segundas, las decisiones de operación, como la apertura o cierre de válvulas. Además, la salida generada por el modelo físico se incorpora explícitamente como una de las entradas del MLP. De este modo, la red neuronal aprende a estimar el sesgo sistemático del modelo hidráulico

Capítulo 3

Diseño e implementación

Todos los capítulos deben comenzar con un breve párrafo introductorio que indique cuál es el contenido que se encontrará al leerlo. La redacción sobre el contenido de la memoria debe hacerse en presente y todo lo referido al proyecto en pasado, siempre de modo impersonal.

- 3.1. Arquitectura General del Software
- 3.2. Implementación del gemelo digital híbrido
- 3.3. Implementación del optimizador de despacho (GA)
- 3.4. Implementación de funciones de costo y penalizaciones

en función de las condiciones reales de operación, actuando como un corrector residual que ajusta las predicciones físicas hacia valores más próximos a las observaciones experimentales.

2.2. Formulación del problema de optimización

Una vez que se dispone de una función de aptitud f_{hybrid} de alta fidelidad, es necesario definir formalmente el problema de optimización.

2.2.1. Formulación matemática formal

El problema de optimización del despacho de riego consiste en encontrar el cronograma de apertura y cierre de válvulas que maximice un conjunto de objetivos, sujeto a las restricciones físicas de la red hidráulica y la disponibilidad de recursos. A continuación, se presenta una nomenclatura y formulación general del modelo.

TABLA 2.1. Nomenclatura del Modelo de Optimización.

Símbolo	Definición
V	Matriz de decisiones de despacho (variable de decisión).
$v_{i,t}$	Variable binaria: 1 si la válvula i está abierta en el tiempo t , 0 si está cerrada.
T	Horizonte de planificación (ej. 24 horas, discretizado en T intervalos).
I	Conjunto de todas las válvulas (sectores de riego).
$f_{hybrid}(V)$	Función de aptitud.
$Q_{entregado,i}(V)$	Caudal total entregado al sector i bajo el plan V .
$Q_{objetivo,i}$	Caudal objetivo para el sector i .
$P_{consumida,t}(V)$	Potencia consumida por la bomba en el tiempo t bajo el plan V .
$P_{disponible,t}$	Potencia disponible pronosticada en el tiempo t .
$N_{recurso,t}(V)$	Nivel del recurso en el tiempo t bajo el plan V .
N_{min}	Nivel mínimo de seguridad del recurso.
w_u, w_t, w_a	Ponderaciones estratégicas para los objetivos de la función.

Función objetivo ponderada

Un enfoque común para problemas multiobjetivo es formular una función de aptitud unificada. Esta función es una suma ponderada que puede incluir objetivos agronómicos, de eficiencia y de sostenibilidad:

max_V (w_u · U(V) - w_t · T_bombeo(V) - w_a · S_acusifero(V)) (2.5)

Donde :

- U(V): Una métrica de uniformidad (ej. minimizar el déficit cuadrático $\sum (Q_{objetivo,i} - Q_{entregado,i})^2$), que se busca maximizar.
- T_bombeo(V): Una métrica del tiempo de bombeo (ej. $\sum v_{i,t}$), que se busca minimizar.

- *Sacuífero*(**V**): Una métrica del estrés sobre el acuífero (ej. la extracción total de agua), que se busca minimizar.

Restricciones del problema

La maximización está sujeta a un conjunto de restricciones físicas y operativas:

1. Restricción de potencia: la potencia consumida por la bomba no puede exceder la potencia disponible en ningún momento.

$$P_{consumida,t}(f_{hybrid}(\mathbf{V})) \leq P_{disponible,t} \quad \forall t \in \mathcal{T} \tag{2.6}$$

2. Restricción de sostenibilidad (ej. nivel freático): el nivel del recurso no debe caer por debajo del límite de seguridad.

$$N_{recurso,t}(f_{hybrid}(\mathbf{V})) \geq N_{min} \quad \forall t \in \mathcal{T} \tag{2.7}$$

3. Restricciones hidráulicas (implícitas): las leyes físicas de la red (Hazen-Williams, Ley de Potencia) están encapsuladas dentro de la evaluación de la función *f_{hybrid}*.

4. Variables de decisión binarias: las decisiones de control de válvulas son binarias.

$$v_{i,t} \in \{0,1\} \quad \forall i \in \mathcal{I}, \forall t \in \mathcal{T} \tag{2.8}$$

2.3. Teoría de optimización: algoritmos genéticos

El algoritmo genético consiste en una búsqueda estocástica global basada en los principios de la evolución natural y la supervivencia del más apto [31, 32]. Este algoritmo es adecuado para navegar espacios de búsqueda complejos, no convexos y de alta dimensionalidad. Su uso acoplado a simuladores como EPANET para la optimización de redes de agua es una técnica documentada en la literatura [26, 27].

2.3.1. Anatomía de un algoritmo genético

El AG opera sobre una población de cromosomas, donde cada cromosoma representa una solución candidata completa al problema de optimización.

Codificación del cromosoma (representación de la solución)

El cromosoma es la estructura de datos que codifica un plan de despacho de riego [33]. La elección de la codificación es fundamental para el rendimiento del algoritmo.

- *Permutation encoding* (codificación por permutación): en este esquema, el cromosoma es una permutación de elementos. Se utiliza para problemas de secuenciación u ordenamiento, como el Problema del Viajante (TSP) [34].
- *Value encoding* (codificación por valor): En este esquema, el cromosoma es una cadena de valores (enteros, reales o binarios) donde cada gen representa un valor específico, no una posición en una secuencia [34].

Capítulo 4

Ensayos y resultados

Todos los capítulos deben comenzar con un breve párrafo introductorio que indique cuál es el contenido que se encontrará al leerlo. La redacción sobre el contenido de la memoria debe hacerse en presente y todo lo referido al proyecto en pasado, siempre de modo impersonal.

- 4.1. Entorno y banco de pruebas
- 4.2. Pruebas del gemelo digital híbrido
- 4.3. Caso de estudio: simulación de despacho

Para el problema de despacho de riego, donde la decisión es asignar un estado (abierto/cerrado) a una válvula en un intervalo de tiempo, la codificación por valor binario es la más adecuada. Si el horizonte de planificación T se discretiza en N_T intervalos y hay N_V válvulas, un cromosoma C es un vector binario de longitud $N_V \times N_T$. Aunque se almacena como un vector, conceptualmente representa una matriz donde cada elemento $v_{i,t}$ es un gen:

$$C = \begin{bmatrix} v_{1,1} & v_{1,2} & \dots & v_{1,N_T} \\ v_{2,1} & v_{2,2} & \dots & v_{2,N_T} \\ \vdots & \vdots & \ddots & \vdots \\ v_{N_V,1} & v_{N_V,2} & \dots & v_{N_V,N_T} \end{bmatrix} \tag{2.9}$$

Donde $v_{i,t} \in \{0,1\}$ representa el estado (cerrado/abierto) de la válvula i en el tiempo t .

Operadores de selección

La selección determina qué cromosomas (soluciones) de la población actual son elegidos para reproducirse y crear la siguiente generación, aplicando el principio de "supervivencia del más apto" [35]. En la tabla 2.2 se comparan los operadores de selección más comunes.

TABLA 2.2. Análisis comparativo de operadores de selección en AG.

Operador	Mecanismo	Ventajas	Desventajas y riesgos
Roulette wheel selection (RWS)	Asigna una probabilidad de selección p_i proporcional a la aptitud f_i : $p_i = f_i / \sum f_k$ [35].	Simple.	Riesgo de convergencia prematura. Un individuo "súper" (con aptitud muy alta) puede dominar la población rápidamente, eliminando la diversidad genética [35].
Tournament selection (TOS)	Elige k (ej. $k = 2$) individuos al azar de la población y selecciona al de mayor aptitud para la reproducción [35].	Computacionalmente eficiente; menor riesgo de convergencia prematura [36, 37].	Si k es muy grande converge más rápido pero con riesgo de caer en óptimos locales.
Rank selection (LRS)	Selecciona individuos basándose en el <i>ranking</i> de su aptitud, no en el valor absoluto. Asigna probabilidades linealmente al ranking [35, 38].	Evita los problemas de escalado de RWS.	Más complejo computacionalmente que TOS; puede ser más lento en converger.

La literatura a menudo reporta que TOS ofrece un equilibrio superior entre exploración y explotación, y un mejor rendimiento general que RWS [36, 37].

Operadores de cruce (crossover / recombinación)

El cruce (*crossover*) es el operador principal de exploración. Combina la información genética de dos cromosomas padres para crear hijos (nuevas soluciones) [32,

[35]. Los operadores dependen críticamente de la codificación del cromosoma, como se observa en la tabla 2.3.

TABLA 2.3. Análisis comparativo de operadores de cruce relevantes.

Operador	Codificación	Mecanismo	Caso de Uso Principal
Single-point crossover	Valor (binario real)	Se elige un punto de corte aleatorio. Los padres intercambian sus colas para crear dos hijos.	El operador de cruce más simple.
Uniform crossover	Valor (binario real)	Por cada gen en el cromosoma, se lanza una moneda (con probabilidad p) para decidir si el hijo hereda el gen del padre 1 o del padre 2 [35].	A menudo es el más efectivo para codificaciones binarias, ya que permite una alta recombinación de bloques de genes.

Operadores de mutación (diversidad)

La mutación es un operador secundario que introduce variaciones aleatorias en un cromosoma. Su propósito es mantener la diversidad genética y prevenir que la población converja prematuramente a un óptimo local [32, 35].

- *bit flip mutation* (para codificación por valor binario): es el operador relevante para este problema. Se selecciona un gen (un bit $v_{i,t}$) al azar y se invierte su valor ($0 \rightarrow 1$ ó $1 \rightarrow 0$).
- *swap mutation* (para codificación por permutación): Se seleccionan dos genes al azar en el cromosoma y se intercambian sus posiciones [39].

2.4. Manejo de restricciones: funciones de penalización

El desafío final en la aplicación de un AG a un problema de ingeniería del mundo real es el manejo de las restricciones. Los operadores de cruce y mutación son estocásticos y no tienen conocimiento de la física del sistema; por lo tanto, generarán inevitablemente cromosomas (planes de riego) que son infactibles.

El método de funciones de penalización es la técnica de manejo de restricciones (CHT) más común para metaheurísticas [40, 27, 41]. La filosofía es transformar un problema de optimización restringida en un problema no restringido [40, 42]. Esto se logra moviendo las restricciones desde el solver hacia la función objetivo [40]. Se añade un término de penalización a la función de aptitud, que castiga a las soluciones infactibles [43, 40]. La formulación matemática general para un problema de minimización es [40, 42]:

$$Fitness_{penalizado}(x) = Fitness_{original}(x) + p(d(x, B)) \tag{2.10}$$

Donde $Fitness_{original}(x)$ es el objetivo, $d(x, B)$ es una métrica de la magnitud de la violación de la restricción, y $p(\cdot)$ es la función de penalización que traduce esa violación en un incremento del costo [42].

Capítulo 5

Conclusiones

Todos los capítulos deben comenzar con un breve párrafo introductorio que indique cuál es el contenido que se encontrará al leerlo. La redacción sobre el contenido de la memoria debe hacerse en presente y todo lo referido al proyecto en pasado, siempre de modo impersonal.

5.1. Resultados obtenidos

5.2. Trabajo futuro

La eficacia del AG depende de cómo se diseña $p(\cdot)$ [43]. Las estrategias varían desde la "pena de muerte" (*death penalty*), que rechaza cualquier solución infactible, hasta enfoques más robustos como las penalizaciones dinámicas, donde la penalización aumenta con cada generación (ej. $p(V, t) = (\lambda \cdot t) \cdot V$) [43, 28].