



UNIVERSIDAD DE CHILE  
FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS  
DEPARTAMENTO DE INGENIERÍA ELÉCTRICA

DISEÑO DE SISTEMAS DE CONTROL PARA LA OPTIMIZACIÓN DEL USO DE  
AGUA DE RIEGO EN FRUTICULTURA

MEMORIA PARA OPTAR AL TÍTULO DE  
INGENIERO CIVIL ELÉCTRICO

IGNACIO BASUALTO CORTÉS

PROFESOR GUÍA:  
DIEGO MUÑOZ CARPINTERO

MIEMBROS DE LA COMISIÓN:  
DORIS SÁEZ HUEICHAPAN  
CARLOS FAÚNDEZ URBINA

Este trabajo ha sido parcialmente financiado por el proyecto MSM2021003 Gestión  
inteligente de recursos hídricos para la agricultura de la Universidad de O'Higgins

SANTIAGO DE CHILE

2023

# Resumen

En la actualidad, el crecimiento demográfico, el calentamiento global y el uso ineficiente del agua marcan un punto de no retorno en términos de la escasez hídrica, lo que genera la necesidad urgente de tomar medidas, desarrollar técnicas y estrategias que permitan optimizar el uso del recurso, en particular en los sectores donde el consumo de agua es mayor y más crítico, como es el caso del sector agrícola.

Por esto, el presente trabajo busca estudiar metodologías para el control del sistema de riego en la fruticultura, en particular en la producción de uva vinífera a partir de la información provista por una Viña en la región del Libertador Bernardo O'Higgins.

Las metodologías actuales de riego en fruticultura se centran en mediciones sobre el estado hídrico de la planta y el terreno cultivado, estimaciones de evapotranspiración y la etapa de desarrollo en la que se encuentra el cultivo. A partir de estos factores se definen los requerimientos hídricos, los que se utilizan para definir la programación de riego para un intervalo de tiempo determinado.

En este trabajo se propone la utilización de técnicas de aprendizaje reforzado (AR) para diseñar controladores que generen programaciones de riego. Para ello se utiliza el software HYDRUS-1D para construir un modelo de transferencias hídricas en Python, sobre este se entrenan los controladores y se obtienen las planificaciones de riego.

El modelo define el entorno de AR, para su diseño y configuración se utiliza la información disponible del terreno, datos meteorológicos y las características particulares del cultivo de uva vinífera. El agente lo constituye el controlador de riego, se establece una recompensa en función del uso de agua y el contenido de agua en el perfil, con la cual se entrena y aprende según lo indicado por los algoritmos *Deep Q-Network* (DQN), *Twin Delayed Deep Deterministic Policy Gradient* (TD3) y *Proximal Policy Optimization* (PPO) utilizados.

Finalmente, las programaciones de riego diario obtenidas deben ser evaluadas respecto de las estrategias de riego que se desean mejorar, las cuales son definidas en función del criterio utilizado por la Viña e implementadas en el entorno.

Se concluye que con aprendizaje reforzado es posible generar las programaciones de riego, donde se logra ajustar el modelo y el aprendizaje del agente para cumplir con los requerimientos del cultivo, sin embargo, se aprecian elementos de mejora para el modelo, principalmente en la aplicación del riego y la estimación de la evapotranspiración.

*Este trabajo se lo dedico a mis padres y abuelos, que siempre me han motivado para llegar  
cada vez más lejos.*

*Y para Agustín el miembro más nuevo de la familia que llegó en mitad de la redacción de  
este informe.*

# Agradecimientos

Primero quiero destacar que este trabajo fue parcialmente financiado por el proyecto MSM2021003 Gestión inteligente de recursos hídricos para la agricultura de la Universidad de O'Higgins, agradeciendo su colaboración en la realización de este trabajo.

Luego, quiero agradecer al profesor Diego Muñoz, por la paciencia y dedicación entregada a lo largo del desarrollo de este trabajo, y por sobretodo por esta oportunidad que me brindo. También al vice-capitán Caba que me impulsó a seguir avanzando aún cuando las cosas se pusieron complicadas y cuando sentía que ya no podía más, gracias por tus palabras de ánimo y todo el apoyo brindado todos estos años.

La realización de este trabajo no habría sido posible sin la ayuda del Pele que me enseñó todo lo que pudo del mundo agrícola, gracias por tu motivación y apañe desde aquel lejano día en la sala de kinesiología de JGM hasta la última reunión por Zoom.

Por lo mismo, quiero dar gracias al Volei, que me formó en la persona que soy actualmente y me dio la oportunidad de conocer a un sinnúmero de gente espectacular, en especial a los equipos de Ingeniería, de la U y Oasis, que me han provisto de innumerables desafíos durante estos años, pero siempre me han impulsado a superarlos y mejorar, en particular quiero destacar al Rai, Diego y Chino que me han soportado y bancado durante este año de viejones.

Quiero dar las gracias de todo corazón a mi familia, en especial a mis padres y abuelos, por siempre entregarme todo lo que pudieron y más, por formar la persona que soy y apoyarme incondicionalmente. Al Seba, Maty y José por mostrarme como luchan por alcanzar sus sueños y objetivos.

Y por último, a la Jenny que ha sido mi compañera incondicional durante toda mi vida universitaria, este trabajo es una pequeña retribución por todo el apoyo que me ha dado este año, por celebrar conmigo mis victorias y ayudar a sobreponerme en las derrotas.

Gracias a todos quienes fueron parte de esta etapa universitaria.

# Tabla de Contenido

|  |           |
|--|-----------|
| <b>1. Introducción</b>   | <b>1</b>  |
| 1.1. Motivación . . . . .  | 1         |
| 1.2. Objetivos . . . . .   | 2         |
| 1.3. Estructura del documento . . . . .                          | 3         |
| <b>2. Estado del arte</b>  | <b>4</b>  |
| 2.1. Estrategias convencionales . . . . .                        | 4         |
| 2.2. Control con aprendizaje reforzado . . . . .                 | 5         |
| 2.2.1. Aplicaciones de AR en la planificación de riego . . . . . | 6         |
| 2.2.2. Variantes en la aplicación de AR . . . . .                | 6         |
| 2.3. Discusión . . . . .   | 8         |
| <b>3. Antecedentes agrícolas</b>                                 | <b>10</b> |
| 3.1. Propiedades del suelo . . . . .                             | 10        |
| 3.1.1. Textura y porosidad . . . . .                             | 10        |
| 3.1.2. Humedad del suelo . . . . .                               | 11        |
| 3.2. Ecuación de balance hídrico . . . . .                       | 12        |
| 3.2.1. Evapotranspiración (ET) . . . . .                         | 13        |
| 3.2.2. Escurrimiento Superficial (RO) . . . . .                  | 17        |
| 3.2.3. Percolación Profunda (DP) . . . . .                       | 17        |
| 3.2.4. Ascenso Capilar (CR) . . . . .                            | 18        |
| 3.3. Características de la producción de uva de vino . . . . .   | 18        |

|           |   |           |
|-----------|---|-----------|
| 3.3.1.    | Etapas de desarrollo del cultivo . . . . .                                  | 18        |
| 3.3.2.    | Estrés hídrico . . . . .  | 19        |
| 3.3.3.    | Rendimiento del cultivo . . . . .   | 20        |
| 3.3.4.    | Potencial Hídrico Xilemático (Px) . . . . .                                 | 20        |
| <b>4.</b> | <b>HYDRUS-1D</b>  | <b>21</b> |
| 4.1.      | Configuración de HYDRUS-1D . . . . .  | 22        |
| 4.1.1.    | Información básica . . . . .  | 22        |
| 4.1.2.    | Flujo de agua ( <i>Water Flow</i> ) . . . . .                               | 25        |
| 4.1.3.    | Absorción de agua en las raíces ( <i>Root Water Uptake</i> ) . . . . .      | 29        |
| 4.1.4.    | Condiciones atmosféricas ( <i>Variable Boundary Conditions</i> ) . . . . .  | 31        |
| 4.1.5.    | Perfil de suelo inicial ( <i>Soil Profile</i> ) . . . . .                   | 32        |
| <b>5.</b> | <b>Aprendizaje Reforzado (AR)</b>   | <b>35</b> |
| 5.1.      | Conceptos básicos . . . . .   | 35        |
| 5.1.1.    | Proceso de decisión de Markov . . . . .                                     | 36        |
| 5.1.2.    | Ecuación de Bellman y funciones de valor . . . . .                          | 38        |
| 5.2.      | Clasificación de algoritmos . . . . .                                       | 39        |
| 5.2.1.    | <i>Model Free</i> . . . . .   | 39        |
| 5.2.2.    | <i>Value Based - Policy Based</i> . . . . .                                 | 40        |
| 5.2.3.    | Monte Carlo y <i>Temporal Difference</i> . . . . .                          | 41        |
| 5.2.4.    | <i>On-policy</i> y <i>Off-policy</i> . . . . .                              | 41        |
| 5.3.      | Algoritmos . . . . .  | 42        |
| 5.3.1.    | <i>Deep Q-Network</i> <b>DQN</b> . . . . .                                  | 42        |
| 5.3.2.    | <i>Twin Delayed Deep Deterministic Policy Gradient</i> <b>TD3</b> . . . . . | 43        |
| 5.3.3.    | <i>Proximal Policy Optimization</i> <b>PPO</b> . . . . .                    | 45        |
| <b>6.</b> | <b>Metodología</b>  | <b>47</b> |
| 6.1.      | Construcción del modelo inicial . . . . .                                   | 48        |

|           |   |           |
|-----------|---|-----------|
| 6.1.1.    | Antecedentes agrícolas para el desarrollo del cultivo . . . . .               | 48        |
| 6.1.2.    | Implementación del sistema en HYDRUS-1D . . . . .                             | 48        |
| 6.1.3.    | Integración en Python utilizando la librería <i>Phydrus</i> . . . . .         | 48        |
| 6.2.      | Ajuste del modelo . . . . .   | 49        |
| 6.2.1.    | Ajuste del modelo para el riego de uva vinífera . . . . .                     | 49        |
| 6.2.2.    | Simulación <i>offline</i> de la primera etapa de riego . . . . .              | 49        |
| 6.3.      | Aplicación de aprendizaje reforzado . . . . .                                 | 50        |
| 6.3.1.    | Modelamiento de la segunda etapa de riego en un ambiente <i>Gym</i> . . . . . | 50        |
| 6.3.2.    | Sistemas de control para el riego . . . . .                                   | 50        |
| 6.3.3.    | Configuración de algoritmos y entrenamiento del agente . . . . .              | 51        |
| 6.3.4.    | Programaciones de riego generadas mediante AR . . . . .                       | 51        |
| 6.4.      | Resultados del experimento . . . . .  | 51        |
| 6.4.1.    | Comparación con técnica de riego convencional . . . . .                       | 52        |
| 6.4.2.    | Análisis de resultados . . . . .  | 52        |
| 6.4.3.    | Conclusiones . . . . .  | 52        |
| <b>7.</b> | <b>Desarrollo de modelos y diseño de controladores</b>                        | <b>53</b> |
| 7.1.      | Construcción del modelo inicial . . . . .                                     | 53        |
| 7.1.1.    | Antecedentes agrícolas para el desarrollo del cultivo . . . . .               | 54        |
| 7.1.2.    | Implementación del sistema en HYDRUS-1D . . . . .                             | 55        |
| 7.1.3.    | Integración en Python utilizando la librería <i>Phydrus</i> . . . . .         | 56        |
| 7.2.      | Ajuste del modelo . . . . .   | 58        |
| 7.2.1.    | Ajuste del modelo para el riego de uva vinífera . . . . .                     | 58        |
| 7.2.2.    | Simulación <i>offline</i> de la primera etapa de riego . . . . .              | 60        |
| 7.3.      | Aplicación de aprendizaje reforzado . . . . .                                 | 62        |
| 7.3.1.    | Modelamiento de la segunda etapa de riego en un ambiente <i>Gym</i> . . . . . | 63        |
| 7.3.2.    | Definición de los sistemas de control para el riego . . . . .                 | 64        |
| 7.3.3.    | Configuración del agente . . . . .  | 66        |

|           |   |            |
|-----------|---|------------|
| 7.3.4.    | Programaciones de riego generadas mediante AR . . . . .                                 | 71         |
| 7.3.5.    | Implementación de técnicas de riego convencional . . . . .                              | 72         |
| <b>8.</b> | <b>Estudio por simulación</b>   | <b>74</b>  |
| 8.1.      | Diseño y configuración del experimento . . . . .  | 74         |
| 8.1.1.    | Configuración del modelo y primera etapa de desarrollo . . . . .                        | 74         |
| 8.1.2.    | Configuración del aprendizaje reforzado en la segunda etapa . . . . .                   | 76         |
| 8.1.3.    | Configuración del entrenamiento . . . . .   | 78         |
| 8.1.4.    | Estrategias de riego convencional . . . . .   | 79         |
| 8.2.      | Resultados del experimento . . . . .  | 79         |
| 8.2.1.    | Simulación <i>Offline</i> de la primera etapa . . . . .                                 | 79         |
| 8.2.2.    | Entrenamiento del sistema de control para la segunda etapa . . . . .                    | 81         |
| 8.2.3.    | Programaciones de riego con AR para el primer método . . . . .                          | 85         |
| 8.2.4.    | Programaciones de riego con AR para el segundo método . . . . .                         | 88         |
| 8.2.5.    | Programaciones de riego con estrategias convencionales para el primer método . . . . .  | 90         |
| 8.2.6.    | Programaciones de riego con estrategias convencionales para el segundo método . . . . . | 92         |
| 8.2.7.    | Programaciones de riego aplicadas por la Viña . . . . .                                 | 94         |
| 8.3.      | Análisis de resultados . . . . .  | 96         |
| 8.3.1.    | Modelo de simulación . . . . .  | 96         |
| 8.3.2.    | Sistema de control . . . . .  | 97         |
| 8.3.3.    | Entrenamiento de los controladores . . . . .  | 98         |
| 8.3.4.    | Resultados obtenidos . . . . .  | 100        |
| 8.3.5.    | Comparación con estrategias convencionales . . . . .                                    | 101        |
| 8.3.6.    | Comparación con riegos aplicados en la Viña . . . . .                                   | 102        |
| <b>9.</b> | <b>Conclusión y trabajo futuros</b>   | <b>103</b> |
| 9.1.      | Conclusiones . . . . .  | 103        |

|   |            |
|---|------------|
| 9.2. Trabajo Futuro . . . . .   | 105        |
| <b>Bibliografía</b>   | <b>106</b> |
| <b>Anexos</b>   | <b>110</b> |
| <b>Anexo A. Cálculo de las variables de la ecuación Penman-Monteith</b> | <b>110</b> |
| <b>Anexo B. Implementación de Phydrus</b>                               | <b>114</b> |
| B.1. HYDRUS-1D en Python . . . . .                                      | 114        |
| B.1.1. Instalación de Phydrus . . . . .                                 | 114        |
| B.1.2. Utilizando Phydrus . . . . .                                     | 115        |
| <b>Anexo C. Algoritmos aprendizaje reforzado</b>                        | <b>117</b> |
| <b>Anexo D. Gym Environment</b>   | <b>122</b> |
| <b>Anexo E. Otros resultados</b>  | <b>124</b> |
| E.1. Comparación HYDRUS-1D y <i>Phydrus</i> . . . . .                   | 124        |
| E.2. Perfil de suelo en la primera etapa . . . . .                      | 128        |
| E.3. Acciones de riego y otras programaciones de riego . . . . .        | 130        |
| E.4. Planificaciones de riego con estrategias convencionales . . . . .  | 134        |
| E.4.1. Tablas de resumen para Método 1 . . . . .                        | 134        |
| E.4.2. Programaciones de riego para Método 1 . . . . .                  | 136        |
| E.4.3. Tablas de resumen para Método 2 . . . . .                        | 139        |
| E.4.4. Programaciones de riego para Método 2 . . . . .                  | 140        |

# Índice de Tablas

|       |   |     |
|-------|---|-----|
| 8.1.  | Resumen de parámetros definidos para la configuración del modelo en <i>Phydrus</i> , para el estado hídrico basado en 1 medición. . . . .   | 75  |
| 8.2.  | Resumen de parámetros definidos para la configuración del modelo en <i>Phydrus</i> , para el estado hídrico basado en 4 mediciones. . . . .   | 76  |
| 8.3.  | Parámetros y características para el ajuste de la simulación con el entorno <i>Gym</i> , para el primer método. . . . .   | 77  |
| 8.4.  | Parámetros y características para el ajuste de la simulación con el entorno <i>Gym</i> , para el segundo método. . . . .  | 78  |
| 8.5.  | Tiempo de entrenamiento del agente para los diferentes experimentos realizados.   | 83  |
| 8.6.  | Tiempo de entrenamiento del agente para los algoritmos PPO. . . . .   | 84  |
| 8.7.  | Resumen del riego aplicado en cada programación, y valor promedio de la diferencia diaria entre el contenido de agua y el umbral de estrés hídrico. . .   | 87  |
| 8.8.  | Resumen del riego aplicado en cada programación, y valor de reposición de $ET_o$ .  | 89  |
| 8.9.  | Resumen del riego aplicado para cada estrategia convencional, y valor promedio de la diferencia diaria entre el contenido de agua y el umbral de estrés hídrico, para la temporada 18-19. . . . .             | 92  |
| 8.10. | Resumen del riego aplicado para cada estrategia convencional y reposición de la $ET_o$ , para la temporada 18-19. . . . .   | 94  |
| 8.11. | Resumen del riego aplicado y reposición de la $ET_o$ para 3 temporadas en la Viña. . . . .  | 95  |
| E.1.  | Resumen de riego aplicado para cada programación de riego, y valor promedio de la diferencia diaria entre el contenido de agua y el umbral de estrés hídrico, con raíces distribuidas homogéneamente. . . . . | 133 |
| E.2.  | Resumen de riego aplicado en estrategias convencionales y valor promedio de la diferencia entre el contenido de agua y el umbral de estrés hídrico, para la temporada 19-20. . . . .                          | 134 |

|   |     |
|---|-----|
| E.3. Resumen de riego aplicado en estrategias convencionales y valor promedio de la diferencia entre el contenido de agua y el umbral de estrés hídrico, para la temporada 20-21. . . . . | 134 |
| E.4. Resumen de riego aplicado en estrategias convencionales y valor promedio de la diferencia entre el contenido de agua y el umbral de estrés hídrico, para la temporada 21-22. . . . . | 135 |
| E.5. Resumen del riego aplicado para cada estrategia convencional y reposición de la $ET_o$ , para la temporada 19-20. . . . .  | 139 |
| E.6. Resumen del riego aplicado para cada estrategia convencional y reposición de la $ET_o$ , para la temporada 20-21. . . . .  | 139 |
| E.7. Resumen del riego aplicado para cada estrategia convencional y reposición de la $ET_o$ , para la temporada 21-22. . . . .  | 139 |

# Índice de Ilustraciones

|   |    |
|---|----|
| 2.1. Metodología de la implementación del control de riego por aprendizaje reforzado en el cultivo de tomates [1]. . . . .                          | 6  |
| 2.2. Estructura de la generación del entorno basado en el software DSSAT con redes neuronales [33]. . . . .   | 7  |
| 3.1. Balance hídrico en la zona de la raíz. . . . .   | 12 |
| 3.2. Distribución de la evapotranspiración, diferenciada en evaporación y transpiración durante el período de crecimiento de un cultivo[2]. . . . . | 13 |
| 3.3. Evolución teórica del parámetro $K_c$ en función del tiempo.[2] . . . . .  | 15 |
| 3.4. Interpretación gráfica de la relación entre $D_r$ , $TAW$ , $RAW$ y $K_s$ . . . . .  | 16 |
| 3.5. Definición de los períodos para los tratamientos de riego en un ciclo anual de la Viña [17]. . . . .   | 19 |
| 4.1. Ejemplos de la visualización del contenido de agua en HYDRUS-1D . . . . .  | 21 |
| 4.2. Interfaz de usuario inicial para HYDRUS-1D . . . . .   | 22 |
| 4.3. <i>Main Processes</i> : Selección de los procesos para el proyecto a simular. . . . .  | 23 |
| 4.4. <i>Geometry Information</i> : Selección de características básicas del perfil de suelo. . . . .  | 23 |
| 4.5. <i>Main Time Information</i> : Selección de características y condiciones atmosféricas temporales del proyecto. . . . .                        | 24 |
| 4.6. <i>Print Information</i> : Configuración de la visualización de resultados. . . . .  | 24 |
| 4.7. <i>Water Flow - Iteration Criteria</i> : Configuración del método iterativo de resolución de la ecuación de Richards. . . . .                  | 25 |
| 4.8. <i>Water Flow - Soil Hydraulic Property Model</i> : Selección del modelo de simulación para el flujo de agua. . . . .                          | 26 |

|   |    |
|---|----|
| 4.9. <i>Water Flow - Soil Hydraulic Parameters</i> : Configuración de parámetros del modelo seleccionado, según los tipos de suelo del perfil. . . . .                              | 27 |
| 4.10. <i>Water Flow - Boundary Conditions</i> : Configuración de las condiciones de borde en los extremos del perfil de suelo. . . . .  | 28 |
| 4.11. <i>Root Water Uptake - Models</i> : Selección del modelo de absorción de agua y solutos por las raíces, . . . . .   | 29 |
| 4.12. <i>Root Water Uptake - Water Stress Reduction</i> : Configuración de los parámetros del modelo Feddes. . . . .  | 30 |
| 4.13. <i>Root Water Uptake - Water Stress Reduction</i> : Configuración de los parámetros del modelo <i>S-Shape function</i> de van Genuchten . . . . .                             | 31 |
| 4.14. <i>Variable Boundary Conditions</i> : Condiciones atmosféricas definidas para el tiempo de duración de la simulación. . . . .   | 32 |
| 4.15. <i>Graphical Editor - Initial Conditions</i> : Definición de las condiciones iniciales en términos de contenido de agua o cabezal de presión para el perfil de suelo. . . . . | 33 |
| 4.16. <i>Graphical Editor - Root Distribution</i> : Definición de la distribución y densidad de las raíces en el perfil de suelo. . . . .   | 33 |
| 4.17. <i>Graphical Editor - Observation Points</i> : Selección de puntos de observación para la visualización de la dinámica del suelo en esos puntos. . . . .                      | 34 |
| 4.18. <i>Soil Profile - Summary</i> : Condiciones iniciales en términos de contenido de agua o cabezal de presión para el perfil de suelo . . . . .                                 | 34 |
| 5.1. Diagrama básico de un algoritmo de aprendizaje reforzado. . . . .  | 35 |
| 5.2. Modelo gráfico de un proceso de Markov [11] . . . . .  | 36 |
| 5.3. Modelo gráfico de un proceso de recompensa de Markov [11] . . . . .  | 37 |
| 5.4. Modelo gráfico de un proceso de decisión de Markov [11] . . . . .  | 37 |
| 5.5. Estructura de organización para los algoritmos de aprendizaje reforzado [26].  | 39 |
| 5.6. Valor de la función $L^{CLIP}$ para un paso, en función del valor de los beneficios $\hat{A}_t$ , partiendo desde $r = 1$ . . . . .  | 45 |
| 6.1. Diagrama de flujo de la metodología aplicada. . . . .  | 47 |
| 7.1. Estructura del sistema de control con aprendizaje reforzada, basada en [22] .  | 53 |
| 7.2. Información meteorológica provista por la Viña entre los años 2018 y 2022 .  | 55 |
| 7.3. Curvas para el coeficiente de cultivo $K_c$ . . . . .  | 55 |

|  |    |
|--|----|
| 7.4. Variables atmosféricas de entrada al modelo en <i>Phydrus</i> . . . . .   | 57 |
| 7.5. Distribución de la densidad de raíces en el perfil de suelo. . . . .  | 61 |
| 7.6. Simulaciones de la etapa 1 con HYDRUS-1D y con el modelo en <i>Phydrus</i> , para el método con 1 medición. . . . .   | 61 |
| 7.7. Simulaciones de la etapa 1 con HYDRUS-1D y con el modelo en <i>Phydrus</i> , para el método con 4 mediciones. . . . . | 62 |
| 7.8. Estructura de aprendizaje reforzado diseñada, basada en [22] . . . . .  | 63 |
| 7.9. Determinación de la acción de control en base al cálculo del riego a capacidad  | 65 |
| 7.10. Cálculo del riego a capacidad para un perfil con diferentes tipos de suelo en la zona de la raíz . . . . .           | 66 |
| 7.11. Costo por riego del cultivo diario . . . . .   | 67 |
| 7.12. Variaciones del costo diario por contenido de agua medido en el perfil . . . .                                       | 68 |
| 7.13. Determinación del riego aplicado en función del espacio de acciones utilizado.                                       | 70 |
| 8.1. Evolución del contenido de agua en la primera etapa, para diferentes datos meteorológicos . . . . .                   | 80 |
| 8.2. Simulaciones de la etapa 1 para cada temporada en <i>Phydrus</i> , con el método con 4 mediciones. . . . .            | 81 |
| 8.3. Curvas de aprendizaje del agente para los algoritmos utilizados, según la función de recompensa utilizada. . . . .    | 82 |
| 8.4. Curvas de aprendizaje de los agente para los algoritmos PPO, según la función de recompensa utilizada. . . . .        | 84 |
| 8.5. Programaciones de riego para la segunda etapa, con los agentes entrenados (1/2). . . . .                              | 85 |
| 8.6. Programaciones de riego para la segunda etapa, con los agentes entrenados (2/2). . . . .                              | 86 |
| 8.7. Programaciones de riego para la segunda etapa, con los agentes entrenados en el segundo método. . . . .               | 88 |
| 8.8. Riego según capacidad de campo para la segunda etapa, en la temporada 18-19.  | 90 |
| 8.9. Riego de cantidad fija para la segunda etapa, en la temporada 18-19. . . . .  | 91 |
| 8.10. Estrategias de riego convencional para la segunda etapa, en la temporada 18-19 para el método 2. . . . .             | 93 |
| 8.11. Programaciones de riego aplicadas por la Viña. . . . .   | 95 |

|   |     |
|---|-----|
| E.2. Simulaciones de la etapa 1 con datos de las diferentes temporadas en HYDRUS-1D, para el método con 1 medición. . . . .               | 125 |
| E.3. Simulaciones de la etapa 1 con datos de las diferentes temporadas en <i>Phydrus</i> , para el método con 1 medición. . . . .         | 125 |
| E.4. Simulaciones de la etapa 1 con datos de las diferentes temporadas en HYDRUS-1D, con el método con 4 mediciones . . . . .             | 126 |
| E.5. Simulaciones de la etapa 1 con datos de las diferentes temporadas en <i>Phydrus</i> , para el método con 4 mediciones. . . . .       | 127 |
| E.6. Evolución del contenido de agua en el perfil de suelo para cada temporada, para el método con 1 medición. . . . .                    | 128 |
| E.7. Evolución del contenido de agua en el perfil de suelo para cada temporada, para el método con 4 mediciones. . . . .                  | 129 |
| E.8. Acciones del controlador para la segunda etapa, obtenidas con el agente entrenado (1/2). . . . .                                     | 130 |
| E.9. Acciones del controlador para la segunda etapa, obtenidas con el agente entrenado (2/2). . . . .                                     | 131 |
| E.10. Acciones del controlador para la segunda etapa, obtenidas con el agente entrenado para el segundo método. . . . .                   | 131 |
| E.11. Programaciones de riego para la segunda etapa, obtenidas con el agente entrenado, con raíces distribuidas homogéneamente. . . . .   | 132 |
| E.12. Acciones del controlador para la segunda etapa, obtenidas con el agente entrenado, con raíces distribuidas homogéneamente . . . . . | 133 |
| E.13. Estrategias de riego convencional para la segunda etapa, en la temporada 19-20 para el método 1. . . . .                            | 136 |
| E.14. Estrategias de riego convencional para la segunda etapa, en la temporada 20-21 para el método 1. . . . .                            | 137 |
| E.15. Estrategias de riego convencional para la segunda etapa, en la temporada 21-22 para el método 1. . . . .                            | 138 |
| E.16. Estrategias de riego convencional para la segunda etapa, en la temporada 19-20 para el método 2. . . . .                            | 140 |
| E.17. Estrategias de riego convencional para la segunda etapa, en la temporada 20-21 para el método 2. . . . .                            | 141 |
| E.18. Estrategias de riego convencional para la segunda etapa, en la temporada 21-22 para el método 2. . . . .                            | 141 |

# Capítulo 1

## Introducción

### 1.1. Motivación

Actualmente el crecimiento poblacional, el calentamiento global y el mal uso del agua generan sequía y escasez hídrica, lo que afecta a la sociedad en todos sus aspectos, pero principalmente en la alimentación, lo que hace necesario buscar métodos efectivos de mantener la producción de alimentos aún ante estas condiciones críticas.

En particular en Chile que es un gran exportador de alimentos, los ingresos por concepto de actividades silvoagropecuarias comprenden alrededor de un 3 – 4 % del PIB [10]. El impacto de la sequía se ha echo notar en los últimos años, mostrando un déficit en las precipitaciones alrededor de un 20 % promedio en Chile para la última década [8], lo que urge en la actualización de los procesos agrícolas en materia del uso eficiente del agua.

La aplicación de riego eficiente en la agricultura chilena, permite aprovechar al máximo el escaso recurso hídrico, reducir las pérdidas de agua por una mala administración del recurso y mantener la producción de alimentos a un costo menor. Todo lo anterior, beneficia a un sector donde predomina la pequeña agricultura o familiar de auto-sustento (alrededor del 90 %, según [9]), donde si se lograra aplicar una estrategia de riego óptima se generaría un impacto más allá de lo económico en este alto porcentaje de familias que depende de la agricultura en el país.

Para llevar técnicas más avanzadas de riego a la pequeña agricultura, estas deben ser aplicadas primero en un ambiente lo más favorable posible, para ello en colaboración con el proyecto MSM2021003 Gestión inteligente de recursos hídricos para la agricultura de la Universidad de O'Higgins, este trabajo se centra en el cultivo de uva vinífera, sector con alta disponibilidad técnica y características del proceso productivo que lo hacen un caso de estudio interesante.

Este tipo de fruticultura posee exigencias particulares respecto de las variaciones de requerimientos hídricos por etapa; técnicas de riego como el sometimiento a estrés hídrico de forma controlada; y la evaluación de rendimiento del cultivo en base al balance entre calidad y cantidad del fruto producido para la fabricación del vino.

De esta manera, este trabajo aborda el diseño de programaciones de riego óptimas para el cultivo de uva vinífera, en una Viña ubicada en la región del Libertador Bernardo O'Higgins. Se busca aprovechar la información disponible del terreno del cultivo, condiciones ambientales y datos históricos de precipitaciones, para el diseño e implementación de un modelo de simulación basado en el software HYDRUS-1D [34], utilizado para el análisis del flujo de agua, temperatura y transporte de solutos, integrado en Python mediante la librería *Phydrus*.

El entorno de simulación basado en HYDRUS, simula el flujo de agua en el suelo a partir de: la configuración de parámetros hidráulicos del suelo, información meteorológica diaria, el riego y la información del terreno del día anterior. El modelo entrega una actualización diaria del contenido de agua y evapotranspiración actual del cultivo a lo largo de la temporada.

Utilizando los algoritmos de aprendizaje reforzado (AR): *Deep Q-Network* (DQN), *Twin Delayed Deep Deterministic Policy Gradient* (TD3) y *Proximal Policy Optimization* (PPO) se diseñan controladores del sistema para generar programaciones de riego óptimas, para ello se utilizan las características del sistema agrícola, la información climática disponible y los requerimientos hídricos del cultivo.

La programación de riego corresponde a la salida de una red neuronal (agente), la cual es entrenada a partir de la interacción con el modelo de simulación del sistema, el aprendizaje viene dado por una función de recompensas que representa la respuesta del entorno ante la acción de riego realizada y una observación del nuevo estado del sistema.

El modelo de simulación reemplaza al sistema real en el proceso de entrenamiento y en la evaluación de resultados. El aprendizaje del agente requiere de miles de realizaciones de la temporada, lo que es inviable para un campo cultivado. Por otro lado, la duración de la temporada y el riesgo económico impiden probar las programaciones de riego en un cultivo real.

La evaluación de los resultados se realiza comparando los antecedentes de riego de la Viña con las programaciones de riego generadas por los controladores, para lo cual se integra la información provista por la Viña en el modelo de simulación.

## 1.2. Objetivos

### Objetivo general

Diseñar sistemas de control con aprendizaje reforzado para la optimización del uso de agua de riego en fruticultura, en particular para generar un plan de riego aplicado al cultivo de uva vinífera en la sexta región.

### Objetivos específicos

- Modelar el suelo cultivado a partir del software de simulación HYDRUS-1D, en base a datos reales del sistema.

- Integrar el modelo generado como entorno de simulación para los sistemas de control por aprendizaje reforzado propuestos.
- Diseñar controladores en base a aprendizaje reforzado, con consideraciones en función de los requerimientos hídricos del cultivo y el uso de agua de riego.
- Evaluar el desempeño de los métodos propuestos y comparar con esquemas tradicionales de riego.

### 1.3. Estructura del documento

El presente informe de trabajo de título está dividido en 9 capítulos y se estructura de la siguiente manera:

- En el Capítulo 1 se desarrolla la motivación para realizar el proyecto de título y los objetivos planteados.
- El Capítulo 2 contiene todos los antecedentes relativos a la generación de planificaciones de riego mediante estrategias convencionales y metodologías basadas en aprendizaje reforzado.
- Los Capítulos 3, 4 y 5 corresponden al marco teórico necesario para la generación del modelo de simulación y los sistemas de control.
- El Capítulo 6 corresponde a la metodología, explica el proceso de generación del modelo del sistema y la integración de los sistemas de control con este.
- El Capítulo 7 resume la ejecución de la metodología, detallando las características del modelamiento y control del sistema.
- En el Capítulo 8 se presentan los resultados obtenidos al aplicar los sistemas de control y se genera un análisis de las planificaciones de riego obtenidas.
- Finalmente, en el Capítulo 9 se establecen las principales conclusiones de lo realizado en los capítulos anteriores. Además de plantear los elementos que quedaron fuera de la realización de esta memoria y pueden ser abordados en trabajos futuros.

Adicionalmente, en este informe se incluye la bibliografía utilizada para el desarrollo de este trabajo de título. A modo de complemento se añade un anexo con información que puede ayudar a la comprensión de este informe, sin embargo, su lectura no es imprescindible.

# Capítulo 2

## Estado del arte

En este trabajo se busca diseñar sistemas de control basados en aprendizaje reforzado (AR) para la optimización del uso de agua de riego en fruticultura, a partir de los requerimientos hídricos y las características del cultivo, en particular para el cultivo de uva vinífera.

En búsqueda de la implementación práctica de los sistemas propuestos, es vital analizar las metodologías convencionales actualmente implementadas, revisando sus fundamentos teóricos a modo de comprender de mejor manera el comportamiento no óptimo que estos puedan tener, a fin de establecer las barreras que deben sortear los controladores diseñados.

Con el objetivo de resolver la problemática establecida, es necesario estudiar e investigar planteamientos previos que buscan dar respuesta a la optimización del riego en la agricultura, específicamente centrándonos en aquellos realizados con las técnicas a las que se va a recurrir, es decir, diferentes algoritmos de AR, a modo de sentar las bases que se requieren para el planteamiento del modelo y las metodologías a utilizar.

### 2.1. Estrategias convencionales

Nos referimos a una estrategia convencional como aquella que programa un riego para cumplir con los requerimientos hídricos del cultivo, donde el criterio que define el cumplimiento se determina a partir de información del estado hídrico del suelo y/o la planta, luego la cantidad de agua suministrada busca satisfacer dicho criterio en un tiempo determinado.

Una formulación del programa de riego se basa en la realización de un balance hídrico en el suelo, en el cual se determinan los ingresos (riegos, precipitaciones y ascenso capilar) y pérdidas (evapotranspiración, escurrimientos y percolación profunda) de agua que afectan al contenido de agua en el suelo. Esta formulación (Sección 3.2) permite establecer condiciones que los controladores deban cumplir, a modo de asegurar el crecimiento adecuado del cultivo.

Si se tiene suficiente información para determinar las variables del modelo, se obtienen buenos resultados con esta estrategia pues puede ajustarse a diferentes intervalos de operación y tipo de cultivo.

## Técnicas de riego en fruticultura

La definición de un criterio de riego utilizando el planteamiento de balance hídrico, nos indica que el riego aplicado se determina en base a una estimación del estrés hídrico en la planta, lo cual puede ser perjudicial en caso de no poseer la información suficiente para generar un buen modelo. Es por esto, que las técnicas de riego convencional se basan en la obtención de datos (potencial hídrico de hoja, contenido de agua del suelo, índice de estrés de agua del cultivo, entre otras) directamente desde la planta y/o el suelo, estableciendo umbrales para estos datos y definiendo programas de riego que permitan alcanzarlos [17].

El estrés hídrico afecta el desarrollo de la planta y el fruto, luego se define el umbral de estrés hídrico que indica el contenido de agua para al cual la planta deja de absorber agua a su tasa máxima, lo que termina afectando a la productividad del cultivo.

La metodología se basa en realizar una evaluación diaria del estado hídrico del cultivo respecto del umbral de estrés hídrico establecido, lo que determinará la necesidad de regar o no la planta conforme a las características particulares del período de tiempo evaluado (diario, semanal, etc) y de la etapa de crecimiento en la que se encuentre el cultivo. La definición del umbral puede ser variable conforme se va cambiando de etapa de desarrollo o puede ser estable a lo largo del proceso completo, también puede ser ajustable dependiendo de las condiciones climáticas.

En general, en fruticultura los umbrales definidos apuntan a maximizar la productividad del cultivo por lo que las estrategias de riego buscan mantener al cultivo alejado de estrés hídrico. Sin embargo, una excepción a la regla es el cultivo de uva vinífera puesto que resultados históricos reflejan la necesidad de alcanzar los umbrales de estrés hídrico debido a los efectos positivos en la calidad del producto final (relación piel/pulpa), lo que da pie para una amplitud de variaciones en la forma de definir la metodología, ya que el estrés hídrico posee diferentes efectos según la etapa en la este haga efecto, alterando así el producto de formas diferentes[17].

## 2.2. Control con aprendizaje reforzado

Las técnicas de AR corresponden a una rama de la inteligencia computacional que intenta simular el aprendizaje obtenido a través de la experiencia adquirida por un individuo al interactuar con su entorno, de esta manera el agente observa el estado en el que se encuentra y reacciona acorde a los conocimientos que tiene, obteniendo una recompensa por parte del entorno y un cambio en el estado. La repetición continua de esta interacción permite al agente generar un aprendizaje, que se traduce en la resolución del problema mediante las acciones realizadas conforme a lo aprendido.

Son múltiples características las que definen una metodología de AR, por lo que su aplicación puede variar considerablemente para un tipo de problema similar, en particular los cambios se encuentran en la definición del entorno para la evolución del estado del sistema, el diseño de la acción con la que el agente interactúa, la recompensa asignada y el algoritmo de resolución utilizado, por mencionar los factores más relevantes.

## 2.2.1. Aplicaciones de AR en la planificación de riego

La generación de planificaciones de riego en base a técnicas de AR, presenta casos de estudio como la gestión de riego de arroz en China, con énfasis en la utilización de predicciones de lluvia [6], de igual manera en la memoria de Joaquín Merino [23] se busca aprovechar las predicciones de lluvia para la aplicación de AR en diversos cultivos en la zona sur de Chile. Por otro lado, utilizando AR y diversas estructuras de redes neuronales para el modelamiento, se genera una planificación de riego para tomates en Portugal, para su comparación con estrategias de riego convencional [1]. Otro enfoques son: uso eficiente de agua de riego en diversos cultivos de trigo y maíz, con ajustes en base al uso de agua y el rendimiento del cultivo [33], y en [3] la utilización de AR para definir el día de inicio del periodo de riego, a partir de información del entorno y la etapa de cultivo, con el fin de reemplazar la componente experta del problema.

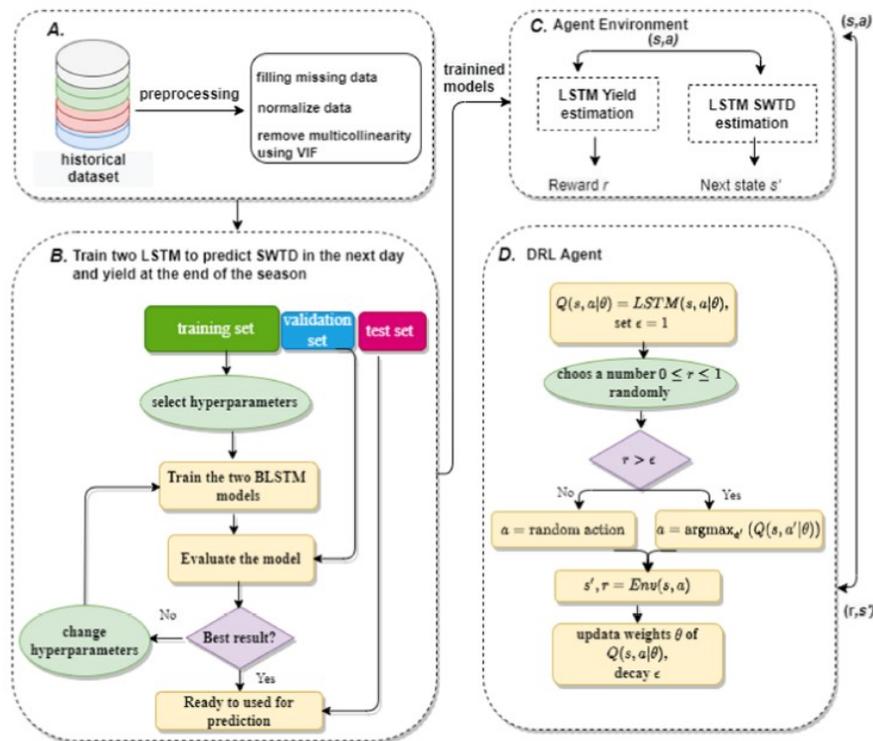


Figura 2.1: Metodología de la implementación del control de riego por aprendizaje reforzado en el cultivo de tomates [1].

## 2.2.2. Variantes en la aplicación de AR

Los distintos casos de estudios presentados se basan en los mismos principios, pero se diferencian en la ejecución de estos, principalmente en la definición de sus entornos, agentes, acciones y recompensas.

## Agente

En general, el agente se asume como el controlador del riego, lo que implica que la acción de control sea definida como un riego en milímetros, pero esta puede ser una decisión que se toma a diario [1][6], periódicamente [33] o sólo al cumplir ciertas condiciones. Por otro lado, la acción de control también puede ser la elección de un día para dar inicio a los riegos del cultivo, como es el caso del problema visto en [3]

## Entorno

La aplicación de AR en agricultura implica que se deben tomar en cuenta las características del cultivo, los antecedentes meteorológicos, las condiciones del suelo y los requerimientos de los agricultores en la definición de los entornos, lo que ofrece un amplio abanico de opciones. El modelamiento en base a una ecuación de balance hídrico (ver el capítulo 3) del sistema, el crecimiento y rendimiento del cultivo [6][23], constituye un entorno cuya dinámica es interpretable a través de las ecuaciones que le dan forma. Otra opción es la utilización de programas computacionales especializados en la simulación del desarrollo de cultivos, en [1] y [33] se utiliza información proveniente del *Decision Support System for Agrotechnology Transfer* (DSSAT) [21] para el entrenamiento de un par de redes neuronales: una predice la cantidad de agua diaria disponible en el suelo en base a la información del entorno, la otra estima el rendimiento final del cultivo en base a los contenidos de agua disponibles diarios de la temporada, ver la Figura 2.2. Finalmente, en el caso [3] se utiliza el modelo MODE-RATO que genera simulaciones del crecimiento del cultivo, genera una estrategia de riego y predicciones de clima para el maíz.

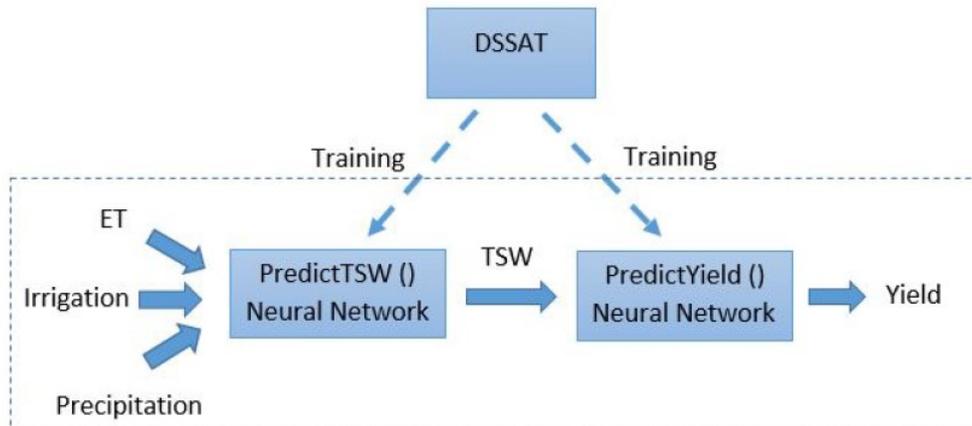


Figura 2.2: Estructura de la generación del entorno basado en el software DSSAT con redes neuronales [33].

## Recompensa

La múltiple variedad de cultivos para estudio, con sus características y requerimientos particulares complejiza la generación de funciones de recompensa, sin embargo, dado que los problemas se centran en el uso eficiente del recurso hídrico sin disminuir la producción, es que se suelen utilizar recompensas de la forma:

$$R_1 = \underbrace{Y \cdot p_Y}_{\text{Rendimiento}} - \underbrace{W \cdot p_W}_{\text{Uso de agua}} \quad (2.1)$$

Donde  $Y$  y  $p_Y$  representan el rendimiento final del cultivo y el precio o valor asociado respectivamente, de manera análoga  $W$  y  $p_W$  representan el agua total consumida y el costo del uso de agua respectivamente, siendo esta formulación la más común [1][33][23]. La formulación [3] es similar, pero agrega el costo operacional ( $p_o$ ) en función de la cantidad de riegos realizados ( $N$ ) y un costo de producción fijo ( $X$ ) que depende del cultivo ( $R_2 = R_1 - N \cdot p_o - X$ ). Por otro lado, en [6] se plantea una formulación más compleja que pondera los efectos del riego en: el aprovechamiento del agua de lluvia, es decir, que cualquier precipitación posterior a una irrigación sea consumida no se pierda por escurrimiento o drenaje, la lógica del riego en términos de técnicas convencionales, básicamente, se desea regar sólo cuando el cultivo se encuentra bajo cierto umbral hídrico, y finalmente una componente del efecto de la irrigación en el rendimiento del cultivo.

## Algoritmos

Finalmente, cuando se tiene la estructura del aprendizaje reforzado se debe seleccionar el algoritmo que mejor se adapte al problema o aquel que mejor aproveche las características e información disponible. En los trabajos [1] y [6] se utiliza DQN, sin embargo, en el primero el enfoque está en las diferentes arquitecturas de redes neuronales que este algoritmo puede utilizar, mientras que en el segundo se destaca la capacidad del algoritmo de utilizar las predicciones de lluvia. En [3] se elige *Q-Learning*, ya que la actualización de las reglas era directo y el rango de acciones limitado. Por otro lado, dado que el factor de rendimiento en la recompensa llega al término de la temporada, en [33] se propone la utilización de SARSA( $\lambda$ ). Finalmente, en la memoria [23] se realizan pruebas con otros algoritmos (PPO, DDPG y TD3) para ver cual se adapta mejor al problema y a la recompensa planteada.

## 2.3. Discusión

En el contexto de cultivo de uva vinífera, se tiene que las técnicas utilizadas actualmente se basan en mediciones del terreno o de la planta para definir un requerimiento hídrico en base a la etapa de desarrollo actual, lo que a su vez es utilizado para generar un programa de riego que satisfaga dicho requerimiento.

De esta manera, se tiene una metodología que carece de una optimización formal del problema, pues pese a tomar en cuenta la disponibilidad de agua, las mediciones y la etapa de desarrollo, no considera las precipitaciones en su formulación, lo que genera una programación reactiva ante estos eventos, lo cual en el contexto de escasez hídrica genera situaciones donde las precipitaciones no son completamente aprovechadas.

Por otro lado, la metodología debe tomar en cuenta el rendimiento del cultivo, el cual para el caso de la fruticultura es complicado pues depende de un balance entre cantidad y calidad de los frutos obtenidos, lo cual en el caso particular de la uva para vino es fundamental tomar en cuenta debido a las técnicas de déficit hídrico que afectan la programación.

En cuanto a la utilización de Aprendizaje Reforzado, se observa en los trabajos estudiados que mediante la entrega de un modelo apropiado, información suficiente y una correcta formulación de las funciones objetivo y restricciones se tiene la capacidad de generar programaciones de riego optimizadas, sin las carencias que las técnicas de riego actuales poseen.

El desafío recae en la generación de una metodología que aproveche la información disponible, las características del cultivo y los requerimientos que tiene la producción de este. Para esto es necesario definir un entorno adecuado para la interacción con el agente, para el cual se debe definir y acotar las acciones de control, por otro lado, se tiene que elegir una recompensa que represente correctamente las implicaciones de las acciones elegidas, y permita la utilización correcta de los algoritmos de AR elegidos.

# Capítulo 3

## Antecedentes agrícolas

En este capítulo se hace una revisión de conceptos y estudios relevantes de sistemas agrícolas, enfocados principalmente en características esenciales sobre el cultivo y producción de uva para vino a tomar en cuenta para la aplicación práctica del problema de riego a solucionar. Siguiendo esta línea es que se aborda la definición del modelo en base al balance hídrico generado a partir de la ecuación del mismo nombre, donde la comprensión de su formulación es esencial para el desarrollo de este trabajo de título.

### 3.1. Propiedades del suelo

La capacidad de un cultivo para desarrollarse y alcanzar su mayor potencial productivo depende en su mayoría de las características del terreno en el que se encuentra, es por esto que se hace fundamental el estudio de los parámetros físicos del terreno que determinan la disponibilidad de agua, oxígeno y nutrientes para la planta [14].

#### 3.1.1. Textura y porosidad

La textura de suelo representa la proporción de partículas de arena, limo y arcilla en la conformación del suelo. Múltiples reacciones físicas, químicas y biológicas en el suelo dependen de la textura, debido a los efectos que tiene en características como: la retención de humedad, plasticidad, permeabilidad, fertilidad, riego y drenaje [32] [14]. A continuación se presentan las diferencias entre los tipos de textura:

- Partículas de arena: entre 0,05 y 2 [mm] de diámetro, aumentan el tamaño de los espacios de los poros entre partículas, facilitando el movimiento de aire y del agua de drenaje, debido a su elevada permeabilidad al agua poseen una escasa retención de agua y nutrientes.
- Partículas de limo: entre 0,002 y 5 [mm] de diámetro, mayor velocidad de intemperización (fractura y descomposición del suelo al estar sometido a la intemperie), mayor

liberación de nutrientes solubles que la arena y gran capacidad de retención de agua disponible para el crecimiento vegetal.

- Partículas de arcilla: menos de 0,002 [mm] de diámetro, capaces de retener agua contra la fuerza de gravedad y cuentan con la presencia de cargas eléctricas negativas en su superficie externa que atraen y retienen cationes de manera reversible ( $\text{Ca}^{2+}$ ,  $\text{Mg}^{2+}$ ,  $\text{Na}^+$ ,  $\text{K}^+$ ). Debido a su baja permeabilidad al agua poseen una elevada retención de agua y nutrientes.

De esta manera, el tamaño de los poros viene dado por la textura, lo que a su vez determina la dinámica de los flujos de agua y solutos al interior del suelo. Adicionalmente, la textura repercute en la porosidad del suelo, referida al volumen total de espacio poroso o número de poros que hay entre partículas sólidas (orgánicas e inorgánicas) del suelo, el cual está ocupado en diferentes porcentajes de aire y agua.

La porosidad es la relación entre el volumen de poros y el volumen de sólidos presentes en el suelo y está determinada por la textura, estructura y grado de compactación (densidad aparente) del mismo, por lo cual sus valores varían en cada suelo.

### 3.1.2. Humedad del suelo

La humedad de suelo o contenido de agua (WC, del inglés *water content*), es una característica expresada adimensionalmente ( $\frac{\text{mm}^3 \text{ de agua}}{\text{mm}^3 \text{ de suelo}}$ ) que refleja la cantidad de agua presente en el mismo, la cual está determinada por las características del suelo y la actividad orgánica presente. En base al contenido de agua se definen una serie de puntos y umbrales hídricos que ayuden a comprender la dinámica del suelo.

- **Punto de Saturación (PS):** es la cantidad máxima de agua que puede retener, depende de la materia orgánica, del tipo y cantidad de arcilla que presente, y se expresa en  $\frac{\text{mm}^3 \text{ de agua}}{\text{mm}^3 \text{ de suelo}}$ . En este punto las raíces son incapaces de extraer agua, por lo que la planta se “ahoga”.
- **Capacidad de Campo o *Field Capacity* (CC o FC):** es la cantidad de agua que puede retener el suelo contra la fuerza de la gravedad, después de un riego o lluvia tal que humedece todo el suelo, se puede estimar a partir de PS:

$$\text{CC} = \frac{\text{PS}}{1,84} - 0,48 [\%].$$

- **Punto de Marchitez Permanente o *Wilting Point* (PMP o WP):** es el porcentaje de humedad de suelo en el que las plantas se marchitan y no se recuperan. Este valor se puede estimar como  $\text{PMP} = \text{CC} \times 0,595$ , pero también se puede ajustar según el cultivo estudiado.
- **Humedad aprovechable (HA):** es la cantidad total de agua en el suelo que la planta puede absorber, y es la diferencia entre la humedad a CC y el PMP ( $\text{HA} = \text{CC} - \text{PMP}$ ), con la HA se puede determinar una lámina de agua aprovechable en el suelo al multiplicar por la profundidad de este.

- **Umbral de Estrés Hídrico (EH):** es el WC al cual la planta deja absorber agua a su máxima velocidad, la estadía del terreno del cultivo por mucho tiempo es de gran impacto para el desarrollo de los cultivos. Se encuentra entre CC y PMP, pero su valor exacto depende de las condiciones del terreno y del cultivo, un buen estimativo es:  $EH = PMP + 0,7 \times HA$ .

La definición de estos puntos permite visualizar el terreno cultivado como un estanque de agua que debe mantenerse en ciertos niveles para garantizar el crecimiento del cultivo, estos son los requerimientos hídricos, las precipitaciones y el riego son claves para cumplirlos. En la Figura 3.1 se pueden apreciar la distribución de los umbrales en el suelo.

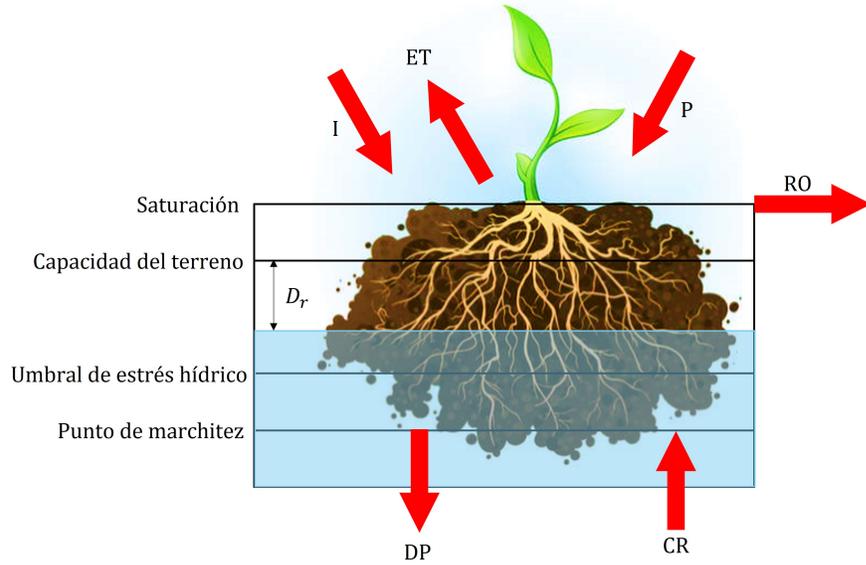


Figura 3.1: Balance hídrico en la zona de la raíz.

## 3.2. Ecuación de balance hídrico

El balance hídrico del suelo representa la evolución de la dinámica del agua almacenada en este, la cantidad de agua disponible se define en términos del déficit de humedad o agotamiento en la zona de las raíces  $D_r$  (*root zone soil moisture deficit*). Luego, se determina el déficit de humedad al final del día  $k$  ( $D_{r,k}$ ) en función del ingreso de agua por: precipitaciones, irrigación y ascenso capilar, y por la pérdida debido a: evapotranspiración, percolación profunda y escurrimiento superficial. Con estos factores se determina la ecuación de balance hídrico:

$$D_{r,k} = D_{r,k-1} + ET_{a,k} + RO_k + DP_k - P_k - CR_k - I_k, \quad (3.1)$$

con  $D_{r,k-1}$  el déficit o agotamiento del final del día anterior,  $ET_{a,k}$  la evapotranspiración,  $RO_k$  el escurrimiento superficial (*runoff*),  $DP_k$  la percolación profunda (*deep percolation*),  $P_k$  las precipitaciones,  $CR_k$  el ascenso capilar (*capillary rise*) e  $I_k$  el agua que ingresa por

concepto de riego. El déficit hídrico y el resto de variables de la ecuación se miden en  $[mm]$ , ya que el balance considera bloques de suelo con un volumen de  $1 [m^3]$  y bajo el supuesto de que el agua se distribuye de manera homogénea en el área de cultivo estudiada.

A continuación, en la Figura 3.1 se representa de manera gráfica el balance hídrico en la zona de la raíz, también se presenta la cotas hídricas asociadas a la máxima capacidad del terreno para almacenar agua, el umbral de estrés hídrico y el punto de marchitez.

Utilizando la Figura 3.1 como referencia se procede a desarrollar con mayor detalle cada una de las variables que definen la dinámica por balance hídrico vista en la ecuación (3.1).

### 3.2.1. Evapotranspiración (ET)

Una de las principales variables a estudiar para determinar la dinámica del nivel de agua disponible en el terreno cultivado es la evapotranspiración (ET), pues está altamente relacionada al rendimiento obtenido al momento de cosechar, además de ser una variable dependiente de las características del cultivo y el entorno.

La ET busca determinar las pérdidas de agua debido a la evaporación del suelo desnudo y las asociadas a la transpiración en las hojas de la planta. Estos procesos ocurren de manera simultánea y no hay una forma sencilla de distinguir uno del otro. El cálculo de la ET viene dado principalmente por las siguientes variables: exposición al sol, temperatura, humedad relativa del aire, velocidad del viento, el tamaño y crecimiento del cultivo.

Lo anterior, permite generar una relación donde en etapas de desarrollo iniciales las pérdidas por evaporación corresponden a casi la totalidad de las pérdidas, pero el crecimiento de la planta revierte la situación, al aumentar la pérdida por transpiración y disminuir la evaporación producto de la sombra generada por la planta en el terreno de cultivo, esta relación se puede apreciar de mejor manera en la Figura 3.2.

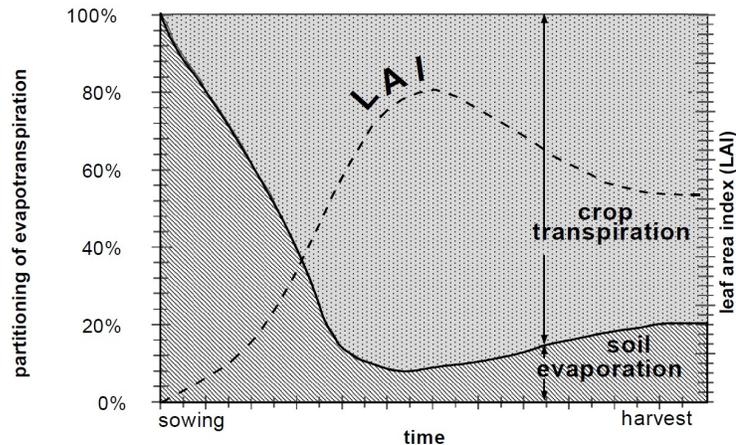


Figura 3.2: Distribución de la evapotranspiración, diferenciada en evaporación y transpiración durante el período de crecimiento de un cultivo[2].

Expresar la evolución de ET en función de la evaporación y la transpiración por separado

es posible, pero posee una alta complejidad debido a que el aporte de cada proceso no es constante en el tiempo. Adicionalmente, los cálculos de pérdidas de agua por evaporación y transpiración también son complejos debido a la alta cantidad de variables que influyen en estos. De esta manera, para calcular la ET diaria, se recurre a metodología específica desarrollada para la estimación de este fenómeno, que se expone en [2].

En dicha guía se propone el cálculo de la ET diaria a partir de otras definiciones particulares de evapotranspiración como: ET de referencia ( $ET_o$ ), ET de cultivo o en condiciones ideales ( $ET_c$ ) y la ET real ( $ET_r$ ). Estas definiciones dependen de las características del cultivo, condiciones del terreno y ambientales, en particular se recurre a las variables meteorológicas: radiación solar, humedad relativa del aire, velocidad del viento y temperatura.

En [2] también se propone la medición de la ET mediante un lisímetro o la estimación de  $ET_o$  con el método del tanque de evaporación, sin embargo, dichos elementos no son tomados en cuenta en este trabajo. De esta manera, la metodología para la determinación de la ET es la que se describe a continuación.

En primer lugar, se calcula la evapotranspiración de referencia, la cual se define a partir de características medioambientales particulares que indican el efecto de la atmósfera en la ET, independiente del tipo de cultivo, esto permite que la  $ET_o$  pueda ser calculada a partir de datos históricos para un lugar y un tiempo específico del año, sin tomar en cuenta el tipo de cultivo y las características del suelo. El método para determinar su valor es el FAO Penman-Monteith [2], correspondiente a una versión estandarizada de la original que fue publicada en 1948 [28], la ecuación estandarizada es la siguiente:

$$ET_o = \frac{0,408\Delta(R_n - G) + \gamma \frac{900}{T+273} u_2 (e_s - e_a)}{\Delta + \gamma(1 + 0,34u_2)} \quad (3.2)$$

con  $ET_o$  la evapotranspiración de referencia en [mm],  $\Delta$  la pendiente de la curva de presión de vapor respecto a la temperatura [ $\frac{kPa}{^\circ C}$ ],  $R_n$  la radiación solar diaria neta en la superficie del cultivo en [ $\frac{MJ}{m^2}$ ],  $\gamma$  la constante psicrométrica en [ $\frac{kPa}{^\circ C}$ ],  $T$  la temperatura media a dos metros de altura en [ $^\circ C$ ],  $u_2$  la velocidad del viento a dos metros de altura en [ $\frac{m}{s}$ ],  $e_s$  la presión de vapor de saturación en [kPa] y  $e_a$  la presión de vapor real en [kPa]. El cálculo de cada una de estas variables se encuentra en el apéndice A. De esta manera, se obtiene una expresión numérica (3.2) para la  $ET_o$  en milímetros, lo cual se logra al ajustar las unidades de las variables utilizadas.

Una vez se calcula la evapotranspiración de referencia ( $ET_o$ ), el siguiente término es la ET de cultivo ( $ET_c$ ), que se refiere a la ET para cultivos libres de enfermedades, fertilizados correctamente, con crecimiento en largos campos y bajo condiciones óptimas de riego, que alcanzan producción máxima bajo las condiciones climáticas dadas. De esta manera, en [2] se define el coeficiente de cultivo  $K_c$  que relaciona  $ET_c$  y  $ET_o$  de la forma:

$$ET_c = K_c ET_o \quad (3.3)$$

El valor de  $K_c$  depende de la etapa de crecimiento del cultivo y las características de terreno y medioambientales en las que se desarrolla. El valor de  $K_c$  se determina día a día,

a partir de valores tabulados para las diferentes etapas de desarrollo ( $K_{c\text{ini}}$ ,  $K_{c\text{mid}}$  y  $K_{c\text{end}}$ ), con los cuales se construye una curva del parámetro en función del tiempo, como se muestra en la Figura 3.3a. La estimación de los parámetros que definen la curva se realiza en base a información histórica respecto de variables como la frecuencia de riego, velocidad del viento, humedad relativa y altura del cultivo.

Lo anterior permite, utilizar una única variable que engloba los efectos de la evaporación y la transpiración, sin embargo, la siguiente formulación permite separarlos en dos nuevos coeficientes:

$$ET_c = K_c ET_o = (K_e + K_{cb}) ET_o, \quad (3.4)$$

con  $ET_e$  coeficiente de evaporación y  $ET_{cb}$  coeficiente de base del cultivo. El  $K_{cb}$  se define como la razón  $\frac{ET_c}{ET_o}$  cuando la superficie está seca, es decir, no hay evaporación pero la transpiración ocurre a su tasa potencial.  $K_e$  representa la componente de evaporación cuando la superficie queda húmeda posterior a precipitaciones o riego, su valor es cercano a cero cuando el suelo está seco. Adicionalmente, el crecimiento del cultivo también afecta a los coeficientes, debido al aumento de transpiración asociado y la disminución de la evaporación producto de la sombra proyectada. En las Figuras 3.3 se aprecian las diferencias entre la utilización del coeficiente singular  $K_c$  o el doble con  $K_{cb}$  y  $K_e$ .

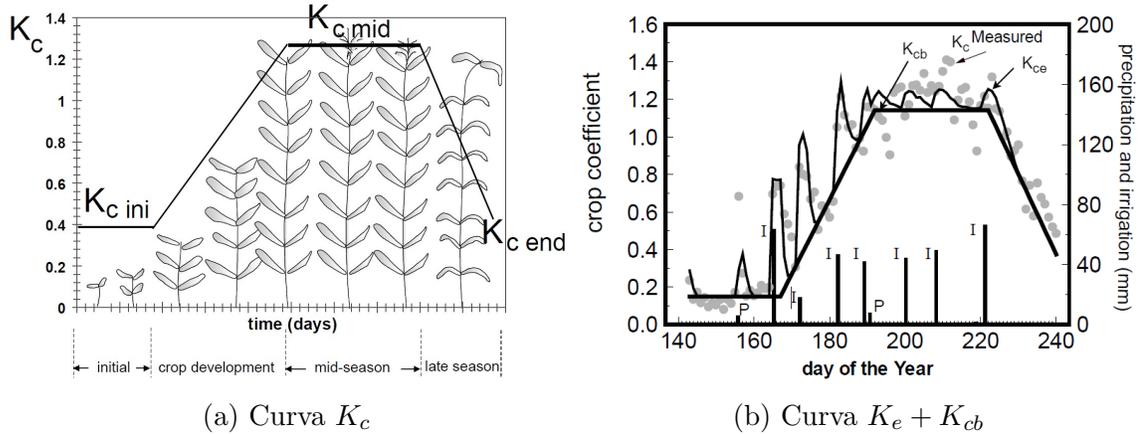


Figura 3.3: Evolución teórica del parámetro  $K_c$  en función del tiempo.[2]

Finalmente, se calcula la evapotranspiración real o actual ( $ET_a$ ), que corresponde a la estimación de las pérdidas de agua diarias por concepto de evapotranspiración del cultivo estudiado. Este valor depende del grado de estrés hídrico en el que se encuentra la planta, el cual se representa mediante el coeficiente de estrés  $K_s$ , que indica la disponibilidad de agua para suplir los requerimientos hídricos del cultivo.

El coeficiente de estrés hídrico  $K_s$ , se determina a partir del total de agua disponible en la zona de la raíz  $TAW$  (*Total Available Water*), el cual se calcula según la siguiente expresión:

$$TAW = 1000 (\theta_{fc} - \theta_{wp}) Z_r, \quad (3.5)$$

con  $\theta_{fc}$  el contenido de agua en la capacidad de campo (*field capacity*) en  $[m^3 m^{-3}]$ ,  $\theta_{wp}$  el contenido de agua en el punto de marchitez (*wilting point*) y  $Z_r$  la profundidad de las raíces

en  $[m]$ . Los parámetros  $\theta_{fc}$  y  $\theta_{wp}$  dependen del tipo y las características del suelo, y vienen dados por tabla. Por otro lado, la profundidad de las raíces debe ser medida o estimada empíricamente a partir de las características y la etapa de crecimiento del cultivo.

De esta manera, se obtiene el  $TAW$  que corresponde a la cantidad de agua que el cultivo puede extraer de la zona de la raíz, y como se ve en la ecuación (3.5) depende del tipo de suelo y la profundidad de sus raíces. A partir de este valor, se puede calcular el agua fácilmente disponible  $RAW$  (*readily available water*), que corresponde a la cantidad de agua que la planta puede absorber en la zona de la raíz sin pasar a un estado de estrés hídrico, se define:

$$RAW = p \cdot TAW \quad (3.6)$$

con  $p$  el porcentaje del total de agua disponible ( $TAW$ ) en el suelo que puede ser absorbida por las raíces, antes de que estas lleguen a la zona de estrés hídrico ( $p \in [0,1]$ ). El valor de  $p$  está dado por tabla y depende del tipo de cultivo, pero también puede ser ajustado en función de la evapotranspiración de cultivo  $ET_c$  de la forma:

$$p_{ajustado} = p + 0,04(5 - ET_c). \quad (3.7)$$

De esta manera, el coeficiente de estrés  $K_s$  se determina como[2]:

$$K_s = \begin{cases} 1 & , \text{ si } D_r \leq RAW \\ \frac{TAW - D_r}{TAW - RAW} & , \text{ si } D_r > RAW \end{cases} \quad (3.8)$$

La ecuación (3.8) muestra una relación entre el agotamiento o déficit hídrico y el contenido de agua disponible en la zona de la raíz, la cual puede ser entendida de mejor manera al observar la Figura 3.4 que representa gráficamente la interacción entre las variables descritas anteriormente.

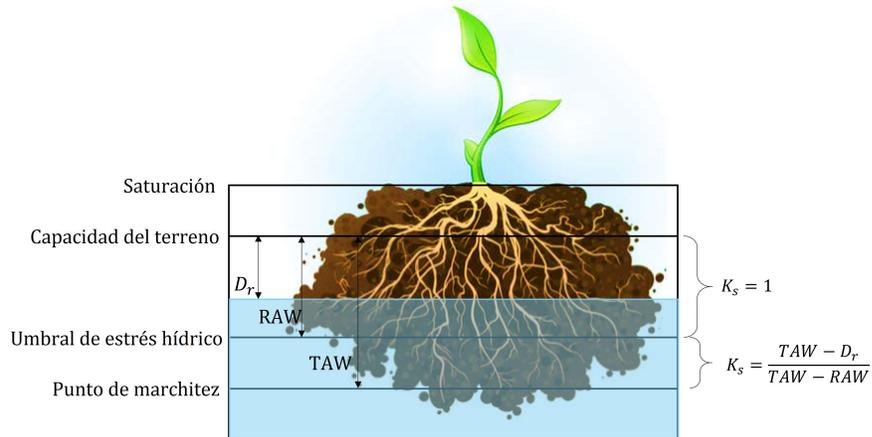


Figura 3.4: Interpretación gráfica de la relación entre  $D_r$ ,  $TAW$ ,  $RAW$  y  $K_s$ .

Ahora que el valor de  $K_s$  posee una expresión definida, se puede determinar el valor de la evapotranspiración real  $ET_a$  mediante la siguiente expresión:

$$ET_a = K_s ET_p \quad (3.9)$$

### 3.2.2. Escurrimiento Superficial (RO)

Es una forma de pérdida de agua en el sistema, que ni siquiera hace ingreso a la zona de la raíz, puesto que el escurrimiento superficial (RO por *runoff*) es el agua de lluvia que cae en el terreno de cultivo y no logra permear hacia capas inferiores al punto de saturación del terreno y termina fluyendo por la superficie de este.

Una de las técnicas para la estimación de esta variable, se basa en un evento de precipitación y en las características físicas de la cuenca (característica del terreno), el Método del Número de la Curva (CN)[20], desarrollado por el *Soil Conservation Service* para los EE. UU. en 1972, que ha sido adaptada para condiciones en otras partes del mundo [4]. El CN representa la impermeabilidad relativa del sistema suelo-vegetación, tal que  $CN \in [0, 100]$ , donde es 0 para una superficie con infiltración infinita y 100 para un terreno completamente impermeable, y en consecuencia con escurrimiento superficial total.

El método comienza con el cálculo del parámetro  $S$  que representa la profundidad máxima de agua retenida durante un evento de precipitación en  $[mm]$ , calculado de la forma:

$$S = 250 \left( \frac{100}{CN} - 1 \right) \quad (3.10)$$

Donde, CN depende de las características del terreno, el tipo de cultivo y las condiciones ambientales, entre otros factores que influyen en cada caso de estudio [4]. Luego, si las precipitaciones  $P_k$  satisfacen  $P_k > 0,2S$ , entonces se produce escurrimiento superficial y se cumple que:

$$RO_k = \frac{(P - 0,2S)^2}{P + 0,8S} \quad (3.11)$$

### 3.2.3. Percolación Profunda (DP)

La pérdida de agua en la zona de la raíz por DP (*Deep Percolation*) es aquella que descienden a capas inferiores a las del punto de marchitez en el terreno cultivado (ver Figura 3.4). Esto ocurre cuando el contenido de agua excede la capacidad del terreno producto del ingreso de agua en exceso tras precipitaciones o una irrigación masiva.

Para determinar su valor, se toma en cuenta la ecuación de balance hídrico (3.1) al final del día en que ocurrió un evento de ingreso de agua, de esta manera se tiene un déficit nulo ( $D_{r,k} = 0$ ) y se puede reordenar la ecuación en torno a la percolación  $DP_i$  (3.12),

$$DP_k = (P_k - RO_k) + I_k - ET_{a,k} - D_{r,k-1}, \quad (3.12)$$

donde se debe cumplir  $DP_k \geq 0$ . Mientras el contenido de agua en la zona de la raíz se encuentre bajo la capacidad del terreno (i.e.,  $D_{k,i} > 0$ ), se asume que no existe percolación y  $DP_k = 0$  [2]. Dado lo anterior, es que la percolación profunda se suele utilizar para poner una cota mínima al agotamiento en la zona de la raíz, la cual se encuentra en la capacidad del terreno, tal cual se ve en la Figura 3.4, con lo que se tiene  $D_{r,k} \in [0, TAW]$ .

### 3.2.4. Ascenso Capilar (CR)

La cantidad de agua que sube producto de ascenso capilar corresponde a aquella que proviene desde una capa de tierra inferior saturada de agua y llega a la zona de la raíz o hasta la superficie. Esta variable depende del tipo de suelo, la profundidad en la que se encuentre la capa de tierra saturada y la humedad de la zona de la raíz [20]. Cuando la distancia de la capa saturada a la zona de la raíz es mayor que 1[m] se considera  $CR = 0$ , en caso contrario se debe revisar el tipo de suelo y revisar los gráficos de flujo por ascenso capilar para cada caso [20][2].

## 3.3. Características de la producción de uva de vino

Una vez se comprenden la dinámica del terreno cultivado y los procesos que influyen en el crecimiento de los cultivos, se pueden revisar los detalles que hacen diferente a cada especie. En esta sección se plantean las características más relevantes de la producción de uva de vino.

### 3.3.1. Etapas de desarrollo del cultivo

En un proceso de producción agrícola, los tiempos y acciones se ajustan a las etapas de desarrollo del cultivo, para el caso de la uva debemos analizar su ciclo de crecimiento en la vid e identificar los requerimientos y sensibilidades de la planta ante eventos como escasez hídrica, cambios de temperatura, entre otros. Las 4 fases de desarrollo de la uva [17] son las siguientes:

1. **Fases I - Crecimiento Vegetativo:** Esta etapa se centra en la división celular, propiciando el desarrollo del fruto en las etapas futuras. Debido al proceso de crecimiento consecuente del aumento de células, es que el cultivo es muy sensible al déficit hídrico afectando la producción final de racimos.
2. **Fase II - Desarrollo Floral:** Se trata de un rápido desarrollo de las semillas y el inicio del proceso de acumulación de taninos, compuestos que según su concentración afectan la calidad del vino producido, en términos de amargura, astringencia, el color, la estabilidad y la palatabilidad.
3. **Fase III - Inicio de crecimiento del Fruto:** En este momento se comienza la acumulación de agua en el fruto, haciendo que este crezca, esto va acompañado con el inicio de el envero (cambio de color de las uvas). Esta etapa es muy susceptible al estrés hídrico debido a la expansión de las células por el agua que ingresa, pero también es sensible al exceso de agua debido a que disminuye la concentración de ácidos y fenoles (flavonoles, procianidinas, taninos y antocianos principalmente) que dan el color, aroma y sabor lo que disminuye la calidad del vino producido.

4. **Fase IV - Desarrollo final del Fruto:** Corresponde a la etapa donde concluye el envero, se acumulan los azúcares y se reduce la presencia de ácidos en el fruto. Dando lugar a la culminación del crecimiento del fruto, obteniendo el color, sabor y olor que caracterizan a la cepa. Finalmente, una vez se cosechan las uvas la vid inicia el proceso de preparación para la siguiente temporada.

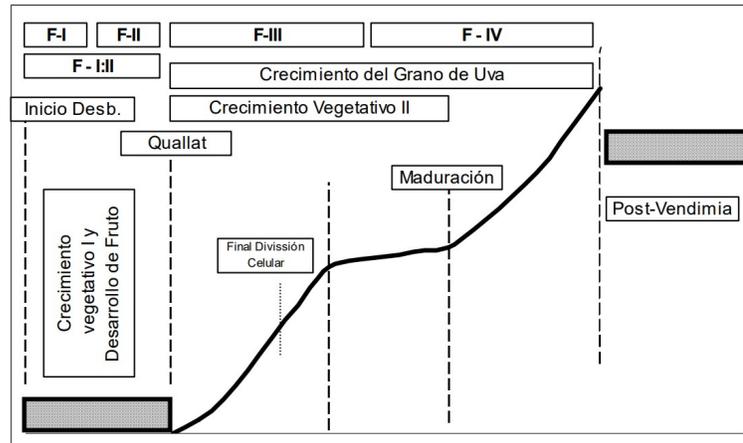


Figura 3.5: Definici3n de los per3odos para los tratamientos de riego en un ciclo anual de la Viña [17].

En la Figura 3.5 se aprecian las cuatro etapas del proceso de desarrollo, sin embargo es com3n encontrarlo como uno de tres, donde las primeras fases I y II se agrupan como un solo per3odo. Cabe destacar que muchos factores afectan al crecimiento de la planta, entre ellos el uso de fertilizantes, condiciones de suelo, clima, exposici3n al sol, estr3s h3drico, enfermedades, y muchos otros.

### 3.3.2. Estr3s h3drico

A continuaci3n se definen brevemente las consecuencias del estr3s h3drico en las diferentes etapas de desarrollo de la planta [13], a modo de caracterizar la variable del agua en el proceso de desarrollo:

- **Estr3s h3drico en Fases I y II:** Esta etapa al ser de divisi3n celular y de desarrollo, la falta de agua disminuye el n3mero de c3lulas de las bayas, por lo tanto, el tama1o de la baya se reduce considerablemente.
- **Estr3s h3drico en Fase III:** A medida que disminuye el proceso de divisi3n celular, en t3rminos del tama1o del fruto, este es menos sensible a la escasez de agua, ya que como se discuti3 antes se pasa el enfoque a la concentraci3n de fenoles, los cuales generan los cambios en el color, aroma y sabor de la uva, y por consecuencia del vino que se produce con ellas. Dependiendo del fenol, cambia el per3odo de estr3s h3drico necesario para su acumulaci3n.

- **Estrés hídrico en Fase IV:** En las etapas finales de desarrollo la escasez de agua se sigue reflejando en la concentraciones de fenoles, pero en menor medida a la fase anterior. Por otro lado, la escasez de agua después de la cosecha y de preparación de la vid, pueden afectar a la floración en las primeras fases de la siguiente temporada.

Además de los efectos mencionados previamente, se ha observado que el sometimiento a estrés hídrico en el cultivo en particular en Fase III tiene consecuencias cuando se acumulan temporadas en las que se aplica riego por déficit hídrico controlado, pues se ve afectada la capacidad de acumulación de nutrientes en la raíces, lo que las hace perder vigor al disminuirse las reservas de carbohidratos.

### 3.3.3. Rendimiento del cultivo

En términos generales, no es posible cumplir los requerimientos hídricos de los cultivos en todo momento, lo que genera que la evapotranspiración real (3.9) sea inferior a la de cultivo (3.3), es decir,  $ET_a < ET_c$ . Bajo esta condición, se genera estrés hídrico en la planta, lo que posee un impacto negativo en el crecimiento y producción del cultivo [12].

Sin embargo, esto no es del todo cierto en el cultivo de uva para vino, pues como hemos visto el estrés hídrico posee diferentes efectos en las etapas de desarrollo, siendo en muchos casos necesario para aumentar la calidad del producto final. Debido a esto, el cálculo del rendimiento del cultivo no está asociado únicamente a la producción final de frutos, sino que también a un estudio experto de la calidad del vino que se puede generar a partir de estos.

Lo anterior impide generar una expresión numérica del rendimiento de la uva, ya que pese a la existencia de formulaciones basadas en la relación entre  $ET_a$  y  $ET_p$  [16], estas buscan que se maximice la cantidad producida, mientras que en el caso de las uvas se puede desear una menor cantidad pero mayor calidad. Este *tradeoff* entre calidad vs cantidad depende del criterio y objetivos de cada Viña, por lo que la evaluación de rendimiento no puede ser expresada de forma genérica.

### 3.3.4. Potencial Hídrico Xilemático (Px)

El potencial hídrico xilemático es una medición en la hoja de la planta, para monitorear el estado hídrico de la planta, pues refleja la tensión del agua en los vasos conductores de la planta, con lo que se define el nivel de estrés de esta, a partir de la disponibilidad de agua en el suelo y la demanda evaporativa de la atmósfera. Es usado principalmente en árboles frutales, lo que ha generado estrategias de riego en base a mediciones realizadas con cámara de presión. [25]

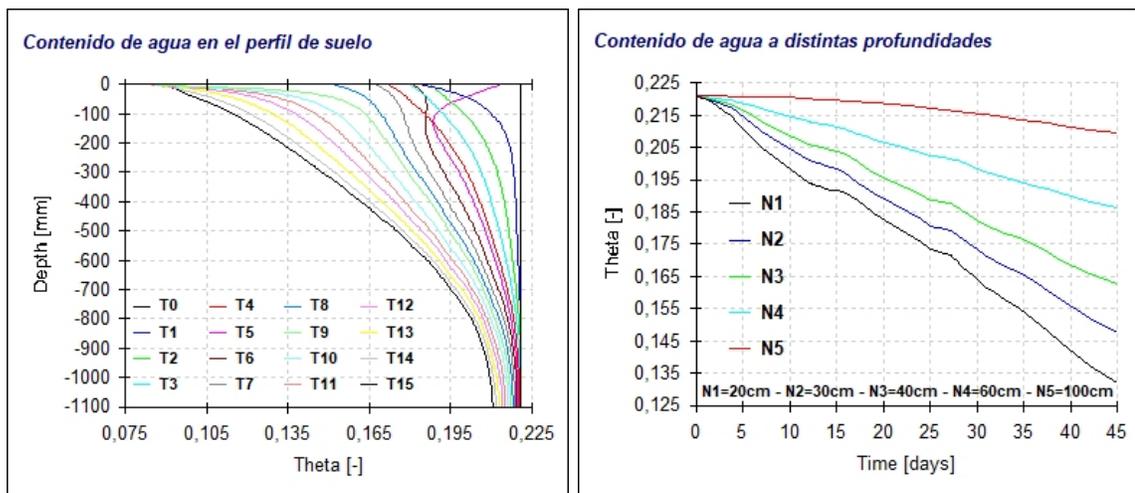
Su principal ventaja es la uniformidad de los datos obtenidos dentro de diferentes partes de la canopia, independiente de la luminosidad que presenta la hoja. El tipo, tamaño, forma y condición fisiológica de la hoja no influyen en la medición de Px. Los elementos que si afectan su medición son las condiciones atmosféricas (radiación solar, temperatura y humedad relativa), hora del día de la medición y la ubicación de la hoja en el árbol.

# Capítulo 4

## HYDRUS-1D

En este capítulo se entregan los conceptos básicos para la comprensión del funcionamiento del software de simulación HYDRUS-1D. El programa simula la dinámica de agua, solutos y calor en el suelo, para ello se configura en función de las características del suelo (textura y contenido de agua, ver sección 3.1), datos meteorológicos (precipitaciones, evapotranspiración, entre otros) y la especie cultivada.

Con HYDRUS-1D se puede observar la evolución de contenido de agua en todo el perfil de suelo, tomando “muestras” en ciertos instantes de tiempo (Figura 4.1a), también permite explorar el comportamiento en puntos particulares del suelo replicando lo que haría un sensor ubicado en un caso real (Figura 4.1b).



(a) Contenido de agua vs profundidad

(b) Contenido de agua vs tiempo

Figura 4.1: Ejemplos de la visualización del contenido de agua en HYDRUS-1D

En términos técnicos, el software resuelve numéricamente la ecuación de Richards [27] para el flujo de agua variablemente saturado, absorción de agua por raíces y ecuaciones de dispersión por advección para el transporte de calor y solutos en un medio unidimensional. [34]

La versión HYDRUS (2D/3D) del programa permiten la configuración en 2 y 3 dimensiones del medio, agregando la capacidad de simular nuevos comportamientos del movimiento de agua, calor y solutos. Adicionalmente, esta versión permite la visualización gráfica de la dinámica de las variables simuladas. Para el desarrollo de este trabajo se utiliza HYDRUS-1D, ya que es de acceso gratuito y puede ser implementado en Python con la librería *Phydrus*.

Al utilizar HYDRUS, la configuración y parámetros seleccionados deben permitir simular el caso real. A continuación se presenta una guía de configuración del software, donde los elementos que se detallan son replicados posteriormente en la implementación en Python donde se obtiene el modelo de simulación.

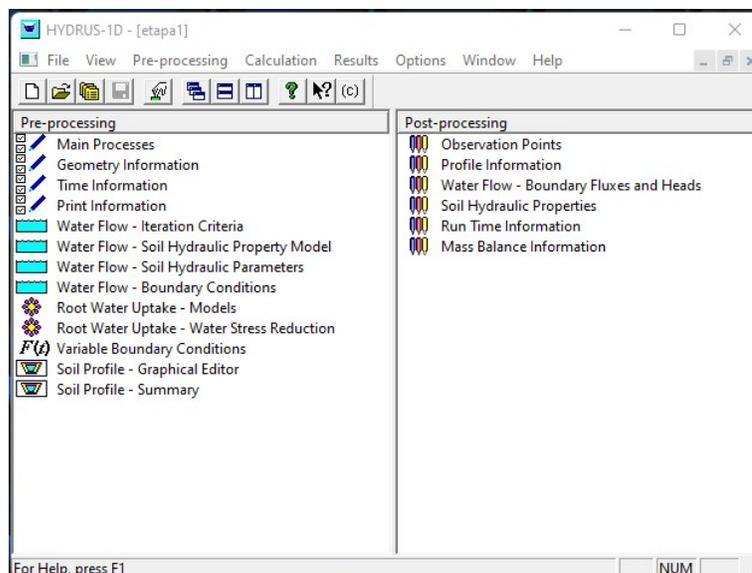


Figura 4.2: Interfaz de usuario inicial para HYDRUS-1D

## 4.1. Configuración de HYDRUS-1D

A continuación, se explican las ventanas de configuración de un proyecto, acotado a los procesos de movimiento de agua y absorción por las raíces en el medio, a través de la interfaz de usuario de HYDRUS-1D. Estas configuraciones constituyen el Menú de Pre-Procesamiento del proyecto, y definen el sistema a simular por el programa.

Cada proceso a simular puede estar caracterizado por diferente funciones, ecuaciones o modelos con una fuerte base teórica, la cual se explica de forma resumida en conjunto con su configuración.

### 4.1.1. Información básica

La utilización de HYDRUS-1D se basa en la definición de proyectos ajustados según las necesidades del medio que se desea simular, los aspectos generales del medio se definen en las ventanas: *Main Processes*, *Geometry Information*, *Time Information* y *Print Information*.

En la primera ventana, se determinan los procesos que se desean incluir en la simulación del proyecto, es decir, el movimiento de agua (*Water Flow*), el flujo de vapor (*Vapor Flow*), el transporte de solutos (*Solute Transport*), el transporte de calor (*Heat Transport*), la absorción de agua por las raíces (*Root Water Uptake*) y el crecimiento de las mismas (*Root Growth*).

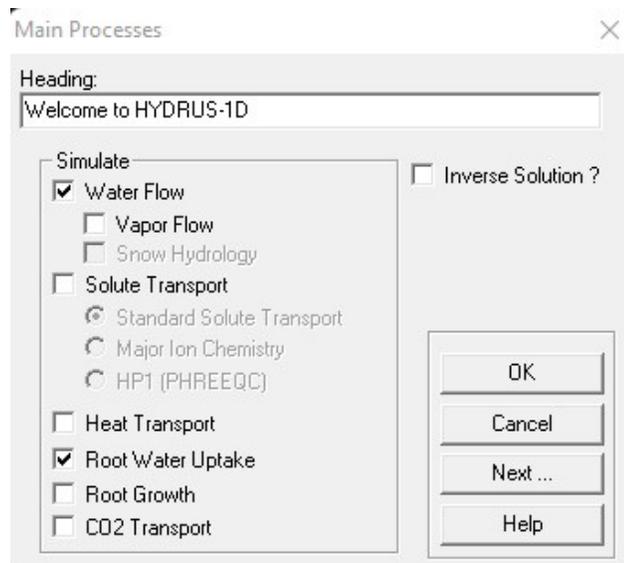


Figura 4.3: *Main Processes*: Selección de los procesos para el proyecto a simular.

En la ventana *Geometry Information* se definen parámetros referentes a las características básicas del perfil de suelo, principalmente la medida de longitud a utilizar en el proyecto, la profundidad del perfil y el número de tipos de suelo diferentes. También se puede configurar la orientación del perfil (vertical u horizontal) y determinar un número de subregiones en las que dividir el suelo para que el programa realice los cálculos.

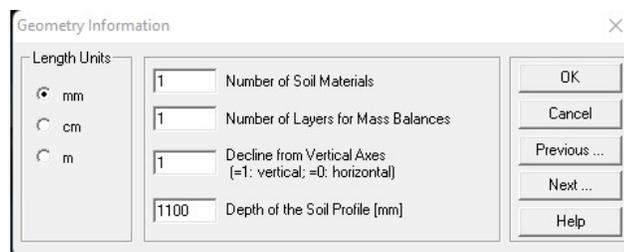


Figura 4.4: *Geometry Information*: Selección de características básicas del perfil de suelo.

Luego, se realiza la configuración temporal del proyecto que también define la unidad temporal para toda la simulación. Se configuran la discretización temporal asociada a los parámetros para la resolución numérica del problema.

Por otro lado, en esta pestaña también se determina la dependencia temporal de las condiciones atmosféricas a utilizar (Precipitaciones, Evaporación, Transpiración, entre otros), de igual manera se puede asignar la utilización de información meteorológica para determinar la evapotranspiración de cultivo. En HYDRUS los datos a utilizar para estas condiciones pueden ser asignados manualmente o generados diariamente por el programa.

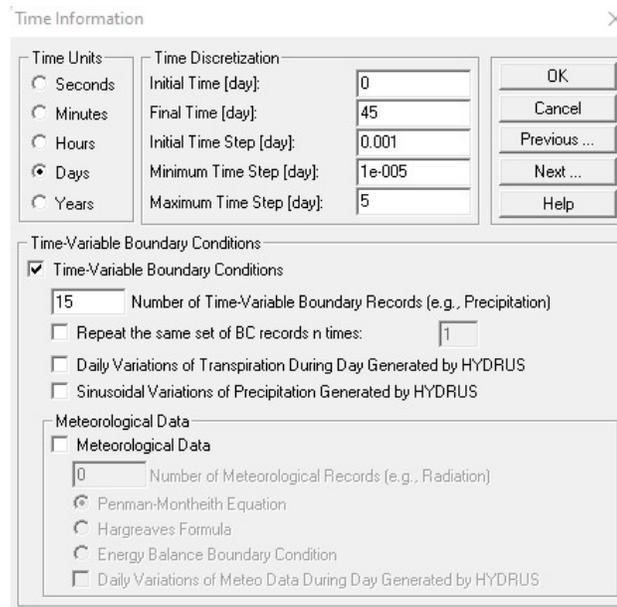


Figura 4.5: *Main Time Information*: Selección de características y condiciones atmosféricas temporales del proyecto.

La última configuración básica a realizar está asociada a la visualización del resultado de la simulación, mediante la selección de instantes de tiempo particulares en los cuales observar el estado del perfil y estudiar la evolución de este.

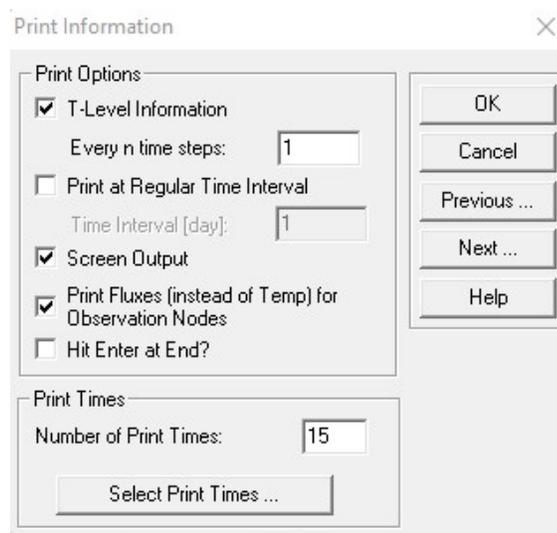


Figura 4.6: *Print Information*: Configuración de la visualización de resultados.

### 4.1.2. Flujo de agua (*Water Flow*)

El movimiento de agua en el perfil es el principal proceso simulado por HYDRUS, por lo cual su configuración es vital para un correcto desempeño de la simulación, partiendo con las características del proceso de resolución numérica (*Iteration Criteria*), la selección del modelo y ecuación que describe el flujo de agua (*Soil Hydraulic Property Model*), los parámetros del suelo asociados al modelo (*Soil Hydraulic Parameters*) y las condiciones de borde del perfil de suelo que caracterizan al proyecto (*Boundary Conditions*).

En primer lugar, se definen los parámetros que configuran el proceso iterativo de resolución de la ecuación no lineal de Richards, se ajusta el número máximo de iteraciones permitido y el cambio máximo en contenido de agua (*Water Content*) y cabezal de presión (*Pressure Head*) entre cada iteración, además de otros parámetros para ajustar el *solver*.

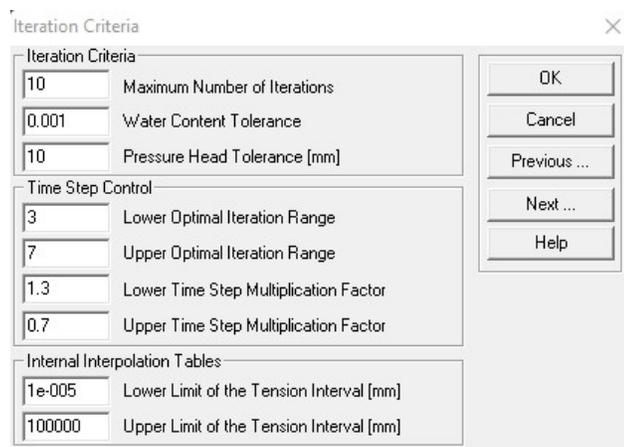


Figura 4.7: *Water Flow - Iteration Criteria*: Configuración del método iterativo de resolución de la ecuación de Richards.

El paso siguiente corresponde a la selección del modelo hidráulico a utilizar en la simulación del problema, HYDRUS permite seleccionar entre 6 tipos de modelo diferentes para la propiedades hidráulicas del suelo, estos modelos son:

1. van Genuchten - Mualem.
2. van Genuchten - Mualem con un valor de entrada de aire de -2 cm.
3. ecuaciones tipo van Genuchten modificadas.
4. ecuaciones de Brooks and Corey.
5. distribución log-normal de Kosuki.
6. doble porosidad de Durner.

La formulación de la mayoría de los modelos se basa en el de van Genuchten - Mualem, cuya formulación básica en función del cabezal de presión/tensión  $h$  (*Pressure Head*) es la

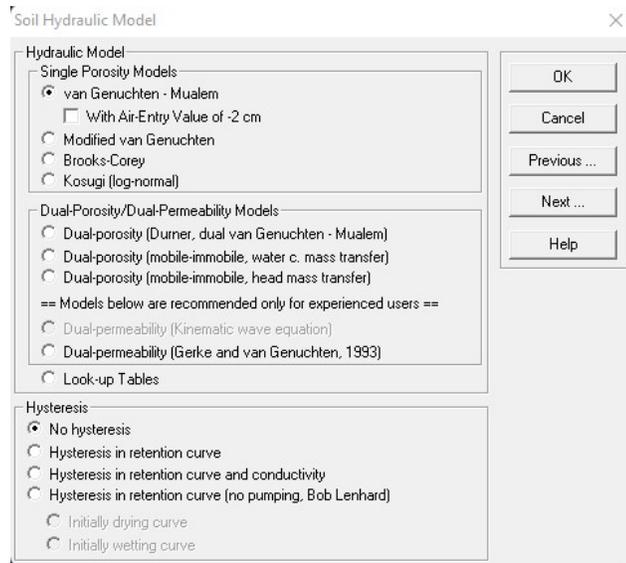


Figura 4.8: *Water Flow - Soil Hydraulic Property Model*: Selección del modelo de simulación para el flujo de agua.

siguiente:

$$\theta(h) = \begin{cases} \theta_r + \frac{\theta_s - \theta_r}{[1 + |\alpha h|^n]^{1-\frac{1}{n}}}, & h < 0 \\ \theta_s, & h \geq 0 \end{cases} \quad (4.1)$$

El cabezal de presión  $h$  es una medición en unidad de largo, que representa la fuerza de atracción cohesiva entre una partícula de suelo y el agua en un espacio poroso. Por otro lado, se tiene la formulación de la conductividad hidráulica en el perfil de suelo, dado por la expresión:

$$K(h) = K_s S_e^l [1 - (1 - S_e^{\frac{1}{n}})^{\frac{1}{n}}]^2 \quad (4.2)$$

Lo anterior, muestra la conversión del valor medido  $h$  (en unidad de largo  $L$ ) al contenido de agua retenido  $\theta$  (ecuación 4.1) y la conductividad hidráulica del suelo  $K$  (ecuación 4.2), en función de parámetros dados para el tipo de suelo:

- $\theta(h)$  : Contenido de agua retenido por el suelo  $\left[\frac{L^3}{L^3}\right]$ .
- $\theta_s$  : Contenido de agua en saturación  $\left[\frac{L^3}{L^3}\right]$ .
- $\theta_r$  : Contenido de agua residual  $\left[\frac{L^3}{L^3}\right]$ .
- $\alpha$  : Parámetros asociado al inverso de la entrada de aire  $\left[\frac{1}{L}\right]$ .
- $n$  : Medida de la distribución del tamaño de los poros  $[-]$ .
- $K_s$  : Conductividad hidráulica en saturación  $\left[\frac{L}{T}\right]$ .
- $K_r$  : Conductividad hidráulica relativa  $[-]$ .

- $S_e$  : Contenido de agua efectivo  $\left[\frac{L}{T}\right]$ .
- $l$  : Parámetro que representa los efectos de de tortuosidad y la conectividad de los poros  $[-]$ .

Estas expresiones definen el primer modelo presentado por HYDRUS, el resto de modelos corresponde en su mayoría a variantes de este, que buscan definir una dinámica más detallada del flujo de agua en el suelo.

Una vez definido el modelo a utilizar, es necesario sintonizar los parámetros de este, al cambiar el modelo cambia el número de parámetros que deben ser elegidos (para el modelo estándar:  $\theta_r$ ,  $\theta_s$ ,  $\alpha$ ,  $n$ ,  $K_s$  y  $l$ ). HYDRUS ofrece 3 alternativas para la elección de las variables: configuración manual por el usuario, valores predeterminados para un catálogo de suelos y una predicción mediante redes neuronales utilizando *Rosetta* (software que estima parámetros hidráulicos del suelo con funciones de pedotransferencia jerárquicas) [30].

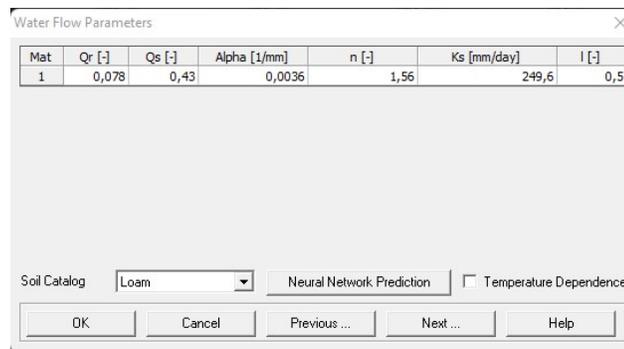


Figura 4.9: *Water Flow - Soil Hydraulic Parameters*: Configuración de parámetros del modelo seleccionado, según los tipos de suelo del perfil.

Finalmente, se deben seleccionar las condiciones de borde para la superficie, para el fondo del perfil y la variable en la que definir las condiciones iniciales (*Water Content* o *Pressure Head*).

Se tiene que para ambos extremos se pueden determinar condiciones de borde asociadas al flujo de agua o a la variable medida, ya sea mantener un valor constante o que este sea variable, sin embargo, existen otras condiciones específicas. En el caso de la superficie, se tiene:

- Condición de borde atmosférica con capa superficial (*Atmospheric boundary condition with surface layer*): Esta condición permite que se acumule agua en la superficie del terreno en función de las condiciones atmosféricas, esta capa de agua aumenta con las precipitaciones y disminuye con la filtración y la evaporación.
- Condición de borde atmosférica con escurrimiento superficial (*Atmospheric boundary condition with surface run off*): Define un flujo de agua en la superficie en función de condiciones de humedad del terreno, la presencia o no de estancamiento de agua y los valores de infiltración y evapotranspiración del suelo.

En el otro extremo del perfil, las condiciones de borde particulares son:

- Drenaje Libre (*Free Drainage*): Se refiere al flujo de agua como drenaje libre vertical, usualmente se utiliza en casos donde el nivel freático se encuentra muy por debajo de la zona de interés estudiada.
- Drenaje Profundo (*Deep Drainage*): Aproximación del drenaje vertical como un flujo de agua dependiente de la posición del nivel freático.
- Cara de infiltración (*Seepage face*): Condición para casos donde el perfil de suelo termina en este borde, lo que implica que mientras no exista saturación del suelo no existe flujo de agua de salida, pero al alcanzar cierto nivel  $h$  se determina un flujo.
- Drenajes Horizontales (*Horizontal Drains*): Flujo de agua en forma de drenaje horizontal, es decir, hacia los contornos del perfil.

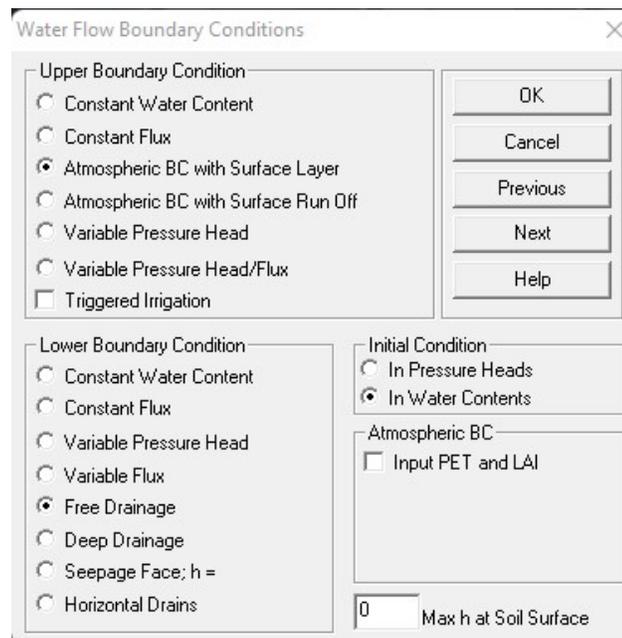


Figura 4.10: *Water Flow - Boundary Conditions*: Configuración de las condiciones de borde en los extremos del perfil de suelo.

Con la selección de las condiciones de borde, la configuración del flujo de agua dentro del perfil de suelo queda definida, a estas configuraciones se les añaden las asociadas a procesos más específicos de cada proyecto, como lo es la absorción de agua en la zona de las raíces.

### 4.1.3. Absorción de agua en las raíces (*Root Water Uptake*)

Los efectos en el suelo producto de la presencia de un cultivo, se reflejan en la simulación al agregar la absorción de agua en las raíces, y de manera opcional la absorción de solutos. Estos procesos responden a curvas de estrés, las cuales son determinadas según los modelos seleccionados, centrando la atención únicamente en el movimiento de agua, se tiene el modelo de Feddes y una función de van Genuchten con forma de S.

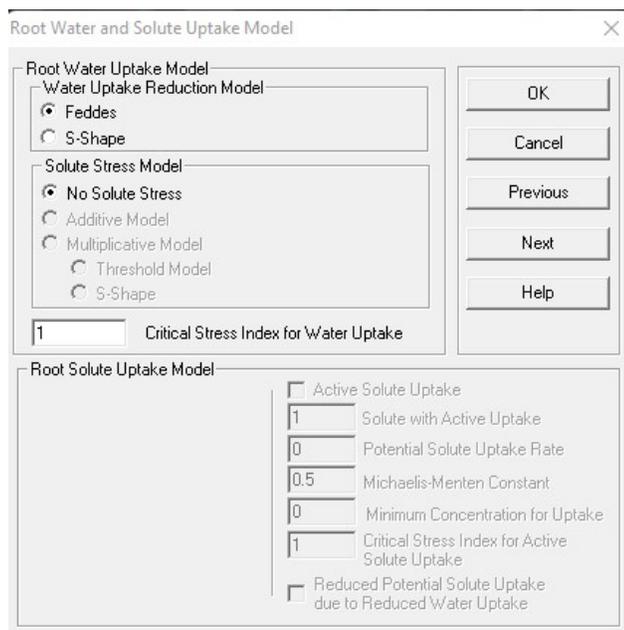


Figura 4.11: *Root Water Uptake - Models*: Selección del modelo de absorción de agua y solutos por las raíces,

El modelo Feddes se configura mediante la selección de valores de cabezal de presión que definen umbrales en el perfil de suelo, la capacidad de extracción de agua de las raíces depende de la posición del perfil con respecto a estos umbrales, los parámetros a definir son:

- **P0**: Valor de presión máxima debajo la cual las raíces comienzan a extraer agua del suelo.
- **POpt**: Valor de presión por debajo del cual las raíces extraen agua a la velocidad máxima posible.
- **P2H**: Valor limitador de presión, debajo del cual las raíces ya no pueden extraer agua a la velocidad máxima.
- **P2L**: Como P2H, pero para una tasa de transpiración potencial de r2L.
- **P3**: Valor por debajo del cual cesa el consumo de agua, se puede tomar como el punto de marchitez permanente.
- **r2H**: Tasa de transpiración potencial en  $\left[\frac{L}{T}\right]$ , según sean las unidades seleccionadas para el modelo.

- **r2L**: Tasa de transpiración potencial en  $[\frac{L}{T}]$ , según sean las unidades seleccionadas para el modelo.

Se puede definir un punto P2 similar a un punto de estrés hídrico, a partir de la definición de una tasa de transpiración potencial  $T_p$  y en función de las tasas de transpiración r2H y r2L definidas por el modelo, de la forma:

$$P2 = P2H + \frac{P2L - P2H}{r2H - r2L}(r2H - T_p), \quad r2L < T_p < r2H, \quad (4.3)$$

$$P2 = P2L, \quad T_p < r2L \quad (4.4)$$

$$P2 = P2H, \quad T_p > r2H \quad (4.5)$$

Los parámetros anteriormente definidos pueden ser sintonizados por el usuario o pueden ser obtenidos por medio de una base de datos de plantas y cultivos que HYDRUS-1D posee.

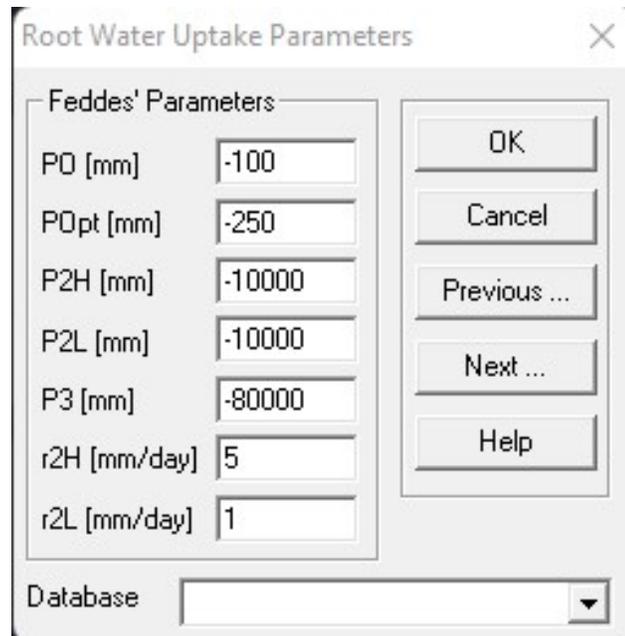


Figura 4.12: *Root Water Uptake - Water Stress Reduction*: Configuración de los parámetros del modelo Feddes.

Por otro lado, para la función de van Genuchten en forma de S, se deben sintonizar dos parámetros solamente:

- **P0**: El exponente,  $p$ , en la función de respuesta de absorción de agua de la raíz asociado al con el estrés hídrico.
- **P50**: El coeficiente,  $h_{50}$ , en la función de respuesta de absorción de agua de la raíz asociado con el estrés hídrico. La absorción de agua de la raíz se reduce en un 50 % a dicho valor de presión.

Una vez definidos los parámetros del modelo de absorción de agua de la raíz, el proceso queda configurado en la simulación.

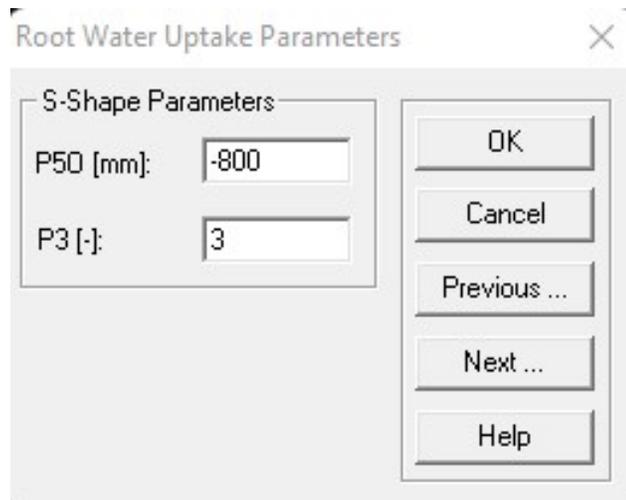


Figura 4.13: *Root Water Uptake - Water Stress Reduction*: Configuración de los parámetros del modelo *S-Shape function* de van Genuchten

#### 4.1.4. Condiciones atmosféricas (*Variable Boundary Conditions*)

Una vez se definen los procesos que abarcan la formulación del caso de estudio, se establecen las condiciones atmosféricas a las que se somete el sistema a lo largo de la simulación, en función a lo definido en la información temporal (ver figura 4.4).

Las variables a definir en esta sección dependen de los procesos, modelos y características definidas en las configuraciones pasadas, sin embargo, en general siempre se cuenta con los parámetros que se aprecian en la figura 4.14, cuya explicación se presenta a continuación:

- **Time:** Tiempo en el cual la información es provista, según la unidad de tiempo elegida  $[T]$ . Este valor indica hasta que instante se aplican los datos provistos en su fila, es decir, marca el fin de un intervalo de tiempo que comienza en el tiempo inicial o en el de la fila anterior.
- **Precip:** Tasa de precipitaciones en el intervalo de tiempo definido, se mide en  $[\frac{L}{T}]$ .
- **Evap:** Tasa de evaporación potencial en  $[\frac{L}{T}]$ .
- **Trans:** Tasa de transpiración potencial en  $[\frac{L}{T}]$ .
- **hCritA:** Mínimo valor de presión permitido en la superficie del perfil en  $[L]$ .

HYDRUS-1D requiere que el usuario entregue los valores potenciales de evaporación y transpiración, debido a que no posee la capacidad de obtener estos valores a partir de la evapotranspiración de cultivo, información que es en general es más fácil de obtener. Sin embargo, existe la opción de generar los valores de transpiración potencial diarios con HYDRUS, lo cual se realiza en la pestaña de *Main Time Information* de la Figura 4.5.

|    | Time<br>[day] | Precip.<br>[mm/day] | Evap.<br>[mm/day] | hCritA<br>[mm] | Transp.<br>[mm/day] |
|----|---------------|---------------------|-------------------|----------------|---------------------|
| 1  | 3             | 0                   | 0,357             | 1000000        | 3,213               |
| 2  | 6             | 0                   | 0,416             | 1000000        | 3,744               |
| 3  | 9             | 0                   | 0,477             | 1000000        | 4,29                |
| 4  | 12            | 0                   | 0,515             | 1000000        | 4,638               |
| 5  | 15            | 0                   | 0,321             | 1000000        | 2,892               |
| 6  | 18            | 0                   | 0,253             | 1000000        | 2,277               |
| 7  | 21            | 0                   | 0,469             | 1000000        | 4,221               |
| 8  | 24            | 0                   | 0,422             | 1000000        | 3,798               |
| 9  | 27            | 0                   | 0,332             | 1000000        | 2,988               |
| 10 | 30            | 0                   | 0,494             | 1000000        | 4,446               |
| 11 | 33            | 0                   | 0,363             | 1000000        | 3,264               |
| 12 | 36            | 0                   | 0,466             | 1000000        | 4,197               |
| 13 | 39            | 0                   | 0,472             | 1000000        | 4,251               |
| 14 | 42            | 0                   | 0,408             | 1000000        | 3,675               |

Buttons: OK, Cancel, Previous ..., Next ..., Help ..., Add Line, Delete Line, Default Time

Figura 4.14: *Variable Boundary Conditions*: Condiciones atmosféricas definidas para el tiempo de duración de la simulación.

#### 4.1.5. Perfil de suelo inicial (*Soil Profile*)

Una vez configurados los procesos del sistema y entregadas las condiciones atmosféricas para el desarrollo de la simulación, el último paso es la configuración del perfil de suelo en términos de condiciones iniciales, distribución de las raíces y la colocación de los puntos de observación.

Para realizar la configuración del perfil de suelo, HYDRUS-1D posee un editor gráfico con una nueva interfaz de usuario, donde se configura el número total de nodos para la discretización del suelo. A cada nodo del perfil se le asigna un tipo de suelo, una condición inicial, una densidad de raíces, entre otros parámetros.

Las condiciones iniciales se definen para cada nodo en función de la variable que se haya definido, por ejemplo, se define el contenido de agua o el cabezal de presión en todo el perfil. Otras condiciones iniciales son temperatura o la concentración de solutos, necesarias para los procesos de transporte de calor y solutos.

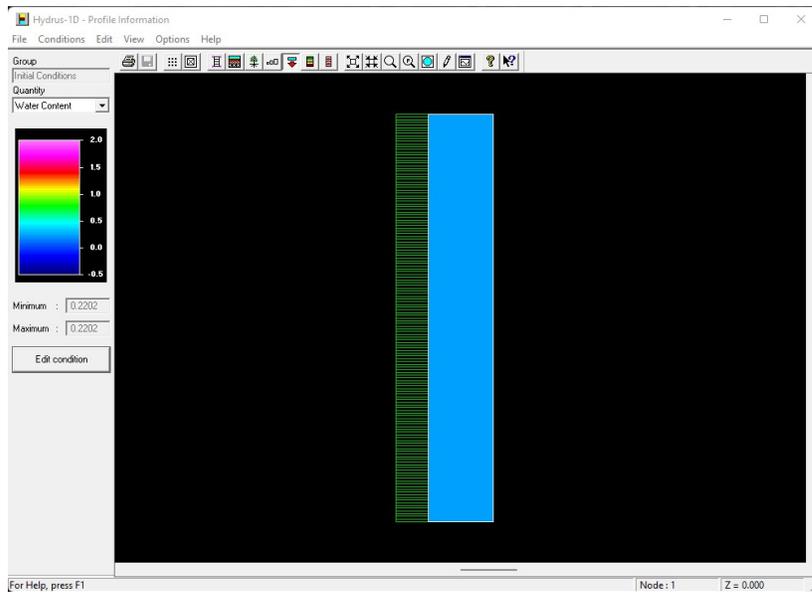


Figura 4.15: *Graphical Editor - Initial Conditions*: Definición de las condiciones iniciales en términos de contenido de agua o cabezal de presión para el perfil de suelo.

Otra configuración necesaria es la distribución de la densidad de raíces en la columna, esto se debe configurar en función de la especie de estudio, donde cada nodo toma un valor entre 0 y 1, donde el 0 corresponde a la completa ausencia de raíces, mientras que el 1 corresponde a una zona de la raíz densa.

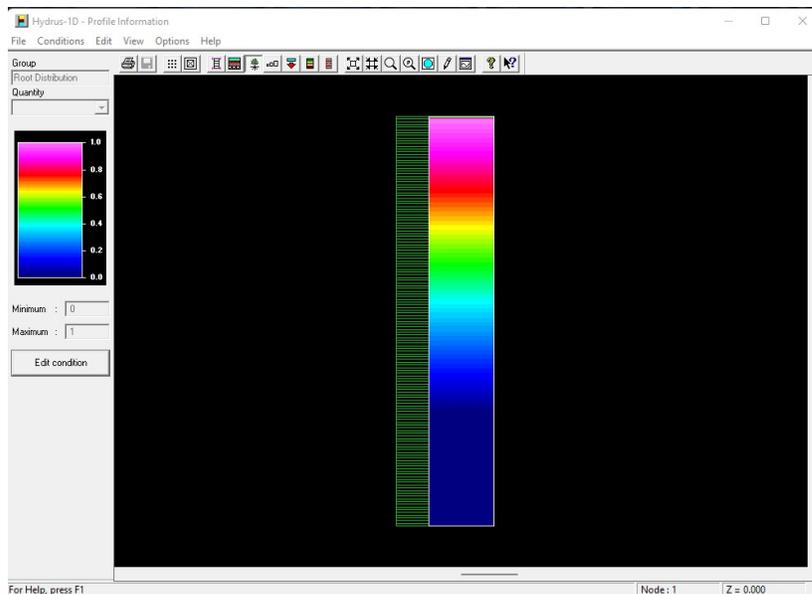


Figura 4.16: *Graphical Editor - Root Distribution*: Definición de la distribución y densidad de las raíces en el perfil de suelo.

Finalmente, cualquier nodo puede ser seleccionado como un punto de observación desde el cual se puede acceder a los cambios de contenido de agua, cabezal de presión y flujo de agua a lo largo del periodo simulado.

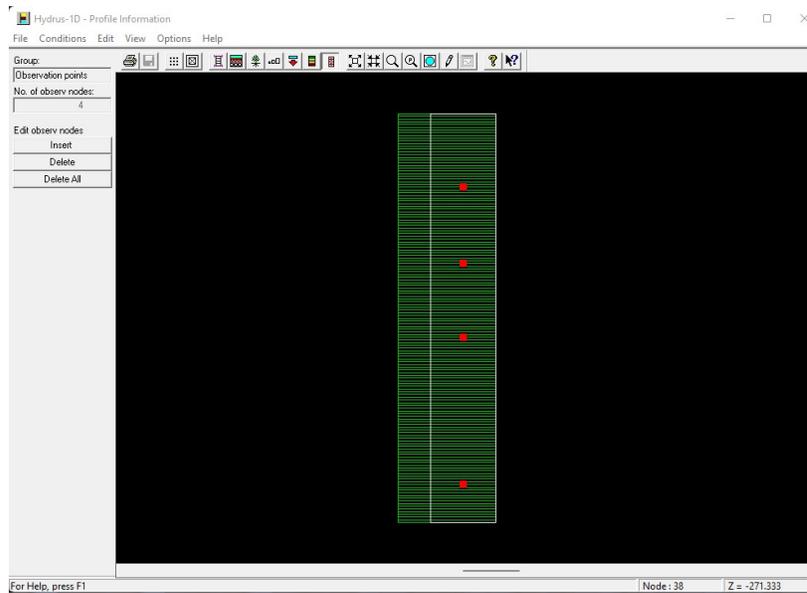


Figura 4.17: *Graphical Editor - Observation Points*: Selección de puntos de observación para la visualización de la dinámica del suelo en esos puntos.

Además de la representación gráfica del perfil, HYDRUS entrega un resumen de las características del perfil que se pueden configurar a modo de tabla, donde se muestran la información de cada nodo: profundidad, condición inicial, distribución de la raíz, entre otros valores. Este resumen también puede ser modificado, pudiendo editar cada dato manualmente y también se puede establecer la condición inicial como capacidad de campo, valor que dependerá de las características de la columna de suelo.

|    | z [mm]  | theta    | Root [1/mm] | Axz | Bxz | Dxz | Mat |
|----|---------|----------|-------------|-----|-----|-----|-----|
| 1  | 0       | 0.220196 | 0           | 1   | 1   | 1   | 1   |
| 2  | 7.33333 | 0.220196 | 1           | 1   | 1   | 1   | 1   |
| 3  | 14.6667 | 0.220196 | 0.990741    | 1   | 1   | 1   | 1   |
| 4  | 22      | 0.220196 | 0.981481    | 1   | 1   | 1   | 1   |
| 5  | 29.3333 | 0.220196 | 0.972222    | 1   | 1   | 1   | 1   |
| 6  | 36.6667 | 0.220196 | 0.962963    | 1   | 1   | 1   | 1   |
| 7  | 44      | 0.220196 | 0.953704    | 1   | 1   | 1   | 1   |
| 8  | 51.3333 | 0.220196 | 0.944444    | 1   | 1   | 1   | 1   |
| 9  | 58.6667 | 0.220196 | 0.935185    | 1   | 1   | 1   | 1   |
| 10 | 66      | 0.220196 | 0.925926    | 1   | 1   | 1   | 1   |
| 11 | 73.3333 | 0.220196 | 0.916667    | 1   | 1   | 1   | 1   |
| 12 | 80.6667 | 0.220196 | 0.907407    | 1   | 1   | 1   | 1   |
| 13 | 88      | 0.220196 | 0.898148    | 1   | 1   | 1   | 1   |
| 14 | 95.3333 | 0.220196 | 0.888889    | 1   | 1   | 1   | 1   |
| 15 | 102.667 | 0.220196 | 0.879629    | 1   | 1   | 1   | 1   |

Figura 4.18: *Soil Profile - Summary*: Condiciones iniciales en términos de contenido de agua o cabezal de presión para el perfil de suelo

Luego, de que se aplicaron todas estas configuraciones se ejecuta la simulación, en la cual se resuelven numéricamente las ecuaciones que definen al sistema. La visualización de los resultados depende de los instantes de tiempo elegidos en la figura 4.6 y en los puntos de observación elegidos.

# Capítulo 5

## Aprendizaje Reforzado (AR)

En aprendizaje reforzado (*Reinforcement Learning*) los sistemas se descomponen en dos elementos, un entorno y un agente. El agente interactúa con el entorno mediante la realización de acciones específicas, a partir de las cuales recibe un *feedback*, en la forma de una actualización del estado y una recompensa, esto induce un proceso de aprendizaje en el agente, con el cual se busca mejorar las acciones para obtener mejores recompensas por parte del entorno [11].

### 5.1. Conceptos básicos

El proceso de aprendizaje a través del *feedback* recibido, genera un lazo entre el agente y el entorno, donde el desarrollo del agente es guiado y determinado por los algoritmos de AR utilizados. Esta técnica esta basada en el interacción de los individuos con su entorno, generando cambios en el individuo a partir de las experiencias adquiridas en sus interacciones, estos cambios apunta a obtener mejores respuestas por parte del entorno, esto constituye el aprendizaje del agente.

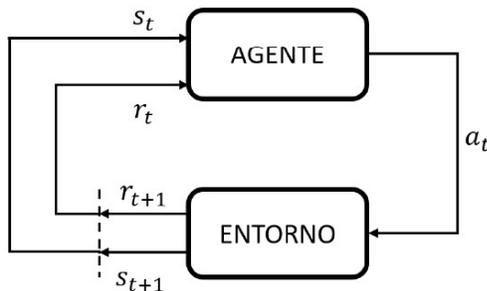


Figura 5.1: Diagrama básico de un algoritmo de aprendizaje reforzado.

La Figura 5.1 representa un esquema básico del funcionamiento del aprendizaje reforzado, donde se muestran las componentes básicas de los algoritmos: agente, entorno, estado, acción y recompensa. Para cada instante  $t$ , el agente observa el estado actual del entorno,  $S_t$  (puede

ser parcial o completamente observable, aunque es difícil hacer la distinción, por lo que se suele considerar completitud en la información), y el valor de recompensa correspondiente,  $r_t$ , dada por la función de recompensas que puede depender del estado actual o de una trayectoria.

Luego, basado en la información obtenida el agente decide según la política de decisión  $\pi$ , una acción,  $a_t$ , para ejecutar en ese instante, lo que genera un nuevo estado,  $s_{t+1}$ , y una nueva recompensa,  $r_{t+1}$ , que corresponden a la información para el agente en el instante siguiente  $t = t + 1$ . Cabe destacar que el nuevo estado puede ser obtenido a través de procesos estocásticos y deterministas.

En el diseño de una metodología por aprendizaje reforzado, es importante destacar los conceptos de exploración y explotación, que determinan el enfoque del entrenamiento del agente, a través de cambios en la política de decisión.

La exploración se refiere a probar múltiples acciones, de modo de aumentar el conocimiento sobre las recompensas disponibles, para encontrar aquellas que tengan un mejor rendimiento. La explotación refuerza la realización de acciones que maximizan el rendimiento, en base al conocimiento actual del sistema. Encontrar un equilibrio óptimo entre ambos enfoques es parte fundamental del proceso de diseño de una método de aprendizaje reforzado.

### 5.1.1. Proceso de decisión de Markov

La dinámica descrita previamente para la interacción entre el agente y su entorno, se define matemáticamente a través de un proceso de decisión de Markov (MDP, del inglés *Markov Decision Process*), que corresponde a una simulación estocástica discreta de un sistema continuo.

La formulación de un MDP corresponde a una mejora del proceso de Markov (MP), que se define a partir, de un conjunto de estados  $\mathcal{S}$  y una matriz de transición de estados  $P$  que define la probabilidad de a un determinado estado dado el estado anterior ( $p(S_t|S_{t-1})$ , con  $S_t \in \mathcal{S}$ ), esto se puede apreciar en la Figura 5.2.

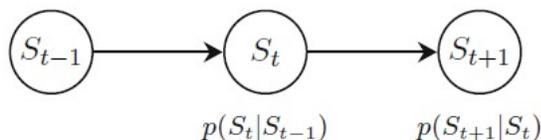


Figura 5.2: Modelo gráfico de un proceso de Markov [11]

Luego, se complementa con la utilización de un proceso de recompensa de Markov (MRP, *Markov Reward Process*), para el cual se utiliza una función de recompensa  $R$  y un factor de descuento de la recompensa  $\gamma$ . Luego, cada estado posee una recompensa asociada, es decir,  $R_t = R(s_t)$ , la representación gráfica de esto se puede apreciar en la Figura 5.3.

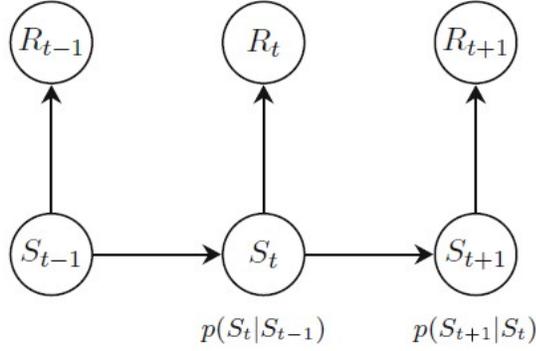


Figura 5.3: Modelo gráfico de un proceso de recompensa de Markov [11]

La recompensa final del proceso, está determinada por aquellas obtenidas en cada instante de tiempo, hasta la etapa terminal  $T$  al seguir una trayectoria de estados  $\tau$ , además de considerar el factor de descuento, que entrega más valor a las primeras recompensas obtenidas, la formulación se tiene en (5.1).

$$R(\tau) = \sum_{t=0}^T \gamma^t R_t. \quad (5.1)$$

Finalmente, con MDP la dependencia de la recompensa obtenida deja de ser únicamente del estado en el instante  $t$ , pasando a considerarse una acción,  $A_t$ , realizada para transicionar de un estado a otro. De esta manera, la formulación pasa a considerar el conjunto de acciones posibles  $\mathfrak{A}$ , definiéndose la nueva función de recompensa como  $R_t = R(S_t, A_t)$ . El modelo gráfico de un MDP se aprecia en la Figura 5.4.

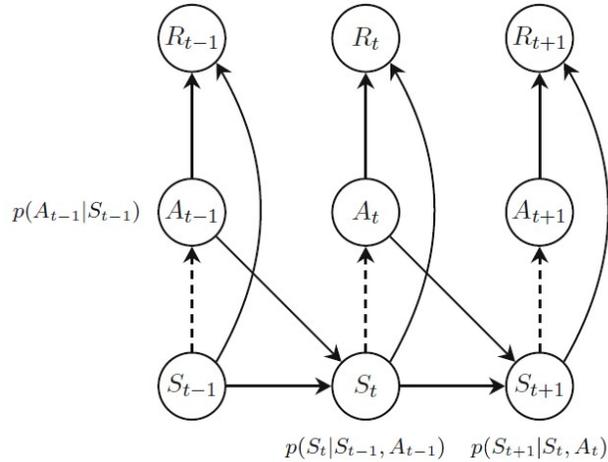


Figura 5.4: Modelo gráfico de un proceso de decisión de Markov [11]

La toma de decisiones del agente que cambia de estados, se define según la política  $\pi$ , basada en las observaciones de los estados del entorno. La política matemáticamente puede definirse como la probabilidad de realizar un acción  $a$  dado un estado  $S$  del sistema, es decir:

$$\pi(A|S) = p(A_t = a|S_t = s) \quad (5.2)$$

De esta manera, se define el retorno esperado como la ponderación de los retornos esperados para todas las posibles trayectorias dadas por una política  $\pi$ . Entonces, el objetivo de aprendizaje reforzado es maximizar el retorno esperado mediante la optimización de la política. Dada una función de recompensa  $R$  y todas las posibles trayectorias  $\tau$ , el retorno esperado  $J(\pi)$  se define como:

$$J(\pi) = \int_{\tau} p(\tau|\pi)R(\tau) = \mathbb{E}_{\tau \sim \pi}[R(\tau)], \quad (5.3)$$

donde la función de retorno definida en (5.1) se reescribe como:

$$R(\tau) = \sum_{t=0}^T \gamma^t R(S_t, A_t). \quad (5.4)$$

Con estos antecedentes, se define el problema de optimización de aprendizaje reforzado, que busca encontrar la política óptima  $\pi^*$  que maximice el retorno esperado, es decir:

$$\pi^* = \arg \max_{\pi} J(\pi) \quad (5.5)$$

### 5.1.2. Ecuación de Bellman y funciones de valor

A partir de la formulación para el retorno esperado, de la Ecuación (5.4), a partir de una política de decisión  $\pi$  dada se definen las funciones:  $q_{\pi}(S_t, A_t)$  valor de tomar una acción  $A$  en el estado  $S$  bajo la política  $\pi$ ,  $v_{\pi}(S_t)$  valor del estado  $S$  bajo la política  $\pi$  y  $a_{\pi}(S_t, A_t)$  beneficios para la acción  $A$  y el estado  $S$ , las cuales se muestran a continuación:

$$q_{\pi}(S_t, A_t) = \mathbb{E}_{\pi}[R_t|S_t, A_t], \quad (5.6)$$

$$v_{\pi}(S_t) = \mathbb{E}_{\pi}[R_t|S_t] \quad (5.7)$$

$$a_{\pi}(S_t, A_t) = q_{\pi}(S_t, A_t) - v_{\pi}(S_t) \quad (5.8)$$

Esta definición de las funciones valor, muestra los beneficios de ejecutar una acción  $A_t$  para un estado  $S_t$ , en el caso de la función de valor para la acción, y los beneficios de encontrarse en el estado  $S_t$  en la función valor del estado. Estas funciones son consideradas en la formulación de algoritmos de aprendizaje reforzada, según sea el enfoque de dicho algoritmo.

De esta manera, las funciones de valor optimizadas según la política de decisión ( $\pi^*$ ),  $q_{\pi^*}(S_t, A_t)$  y  $v_{\pi^*}(S_t)$ , permiten establecer las siguientes expresiones para las ecuaciones de Bellman:

$$v_{\pi^*}(S_t) = \max_a \mathbb{E}_{\pi^*}[R(S_t, A_t) + \gamma v_{\pi^*}(S_{t+1})], \quad (5.9)$$

$$q_{\pi^*}(S_t, A_t) = \mathbb{E}_{\pi^*}[R(S_t, A_t) + \gamma q_{\pi^*}(S_{t+1}, A_{t+1})] \quad (5.10)$$

Estas ecuaciones se basan en un criterio de optimalidad sobre la política de decisión  $\pi$ , lo que permite calcularlas de forma recursiva durante el entrenamiento para estimar el criterio de decisión óptimo, en lo que se conoce como *Policy Iteration*.

## 5.2. Clasificación de algoritmos

En Aprendizaje Reforzado los algoritmos son clasificados y organizados según diferentes perspectivas, incluyendo metodologías *model-based* y *model-free*, métodos *value-based* y *policy-based* (o una combinación de ambos), métodos de Monte Carlo y Diferencia Temporal, métodos *On-Policy* y *Off-Policy*, estos criterios constituyen la base del orden de los algoritmos [11].

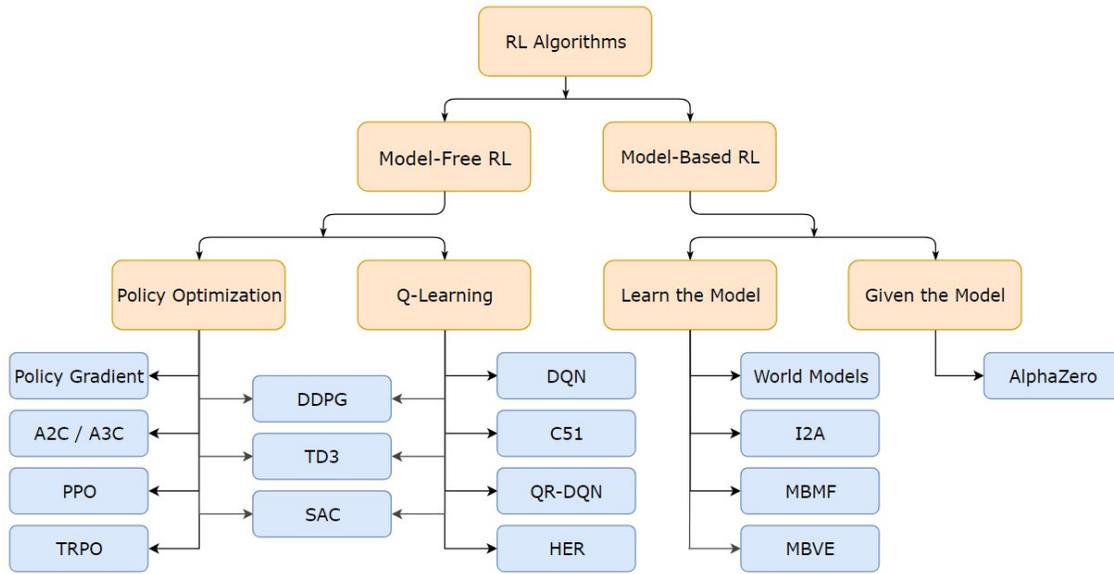


Figura 5.5: Estructura de organización para los algoritmos de aprendizaje reforzado [26].

### 5.2.1. *Model Free*

Se refiere a la rama de algoritmos que no requieren de un modelo entrenado del entorno o que no aprenden/construyen uno para interactuar con él, es decir, son aquellos donde el agente realiza acciones de forma directa con el entorno y mejora su desempeño en función de los estados y recompensas explorados.

Esto permite que su implementación sea directa y no dependa de la complejidad del entorno, sin embargo, su aplicación se restringe en casos donde el entorno es una situación real, debido a lo costoso de la exploración con equipos reales, pudiendo tener riesgos asociados a las respuestas, además que la obtención de suficientes muestras puede llevar demasiado tiempo.

### 5.2.2. Value Based - Policy Based

Identificados principalmente por los algoritmos de tipo *Q-Learning*, los métodos *value-based* se basan en la optimización de la función acción-valor  $q_\pi(s, a)$ . Donde el valor óptimo de la función después de la optimización es  $q_{\pi^*}(s, a) = \max_a q_\pi(s, a)$ , por lo tanto, la política óptima se deriva como  $\pi^* \approx \operatorname{argmax}_\pi q_\pi$ .

Las ventajas de los métodos basados en la función valor se centran en su eficiencia de muestreo, una baja varianza de la función valor y que dada su formulación es difícil caer en óptimos locales. Los mayores problemas que poseen estos métodos es su incapacidad de manejar espacios de acciones continuos y su facilidad de caer en sobre-estimaciones productos de las estrategias *greedy*.

Por otro lado, los métodos *policy-based* o realizan una optimización directamente sobre la política  $\pi_\theta(a|s)$ , donde básicamente se itera la política hasta maximizar el retorno acumulado 5.3, con lo que se logra la optimización de la parametrización  $\theta$  de la política.

Las principales ventajas de los métodos de (*Policy Optimization*) son la parametrización simple de la política, una mejor convergencia al óptimo y principalmente el poder su utilizada en espacios de acción continuos o de alta dimensionalidad.

La actualización de la política se realiza con el método del ascenso del gradiente, dado por:

$$\nabla_\theta J(\pi_\theta) = \mathbb{E}_{\pi_\theta}[\nabla_\theta \log(\pi_\theta(a|s))q_\pi(s, a)], \quad (5.11)$$

donde  $J_{\pi_\theta}$  es la función de desempeño a maximizar, dada por el retorno esperado. De esta forma, se puede obtener la actualización de la parametrización  $\theta$  de la política como:

$$\theta = \theta + \alpha \nabla_\theta \hat{J}(\pi_\theta), \quad (5.12)$$

donde  $\alpha$  es la tasa de aprendizaje y  $\hat{J}(\pi_\theta)$  corresponde a la función de desempeño estimada.

#### Actor-crítico

Estos métodos surgen como la combinación de los métodos *value-based* y *policy-based*, se utiliza *Q-Learning* para mejorar la eficiencia del muestreo y *Policy Optimization* para aprender la función de la política, permitiendo su utilización en espacios discretos y continuos.

Se denomina actor a una parametrización  $\theta$  de la política  $\pi$ , mientras que el crítico corresponde a una parametrización  $\omega$  de la función de la función de valor  $v_\pi$  utilizada en la función de desempeño  $J$ .

En esta estructura, el actor interactúa con el entorno siguiendo la política  $\pi_\theta$ , generando tuplas de la forma  $(S_t, A_t, r_t, S_{t+1}, A_{t+1})$ , mientras que el crítico estima la calidad del par estado-acción de acuerdo a los parámetros  $\omega$ . Luego, de acuerdo a la estimaciones realizadas por el crítico, se realiza la actualización de los parámetros  $\theta$  por ascenso del gradiente (Ecuación 5.11), por otro lado, los parámetros  $\omega$  mediante el descenso del gradiente del promedio

cuadrático (para aproximar el valor esperado) de la diferencia temporal  $\delta_t$ , derivada de la ecuación de Bellman (Ecuación 5.9):

$$\delta_t = r_t + \gamma v_\omega(S_{t+1}, A_{t+1}) - v_\omega(S_t, A_t), \quad (5.13)$$

con  $v_\omega(S, A)$  la función de valor correspondiente.

### 5.2.3. Monte Carlo y *Temporal Difference*

Las metodologías basadas en Monte Carlo (MC) y Diferencia Temporal (TD) buscan estimar la función valor  $v_\pi$ , sin necesidad de tener completo conocimiento del entorno y utilizando la experiencia obtenida en cada episodio para su aprendizaje, sin embargo estos métodos se diferencian en su aprendizaje.

En primer lugar MC, utiliza el retorno obtenido por una política  $\pi$  (Ecuación 5.3) según la secuencia de estados y acciones tomadas hasta alcanzar el estado terminal, de esta forma se evalúa la política en base al desempeño en múltiples episodios, definiendo la función valor como:

$$v_\pi(s) = \mathbb{E}_\pi[R_t(\tau)|S_t = s] \quad (5.14)$$

Por otro lado, TD también utiliza las secuencias de estados y acciones tomadas por la política  $\pi$ , sin embargo, no es necesario que los episodios alcancen un estado terminal, ya que utilizan las muestras de episodios incompletos para inferir el comportamiento de uno completo, utilizando *bootstrapping*. TD realiza el aprendizaje de la función valor aprovechando la diferencia entre un valor objetivo y el valor estimado para diferentes instantes, donde el valor objetivo se define en base al retorno observado en ese instante y una estimación del valor para el instante siguiente, generando una actualización de la función valor de la forma:

$$V(S_t) = V_{S_t} + \alpha[R_{t+1} + \gamma V(S_{t+1}) - V(S_t)] \quad (5.15)$$

De esta forma, se aprecia que la principal diferencia recae en que MC necesita la información de un episodio completo, mientras que TD puede realizar actualizaciones y aprender en cada una de las etapas del episodio. Esto genera que TD tienda a tener mayores *bias*, mientras que en MC aumenta la varianza.

### 5.2.4. *On-policy* y *Off-policy*

*On-policy* se basa en que durante el entrenamiento, el agente utilice la misma política de decisión al interactuar con el entorno (observar  $s_t$  y  $r_t$ , ejecutar  $a_t$  y recibir la actualización  $s_{t+1}$  y  $r_{t+1}$ ) y para realizar la actualización de política (cálculo de las funciones de valor, ecuación de Bellman y *Policy Iteration*).

Por otro lado, *Off-policy*, posee políticas de decisión separadas para cada acción, la usada para interacción con el entorno se llama conductual y la utilizada para la actualización es la objetivo, esta última es la que se aplica posterior al entrenamiento.

## 5.3. Algoritmos

A continuación, se presentan algoritmos *model-free* de las diferentes categorías definidas en las secciones anteriores, se entrega un pequeño resumen de sus características, consideraciones de implementación y un pseudo-código de su implementación.

### 5.3.1. *Deep Q-Network* DQN

El algoritmo *Deep Q-Network* (DQN) es una metodología *value-based* basada en *Temporal Difference* y *Off-Policy*. Las características de DQN limitan su uso a entornos de acciones discretas, donde se busca la actualización de la función  $q_\pi$  dada por la regla definida en *Q-Learning*:

$$Q(S_t, A_t) = Q(S_t, A_t) + \alpha [R_t + \gamma \max_{A_{t+1}} Q(S_{t+1}, A_{t+1}) - Q(S_t, A_t)] \quad (5.16)$$

Esta regla de actualización muestra el carácter *Off Policy* de los algoritmos basados en *Q-Learning*, ya que la acción se obtiene de la maximización de la función  $Q$  con la política *greedy*, diferente de la política que se actualiza.

DQN constituye una mejora de otras técnicas de *Q-Learning* pues agrega un mecanismo conocido como *replay buffer*, el cual en cada instante  $t$ , almacena la experiencia del agente  $(S_t, A_t, R_t, S_{t+1})$ , y luego extrae un mini-batch de muestras del *buffer* uniformemente para aplicar en la actualización *Q-Learning* [11].

Esto tiene ciertas implicaciones, primero, la experiencia de cada instante puede ser reutilizada para aprender la función  $Q$ , si este mecanismo no existe las muestras se obtienen de manera consecutiva, implicando una alta correlación, lo que aumenta la varianza de las actualizaciones. Finalmente, la experiencia reutilizada asegura que las muestras de entrenamiento siempre posean parámetros antiguos, reduciendo las oscilaciones y mejorando las curvas de entrenamiento.

Por otro lado, DQN propone la utilización de una red neuronal diferente para generar los valores objetivo de la función  $Q$ , denominada *target network*. Además, cada  $C$  pasos, esta red estará sincronizada con la red  $Q$  principal. Seleccionando muestras del mini-batch se calcula el objetivo  $y$  de la forma:

$$y = r + \gamma \max_{\hat{a}} \hat{Q}(\phi', a'; \hat{\theta}), \quad (5.17)$$

con  $\phi'$  la función que representa la información obtenida de la secuencia pasos realizados,  $\hat{a}$  la acción y  $\hat{\theta}$  los parámetros de la función  $Q$  al interior del mini-batch.

La *target network*, disminuye las oscilaciones y la divergencia de la actualización para la función  $Q$ , puesto que genera el retraso de la función objetivo con parámetros antiguos. El algoritmo 1 presenta un pseudo-código de la aplicación de DQN, con la utilización del *replay buffer*, la aproximación de la función  $Q$  con redes neuronales y las actualizaciones de las redes. Todo esto aplicado en el problema planteado en el paper original de DQN [24].

---

**Algorithm 1** DQN

---

```
1: Inicializar el replay buffer  $\mathcal{D}$  en capacidad  $N$ 
2: Inicializar la función acción-valor objetivo  $Q$  con pesos aleatorios
3: for episodios = 1,  $M$ . do
4:   Inicializar secuencia  $s_1 = \{x_1\}$  y secuencia de preprocesamiento  $\phi_1 = \phi(s_1)$ 
5:   for  $t = 1, T$  do
6:     Con probabilidad  $\varepsilon$  aplicar una acción aleatoria  $a_t$ , de lo contrario elegir  $a_t = \max_a Q^*(\phi(s_t), a; \theta)$ 
7:     Ejecutar la acción  $a_t$  en el simulador y observar la recompensa  $r_t$  y la imagen  $x_{t+1}$ 
8:     Establecer  $s_{t+1} = \{s_t, a_t, x_{t+1}\}$  y preprocesamiento  $\phi_{t+1} = \phi(s_{t+1})$ 
9:     Guardar la transición  $(\phi_t, a_t, r_t, D_t, \phi_{t+1})$  en  $\mathcal{D}$ 
10:    Muestrear un minibatch aleatorio de transición  $(\phi_j, a_j, r_j, \phi_{j+1})$  desde  $\mathcal{D}$ 
11:    if  $\phi_{j+1}$  es terminal then
12:      Aginar  $y_j = r_j$ 
13:    else
14:      Asignar  $y_j = r_j + \gamma \max_{a'} Q(\phi_{j+1}, a'; \theta)$ .
15:    end if
16:    Aplicar un paso de gradiente descendente en  $(y_j - Q(\phi_j, a_j; \theta))^2$  con respecto a  $\theta$ 
17:  end for
18: end for
```

---

### 5.3.2. *Twin Delayed Deep Deterministic Policy Gradient* TD3

El algoritmo *Twin Delayed Deep Deterministic Policy Gradient* (TD3) es una metodología **actor-crítico** basada en *Temporal Difference* y *Off-Policy* centrado en el uso de principios derivados de *Q-Learning*, donde se busca una política aprendida determinística de la forma  $a = \pi(s)$ . Las características de TD3 permiten su uso en entornos de acciones continuas y de mayor complejidad que DQN.

TD3 es una derivación de DDPG (del inglés *Deep Deterministic Policy Gradient*), cuya estructura está basada en un par actor  $\pi$  y crítico  $q_\theta$  de exploración, y un segundo par actor-crítico objetivo ( $\pi'_\theta$  -  $Q'_{\theta}$  respectivamente). La modificación que propone TD3 se basa en tres puntos, en primer lugar utiliza dos funciones  $Q_{\theta_1}$  y  $Q_{\theta_2}$  en el aprendizaje (*Double Q-Learning*), cada una con su respectiva función objetivo  $Q'_{\theta_1}$  y  $Q'_{\theta_2}$ , con el objetivo de resolver un problema de sobre-estimación que DDPG y DQN poseen al actualizar la política debido a la utilización del operador max en:

$$Q(s, a) = R_s^a + \gamma \max_{\hat{a}} Q(s', a'). \quad (5.18)$$

De esta forma, TD3 surge de incorporar *Double Q-Learning* en DDPG, lo que permite el cálculo del valor de la siguiente etapa en base a las funciones objetivo  $Q'_{\theta_1}$  y  $Q'_{\theta_2}$  de la forma:

$$Q_{\theta'_1}(s', a') = Q_{\theta_1}(s', \pi_{\phi_1}(s')) \quad (5.19)$$

$$Q_{\theta'_2}(s', a') = Q_{\theta_2}(s', \pi_{\phi_2}(s')) \quad (5.20)$$

Al utilizar el mínimo de los dos valores (*clipped*) para reescribir el cálculo visto en la Ecuación 5.17, obteniendo:

$$y = r + \gamma \min_{i=1,2} Q_{\theta'_i}(s', \pi_{\phi_i}(s')) \quad (5.21)$$

La segunda modificación busca disminuir la varianza de las estimaciones, para ello TD3 disminuye la frecuencia de actualización de la política  $\pi_\theta$ , solo actualiza la política y las redes objetivo con respecto al crítico después de un determinado número de pasos  $d$ , generando un retraso (*delay*) en la actualización de los parámetros, entre mayor sea este retraso menor será la varianza, logrando obtener una mejor política.

Finalmente, la última modificación se basa en la tendencia de las políticas determinísticas en sobre-ajustarse a *peaks* del valor estimado, de modo que se propone ajustar la política a una pequeña área alrededor del valor objetivo. Para ello se agrega un ruido de distribución normal truncado, de modo que el cálculo de los valores- $Q$  pueda ser suavizado, evitando el sobreajuste:

$$y = r + \gamma Q_{\theta'}(s', \pi_{\phi'}(s') + \varepsilon), \quad \varepsilon \sim \text{clip}(\mathcal{N}((0, \sigma), -c, c)) \quad (5.22)$$

El algoritmo 2 presenta un pseudo-código de la aplicación de TD3, como una modificación de DDPG, con dos redes acción-valor, la actualización retrasada de la política hacia el crítico y la aplicación de ruido en la ajuste de la política. Todo esto aplicado en el problema planteado en el paper original de TD3 [15].

---

#### Algorithm 2 TD3

---

- 1: Inicializar los críticos  $Q_{\theta_1}$ ,  $Q_{\theta_2}$ , el actor  $\pi_\phi$  con parámetros aleatorios  $\theta_1$ ,  $\theta_2$ ,  $\phi$
  - 2: Inicializar las redes objetivo  $\theta'_1 = \theta_1$ ,  $\theta'_2 = \theta_2$ ,  $\phi' = \phi$
  - 3: Inicializar *replay buffer*  $\mathcal{B}$
  - 4: **for**  $t = 1$  a  $T$  **do**
  - 5:     Realizar acción con ruido de exploración  $a_t \sim \pi_\phi(s) + \varepsilon$ ,  $\varepsilon \sim \mathcal{N}(0, \sigma)$
  - 6:     Observar recompensa  $r_t$  y nuevo estado  $s'$
  - 7:     Guardar la tupla de transición  $(s, a, r, s')$  en  $\mathcal{B}$
  - 8:     Muestrear un *minibatch* de  $N$  transiciones  $(s, a, r, s')$  desde  $\mathcal{B}$
  - 9:      $\tilde{a} = \pi_{\phi'}(s') + \varepsilon$ ,  $\varepsilon \sim \text{clip}(\mathcal{N}((0, \sigma), -c, c))$
  - 10:      $y = r + \gamma \min_{i=1,2} Q_{\theta'_i}(s', \tilde{a})$
  - 11:     Actualizar el crítico  $\theta_i = \arg \min_{\theta_i} N^{-1} \sum (y - Q_{\theta_i}(s, a))^2$
  - 12:     **if**  $t$  módulo  $d$  **then**
  - 13:         Actualizar  $\phi$  según gradiente determinístico de la política:
  - 14:          $\nabla_\phi J(\phi) = N^{-1} \sum \nabla_a Q_{\theta_1}(s, a)|_{a=\pi_\phi(s)} \nabla_\phi \pi_\phi(s)$
  - 15:         Actualizar redes objetivo:
  - 16:          $\theta'_i = \tau \theta_i + (1 - \tau) \theta'_i$
  - 17:          $\phi' = \tau \phi + (1 - \tau) \phi'$
  - 18:     **end if**
  - 19: **end for**
-

### 5.3.3. Proximal Policy Optimization PPO

El algoritmo *Proximal Policy Optimization* (PPO) es una metodología *policy-based* basada en Monte Carlo y *On-Policy*, donde se busca una política aprendida estocástica de la forma  $\mathbb{P}(a) = \pi(a|s)$ . Las características de PPO lo hacen sumamente versátil y permiten su uso en entornos de acciones discretas o continuas.

Se define la función de desempeño  $J(\pi_\theta)$  a maximizar como el sustituto  $L^{CLIP}$  de la forma:

$$J(\pi_\theta) = L^{CLIP} = \mathbb{E} \left[ \min \left( \frac{\pi_\theta(a_t|s_t)}{\pi_{\theta_{old}}(a_t|s_t)} \hat{A}_t, \text{clip} \left( \frac{\pi_\theta(a_t|s_t)}{\pi_{\theta_{old}}(a_t|s_t)}, 1 - \varepsilon, 1 + \varepsilon \right) \hat{A}_t \right) \right], \quad (5.23)$$

con  $\varepsilon$  un hiper-parámetro y  $\hat{A}_t$  una estimación de la función de beneficios (Ecuación 5.8) definida para la función valor  $V_{\phi_k}$  como:

$$\hat{A}_t = \sum_{k=0}^T \gamma^k \delta_{t+k}, \quad (5.24)$$

con  $\delta_{t+k} = r_t + V_{\phi_k}(s_{t+1}) - V_{\phi_k}(s_t)$  y  $T$  el número de transiciones realizadas.

Lo anterior, permite limitar el paso de actualización del gradiente de la política para evitar una caída del desempeño, es decir, la variación de parámetros  $\theta$  es limitada por el intervalo dado por el parámetro  $\varepsilon$ . En el paper [31] el autor ejemplifica gráficamente lo anterior, ver Figura 5.6, donde a  $J(\pi_\theta)$  se le llama  $L^{CLIP}$  y  $r = \frac{\pi_\theta(a_t|s_t)}{\pi_{\theta_{old}}(a_t|s_t)}$ .

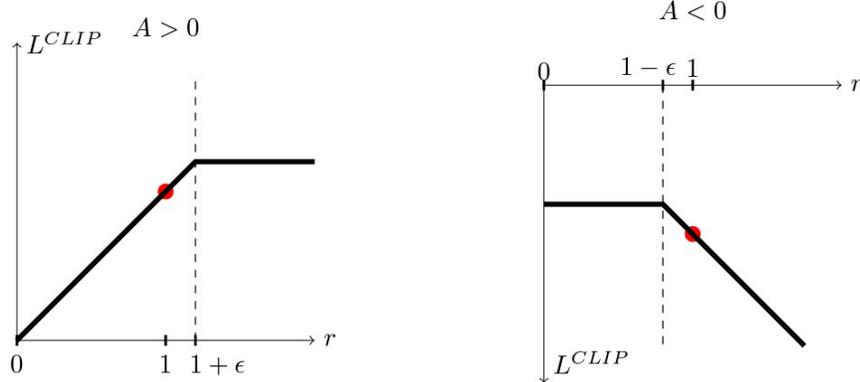


Figura 5.6: Valor de la función  $L^{CLIP}$  para un paso, en función del valor de los beneficios  $\hat{A}_t$ , partiendo desde  $r = 1$ .

Si el beneficio  $\hat{A}_t$  es positivo, entonces el desempeño está acotado superiormente por  $1 + \varepsilon$ . Por otro lado, si  $\hat{A}_t < 0$ , entonces está acotado inferiormente por  $1 - \varepsilon$ .

El algoritmo 3 presenta un pseudo-código de la aplicación de PPO, donde para cada iteración, cada uno de los  $N$  actores recolecta datos de  $T$  pasos. Luego, construye la función de desempeño para los  $NT$  pasos, y optimiza con un *minibatch*. Todo esto explicado en la implementación original de PPO en [31].

---

**Algorithm 3** PPO-*Clip*

---

```
1: for iteration = 1, 2, ... do
2:   for actor = 1, 2, ... ,  $N$  do
3:     Ejecutar la política  $\pi_{\theta_{old}}$  en el entorno durante  $T$  pasos
4:     Calcular los beneficios estimados  $\hat{A}_1, \dots, \hat{A}_T$ 
5:   end for
6:   Optimizar el sustituto  $L^{CLIP}$  con respecto a  $\theta$ , con  $K$  épocas y un minibatch de
   tamaño  $M \leq NT$ 
7:    $\theta_{old} = \theta$ 
8: end for
```

---

En el algoritmo 7 del Anexo C se presenta el pseudo-código de PPO utilizado en el libro [11], en el cual se entregan más detalles de la aplicación de esta metodología, y como realiza las actualizaciones de la política.

# Capítulo 6

## Metodología

La problemática planteada en este trabajo es la generación de programaciones de riego para el cultivo de uva vinífera con estrategias de aprendizaje reforzado. Para su resolución se plantea la siguiente metodología:

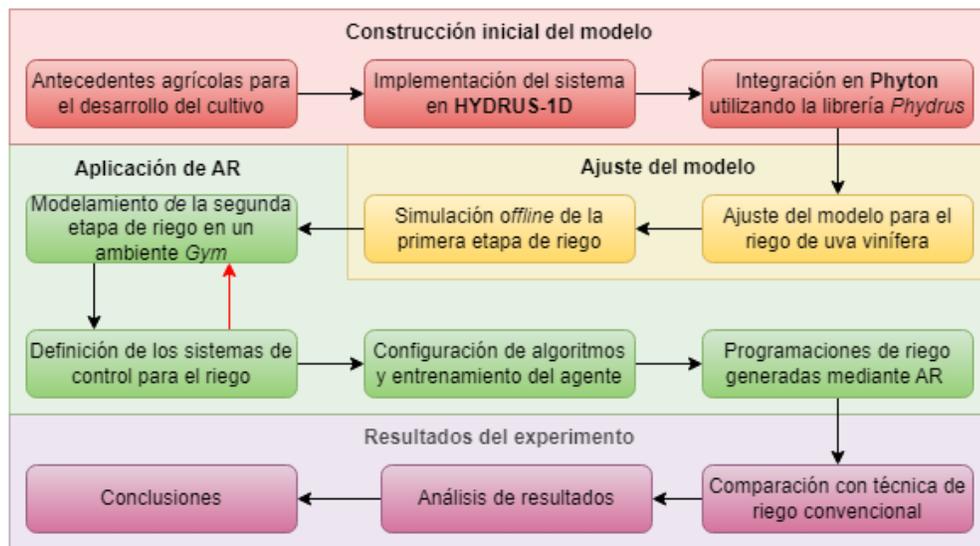


Figura 6.1: Diagrama de flujo de la metodología aplicada.

Esta secuencia de procesos busca cumplir con los objetivos propuestos: modelamiento del suelo de cultivo en base a las características del mismo, integración del modelo con AR, diseño y aplicación de controladores con AR y generación de planificaciones de riego comparables a técnicas existentes.

Como se aprecia en la Figura 6.1, el desarrollo del trabajo se compone de 4 partes: generación inicial del modelo, ajuste del modelo, aplicación de AR y los resultados del experimento. A continuación se describe cada una de las etapas, su aporte en la resolución del problema, y los puntos claves de su ejecución.

## 6.1. Construcción del modelo inicial

Esta fase es una introducción a la problemática y al programa de simulación HYDRUS-1D, aquí se sientan las bases del trabajo con el establecimiento de un modelo inicial del sistema. Son 3 los procesos que constituyen esta etapa: El estudio de antecedentes agrícolas para el desarrollo del cultivo, la implementación del sistema en HYDRUS-1D y la integración en Python utilizando la librería *Phydrus*.

### 6.1.1. Antecedentes agrícolas para el desarrollo del cultivo

Los antecedentes agrícolas se obtienen mediante una investigación bibliográfica del problema, consulta con especialistas agrónomos dedicados al estudio de suelos y al desarrollo de cultivos. Se busca dar respuestas a preguntas como:

- ¿Cuáles son las etapas de desarrollo de un cultivo?
- ¿Qué procesos definen la dinámica del movimiento de agua en el cultivo?
- ¿Qué variables meteorológicas influyen en el desarrollo del cultivo?
- ¿Cuáles son los criterios usados para tomar las decisiones de riego?
- ¿Cómo se determinan las características del riego (cantidad, periodicidad, método, etc)?
- ¿Cómo se define el rendimiento de los cultivos?

Las respuestas a estas preguntas dan forma a un esquema general del sistema, que luego permite abordar la implementación en HYDRUS-1D, para la cual es necesario estudiar su funcionamiento, de modo de realizar pruebas iniciales de las capacidades del software, a fin de evaluar su utilización en las etapas siguientes.

### 6.1.2. Implementación del sistema en HYDRUS-1D

El capítulo 4 presenta una guía básica con los pasos necesarios para configurar el sistema en HYDRUS, con la cual se realizan pruebas para un análisis de las funciones, parámetros y variables del software, su significado y como aprovechar la información que entrega para la generación de un modelo que se pueda utilizar en aprendizaje reforzado, el cual debe ser implementado en Python, lo que nos lleva al siguiente paso.

### 6.1.3. Integración en Python utilizando la librería *Phydrus*

La integración de HYDRUS-1D en Python es necesaria para la definición de un entorno de aprendizaje reforzado, para ello se aprovecha el trabajo realizado con la librería *Phydrus* [7]. En términos simples se convierte la interfaz de usuario de HYDRUS en funciones que

van configurando el modelo. Una vez se logra la integración del modelo básico en Python, se puede continuar con la siguiente etapa, donde se ajusta el modelo según las características particulares del cultivo de uva para vino, su desarrollo y los requerimientos de la Viña.

## 6.2. Ajuste del modelo

Al contar con un modelo básico implementado en Python, se procede a mejorarlo integrando la información recabada respecto del cultivo de uva para vino, esto buscando que el modelo represente de forma lo más fiel posible al sistema real, dado que este trabajo se enfoca en la uva vinífera, los datos y recomendaciones de la Viña son vitales.

### 6.2.1. Ajuste del modelo para el riego de uva vinífera

En primer lugar, se identifican las peculiaridades que tienen los procesos de producción de uva para vino, las cuales sirven de complemento a las respuestas realizadas en la Etapa 6.1. Esta información se integra en el modelo mediante la configuración de las funciones y variables de la implementación con *Phydrus*. Lo anterior, corresponde al modelamiento de la simulación realizada, es decir, la parte asociada al desarrollo de la vid y la dinámica del agua en el suelo cultivado.

Por otro lado, se debe realizar el modelamiento del proceso general, asociado a la duración de las etapas, la información meteorológica disponible, las propiedades y parámetros hidráulicos del suelo, la integración del riego en el modelo y la utilización de los resultados de la simulación para evaluar el desarrollo del cultivo. Para este modelamiento es fundamental el trabajo en Python, con sus herramientas de análisis de datos y la capacidad de generar funciones que complementen lo realizado con *Phydrus*.

### 6.2.2. Simulación *offline* de la primera etapa de riego

Previo a la aplicación de AR, es necesario evaluar el modelo generado, para ello se aprovechan la información provista por la Viña, que indica la separación del desarrollo de la uva en dos etapas. En la primera no se aplica riego, es decir, no se requiere la acción del controlador y la única entrada de agua proviene de las precipitaciones. La segunda, constituye el objetivo de control de este trabajo, es aquí donde se aplican los controladores de riego para cumplir con los requerimientos hídricos del sistema.

De esta manera, se trabaja con la primera etapa de desarrollo y se realizan simulaciones con Python sin acciones del controlador (*offline*), utilizando el modelo ajustado en esta etapa para diferentes condiciones meteorológicas, se evalúa su desempeño comparando los resultados con simulaciones realizadas en HYDRUS-1D con configuraciones similares, y finalmente en caso de tener la información completa de la Viña se compara con sus datos históricos para la primera etapa.

Además de la evaluación del modelo para su utilización con el aprendizaje reforzado, los resultados de la simulación de la primera etapa de desarrollo son las condiciones iniciales de la segunda etapa, en particular el contenido de agua disponible en el perfil de suelo, después de la etapa sin riego. Con el modelo ajustado, las primeras simulaciones y resultados obtenidos se puede pasar al diseño y aplicación del aprendizaje reforzado.

### **6.3. Aplicación de aprendizaje reforzado**

El principal objetivo de esta etapa es la utilización del modelo construido para el diseño y aplicación de los controladores de riego, con los cuales se puedan obtener las planificaciones de riego deseadas.

#### **6.3.1. Modelamiento de la segunda etapa de riego en un ambiente *Gym***

La primera mitad de la aplicación de AR, se basa en un proceso iterativo de prueba y error, basado en adaptar el modelo de la segunda etapa de desarrollo del cultivo al entorno de simulación *Gym*, dándole la estructura de la Figura 5.1, definiéndose las acciones, la recompensa y las observaciones del estado, esto da lugar a un sistema de control con AR. Estos elementos cambia a medida que se busca la estructura que cumpla de mejor manera con los requerimientos del problema.

#### **6.3.2. Sistemas de control para el riego**

El sistema de control basado en AR se aplica en base a dos métodos diferentes para definir el estado hídrico del sistema. La definición de el estado en cada método repercute en la características del modelo de simulación, la configuración de los algoritmos y el cálculo de la recompensa que depende del estado. A continuación, se describen brevemente las principales características de cada método:

- En el primer método el estado hídrico del sistema se define a partir de la medición del contenido de agua en un único punto del perfil. El modelo en este caso es configurado con parámetros predeterminados de HYDRUS y características del cultivo obtenidos de la FAO. Por otro lado, en el modelo el riego aplicado es distribuido de manera uniforme en el día.
- En el segundo método se utilizan cuatro mediciones de contenido de agua para definir el estado hídrico del sistema replicando la utilización de cuatro sensores en el caso real. la configuración del modelo se realiza en base a la información de la Viña, de forma que la simulación se realiza con características y parámetros reales para el perfil de suelo y el cultivo. En este modelo, el riego aplicado es concentrado en un determinado número de horas.

Luego, el sistema de control quedará definido por el método utilizado para la aplicación del AR, la cual también posee diferencia según sea el método aplicado.

### **6.3.3. Configuración de algoritmos y entrenamiento del agente**

Una vez definido el sistema de control, la configuración del AR se basa principalmente en la selección de los algoritmos para el entrenamiento del agente, dichos algoritmos dependen principalmente de la definición de la recompensa y del espacio de acciones para su aprendizaje, es por esto, que para asegurar un correcto aprendizaje se deben probar distintas variaciones en la estructura de AR.

El proceso de diseño, termina con el establecimiento de diferentes configuraciones que pasan a la parte del entrenamiento de los controladores, esto se hace definiendo una serie experimentos, donde se van probando los diferentes controladores, con diferentes estructuras del entorno *Gym*. El proceso de entrenamiento es vital para comprender el proceso de aprendizaje que experimenta el controlador, puesto que muestra las diferencias entre los algoritmos utilizados.

Según sea método con el que se define el estado hídrico, la definición del sistema de control y su aplicación con AR cambia. En el primer método se exploran los mejores algoritmos y funciones de recompensa en un modelo diferente al caso real. Al aplicar el segundo método con características similares a un cultivo real, se restringen los algoritmos y recompensas a aquellos que generan un mejor desempeño con el primer método.

### **6.3.4. Programaciones de riego generadas mediante AR**

El entrenamiento da paso a la obtención de las planificaciones de riego deseadas, con su correspondiente evolución de la dinámica del cultivo y la recompensa asociada a esa programación. Esta información permite pasar al proceso de evaluación del trabajo realizado, donde se define el cumplimiento o no de los objetivos.

## **6.4. Resultados del experimento**

Se puede resumir este trabajo como: utilizar técnicas de modelamiento y control de sistemas en el riego de cultivos, generando alternativas a estrategias convencionales. Luego, es necesario revisar los resultados obtenidos, estableciendo elementos de comparación y métricas para determinar el cumplimiento de los objetivos.

### **6.4.1. Comparación con técnica de riego convencional**

A partir de los resultados obtenidos en la experimentación se evalúan y comparan las planificaciones de riego generadas con estrategias convencionales aplicadas al mismo modelo, las cuales se construyen utilizando la información proveniente del modelo, de modo de determinar la cantidad de riego y el día de aplicación en base a las mediciones de contenido de agua disponibles.

### **6.4.2. Análisis de resultados**

Los principales aspectos del trabajo son el modelo de simulación construido y el sistema de control aplicado con AR para la obtención de las programaciones de riego.

De esta manera, se busca evaluar el desempeño del modelo en representar el cultivo estudiado, basado en la información provista por la Viña, consistente en datos históricos de riego y mediciones del suelo. En cuanto a las programaciones de riego, las obtenidas a partir del primer método, no puede ser comparadas directamente con el caso real, sin embargo, sus resultados permiten identificar las mejores configuraciones del aprendizaje reforzado.

En el segundo método, las programaciones se obtienen con los algoritmos y recompensas encontrados con el primer método y un modelo de simulación que aprovecha la información del cultivo real. Luego, las planificaciones de riego dadas por el controlador son comparadas con las realizadas por la Viña.

### **6.4.3. Conclusiones**

Tomando en consideración el trabajo realizado y los resultados obtenidos, se realiza un análisis de estos, concluyendo respecto del cumplimiento de los objetivos planteados, identificando puntos favorables y aspectos de mejora.

# Capítulo 7

## Desarrollo de modelos y diseño de controladores

Este trabajo busca establecer un sistema de control como el visto en la Figura 7.1 para el riego de uva vinífera, para ello se construye un modelo de simulación que represente al sistema real y el controlador basado en aprendizaje reforzado.

En el capítulo anterior se planteo una pequeña descripción del trabajo a realizar para lograr el cumplimiento de los objetivos propuestos, y también en como cada parte del trabajo es fundamental para el éxito de la siguiente. A continuación, se plantea una versión detallada del trabajo realizado.

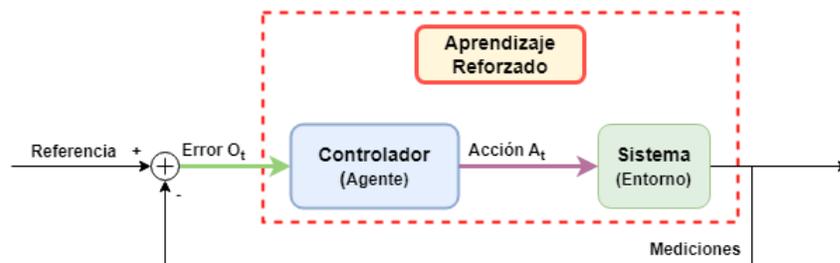


Figura 7.1: Estructura del sistema de control con aprendizaje reforzada, basada en [22]

### 7.1. Construcción del modelo inicial

La obtención de un modelo funcional del sistema, inicia con la identificación de los factores y variables que definen la dinámica del cultivo, sus etapas de desarrollo y el comportamiento del agua en el suelo. A partir de esta información, se utiliza el software de simulación HYDRUS-1D en la implementación, primero en el programa base y luego con la librería *Phydrus* en Python.

### 7.1.1. Antecedentes agrícolas para el desarrollo del cultivo

El primer paso para generar un modelo, es estudiar y conocer el sistema de producción de uva vinífera a modelar, separando el problema en tres partes: dinámica del suelo, características del cultivo y variables del ambiente.

El estudio de la dinámica de suelo se basa en la visión entregada por la ecuación de balance hídrico (Ecuación 3.1). En ella se define el sistema a partir de las entradas y salidas de agua, y el agotamiento o déficit de la zona de la raíz. Todo esto en función de las características del terreno cultivado (tipo de suelo, profundidad hasta nivel freático, pedregosidad, posición geográfica, entre otros).

En el desarrollo del cultivo se identifican como puntos claves: la duración de cada etapa, los requerimientos hídricos asociados y el rendimiento del cultivo para la temporada, estas características varían según la especie, Sección 3.3. Sin embargo, la variable más relevante del estudio corresponde a la evapotranspiración de cultivo ( $ET_c$ ), variable que representa el efecto combinado de la evaporación del agua en la superficie y la transpiración de la planta.

La  $ET_c$  se determina a partir de la evapotranspiración de referencia  $ET_o$  y el coeficiente de cultivo ( $K_c$ ). El primer término puede ser determinado mediante la utilización de la ecuación de Penman-Monteith (más información en el Anexo A), a partir de información meteorológica del lugar y terreno cultivado. Por otro lado,  $K_c$  es una variable que depende del tipo de cultivo, su etapa de desarrollo y las características del terreno.

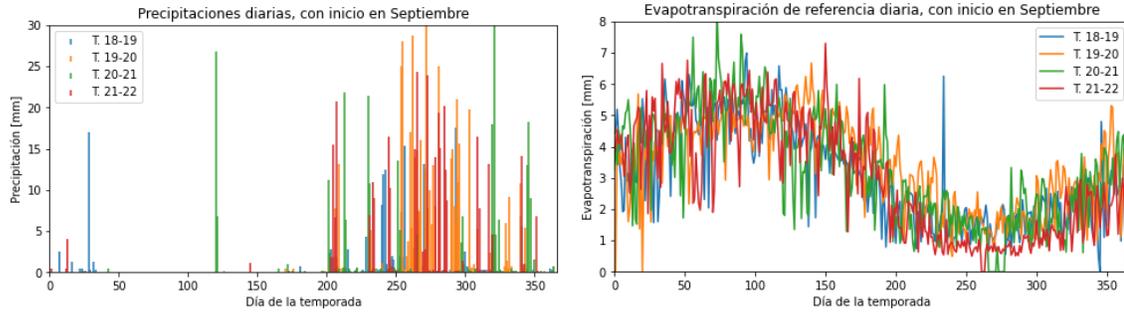
En resumen, las variables y parámetros más relevantes para la generación del modelo son:

- Características del suelo cultivado.
- Características del cultivo.
- Movimientos de agua de ingreso, salida y dentro del suelo.
- Evapotranspiración de referencia ( $ET_o$ ).
- Coeficiente de cultivo ( $K_c$ )
- Precipitaciones diarias en  $[mm]$ .

Esta información es recabada desde la estación meteorológica de la Viña, estudios de sus cultivos y programas de medición del estado de la plantación. El resto de datos puede ser obtenido mediante una búsqueda bibliográfica de otros casos de estudio del cultivo.

A modo de ejemplo se presenta la información de precipitaciones y evapotranspiración de referencia, provista por la Viña. La cual consta de mediciones realizadas desde el 2018 hasta el presente, las cuales son separadas en 4 temporadas.

Por otro lado, la curva del coeficiente de cultivo  $K_c$  para la obtención de la  $ET_c$  se aborda de dos maneras:

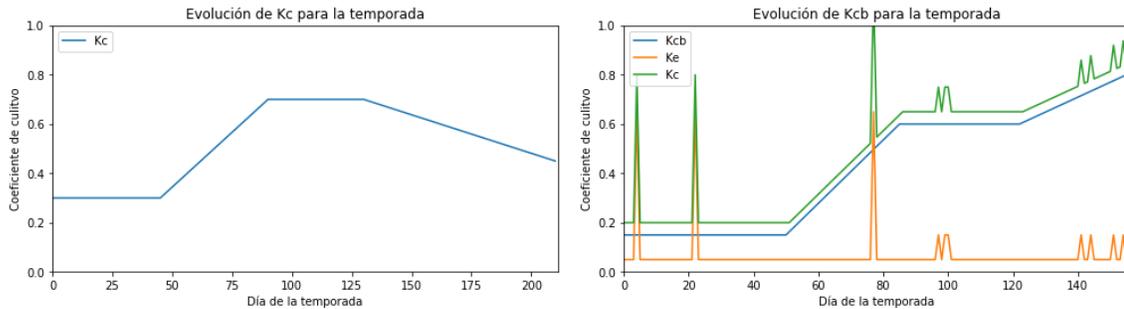


(a) Precipitaciones

(b) Evapotranspiración de referencia

Figura 7.2: Información meteorológica provista por la Viña entre los años 2018 y 2022

- Un único valor de  $K_c$  que agrupa los efectos de la evaporación y transpiración, en base a la información de las etapas y necesidades hídricas del cultivo provistas por la FAO, según lo visto en el Capítulo 6 de la FAO [2].
- La separación de los efectos de evaporación y transpiración base del cultivo en los coeficientes  $K_e$  y  $K_{cb}$  respectivamente, de la forma  $ET_c = ET_o \cdot K_c = ET_o \cdot (K_e + K_{cb})$ , indicada en el Capítulo 7 de la FAO [2].



(a) Curva  $K_c$

(b) Curva  $K_e + K_{cb}$

Figura 7.3: Curvas para el coeficiente de cultivo  $K_c$

La Figura 7.3a muestra la evolución del coeficiente de cultivo según la información provista por la FAO para la uva vinífera. La Figura 7.3b se construye en base a la información provista por la Viña del coeficiente base de cultivo  $K_{cb}$ , por otro lado, el coeficiente de evaporación  $K_e$  es una variable que depende del ingreso de agua para la presencia de evaporación superficial.

### 7.1.2. Implementación del sistema en HYDRUS-1D

Una vez se ha identificado el sistema y se recopila la información relevante a utilizar, se procede a la utilización del software de simulación HYDRUS-1D.

Este programa de modelamiento permite resolver la dinámica del terreno del cultivo, a partir de un conjunto de ecuaciones que simulan el movimiento de agua. Esto permite aprovechar la información disponible para la configuración de la simulación, logrando aumentar

la similitud con el caso real. Además, HYDRUS cuenta con una base de datos de parámetros en caso de contar con esa información.

De esta manera, se sigue el procedimiento descrito en el Capítulo 4 para la simulación de un proyecto que consta con los procesos de flujo de agua y absorción de agua por las raíces, ver Figura 4.3. Y desde aquí se siguen configurando los parámetros de la simulación.

Este paso busca estudiar el programa y aprender su funcionamiento, determinar como utilizar la información disponible y como aprovechar la capacidad computacional de este en la generación del modelo en Python.

A continuación, se mencionan brevemente los pasos de configuración conforme a la guía vista en el Capítulo 4, a modo de resumen se muestra como utilizar la información en ciertas etapas, mencionando la imagen de configuración asociada y los parámetros clave de esa parte:

- En la Figura 4.4, se utilizan  $[mm]$  como unidad de longitud, ya que la información de riego, precipitaciones y ET se entregan en esa unidad. También se determina el/los tipo/s de suelo presentes en el terreno.
- En la Figura 4.5, la unidad de tiempo se define en días, puesto que el objetivo es generar programaciones de riego diario.
- En la Figura 4.8, se elige el modelo hidráulico de van Genuchten - Mualem.
- En la Figura 4.9, de la base de datos de HYDRUS se obtienen los parámetros asociados al/los tipos de suelo presentes en el terreno.
- En la Figura 4.10, se elige drenaje en el fondo del perfil y generación de capa de agua en la superficie como condiciones de borde. Y se define el uso de contenido de agua como variable principal en el modelo.
- En la Figura 4.11, se selecciona el modelo Feddes, y sus parámetros en la Figura 4.12 son los del cultivo estudiado, obtenidos de la base de datos.
- En la Figura 4.14, se muestra que la información de precipitaciones, evaporación y transpiración, se entregan en  $\left[\frac{mm}{dia}\right]$ , debido a las unidades definidas anteriormente.
- En la Figura 4.18, se configura la condición inicial del contenido de agua del perfil, con la opción de encontrarse en capacidad de campo.

Los elementos aquí listados son usados directamente en la implementación en Python, la generación del modelo se basa en completar el resto de la configuración, y dar forma al modelo de simulación para la temporada.

### 7.1.3. Integración en Python utilizando la librería *Phydurs*

La implementación de HYDRUS-1D parte de la instalación de la librería *Phydurs* (explicación en el Anexo B). Como se menciona en la parte anterior, hay información que es llevada directamente desde la interfaz de usuario a su función asociada. La configuración restante se

divide en dos grupos, la primera asociada a como efectuar la simulación del proceso y la otra a información que debe ir cambiando diariamente según avanza la temporada.

El primer grupo de configuraciones se resuelve a partir de definir la realizaciones de simulaciones con duración de 1 día, esto pues el sistema de control busca saber cuanto regar en cada día que actúe, esto implica que se realizara una simulación en HYDRUS para cada día de la temporada, a partir de los resultados del día anterior y la información meteorológica de ese día.

Esto nos lleva a la información que cambia diariamente, en primer lugar el contenido de agua en el perfil, correspondiente a la variable de interés, se obtiene al extraer el estado final del perfil del día anterior. Según sea el método con el que se defina el estado hídrico cambia las mediciones de contenido de agua utilizadas.

En cuanto a la información meteorológica, en primer lugar se determina  $ET_c$  a partir de los valores de  $ET_o$  y  $K_c$  diarios, para obtener con  $K_e$  y  $K_{cb}$  la evaporación potencial y transpiración potencial, variables de entrada que requiere HYDRUS. En el caso de las precipitaciones, estas constituyen la única forma de ingreso de agua al perfil, por lo que, además de la lluvia diaria se incorpora el riego como una precipitación. Esto se puede ver de mejor manera en la Figura 7.4.

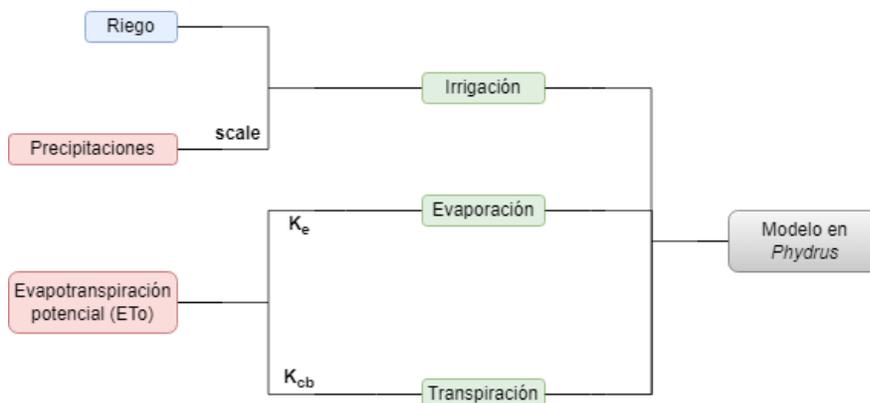


Figura 7.4: Variables atmosféricas de entrada al modelo en *Phydrus*

A partir de estas configuraciones se define la base del modelo, correspondiente a la función `hydrus_day` que simula la dinámica del suelo para un día. Esta función requiere de los siguientes parámetros:

- `profile`: perfil de suelo utilizado para la simulación en formato `DataFrame`.
- `ETo`: evapotranspiración de referencia del día en  $[mm]$ .
- `prec`: precipitación del día en  $[mm]$ , ajustada por `scale`.
- `riego`: riego del día en  $[mm]$ .
- `horas_riego`: duración del riego aplicado.
- `dia` : día de la simulación.

- $K_{cb}$  : coeficiente de base de cultivo del día.
- $K_e$  : coeficiente evaporación de suelo.
- RWU: selección de parámetros para modelo Feddes.
- Soil: lista con los tipos de suelo (*loam, silt, sand, clay*) del perfil.
- cotas: profundidad de los tipos de suelo definidos en *soil*.
- Folder: carpeta en la que se guardan los archivos de simulación diarios de *Phydrus*.
- metric: selección de la variable (contenido de agua o cabezal de presión).

Esta función entrega como salida el perfil actualizado, la información del perfil y el valor de las mediciones de contenido de agua de interés. El modelo se completa con funciones auxiliares para la generación del perfil de suelo, el cálculo de  $K_c$ , la implementación de la Ecuación 4.1 de Genuchten y la obtención de mediciones del contenido de agua.

Lo anterior constituye la generación básica del modelo de simulación, en la que se debe realizar un siguiente ajuste asociado a las características del cultivo de uva vinífera, de modo de entrar a las especificaciones necesarias para el problema planteado.

## 7.2. Ajuste del modelo

El objetivo de los procesos previos era la generación de un modelo genérico para cualquier tipo de cultivo, en la presente sección se da el siguiente paso para la sintonización del modelo a los datos del cultivo de uva vinífera.

### 7.2.1. Ajuste del modelo para el riego de uva vinífera

La comparación de los resultados de este trabajo esta sujeta a la capacidad del modelo de replicar las condiciones reales del sistema, para lograr esto se realiza el ajuste de los parámetros según la información obtenida de la Viña y del cultivo de uva vinífera.

#### Ajustes del modelo para el primer método

La configuración del modelo varía según el método de definición del estado hídrico a utilizar, en el primer caso este se define a partir de la medición del contenido de agua a los 30 [cm] de profundidad, luego el modelo se configura con parámetros predeterminados de HYDRUS-1D e información del cultivo proveniente de la FAO, de esta forma las variables del modelo para el primer método son:

- Día de inicio de la temporada: 1 de Octubre.
- Profundidad del terreno: 1.1 [m] o 1100 [mm].
- Tamaño de la zona de la raíz (RZL): desde los 5 [cm] aproximadamente hasta los 80 [cm] de profundidad.
- Tipo de suelo: Franco (*loam*) homogéneo en todo el perfil (sin el dato de la Viña, se utiliza la opción predeterminada de HYDRUS).
- Parámetros de van Genuchten: de la base de datos de HYDRUS según el tipo de suelo, ver Figura 4.9.
- Parámetros de Feddes: para uva en período de maduración, valores según base de datos de HYDRUS, ver Figura 4.12.
- Punto de Saturación:  $0.3603 \left[ \frac{m^3}{m^3} \right]$ .
- Capacidad de campo:  $0.2202 \left[ \frac{m^3}{m^3} \right]$ .
- Punto de Marchitez:  $0.0937 \left[ \frac{m^3}{m^3} \right]$ .
- Umbral de estrés hídrico:  $0.1921 \left[ \frac{m^3}{m^3} \right]$ .
- Distribución de las raíces (*Beta*): decaimiento lineal en la zona de la raíz, tal que:  $\beta_5 = 1 \rightarrow \beta_{80} = 0$ , ver Figura 7.5.
- Curva de  $K_c$ : sin información de la Viña, es calculada según la duración de etapas y valores estándares de la FAO, ver Figura 7.3a.
- Evaporación y Transpiración: al dividir la evapotranspiración de cultivo como:  $ET_c = \text{Evaporación} \times (1 - \alpha) + \text{Transpiración} \times \alpha$ , con  $\alpha$  un parámetro por definir.

La programación de riego implementada por la Viña consta de dos etapas, en primer lugar un periodo que inicia en octubre de aproximadamente 45 días donde no se ejecuta riego alguno y la condición de inicio del suelo es un perfil a capacidad de campo (FC). La segunda etapa comienza a mediados de noviembre y concluye en abril (165 días aproximadamente), en este periodo se aplican riegos de manera periódica al cultivo, para cumplir sus requerimientos hídricos.

Los datos meteorológicos usados corresponden a las mediciones de precipitaciones y evapotranspiración de referencia entre los años 2018 y 2022, estos datos son divididos en 4 temporadas, cada una iniciando el 1 de Octubre.

### Ajustes del modelo para el segundo método

En el segundo método el estado hídrico del sistema se determina a partir de 4 mediciones del contenido de agua en el perfil de suelo (20, 40 , 60 y 100 [cm]), replicando los sensores instalados por la Viña.

De igual manera, la configuración del modelo utiliza la información disponible de la Viña, proveniente de estudios de suelo y características del cultivo de uva vinífera. Los cambios implementados al modelo tomando en cuenta esta información son los siguientes:

- Día de inicio de la temporada: 23 de Septiembre
- Tipo de suelo: 3 secciones de suelo (350 [mm] de franco arcillo arenoso (FAa), 150 [mm] de franco arcilloso (FA) y los 600 [mm] restantes de franco arcillo arenoso (FAa)).
- Parámetros de van Genuchten: de la base de datos de HYDRUS para cada uno de los tipos de suelo, ver Figura 4.9.
- Capacidad de campo y umbral de estrés hídrico para cada tipo de suelo aproximados según curvas de contenido de agua de la Viña.
- Punto de Saturación:  $[0.308, 0.387, 0.308] \left[ \frac{m^3}{m^3} \right]$
- Capacidades de campo:  $[0.2, 0.275, 0.215] \left[ \frac{m^3}{m^3} \right]$ .
- Punto de Marchitez:  $[0.115, 0.161, 0.115] \left[ \frac{m^3}{m^3} \right]$
- Umbrales de estrés hídrico:  $[0.14, 0.1925, 0.1505] \left[ \frac{m^3}{m^3} \right]$ .
- Distribución de las raíces (*Beta*): decaimiento lineal en la zona de la raíz, tal que:  $\beta_5 = 1 \rightarrow \beta_{80} = 0,4$ , ver Figura 7.5.
- Curva de  $K_c$ : generada a partir de las curvas de  $K_e$  y  $K_{cb}$ , ver Figura 7.3b.
- Evaporación y Transpiración: a partir de los coeficientes  $K_e$  y  $K_{cb}$ ,  $K_e$  se determina día a día y  $K_{cb}$  estimado a partir de la información provista por la Viña.

En este caso, la duración de la primera etapa es de aproximadamente 50 días (estado de brotación a flor), mientras que la segunda etapa en torno a 105 días (estados de flor a pepa dura, pepa dura a pinta y pinta hasta Marzo) según lo indicado por la Viña. Por otro lado, los datos meteorológicos se siguen dividiendo en las mismas temporadas, pero deben iniciar en Septiembre.

### 7.2.2. Simulación *offline* de la primera etapa de riego

Una vez ajustado el modelo se debe realizar una evaluación del desempeño de este y también se debe ejecutar la primera etapa planificada por la Viña, en la cual no se realizan riegos. Se decide realizar ambas cosas con el mismo experimento de modo que se puedan realizar cambios al modelo evaluando su desempeño al simular la primera etapa.

La evaluación se realiza comparando el modelo de Python con simulaciones de la etapa en el software HYDRUS-1D. Para ello se evalúa la evolución del contenido de agua en el perfil para los datos meteorológicos de cada una de las cuatro temporadas. Por otro lado,

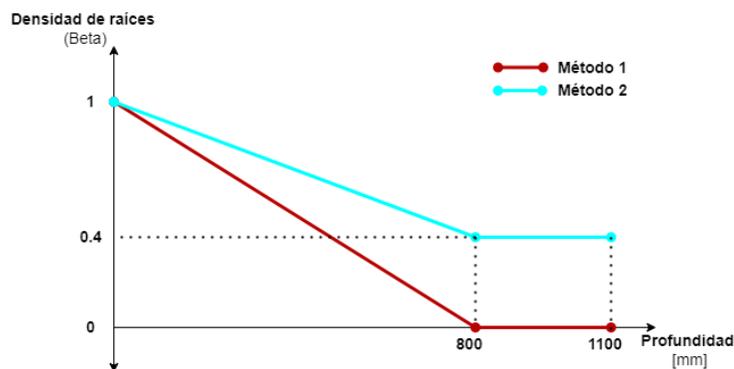


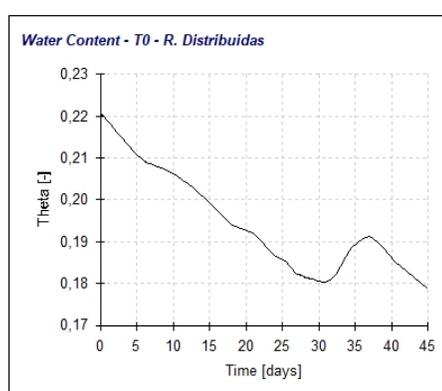
Figura 7.5: Distribución de la densidad de raíces en el perfil de suelo.

es necesario generar perfiles iniciales para la segunda etapa, se generan 16 perfiles de suelo producto del mismo número de posibles combinaciones al elegir un dato de precipitaciones y uno de  $ET_o$ .

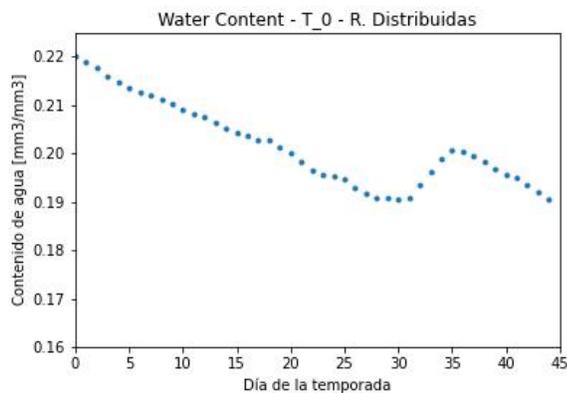
Al realizar cambios en el modelo, la simulación de la primera etapa debe ser realizada tomando en consideración dichos cambios, es por esto que se tienen dos ejecuciones de este paso, asociadas a los diferentes métodos de definición del estado hídrico.

### Simulación del modelo con medición a los 30 [cm]

En este método se caracteriza el estado hídrico del sistema a partir de una única medición en el perfil de suelo, esto se debe a la configuración del modelo utilizada. En este modelo la concentración de raíces es mayor al acercarse a la superficie, el tipo de suelo es homogéneo en todo el perfil y una profundidad de 30 [cm] es un punto medio entre dos de los sensores utilizados por la Viña. Con estos antecedentes, se tienen las implementaciones en HYDRUS-1D y *Phydrus*:



(a) Etapa 1 en HYDRUS-1D



(b) Etapa 1 en *Phydrus*

Figura 7.6: Simulaciones de la etapa 1 con HYDRUS-1D y con el modelo en *Phydrus*, para el método con 1 medición.

## Simulación del modelo con mediciones a los 20, 40, 60 y 100 [cm]

La caracterización del estado hídrico a través de diferentes mediciones simula los sensores instalados por la Viña en el campo. Se utiliza un estudio de suelo para configurar el perfil, el cual indica la presencia de distintos tipos de suelo lo que genera diferentes dinámicas del flujo de agua que son observadas a través de las distintas mediciones, la implementación de este método en la primera etapa se aprecia a continuación:

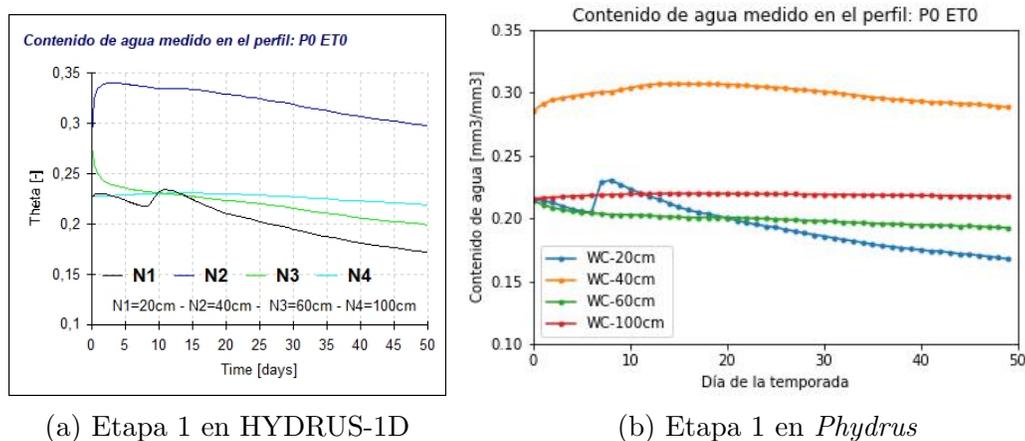


Figura 7.7: Simulaciones de la etapa 1 con HYDRUS-1D y con el modelo en *Phydrus*, para el método con 4 mediciones.

Estas simulaciones son de vital importancia para terminar de ajustar el modelo antes de su utilización en el aprendizaje reforzado, ya que sirven como *checkpoint* para asegurar la correcta implementación de HYDRUS-1D en Python, se pueden revisar las simulaciones para el resto de temporadas en la Sección E.1 del Anexo E.

Por otro lado, también permiten observar un pequeño efecto del cambio climático para el mismo período de tiempo, disminución en las precipitaciones y aumento en la evapotranspiración, por lo tanto, el sistema de control a diseñar debe ser capaz de anticiparse a estos datos.

### 7.3. Aplicación de aprendizaje reforzado

Una vez se tiene el modelo ajustado para el cultivo de uva vinífera se aplica AR para el control de riego en la segunda etapa. Para ello se define la estructura de AR, basada en un entorno, un agente, recompensas y observaciones. Las decisiones del agente se determinan según las características del algoritmo elegido para su entrenamiento, para ello se debe definir la estructura y configuración de estos.

### 7.3.1. Modelamiento de la segunda etapa de riego en un ambiente *Gym*

En la Figura 7.8 se presenta la estructura de aprendizaje reforzado aplicada al problema de control de riego, con todos sus elementos constituyentes (agente, acción, entorno, recompensa, observación, política y algoritmo de aprendizaje).

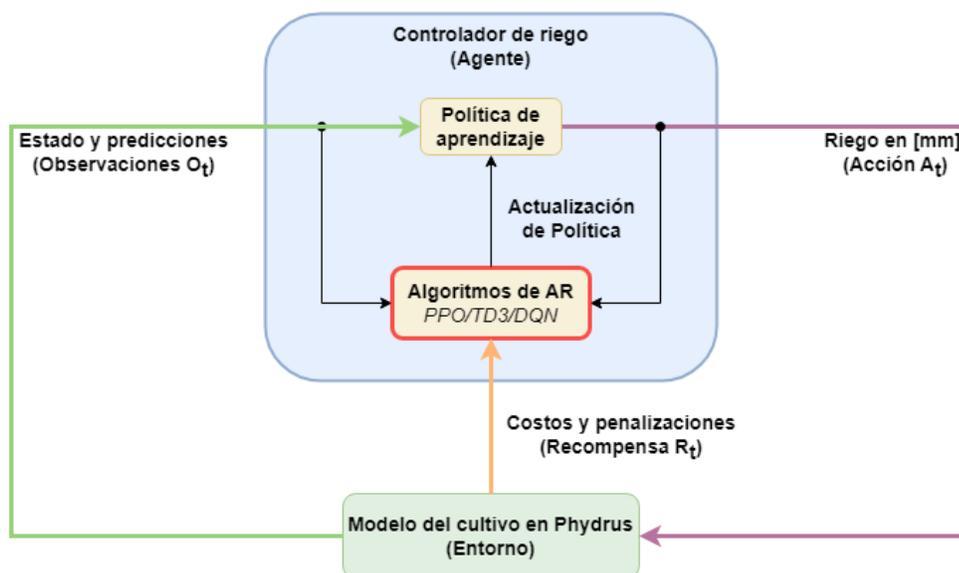


Figura 7.8: Estructura de aprendizaje reforzado diseñada, basada en [22]

En esta sección se determinan el entorno y las observaciones. La configuración y diseño del resto de componentes se explican en las siguientes, sin embargo toda la estructura se diseña a la vez, probando diferentes configuraciones hasta encontrar la definitiva para los experimentos.

Para la definición del entorno se utiliza el paquete *Gym* de Python (más información en el Anexo D), en el cual se adapta el modelo generado en *Phydru* para su utilización con aprendizaje reforzado en la segunda etapa de desarrollo del cultivo.

Un entorno *Gym* define inicialmente sus parámetros y los espacios de acción y observación. Los primeros en este caso están asociados a los datos meteorológicos, a la recompensa definida y al algoritmo elegido. Los espacios se definen en base a si son discretos o continuos, a estos últimos se les asignan los límites inferiores y superiores.

Luego, se realiza la configuración de las condiciones iniciales del entorno (*reset*) para el primer día de la segunda etapa, los datos y valores de parámetros para ese día, el contenido de agua en el perfil de suelo y sus características obtenidas de la primera etapa.

Dependiendo del método para definir el estado hídrico el conjunto de datos meteorológicos cambio, sin embargo se usa el mismo que en la simulación de la primera etapa para la entrega de las condiciones iniciales. Donde se añade el factor de escalamiento de lluvias (*scale*), asociado a que el agua caída no es aprovechada en su totalidad por el cultivo.

De igual manera, la curva de  $K_c$  depende del método, en el primero se utiliza la curva calculado con datos de la FAO y se determina el parámetro  $alpha$  para el cálculo de la evaporación potencial y transpiración potencial. En el segundo método se utiliza la curva de  $K_{cb}$  obtenida con los datos de la Viña para la transpiración y se determina  $K_e$  para la evaporación.

El entorno utiliza estos datos y parámetros en cada paso de simulación (*step*), además se aplica el riego determinado por la acción elegida por el agente y se realiza una simulación diaria. Según sea el método utilizado se genera un nuevo estado hídrico del sistema correspondiente a la actualización del contenido de agua en el terreno, en forma de la medición a los 30 [cm] de profundidad para el primer método y de las mediciones a los 20, 40, 60 y 100 [cm] para el segundo.

La observación se compone de elementos que ayuden al entrenamiento y aprendizaje del agente, para este problema se utiliza el estado hídrico del sistema definido, el número de día, las predicciones de de los próximos 6 días de las precipitaciones y de la  $ET_o$ , y por último el número de días sin regar, este último dato es muy relevante al definir el sistema de control, dando un total de 16 o 19 (según método utilizado) variables que definen el espacio de observaciones.

Finalmente, se definen dos condiciones que terminan la simulación, la primera correspondiente a llegar al último día de la temporada y la segunda alcanzando un estado asociado a la muerte del cultivo por llegar al punto de marchitez permanente en las mediciones de los 30 y 20 [cm], para el primer y segundo método respectivamente. Llegar a estas condiciones implica reiniciar el entorno a su condición inicial y empezar un nuevo episodio.

### 7.3.2. Definición de los sistemas de control para el riego

Los elementos antes definidos permiten establecer el sistema de control, siendo dependiente del estado hídrico definido y la configuración del modelo asociada a este. En particular las observaciones ya que se elijen en base a lo necesario para que el agente aprenda y genere una mejor acción, a continuación se plantean los criterios y límites para la definición de esta.

#### Criterio de riego

En primer lugar, para la definición del sistema de control es necesario revisar la estrategia de riego de la Viña, la cual se basa en mediciones del potencial hídrico xilemático de la hoja del cultivo y mediciones del contenido de agua (WC) dadas por sensores en el suelo, lo riegos se encuentran espaciados por una semana aproximadamente y buscan cumplir un requerimiento hídrico basado en las mediciones.

La estrategia de la Viña se basa en que después de unos 7 días después del riego, se mida el potencial hídrico xilemático y su valor se encuentre entre los 6-7 [bar], la cual se realiza al medir determinados valores de contenido de agua. Esto pretende que la planta sufra de estrés hídrico cada cierto tiempo, mejorando la calidad del fruto para la producción de vino.

El modelo carece de la capacidad de generar información respecto del estado de la planta, pero el contenido de agua en el suelo es la variable base, es así como la decisión de riego se define en función del estado hídrico definido.

Luego, en el primer método la medición a los 30 [cm] representa el estado hídrico de un perfil con un tipo de suelo homogéneo. El segundo método utiliza información real de la Viña en la configuración del modelo, en particular la presencia de diferentes tipos de suelo en el perfil, por lo que una única medición no sería representativa del estado hídrico, es por esto, que se utilizan 4 mediciones que asemejan a los 4 sensores instalados a los 20, 40, 60 y 100 [cm] de profundidad.

Por otro lado, los requerimientos hídricos del cultivo en la Viña son particulares en comparación a otros cultivos, principalmente por el efecto del estrés hídrico en el rendimiento de la uva vinífera. De modo que en base a la información del cultivo disponible, se establece un criterio de riego basado en la optimización del uso de agua, la supervivencia del cultivo y un sometimiento del cultivo a estrés hídrico, por medio de la acción del agente.

### Acción de riego

Con respecto a la acción de riego, la utilización del contenido de agua como variable base del entorno y el conocimiento del valor de capacidad de campo del terreno, permiten definir la acción de riego en función de la diferencia de agua entre WC y FC, y el tamaño de la zona de la raíz (RZL), como se muestra en la Figura 7.9a.

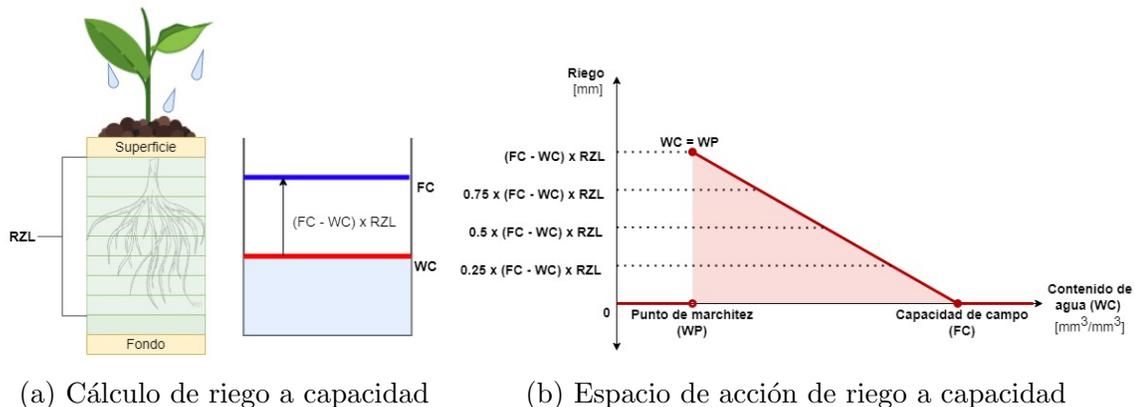


Figura 7.9: Determinación de la acción de control en base al cálculo del riego a capacidad

El riego a capacidad de campo (RCC) pretende aplicar un riego en [mm] que suministre la cantidad de agua faltante para que el terreno cultivado llegue a FC, para eso hace uso del tamaño de la zona de la raíz. El motivo detrás de esto es que FC marca la cantidad límite de agua aprovechable, es decir, cualquier excedente es agua desaprovechada.

En base a esto, el riego a capacidad establece límites para la acción de control, esto se puede ver en la Figura 7.9b, donde se determina un riego nulo en caso de que el contenido de agua sea mayor a FC, por otro lado, si se alcanza el punto de marchitez el cultivo muere por la falta de agua, por lo que para valores de WC inferiores también se tiene riego nulo y el valor máximo teórico de riego es alcanzar FC estando justo en PMP.

Según sea la definición del estado hídrico del sistema, el riego es calculado y aplicado de forma diferente. En el primer método, el suelo es homogéneo por lo que el valor de FC es el mismo para toda la zona de la raíz y el cálculo de RCC es exactamente el visto en la Figura 7.9, adicionalmente en el modelo la irrigación aplicada se distribuye en las 24 horas del día.

En el segundo método, la presencia de diferentes suelos en el perfil implica diferentes valores de FC, por lo tanto, el cálculo de RCC se realiza a partir de los horizontes de los tipos de suelo presentes, el tamaño de la zona de la raíz en ellos y la medición de WC que represente esa porción de suelo, como se muestra en la Figura 7.10. Luego, el riego calculado es aplicado en una fracción del día, 6 horas en este caso, lo que simula el riego en  $\left[\frac{mm}{hr}\right]$  de un equipo instalado en el campo, esto busca que el agua logre alcanzar una mayor profundidad en el perfil de suelo.

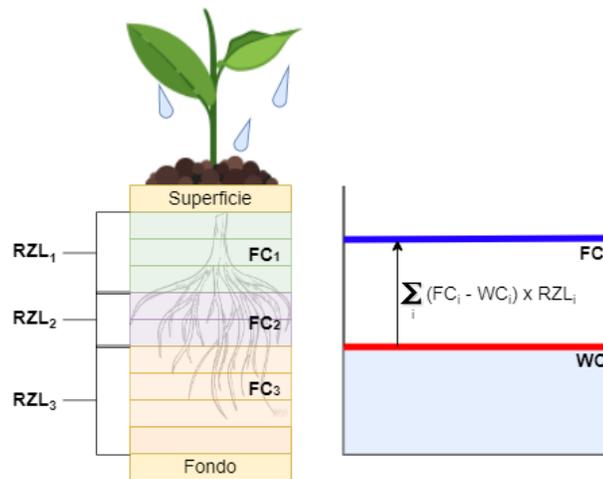


Figura 7.10: Cálculo del riego a capacidad para un perfil con diferentes tipos de suelo en la zona de la raíz

Una vez definidos el criterio de riego y los límites teóricos del riego, se procede a realizar la configuración del agente y su acción, elementos que dependen del algoritmo de aprendizaje reforzado utilizado.

### 7.3.3. Configuración del agente

El agente en el aprendizaje reforzado es el controlador de riego, el cual ejecuta acciones en función de una política de decisión, la cual es actualizada por los algoritmos en base a una función de recompensa, esta se diseña para que en el entrenamiento, el agente aprenda a cumplir los criterios de riego establecidos.

## Recompensa

La recompensa entonces debe buscar la optimización del uso de agua para riego, la supervivencia de la planta y episodios de estrés hídrico que mejoren la calidad del fruto. De esta forma, se propone una recompensa compuesta de dos funciones de costos, una asociada al consumo de agua y el costo operacional de regar, y la otra al estado del cultivo en función del contenido de agua medido.

La función de costos asociada al uso de agua se presenta en la Figura 7.11, la cual se compone por un costo operacional ( $CO$ ) fijo por la acción de regar y otro asociado a la cantidad de agua utilizada para regar. Esta última componente se representa como el producto entre un costo por uso de agua ( $C_{RCC}$ ) y la razón  $\frac{A_t}{RCC}$ , donde  $A_t$  es el riego efectuado y  $RCC$  es el riego a capacidad de campo, esto busca que el agente busque una cantidad de riego óptima, pues penaliza que se riegue el máximo teórico dado por el  $RCC$ .

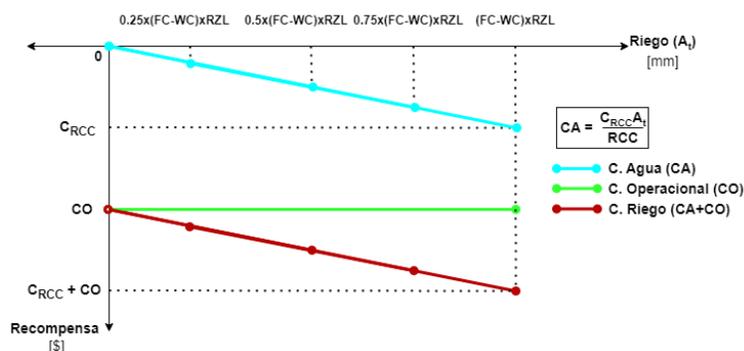
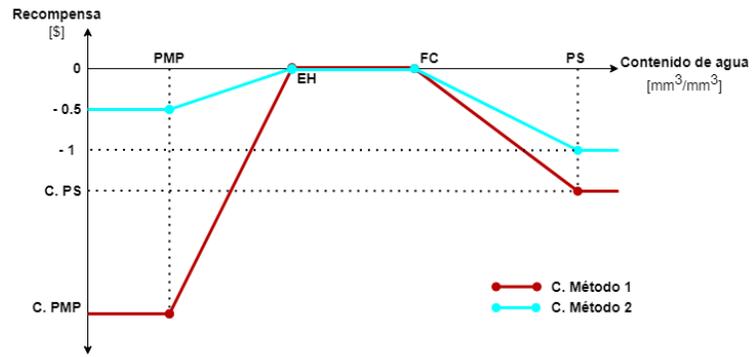


Figura 7.11: Costo por riego del cultivo diario

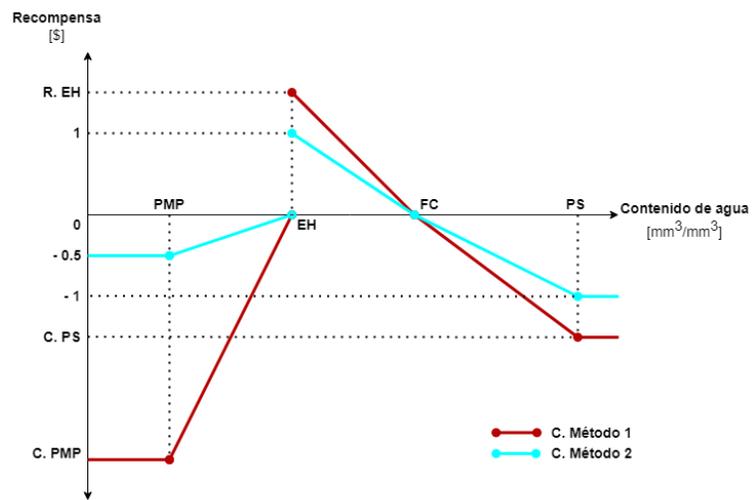
La otra parte de la función de recompensa viene dada por el estado hídrico actual, es decir, por el contenido de agua medido por el o los sensores. Tiene como objetivo penalizar los estados no favorables para el desarrollo del cultivo y bonificar aquellos que sí lo son. En la Figura 7.12 se aprecian las 3 funciones de costos propuestas para el contenido de agua medido, se hace una diferenciación entre la utilizada para cada definición del estado hídrico.

Las penalizaciones presentadas en las funciones de costos se pueden ordenar según la gravedad del estado alcanzado, lo que se refleja como mayor impacto en la recompensa:

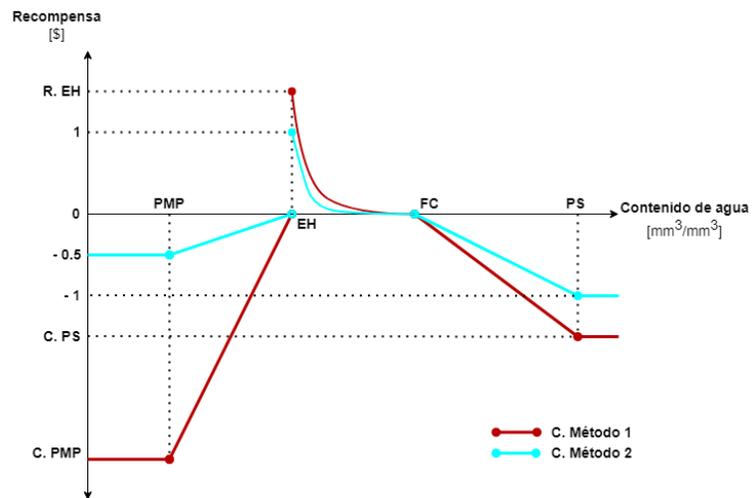
- Muerte del cultivo por alcanzar el punto de marchitez permanente (PMP o WP).
- Muerte del cultivo por alcanzar el punto de saturación (PS), es la mitad que la del PMP.
- Sometimiento a estrés hídrico grave al encontrarse bajo el umbral de estrés hídrico.
- Agua desaprovechada en el terreno al encontrarse por sobre capacidad de campo.



(a) Costo 1: Constante



(b) Costo 2: Lineal



(c) Costo 3: Cuadrático

Figura 7.12: Variaciones del costo diario por contenido de agua medido en el perfil

Por otro lado, para el intervalo entre el umbral de estrés hídrico y capacidad de campo, las funciones de costos abordan la cuestión de manera diferente:

- El Costo 1 (Figura 7.12a), no diferencia entre encontrarse más cerca de EH o FC, se espera que el acercamiento a estrés hídrico ocurra debido al menor uso de agua que requiere mantener ese nivel de contenido de agua.
- El Costo 2 (Figura 7.12b), aplica una alta bonificación a los estados cercanos al umbral de estrés hídrico, la cual decae linealmente al acercarse a FC, esto busca favorecer el acercamiento a EH, sin quitar lo favorable de estar en estados que favorecen el desarrollo del cultivo.
- El Costo 3 (Figura 7.12c), al igual que el anterior se busca favorecer los estados cercanos a EH, pero se utiliza una función cuadrática que genera una gran diferencia de bonificación entre los estados cercanos EH y las cercanías de FC.

En el método con medición a los 30 [cm] el valor entregado por la función corresponde al costo por el estado, ya que es la única medición la que representa el estado. En el otro método, se utilizan los sensores a los 20, 40 y 60 [cm] para representar el costo del estado, por esto las funciones se encuentran acotadas entre [-1,1] y el valor total del costo se define ponderando el valor de la función de costo de cada sensor ( $C_i$ ) según la densidad de raíces ( $\beta$ ) que posea el sector de la zona de la raíz que represente ese sensor como se ve en la siguiente función:

$$\text{Costo Estado} = \frac{\sum_{i=1}^3 C_i * \beta_i}{\sum_{i=1}^3 \beta_i} \quad (7.1)$$

La medición de los 100 [cm] no se considera pues se encuentra fuera de la zona de la raíz, sin embargo, si llega presentar un aumento del contenido de agua es una muestra de un mal uso de agua, debido a que no hay extracción de agua para el desarrollo del cultivo en ese punto. Luego, la recompensa se obtiene al multiplicar el costo total obtenido por un peso entregado como parámetro. Estos parámetros, que asignan los valores a estas funciones son definidos al momento de implementar el entorno *Gym*, por lo que, deben ser sintonizados al entrenar al agente.

## Espacio de acción

El entrenamiento del agente viene dado por los algoritmos de aprendizaje reforzado que se apliquen, pues tienen diferentes formas de actualizar la toma de decisiones (política) del agente. Además, los diversos algoritmos requieren definiciones de acción diferentes según puedan ejecutarse en espacios continuos o discretos.

En el caso continuo (TD3 y PPO), la acción de riego puede tomar cualquier valor entre 0 y RCC para el estado en el que se encuentre, es decir, la recta completa que se aprecia en la Figura 7.13, en la práctica el controlador determina un valor de acción ( $a_t$ ) en  $[-1,1]$ , al cual se le aplica la conversión tal que:

$$\text{Riego} = \frac{a_t \cdot \text{RCC}}{2}, \quad \text{con } a_t \in [-1, 1] \quad (7.2)$$

Por otro lado, en el caso discreto (DQN y PPO) se definen 5 acciones basadas en RCC, sin embargo, para cada método una de las acciones es diferente. En el primer método, se presentaba la opción de regar como si se estuviera en PMP, en el segundo esta acción es reemplazada por el riego al 25 [%] de RCC, el detalle de las acciones se muestra a continuación:

$$\text{Riego}_1 = \begin{cases} 0 & , \text{ si } a_t = 0 \\ 0,5 \cdot \text{RCC} & , \text{ si } a_t = 1 \\ 0,75 \cdot \text{RCC} & , \text{ si } a_t = 2 \\ \text{RCC} & , \text{ si } a_t = 3 \\ \text{RCC en PMP} & , \text{ si } a_t = 4 \end{cases} \quad \text{Riego}_2 = \begin{cases} 0 & , \text{ si } a_t = 0 \\ 0,25 \cdot \text{RCC} & , \text{ si } a_t = 1 \\ 0,5 \cdot \text{RCC} & , \text{ si } a_t = 2 \\ 0,75 \cdot \text{RCC} & , \text{ si } a_t = 3 \\ \text{RCC} & , \text{ si } a_t = 4 \end{cases} \quad (7.3)$$

Estas acciones se pueden apreciar gráficamente en la Figura 7.13, luego en el entorno *Gym* el espacio de acción se define a partir de un parámetro que indica el algoritmo elegido.

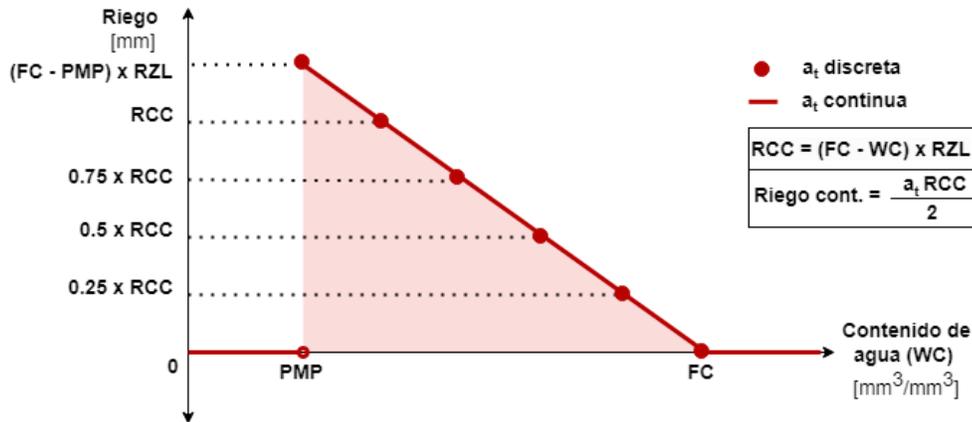


Figura 7.13: Determinación del riego aplicado en función del espacio de acciones utilizado.

En el entorno también se define cuándo aplicar la acción de riego, en base al criterio de riego y las características del modelo. Luego, el controlador no actúa bajo dos condiciones: han pasado menos de 5 días desde el último riego y si el terreno se encuentra sobre capacidad de campo no se debe regar, luego no se aplica ninguna acción.

## Entrenamiento

Con esto ya se ha definido el entorno, el agente, las observaciones, la recompensa y la acción de control, faltando únicamente el entrenamiento y actualización de la política de decisión. Para ello se debe realizar la configuración de los algoritmos en términos de los parámetros y variables que caracterizan su funcionamiento, y al igual que en el diseño del modelo ir probando configuraciones hasta encontrar la deseada.

Uno de los principales elementos a cambiar son las arquitecturas de las redes neuronales para las políticas de decisión (funciones de activación, número de capas ocultas y número de neuronas), el diseño de las arquitecturas difiere entre los algoritmos según su categoría sea *Off-Policy* u *On-Policy*. Además existen otros ajustes asociados a si la estructura es actor-crítico.

Por otro lado, cada algoritmo posee sus propios parámetros que sintonizar, en DQN se debe ajustar el ratio de exploración para el entrenamiento, en TD3 se puede aplicar algún tipo de ruido para evitar el sobreajuste y también se puede establecer un periodo acciones aleatorias durante el entrenamiento a modo de exploración.

Otro factor relevante en el entrenamiento es el número de pasos que dura este, es decir, cuanto tiempo voy a estar entrenando el agente antes de generar una planificación de riego. En general, los *timesteps* rondan por el millón, sin embargo, según sea el comportamiento del agente y su curva de aprendizaje esto puede disminuir para ahorrar recursos computacionales.

Una vez realizado el entrenamiento de los agentes, con las mejores selecciones de parámetros y configuraciones de los algoritmos se procede a la obtención de las programaciones de riego para el cultivo.

### 7.3.4. Programaciones de riego generadas mediante AR

El objetivo de este trabajo es la generación de planificaciones con los agentes entrenados, para ello se realizan experimentos que difieren en las funciones de recompensa y en los algoritmos de aprendizaje reforzado utilizados.

La evaluación de estas planificaciones de riego se realiza en base al uso de agua y a la evolución del contenido de agua en la temporada, enfocado en el sometimiento a estrés hídrico. Bajo estos mismos criterios se realiza la comparación de los resultados con programaciones de riego convencionales. En base a lo anterior, para el primer método se eligen 4 métricas que resuman lo realizado por el controlador en la programación de riego, estas métricas son:

- **Riego total:** Es la cantidad total de agua en milímetros utilizada para regar al cultivo.
- **N° de riegos:** Es la cantidad de veces que el controlador determina regar el cultivo.
- **Riego promedio:** Estimativo de la cantidad de agua suministrada en cada riego.

- **Distancia promedio al umbral de estrés hídrico ( $\bar{z}$ ):** Busca estimar que tan alejado estuvo el cultivo de valores cercanos al umbral deseado, se calcula como:  $\bar{D} = \frac{\sum_{t=0}^T |wc_t - EH|}{T}$ , con  $wc_t$  contenido de agua en el día  $t$  y  $T$  total de días de la temporada.

Con múltiples mediciones de contenido de agua, utilizar la distancia de los sensores al estrés hídrico se vuelve complicado, es por esto que para el segundo se reemplaza esa métrica por la reposición de la  $ET_o$ :

- **Reposición de  $ET_o$ :** Muestra una relación entre el agua pérdida por la evapotranspiración y la ingresada mediante riegos, su valor debe estar cercano a  $K_c$ . Se calcula como:  $\text{Reposición} = \frac{\text{Riego total}}{ET_o \text{ total}}$

### 7.3.5. Implementación de técnicas de riego convencional

Las técnicas de riego convencional son aquellas que se busca mejorar, en primer lugar se intenta replicar la estrategia de riego de la Viña, asumiendo riegos espaciados por al menos 5 días, acompañado de un criterio que espera a encontrarse en estrés hídrico para aplicar el siguiente riego.

Los riegos implementados corresponden al riego a capacidad (RCC) o un porcentaje de este y a la aplicación de cantidades físicas al alcanzar un contenido de agua determinado. Para someter a estrés hídrico a la planta, los riegos propuestos siguen la regla de regar sólo al alcanzar el umbral de estrés hídrico, siempre y cuando no se haya regado ya en los 5 días anteriores, si esa condición se cumple entonces se aplica un riego, las cantidades de agua a regar son las siguientes:

- **100-RCC:** Se aplica el 100 % del riego a capacidad de campo.
- **75-RCC:** Se aplica el 75 % del riego a capacidad de campo.
- **50-RCC:** Se aplica el 50 % del riego a capacidad de campo.
- **25-RCC:** Se aplica el 25 % del riego a capacidad de campo.
- **30mm:** Se aplican 30 [mm] de riego.
- **25mm:** Se aplican 25 [mm] de riego.
- **20mm:** Se aplican 20 [mm] de riego.
- **15mm:** Se aplican 15 [mm] de riego.

Las técnicas de riego planteadas son utilizadas para la evaluación del desempeño de las programaciones de riego generadas con el primer método de definición del estado hídrico. En el segundo método se aprovecha la información del terreno en la configuración del modelo, por lo que se puede realizar la comparación con la estrategia de riego implementada por la viña en las temporadas 18-19, 19-20 y 20-21, de todas formas se aplican 4 métodos convencionales como los ya mencionados:

- **100-RCC:** Se aplica el 100 % del riego a capacidad de campo.
- **75-RCC:** Se aplica el 75 % del riego a capacidad de campo.
- **45mm:** Se aplican 45 [mm] de riego.
- **40mm:** Se aplican 40 [mm] de riego.

# Capítulo 8

## Estudio por simulación

En el presente capítulo se presenta todo el procedimiento de obtención de los resultados del trabajo, comenzando con la descripción de los experimentos realizados, las planificaciones de riego obtenidas y el análisis y evaluación de estas.

### 8.1. Diseño y configuración del experimento

Los experimentos que se realizan para la obtención de los resultados se basan en el modelo y la estructura de aprendizaje reforzado cuyo diseño se detalla en el capítulo 7, los experimentos buscan dar respuesta al cumplimiento de los objetivos y entregar la capacidad de evaluar el desempeño de la estrategia de control propuesta. A continuación se detalla el proceso de diseño.

#### 8.1.1. Configuración del modelo y primera etapa de desarrollo

##### Simulación del modelo con medición a los 30 [cm]

Para realizar las simulaciones de la primera etapa de riego, se configura el modelo en *Phydrus* con los parámetros definidos en la Sección 7.2.1, optando por configurar la zona de la raíz distribuida en el perfil de suelo como se muestra en la Figura 7.5. En la Tabla 8.1 se presenta un resumen de los parámetros utilizados para la configuración del modelo:

La información meteorológica es ajustada por diversos parámetros, *scale* para el escalamiento de las precipitaciones,  $K_c$  para la determinación de la evapotranspiración de cultivo  $ET_c$  y *alpha* para la separación de esta en evaporación ( $E_p$ ) y transpiración potenciales ( $T_p$ ), el valor de esta variable se define como:

$$alpha = \begin{cases} 0,4 & , \text{ si el día anterior llovió} \\ 0,7 & , \text{ si el día anterior se regó} \\ 0,9 & , \text{ para cualquier otro caso} \end{cases} \quad (8.1)$$

| Parámetro              | Valor                        | Descripción                                    |
|------------------------|------------------------------|--|
| Día de inicio          | 1 de Octubre                 | Inicio del desarrollo de la uva                |
| Duración 1° etapa      | 45 días                      | Período sin aplicación de riegos               |
| Duración 2° etapa      | 165 días                     | Aplicación de riegos periódicos                |
| Profundidad del perfil | 1100 [mm]                    | Distancia entre la superficie y la tosca       |
| Zona de la raíz        | 50 - 800 [mm]                | Extensión de raíces en el perfil de suelo      |
| Tipo de suelo          | Franco                       | Define parámetros de van Genuchten             |
| Punto de Saturación    | 0.3603 [ $\frac{m^3}{m^3}$ ] | (PS), punto característico del terreno         |
| Capacidad de campo     | 0.2202 [ $\frac{m^3}{m^3}$ ] | (FC), punto característico del terreno         |
| Punto de Marchitez     | 0.0937 [ $\frac{m^3}{m^3}$ ] | (PMP), punto característico del terreno        |
| Estrés Hídrico         | 0.1821 [ $\frac{m^3}{m^3}$ ] | (EH), punto característico del terreno         |
| Densidad de raíces     | Distribuidas                 | Concentración de las raíces en el terreno      |
| Estado de la uva       | Maduración                   | Define parámetros para modelo Feddes           |
| Valores de $K_c$       | 0.3-0.7-0.45                 | Coefficientes para cálculo de curva $K_c$      |
| Etapas para $K_c$      | 45-45-40-80                  | Duración de etapas para cálculo de curva $K_c$ |
| Escalamiento (scale)   | 80 [%]                       | Factor de aprovechamiento de lluvia            |

Tabla 8.1: Resumen de parámetros definidos para la configuración del modelo en *Phydrus*, para el estado hídrico basado en 1 medición.

### Simulación del modelo con mediciones a los 20, 40, 60 y 100 [cm]

La configuración en este caso, sigue las diferencias planteadas en la Sección 7.2.1, en particular respecto del día de inicio de la temporada, la duración de las etapas, la estructura del suelo, los umbrales hídricos, la distribución de raíces y el cálculo de evaporación y transpiración con  $K_e$  y  $K_{cb}$  respectivamente. Luego, el resumen de la configuración del modelo es la siguiente:

En este método se determinan  $E_p$  y  $T_p$  por medio de los coeficientes  $K_e$  y  $K_{cb}$  respectivamente, el valor diario del segundo se obtiene de la curva generada a partir de los datos de la Viña. Sin embargo,  $K_e$  varía en función de los ingresos de agua del día anterior y la etapa del cultivo, según se muestra en la Ecuación 8.2:

$$K_e = \begin{cases} 0,65 & , \text{si el día anterior ingreso agua y es previo al día 85 de la temporada} \\ 0,15 & , \text{si el día anterior ingreso agua y es posterior al día 85 de la temporada} \\ 0,05 & , \text{para cualquier otro caso} \end{cases} \quad (8.2)$$

La curvas que generan estos coeficientes se aprecia en la Figura 3.3b, en ellas se puede ver el cambio en  $K_e$  posterior al día 85, donde la evaporación disminuye producto del crecimiento del cultivo.

Luego, recordando que la condición inicial del terreno de suelo es la de un contenido de agua igual a capacidad de campo para todo el perfil, se utiliza la información meteorológica provista por la Viña para simular la primera etapa, seleccionando cada una de las temporadas

| Parámetro              | Valor               | Descripción                                  |
|------------------------|---------------------|--|
| Día de inicio          | 23 de Septiembre    | Inicio del desarrollo de la uva              |
| Duración 1° etapa      | 50 días             | Período sin aplicación de riegos             |
| Duración 2° etapa      | 105 días            | Aplicación de riegos periódicos              |
| Profundidad del perfil | 1100 [mm]           | Distancia entre la superficie y la tosca     |
| Zona de la raíz        | 50 - 800 [mm]       | Extensión de raíces en el perfil de suelo    |
| Tipos de suelo         | [FAA/FA/FAA]        | Estructura del perfil de suelo               |
| Extensión de suelos    | [350/150/600]       | Tamaño de la sección de suelo                |
| Puntos de Saturación   | [0.308/0.387/0.308] | (PS), punto característico del terreno       |
| Capacidades de campo   | [0.2/0.275/0.215]   | (FC), punto característico del terreno       |
| Puntos de Marchitez    | [0.115/0.161/0.115] | (PMP), punto característico del terreno      |
| Estrés Hídrico         | [0.140/1925/0.1505] | (EH), punto característico del terreno       |
| Densidad de raíces     | Distribuidas        | Concentración de las raíces en el terreno    |
| Estado de la uva       | Maduración          | Define parámetros para modelo Feddes         |
| Valores de $K_{cb}$    | 0.15-0.6-0.8-0.8    | Coefficientes para cálculo de curva $K_{cb}$ |
| Etapas para $K_c$      | 50-35-37-33         | Duración de etapas en curva $K_{cb}$         |
| Escalamiento (scale)   | 80 [%]              | Factor de aprovechamiento de lluvia          |

Tabla 8.2: Resumen de parámetros definidos para la configuración del modelo en *Phydrus*, para el estado hídrico basado en 4 mediciones.

de precipitaciones con cada una de las temporadas de  $ET_o$ , esto da como resultado un total de 16 perfiles de suelo que sirven de condición inicial para la segunda etapa de desarrollo.

Por último, antes de continuar con el aprendizaje reforzado se debe definir la utilización de la información meteorológica en el entrenamiento y obtención de las programaciones para la segunda etapa:

- En el primer método, la información meteorológica (precipitaciones y  $ET_o$ ) utilizada para la etapa 2, corresponden a las que generaron el perfil inicial, es decir, si el perfil de inicio fue generado con los datos de precipitaciones  $X$  y de  $ET_o$   $Y$ , entonces son esos datos los ocupados para el entrenamiento y obtención de la programación de riego.
- En el segundo método, los datos de precipitaciones y  $ET_o$  utilizados en los episodios de entrenamiento del agente se seleccionan día a día de forma aleatoria entre las 4 temporadas disponibles, esto permite aumentar la aleatoriedad de cada día simulado para el aprendizaje. Sin embargo, en la obtención de las programaciones se selecciona una temporada aleatoriamente y se utilizan sus datos de lluvia y  $ET_o$ , de modo de permitir la comparación con las técnicas convencionales.

### 8.1.2. Configuración del aprendizaje reforzado en la segunda etapa

El ajuste del entorno *Gym* se realiza con los mismos parámetros del modelo para la primera etapa, ver Tabla 8.1 y 8.3, la diferencia recae en la condición inicial del perfil de suelo, correspondiente al estado de uno de los 16 perfiles obtenidos de simular la primera etapa, elegido de manera aleatoria.

## Configuración del AR para el primer método

La información meteorológica de la simulación depende del perfil seleccionado como condición inicial, el resto del diseño corresponde a la definición de los espacios de acción y observación, y la definición de los parámetros que configuran al agente, su acción de control y la función de recompensa. En la Tabla 8.2 se muestran los parámetros que definen al entorno para la simulación con *Gym* para el método basado en la medición de a los 30 [cm]:

| Parámetro               | Valor                                | Descripción                             |
|-------------------------|--------------------------------------|---|
| control                 | DQN-TD3-PPO/2                        | Algoritmo elegido para el entrenamiento |
| scale                   | 0.8                                  | Escalamiento de la lluvia               |
| rwd                     | 1, 2 o 3                             | Función de costos elegida               |
| c1                      | 1                                    | Costo por cantidad de riego             |
| c2                      | 3                                    | Costo operacional por regar             |
| c3                      | 100                                  | Costo para el estado hídrico            |
| action_space (discreto) | 5                                    | N° de acciones posibles con DQN y PPO2  |
| action_space (continuo) | $[-1,1] \rightarrow [0, RCC]$        | Intervalo de acción con TD3 y PPO       |
| observation_space       | $[\text{inf}, \text{sup}] \times 16$ | Límites de las 16 observaciones         |

Tabla 8.3: Parámetros y características para el ajuste de la simulación con el entorno *Gym*, para el primer método.

La configuración de estos parámetros establece la estructura de los elementos del aprendizaje reforzado para el primer método, definiendo: el entorno, el sistema de control y el agente según lo indicado en la Sección 7.3. Los elementos que no poseen un valor fijo (control y rwd) establecen el número de experimentos, puesto que con cada uno de los 4 algoritmos (DQN, TD3, PPO continuo (PPO) y discreto (PPO2)) se entrena al agente con cada una de las 3 funciones de costos definidas, siendo un total de 12 experimentos a los cuales definir su entrenamiento.

## Configuración del AR para el segundo método

En la aplicación del segundo método para el estado hídrico se restringen los algoritmos y las recompensas utilizadas, en base a los resultados del AR en el primer método. Por otro lado, la utilización de 4 mediciones del contenido de agua cambia el espacio de observaciones, luego, las configuraciones del AR para este método se presentan en la Tabla 8.4.

Para este método se utiliza únicamente PPO como algoritmo, en sus versiones continua y discreta. El agente es entrenado en base a las funciones de costos 2 y 3 para el estado hídrico. Lo que da un total de 4 experimentos en los que realizar el entrenamiento.

| Parámetro               | Valor                                | Descripción                             |
|-------------------------|--------------------------------------|---|
| control                 | PPO - PPO2                           | Algoritmo elegido para el entrenamiento |
| scale                   | 0.8                                  | Escalamiento de la lluvia               |
| rwd                     | 2 o 3                                | Función de costos elegida               |
| c1                      | 2                                    | Costo por cantidad de riego             |
| c2                      | 5                                    | Costo operacional por regar             |
| c3                      | 10                                   | Costo para el estado hídrico            |
| action_space (discreto) | 5                                    | N° de acciones posibles para PPO2       |
| action_space (continuo) | $[-1,1] \rightarrow [0, RCC]$        | Intervalo de acción para PPO            |
| observation_space       | $[\text{inf}, \text{sup}] \times 19$ | Límites de las 19 observaciones         |

Tabla 8.4: Parámetros y características para el ajuste de la simulación con el entorno *Gym*, para el segundo método.

### 8.1.3. Configuración del entrenamiento

El entrenamiento para los experimentos difiere principalmente del algoritmo utilizado, ya que el diseño de la arquitectura depende de la estructura que tenga el algoritmo, además de que cada método posee sus propios parámetros que ajustar.

#### Configuración del entrenamiento para el primer método

En el caso de DQN y PPO, se utiliza una arquitectura *fully-connected* que separa *value network* y *policy network*, asignando 2 capas ocultas, 64 neuronas y función de activación *tanh* para cada red. En el caso de TD3, la arquitectura es similar pero la separación es de las redes de actor y crítico, manteniendo las 2 capas ocultas, las 64 neuronas y *tanh* como función de activación.

Además de las arquitecturas de red, los algoritmos tienen otros parámetros que pueden ser configurados según como afecten al entrenamiento, en PPO se definió una tasa de aprendizaje de 0.0001, en DQN se determina un 50 % del tiempo de entrenamiento para la exploración y en TD3 se agrega ruido a la acción para evitar el sobreajuste, se define una tasa de aprendizaje de 0.0003 y se define una exploración de acciones aleatorias por 50000 *timesteps*. Las tasas de aprendizaje son menores a las inicialmente definidas debido a que PPO y TD3 presentaban una convergencia demasiado rápida en sus entrenamientos.

Lo anterior, se asocia con la última variable a definir, el número de pasos (*timesteps*) con los que se entrenarán los modelos, en general, este número es del orden del millón, pero en las primeras pruebas se hizo evidente que no era factible utilizar esa cantidad. Definiéndose 200000 pasos para TD3 y PPO y 500000 para DQN, la decisión tras esta definición se discutirá más adelante.

La arquitectura y los parámetros deben ser ajustados realizando múltiples entrenamientos de prueba, sin embargo, debido a los largos tiempos de entrenamiento y constantes ajustes al modelo de simulación, no se realizó un número adecuado de pruebas, utilizándose arquitecturas vistas en otros trabajos y no unas particulares para el problema en cuestión.

## Configuración del entrenamiento para el segundo método

Para el segundo método sólo se debe realizar una configuración para PPO, puesto que es el único método utilizado para este entrenamiento, luego, se mantiene una arquitectura *fully-connected* que separa *value network* y *policy network*, asignando 2 capas ocultas, 64 neuronas y función de activación *tanh* para cada red.

El principal cambio es el aumento a 500.000 los *timesteps* puesto que la duración de la temporada para el modelo disminuye y con ella los días no controlados (debido a la condición de no regar antes de 5 días desde el último riego) que son la principal causa de demora en el entrenamiento, pues cada uno de esos día no se cuenta como *timesteps*. Adicionalmente, la utilización de un mayor número de variables para la definición del estado hídrico requiere un mayor tiempo de entrenamiento, por lo que también se aumenta la tasa de aprendizaje a su valor predefinido en 0.0003.

### 8.1.4. Estrategias de riego convencional

Las estrategias de riego convencional establecen los puntos de comparación con las planificaciones de riego generadas con aprendizaje reforzado. El criterio de riego que siguen las estrategias es el mismo, los riegos están distanciados por al menos 5 días y se rega sólo si se llega al umbral de estrés hídrico.

La diferencia es la cantidad de agua suministrada al momento del riego, la primera técnica se basa en el riego a capacidad de campo, con variaciones en el porcentaje de esta cantidad. La segunda se basa en suministrar cantidades fijas de agua al llegar a estrés hídrico, se prueba con diferentes valores.

Cuando el estado hídrico es definido en base a 4 mediciones, el modelo se asemeja al cultivo real regado por la Viña, es por esto, que se añade la comparación con los riegos realizados en la Viña entre los años 2018 y 2021.

## 8.2. Resultados del experimento

Con las configuraciones y parámetros definidos para los experimentos a realizar, se procede a la presentación de los resultados para su posterior análisis.

### 8.2.1. Simulación *Offline* de la primera etapa

Mediante la utilización del modelo generado en *Phydrus* para la simulación de la primera etapa, se busca generar los perfiles de suelo iniciales para la etapa con riegos, adicionalmente se puede evaluar el comportamiento del contenido de agua cuando no se aplican riegos, y la única entrada de agua es por medio de precipitaciones.

## Primera etapa con medición del contenido de agua a los 30 [cm]

En las Figuras 8.1 se aprecia la evolución del contenido de agua a los 30 [cm] para los 16 perfiles generados por la simulación, estos gráficos se pueden complementar con la dinámica del perfil completo, la cual fue calculada para los datos de las 4 temporadas, sin combinaciones, cuyos resultados se puede ver en la Sección E.2 del anexo.

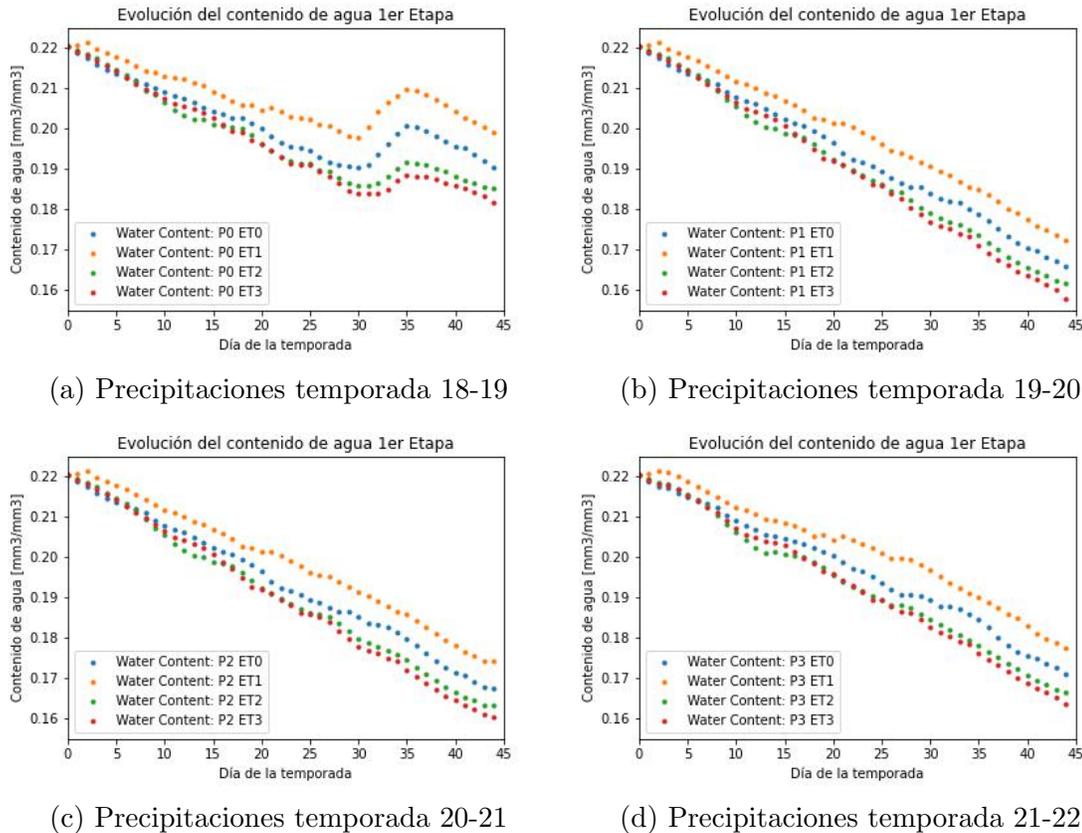


Figura 8.1: Evolución del contenido de agua en la primera etapa, para diferentes datos meteorológicos

Las Figuras 8.1 muestran el contenido de agua para cada temporada de precipitaciones, con los diferentes datos de evapotranspiración de referencia. Se puede notar que el contenido de agua disminuye a medida que las temporadas de  $ET_o$  se acercan al presente, lo que se puede asociar al efecto del cambio climático, principalmente en el aumento de las temperaturas, uno de los factores que determina el valor de la  $ET_o$ .

## Primera etapa con mediciones del contenido de agua a los 20, 40, 60 y 80 [cm]

En las Figuras 8.2 se aprecia la evolución del contenido de agua en los 4 puntos del perfil elegidos para definir el estado hídrico. En lugar de revisar las 16 combinaciones que dan lugar a los perfiles de inicio de la segunda etapa, se aprecia la evolución del contenido de agua para cada temporada de datos meteorológicos disponible, a modo de complemento en las Figuras E.7 del anexo se aprecia la evolución del contenido de agua en todo el perfil.

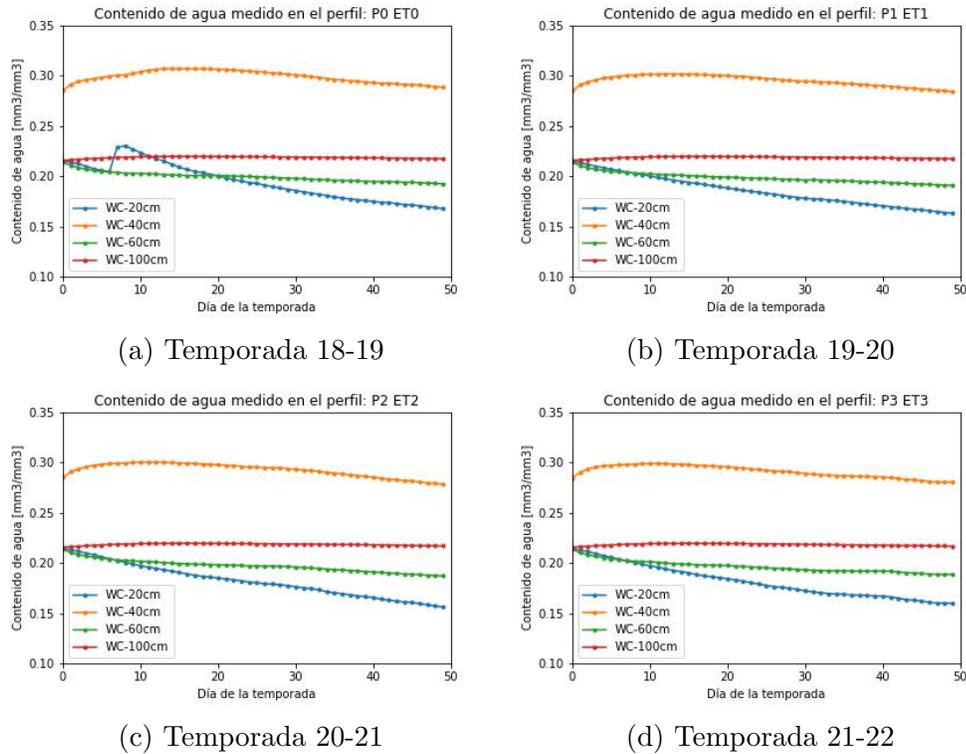


Figura 8.2: Simulaciones de la etapa 1 para cada temporada en *Phydrus*, con el método con 4 mediciones.

En la evolución del contenido de agua visto en las Figuras 8.2a se pueden apreciar la presencia de una precipitación al comienzo de la etapa, que genera un aumento en el contenido de agua a los 20 [cm], pero que es apenas apreciado a los 40 [cm].

También se puede notar que el contenido de agua en las zonas más profundas disminuye lentamente, y el cambio más notorio ocurre a los 20 [cm] que es el más cercano a la superficie y donde existe una mayor concentración de raíces.

## 8.2.2. Entrenamiento del sistema de control para la segunda etapa

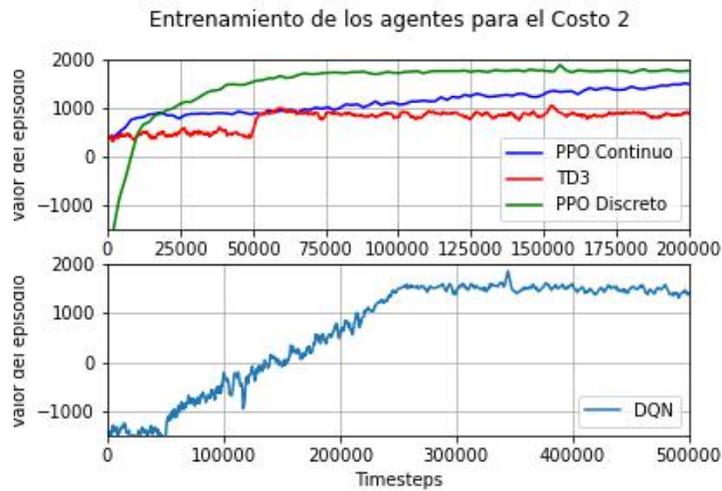
Utilizando la estructura de aprendizaje del entorno *Gym* configurado, se realizan los experimentos para las diferentes recompensas con los distintos algoritmos de aprendizaje reforzado.

### Entrenamiento de los agentes para el primer método

En las Figuras 8.3 se aprecian las curvas de entrenamiento de los diferentes algoritmos para cada una de las funciones de costos planteadas, debido al mayor número de pasos necesarios para el entrenamiento de DQN, el aprendizaje de este se expone en un gráfico diferente de TD3 y PPO.



(a) Entrenamiento del agente para el costo 1.



(b) Entrenamiento del agente para el costo 2.



(c) Entrenamiento del agente para el costo 3.

Figura 8.3: Curvas de aprendizaje del agente para los algoritmos utilizados, según la función de recompensa utilizada.

A partir de estos gráficos se pueden observar ciertas características de las curvas de aprendizaje para los diferentes algoritmos, las cuales se asocian directamente a sus configuraciones.

En PPO, se aprecia una rápida convergencia de la curva, el caso discreto es ligeramente más lento pero alcanza valores de recompensa mayores, el caso continuo presenta una leve pero constante mejora a lo largo del entrenamiento, pudiendo beneficiarse de una mayor cantidad de pasos.

En TD3 se aprecia claramente la ejecución de acciones aleatorias durante los primeros 50000 pasos, luego de eso se aprecia una rápida convergencia, con oscilaciones producto del ruido en las acciones implementado.

En DQN se aprecia un aprendizaje más lento que en los otros casos, lo que justifica la utilización de más *timesteps*. Adicionalmente, se puede apreciar el cambio en el aprendizaje cuando termina la fase de exploración (250000 pasos), en la Figura 8.3a se genera una evidente mejora en el desempeño, caso para el cual hubieran sido útiles unas cuantas iteraciones extras. Por otro lado, en las Figuras 8.3b y 8.3c se ve que luego de la exploración la curva comienza a converger.

Adicionalmente, a las curvas se presenta una tabla resumen de los tiempos de entrenamiento del agente en cada uno de los experimentos.

|                          | <b>Algoritmos</b>   |            |            |                     |
|--------------------------|---------------------|------------|------------|---------------------|
| <b>Función de costos</b> | <b>PPO continuo</b> | <b>DQN</b> | <b>TD3</b> | <b>PPO discreto</b> |
| C1. Constante            | 36 horas            | 48 horas   | 34 horas   | 29 horas            |
| C2. Lineal               | 32 horas            | 77 horas   | 32 horas   | 31 horas            |
| C3. Cuadrático           | 30 horas            | 73 horas   | 33 horas   | 32 horas            |

Tabla 8.5: Tiempo de entrenamiento del agente para los diferentes experimentos realizados.

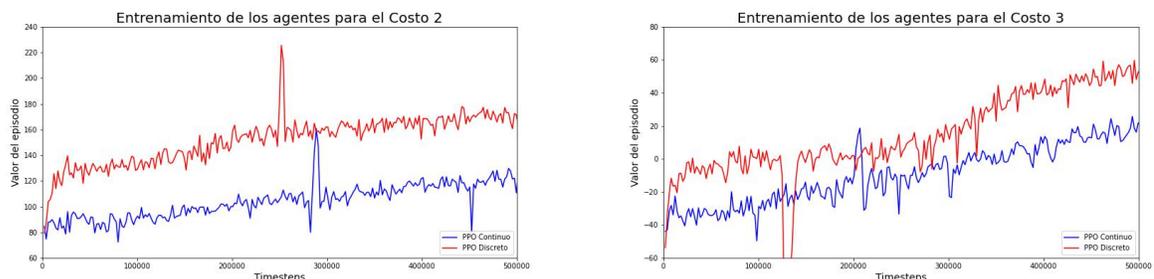
Los resultados vistos en la Tabla 8.5 muestran una relación con el número de pasos realizados, los entrenamientos con TD3 y PPO poseen tiempos similares, mientras que DQN que hace 2.5 veces las iteraciones de los otros algoritmos, sigue esa tendencia para los costos C2 y C3, pero no para C1.

Los tiempos de ejecución dependen de dos factores, la utilización de la memoria del servidor donde se realizan las simulaciones, dado que los experimentos se ejecutan simultáneamente, en ocasiones se saturaba la memoria, evitando la ejecución de los pasos, retrasando el entrenamiento.

El otro es la configuración de la simulación en *Gym*. En el entrenamiento cada paso representa una acción de riego, pero las decisiones de riego no son tomadas cada día, luego en un episodio o temporada el número de acciones tomadas es variable, de este modo entre menos pasos tenga un episodio, más episodios y más simulaciones de *Phydrus* sin riego, luego un mayor tiempo de entrenamiento.

## Entrenamiento de los agentes para el segundo método

En las Figuras 8.4 se aprecian las curvas de entrenamientos para los agentes en base a PPO continuo y PPO discreto.



(a) Entrenamiento de los algoritmos PPO para el costo 2. (b) Entrenamiento de los algoritmos PPO para el costo 3.

Figura 8.4: Curvas de aprendizaje de los agente para los algoritmos PPO, según la función de recompensa utilizada.

El entrenamiento de los algoritmos PPO cambia en lo visto para el primer método en las Figuras 8.3, se puede ver en las Figuras 8.4 que la curva de aprendizaje presenta oscilaciones y un aumento de la recompensa constante, sin lograr una convergencia en la cantidad de *timesteps* utilizados.

Respecto de los tiempos de entrenamiento para cada caso, se presenta a continuación en la Tabla 8.6:

| Función de costos | Algoritmos   |              |
|-------------------|--------------|--------------|
|                   | PPO continuo | PPO discreto |
| C2. Lineal        | 64 horas     | 49 horas     |
| C3. Cuadrático    | 73 horas     | 60 horas     |

Tabla 8.6: Tiempo de entrenamiento del agente para los algoritmos PPO.

Los tiempos de ejecución vistos en la Tabla 8.6 se encuentran dentro de lo esperado para la cantidad de *timesteps*. Al observar los gráficos se aprecia que más entrenamiento beneficiaría el aumento en las recompensas, para esto, en lugar de aumentar el número de *timesteps* y volver a entrenar, antes de obtener las programaciones de riego se vuelven a entrenar los agentes por 100.000 *timesteps* más, lo que permite mejorar el desempeño del agente sin la necesidad de realizar un ciclo de entrenamiento completamente nuevo.

### 8.2.3. Programaciones de riego con AR para el primer método

Utilizando los sistemas de control entrenados a partir de diferentes recompensas y según los distintos algoritmos elegidos, se procede a generar las planificaciones de riego y obtener métricas que ayuden a determinar su desempeño. Para facilitar la visualización de los resultados, se ajusta el eje para que empiece justo ligeramente por debajo del umbral de estrés hídrico.

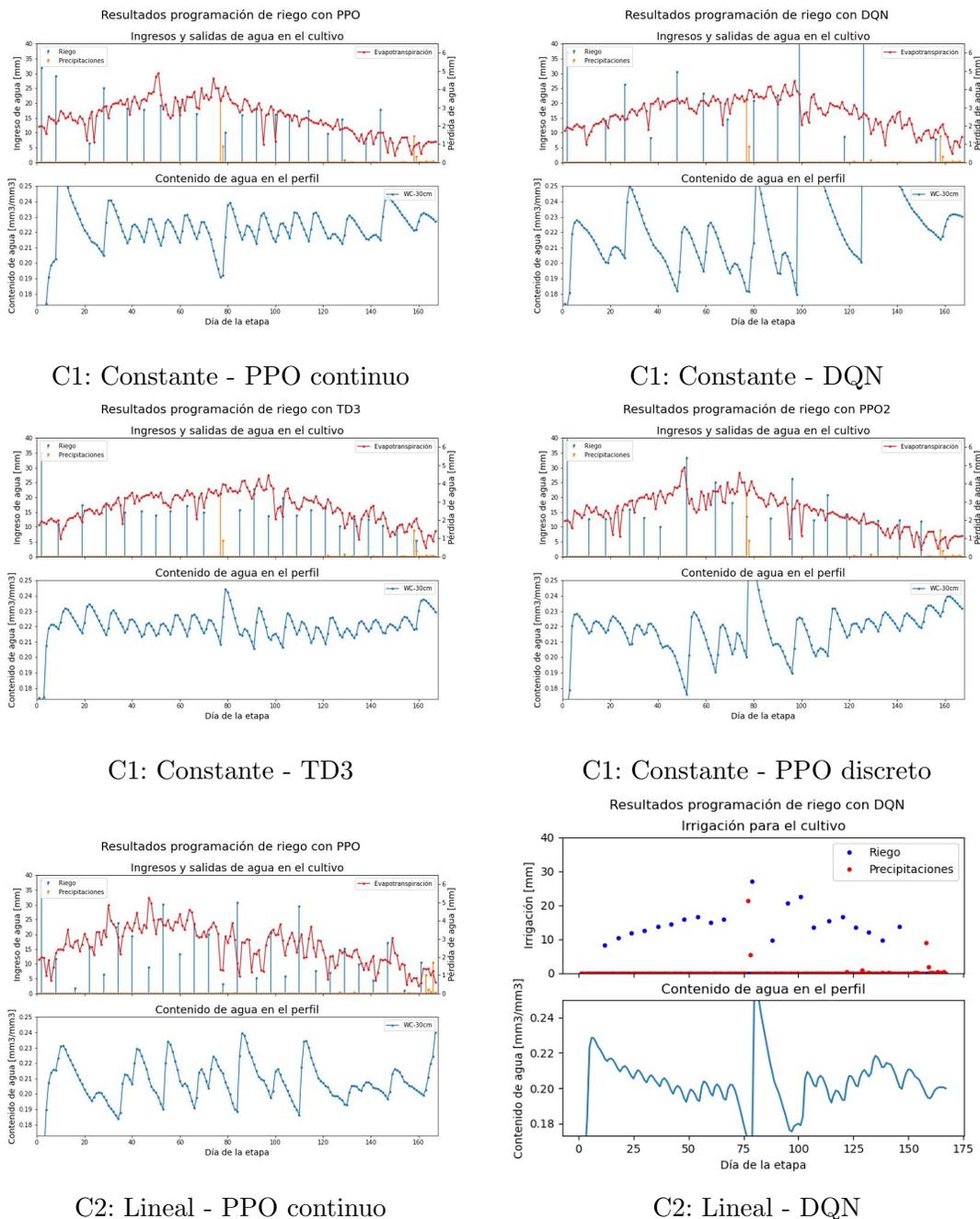


Figura 8.5: Programaciones de riego para la segunda etapa, con los agentes entrenados (1/2).

En las Figuras 8.5 y 8.6 se muestran algunas planificaciones de riego obtenidas, donde las cantidades de riego están definidas por las acciones tomadas por el controlador, las secuencias de control asociadas se pueden ver en las Figuras E.8 y E.9 del anexo.

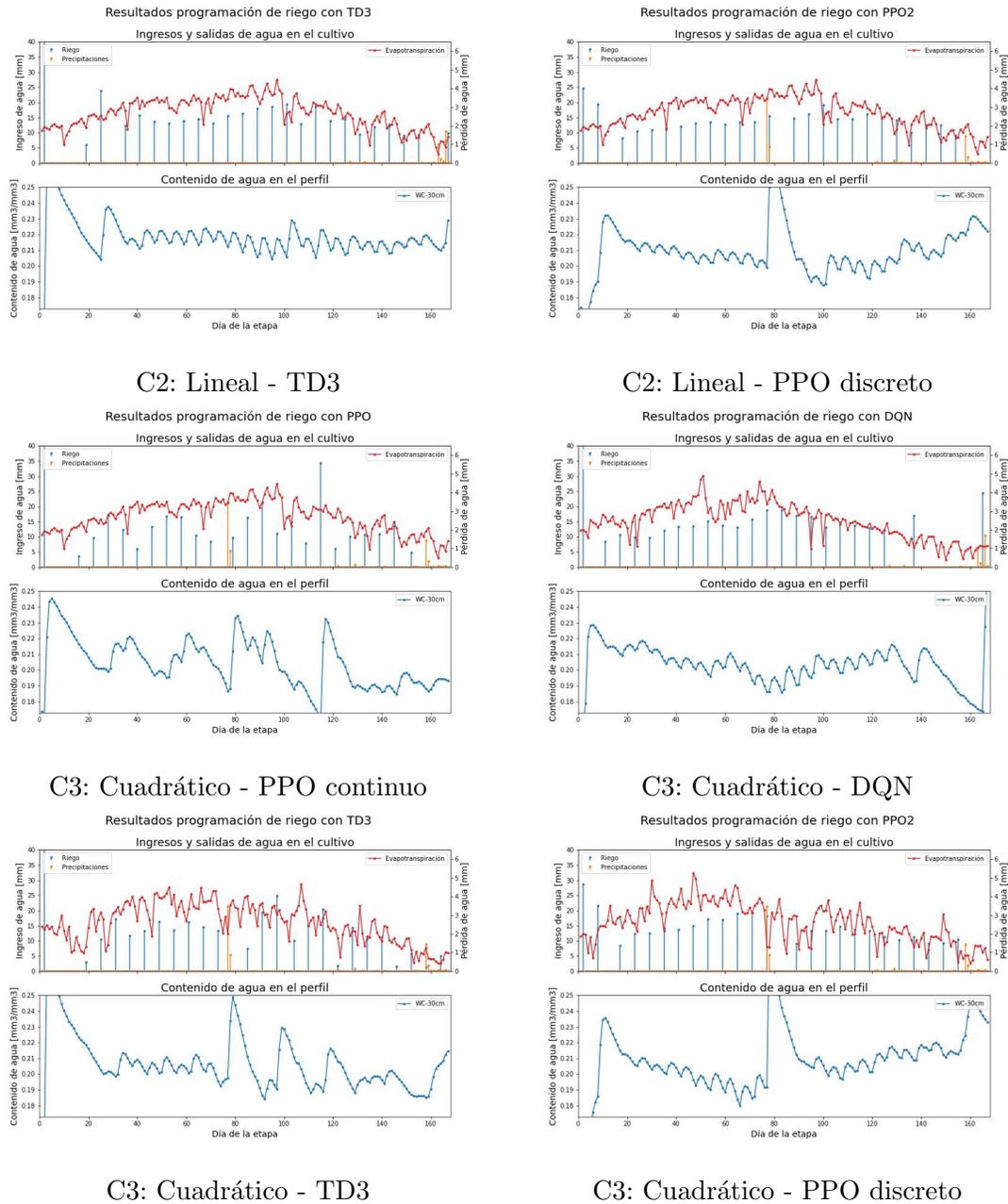


Figura 8.6: Programaciones de riego para la segunda etapa, con los agentes entrenados (2/2).

Se pueden notar que los contenidos de agua iniciales obtenidos en la primera etapa, son considerablemente menores que el rango de “operación” deseado lo que resulta en la necesidad de aplicar una cantidad considerable de agua al comienzo de la etapa, un riego que supera los 40 [mm] en la mayoría de los casos.

La presencia de lluvias en algunas temporadas permite estudiar la adaptabilidad de los controladores ante ellas, pues se observa que el contenido de agua medido sube rápidamente,

lo que impide la actuación del sistema de control hasta retomar valores normales.

Respecto de las funciones de costos, se puede ver que salvo por DQN, los controladores para C1 poseen un comportamiento relativamente estable en torno a capacidad de campo. En el caso de C2, se aprecia una tendencia por la aplicación de pequeños riegos, que genera ligeras oscilaciones en torno a FC, excepto para PPO continuo, que presenta un comportamiento cercano al esperado. Finalmente, para el caso de C3 no es posible distinguir ningún comportamiento particular.

La visualización del contenido de agua en el perfil, no es suficiente para evaluar el desempeño de los controladores, es por esto que se hace necesaria la generación de métricas que complementen la información de los gráficos. La Tabla 8.7 muestra la cantidad de agua usada para riego, el número de acciones de riego, la cantidad de agua promedio de cada riego y el promedio de la distancia absoluta del contenido de agua diario a la zona de estrés hídrico.

|       |           | Información de la programación de riego |           |              |                |
|-------|-----------|---|-----------|--------------|----------------|
| Costo | Algoritmo | R. total [mm]                           | N° riegos | R. prom [mm] | Dist. prom. EH |
| C1    | PPO       | 386.36                                  | 20        | 19.32        | 0.050          |
|       | DQN       | 370.95                                  | 14        | 26.50        | 0.030          |
|       | TD3       | 378.94                                  | 23        | 16.48        | 0.046          |
|       | PPO2      | 379.83                                  | 20        | 18.99        | 0.043          |
| C2    | PPO       | 376.29                                  | 25        | 15.05        | 0.036          |
|       | DQN       | 351.58                                  | 22        | 15.98        | 0.031          |
|       | TD3       | 395.49                                  | 24        | 16.48        | 0.045          |
|       | PPO2      | 351.00                                  | 25        | 14.04        | 0.036          |
| C3    | PPO       | 357.73                                  | 24        | 14.91        | 0.030          |
|       | DQN       | 345.42                                  | 22        | 15.7         | 0.028          |
|       | TD3       | 369.37                                  | 24        | 15.39        | 0.036          |
|       | PPO2      | 384.03                                  | 22        | 17.46        | 0.044          |

Tabla 8.7: Resumen del riego aplicado en cada programación, y valor promedio de la diferencia diaria entre el contenido de agua y el umbral de estrés hídrico.

Considerando que la distancia en contenido de agua entre capacidad de campo y el umbral de estrés hídrico es de 0.038 y las curvas vistas en los gráficos, la distancia promedio a EH muestra la poca tendencia de la planificación de riego de acercar el cultivo a estrés hídrico.

Se puede notar que los riegos totales de los diferentes controladores se encuentran entre los 350 y 400 milímetros. Por otro lado, los controladores deciden regar entre 20 y 25 veces por temporada, lo que se puede traducir a riegos separados entre 6 a 9 días. El riego promedio para los experimentos permite establecer un buen punto de comparación con una estrategia convencional que suministre cantidades fijas de agua de 15 o 20 milímetros.

Inicialmente para la generación de las programaciones se agregarían los mismos experimentos pero para una distribución homogénea de las raíces, sin embargo, debido a los tiempos de entrenamiento sólo se alcanzaron a implementar los sistemas de control con TD3 y PPO para la primera forma de la recompensa, los resultados se pueden ver en las Figuras E.11 y en la Tabla E.1.

## 8.2.4. Programaciones de riego con AR para el segundo método

Las programaciones de riego para el método con un estado hídrico definido por el contenido de agua a los 20, 40, 60 y 100 [cm], se obtienen entrenando agentes con PPO continuo y PPO discreto a partir de las funciones de costos por estado hídrico 2 y 3.

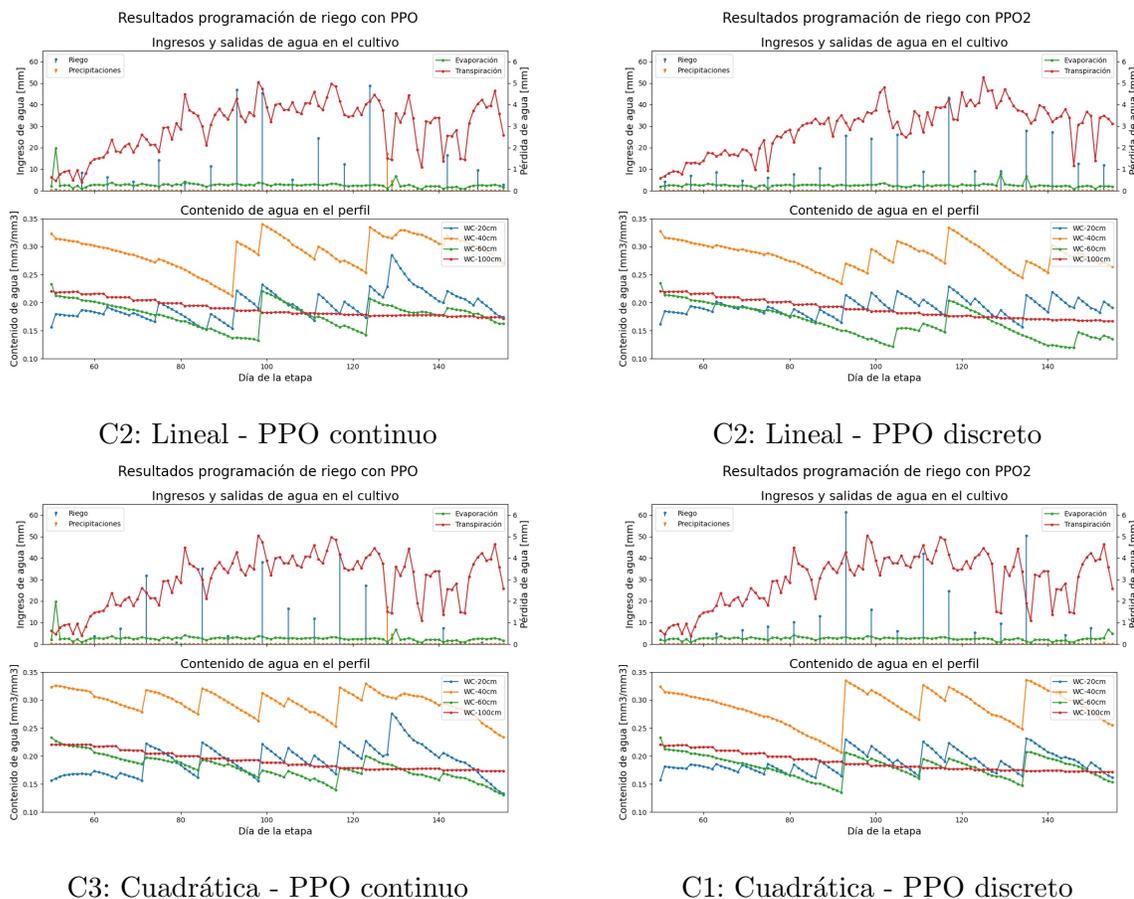


Figura 8.7: Programaciones de riego para la segunda etapa, con los agentes entrenados en el segundo método.

En las Figuras 8.7 se observan las programaciones de riego obtenidas, graficando comportamiento de cada una de las mediciones de contenido de agua y la información meteorológica para la temporada, donde además de las precipitaciones y el riego se añade la evolución de la evaporación y transpiración. Las acciones tomadas por los controladores que definen la cantidad de agua aplicada se pueden ver en las Figuras E.10 del Anexo E.

Se puede apreciar en las programaciones de riego obtenidas que los controladores aplican riegos pequeños al comenzar la segunda etapa, estos evitan que a los 20 [cm] se llegue a estrés hídrico, pero el agua no profundiza más abajo, pues el contenido de agua a los 40 y 60 [cm] no aumenta con bajas cantidades de agua. Para que el agua profundice en el perfil, el tamaño de los riegos debe aumentar, esto ocurre cuando el estado hídrico en los 40 y 60 [cm] se acerca al punto de estrés. Por otro lado, a los 100 [cm] no se aprecian aumentos en el contenido de agua, lo que es positivo pues no llega agua a un lugar donde no hay raíces, de modo que no se desperdicia el agua.

En las programaciones de PPO continuo de las Figuras 8.7 se puede ver el efecto de la precipitación en la temporada, en primer lugar, agua ingresada se refleja principalmente a los 20 [cm], afectando ligeramente a las mediciones más profundas. Esto se debe a que las precipitaciones se distribuyen a lo largo del día, lo que dificulta su concentración y profundización en el perfil. En cuanto, a la respuesta del modelo ante la precipitación, se observa que el próximo riego se ve aplazado producto de la lluvia.

Al comparar las programaciones según la recompensa utilizada, se aprecia que al utilizar C3 el controlador evita de mejor manera el estrés hídrico a los 60 [cm], manteniendo un buen control en el contenido de agua. En el caso de C2, se puede considerar que espera hasta que los sensores a mayor profundidad detecten el estrés hídrico para aplicar el riego.

Respecto de la nueva configuración del modelo, se puede apreciar el crecimiento de la transpiración a medida que se desarrolla el cultivo. Asociado a esto, se aprecia que la disminución en el contenido de agua es más rápida en las zonas más cercanas a la superficie debido a la mayor densidad de raíces.

Para la correcta evaluación de las programaciones se utilizan las métricas discutidas en la Sección 7.3.4, donde se incluye la reposición de  $ET_o$  en reemplazo de la distancia promedio al EH. La Tabla 8.4 resumen la información del riego de las programaciones.

|              |                  | <b>Información de la programación de riego</b> |                  |                     |                                      |
|--------------|------------------|--|------------------|---------------------|--------------------------------------|
| <b>Costo</b> | <b>Algoritmo</b> | <b>R. total [mm]</b>                           | <b>N° riegos</b> | <b>R. prom [mm]</b> | <b>Reposición. <math>ET_o</math></b> |
| <b>C2</b>    | <b>PPO</b>       | 260.3474                                       | 16               | 16.2717             | 0.4993                               |
|              | <b>PPO2</b>      | 273.6833                                       | 18               | 15.2046             | 0.5380                               |
| <b>C3</b>    | <b>PPO</b>       | 225.6726                                       | 13               | 17.3594             | 0.4328                               |
|              | <b>PPO2</b>      | 277.7231                                       | 17               | 16.3367             | 0.5326                               |

Tabla 8.8: Resumen del riego aplicado en cada programación, y valor de reposición de  $ET_o$ .

En primer lugar, la reposición de  $ET_o$  se acerca al valor de  $K_c$  cuando el riego satisface correctamente el requerimiento hídrico del cultivo. Se puede observar que la reposición lograda por las programaciones de riego no está lejos de 0.6 y 0.8 que son los valores de  $K_{cb}$  utilizados. Esto puede indicar que se está regando menos de lo necesario, sin embargo, los visto en las Figuras 8.7 muestra que no se alcanza estrés hídrico en el perfil de suelo, luego la otra alternativa es que se estaba subestimando la evapotranspiración, es decir, se pierde menos agua de la que indica la  $ET_o$ .

Por otro lado, se aprecia que el riego total se encuentra en un rango entre 200 y 300 milímetros. El número de riegos realizados en el periodo de 105 días de la etapa, muestra que el controlador aplica en promedio un riego en un rango de 6 a 9 días, al igual que para el primer método.

## 8.2.5. Programaciones de riego con estrategias convencionales para el primer método

Para evaluar el desempeño de las planificaciones de riego se genera una base con 8 estrategias convencionales, basadas en cantidades fijas y porcentajes del llenado del “estanque” a capacidad.

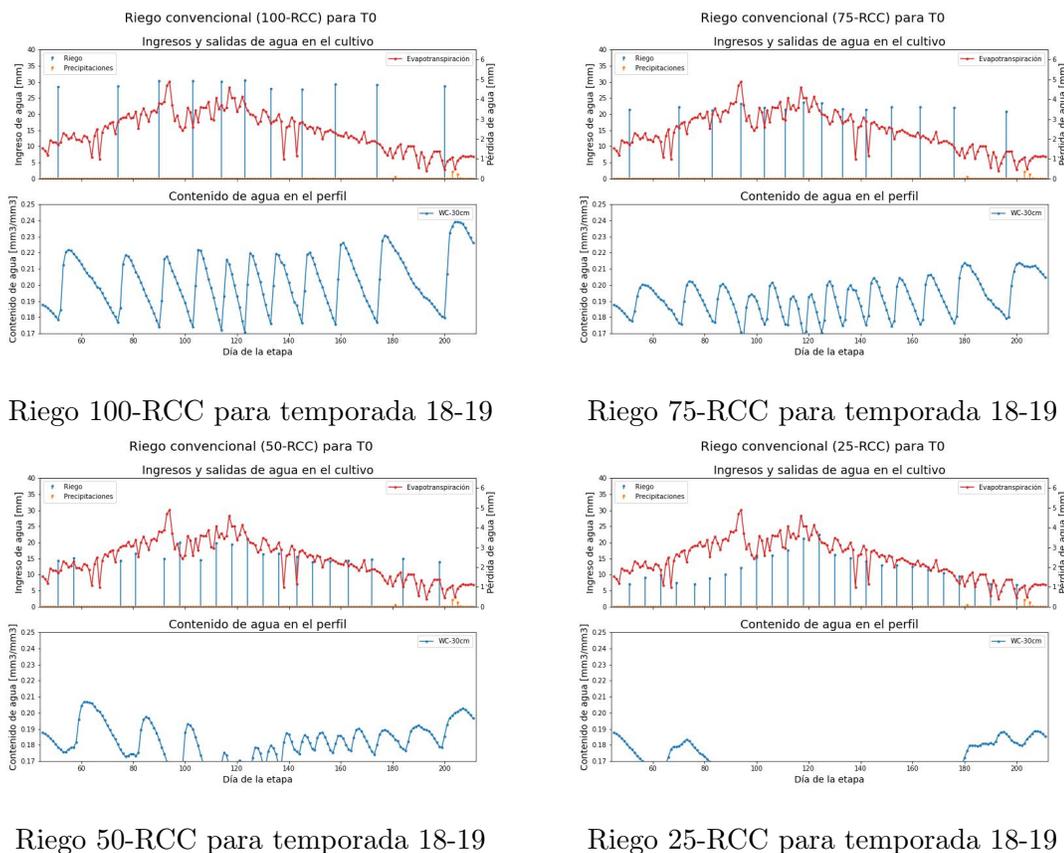
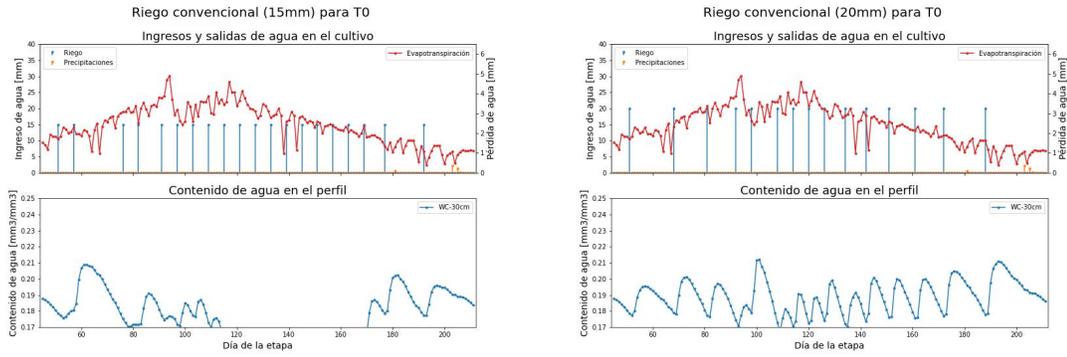


Figura 8.8: Riego según capacidad de campo para la segunda etapa, en la temporada 18-19.

Para la comparación con las planificaciones de riego obtenidas con AR, la escala de los ejes se ajusta al rango de contenido de agua de interés (estrés hídrico y contenido de agua). Adicionalmente en la Sección E.4 del anexo, se presentan la aplicación para las temporadas del año 2019 al 2022.

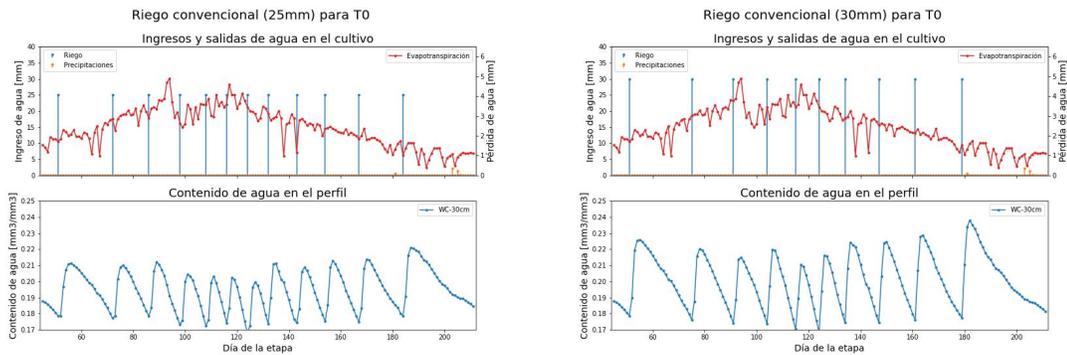
De las estrategias basadas en riego a capacidad de campo, los riegos al 100 % y 75 % son capaces de mantener los niveles de contenido de agua dentro de los límites deseados, mientras que los riegos al 50 % y 25 % someten a un constante estrés hídrico a la planta, lo que pondría en riesgo su desarrollo.

Por otro lado, en el caso de los riegos de cantidad fija, el único caso que no logra mantener un buen estado hídrico es el de 15 [mm]. Las otras estrategias presentan ligeras diferencias en la cantidad de agua disponible en el perfil, pero todas pueden ser estrategias válidas de riego bajo los supuestos planteados.



Riego de 15 [mm] para temporada 18-19

Riego de 20 [mm] para temporada 18-19



Riego de 25 [mm] para temporada 18-19

Riego de 30 [mm] para temporada 18-19

Figura 8.9: Riego de cantidad fija para la segunda etapa, en la temporada 18-19.

Si se revisan los resultados obtenidos para otras temporadas, ver Figuras E.14, se puede observar, que esta estrategia de riego convencional no posee la capacidad de predecir la lluvia, lo que ocasiona un gran efecto de estas en el desarrollo de la temporada.

Por otro lado, se puede ver el efecto del cambio climático al evaluar las temporadas más cercanas a la actualidad, las que como se ve en las Figuras 8.1a existe un aumento en la evapotranspiración de referencia, lo que genera la necesidad de mayor agua en la mantención de los niveles de contenido de agua.

Al igual que para las programaciones de riego con AR, se complementa la información gráfica con las métricas que resumen las características de la planificación implementada respecto del uso de agua y el sometimiento a estrés hídrico.

|                        | <b>Información de riego convencional (Temporada 18-19)</b> |                  |                    |                       |
|------------------------|--|------------------|--------------------|-----------------------|
| <b>R. Convencional</b> | <b>R. total[mm]</b>  | <b>N° riegos</b> | <b>R. prom[mm]</b> | <b>Dist. prom. EH</b> |
| <b>100-RCC</b>         | 320.976  | 11               | 29.179             | 0.029                 |
| <b>75-RCC</b>          | 308.940  | 14               | 22.024             | 0.018                 |
| <b>50-RCC</b>          | 304.020  | 19               | 16.001             | 0.011                 |
| <b>25-RCC</b>          | 300.636  | 25               | 12.025             | 0.023                 |
| <b>30mm</b>            | 300  | 10               | 30                 | 0.027                 |
| <b>25mm</b>            | 300  | 12               | 25                 | 0.021                 |
| <b>20mm</b>            | 300  | 15               | 20                 | 0.016                 |
| <b>15mm</b>            | 300  | 20               | 15                 | 0.027                 |

Tabla 8.9: Resumen del riego aplicado para cada estrategia convencional, y valor promedio de la diferencia diaria entre el contenido de agua y el umbral de estrés hídrico, para la temporada 18-19.

Revisando la Tabla 8.9, se puede ver que ambos métodos son bastante similares en cuanto al agua total utilizada y al número de riegos, sin embargo, se puede notar que el riego a capacidad de campo es ligeramente más eficiente.

En cuanto al sometimiento a estrés hídrico, las estrategias que gráficamente se muestran válidas para el manejo de riego poseen valores similares de distancia al umbral de estrés hídrico, siendo ligeramente mejores para los métodos RCC.

Se puede apreciar en los gráficos y en las tablas que los riegos convencionales tienen un mejor desempeño en el riego del cultivo, son más eficientes en el uso de agua y cumplen mejor los requerimientos establecidos.

### **8.2.6. Programaciones de riego con estrategias convencionales para el segundo método**

Al revisar el desempeño de las estrategias convencionales para el primer método y la opción de realizar comparaciones con los riegos reales aplicados por la Viña, se reduce el número de riegos convencionales aplicados, dejando aquellos que suministran una mayor cantidad de agua una vez alcanzado el EH a los 20 [cm]. Las programaciones de riego presentadas a continuación son realizadas con datos meteorológicos de la primera temporada, el resto pueden ser estudiadas en el Anexo E.4.

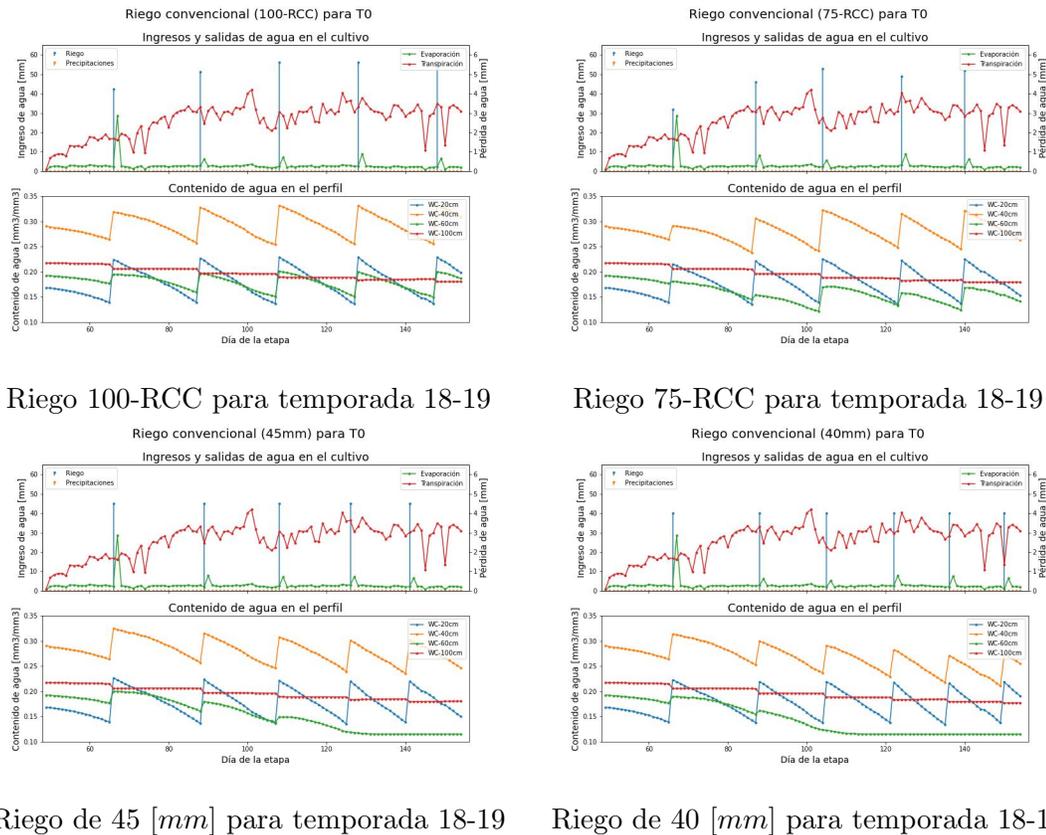


Figura 8.10: Estrategias de riego convencional para la segunda etapa, en la temporada 18-19 para el método 2.

El comportamiento del contenido de agua a los 20 [cm] observado en las Figuras 8.10, muestra que se alcanza el estrés hídrico 5 veces, lo que implica regar cada 21 días aproximadamente. Esta demora en alcanzar un estado de estrés hídrico, puede indicar que se subestima la evapotranspiración o la necesidad de un ajuste en el valor del umbral de estrés hídrico.

Al esperar tanto tiempo entre cada riego, la cantidad de agua aplicada es alta, lo que permite que esta profundice en el perfil y sus efectos se vean en las mediciones de 40 y 60 [cm]. De las 4 programaciones estudiadas, el riego de 40 [mm] es el único que termina por estresar el cultivo en alguno en alguna profundidad.

Estos riegos se ejecutan cuando se llega a EH a los 20 [cm] y dado que la extracción de agua a mayor profundidad es más lenta, al llegar el momento de regar a los 40 [cm] aún no se llega a EH cuando se repone el agua, lo que genera una ligera acumulación de agua en esa zona entre cada riego, lo cual muestra lo ineficiente que pueden ser estos métodos en el uso de agua.

A diferencia de lo visto con las estrategias de riego convencional para el método 1, las programaciones de riego obtenidas se alejan de los requerimientos hídricos establecidos por la Viña puesto que el tiempo entre cada riegos es demasiado. Lo anterior, también se puede ver reflejado en las métricas, en particular en la reposición de la  $ET_o$ , que se muestra en la Tabla 8.10.

|                 | Información de riego convencional (Temporada 18-19) |           |              |                   |
|-----------------|---|-----------|--------------|-------------------|
| R. Convencional | R. total [mm]                                       | N° riegos | R. prom [mm] | Reposición $ET_o$ |
| <b>100-RCC</b>  | 262.7505  | 5         | 52.5501      | 0.5164            |
| <b>75-RCC</b>   | 231.525   | 5         | 46.305       | 0.4550            |
| <b>45mm</b>     | 225   | 5         | 45           | 0.4422            |
| <b>40mm</b>     | 200   | 5         | 40           | 0.4717            |

Tabla 8.10: Resumen del riego aplicado para cada estrategia convencional y reposición de la  $ET_o$ , para la temporada 18-19.

Debido a los pocos riegos aplicados (se riega cada 21 días aproximadamente), el riego total es similar a la cantidad obtenida con AR, por lo mismo al evaluar la reposición se puede ver que las programaciones de riego no son capaces de cubrir los requerimientos hídricos del cultivo.

### 8.2.7. Programaciones de riego aplicadas por la Viña

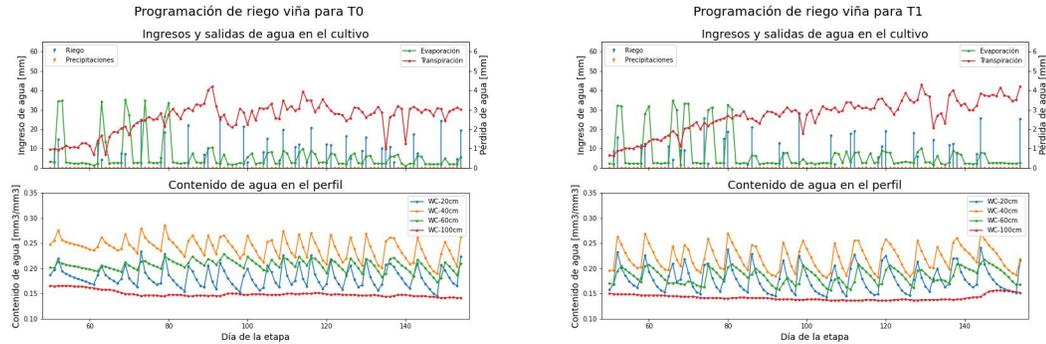
La definición del estado hídrico en el segundo método replica los sensores instalados en la Viña para el monitoreo del contenido de agua en el suelo. Luego, a partir de la información provista (estudio de suelo, información meteorológica y datos del cultivo) se configura el modelo para replicar las condiciones del cultivo real. Lo anterior, permite realizar comparaciones con las programaciones de riego aplicadas por la Viña.

Es importante notar, que las temporadas de riego realizadas por la Viña no coinciden en el día de inicio y la duración de la temporada configurada para este trabajo. Por lo que, para realizar las comparaciones se identifica el día anterior al primer riego realizado por la Viña y se toman en cuenta los siguientes 105 días replicar la duración definida para la segunda etapa. Siendo los días de inicio y término de la segunda etapa:

- Temporada *Phydrus*: Inicio 12/11 - Término 24/02
- Temporada 1 Viña: Inicio 23/11/2018 - Término: 7/03/2019
- Temporada 2 Viña: Inicio 1/12/2019 - Término: 14/03/2020
- Temporada 3 Viña: Inicio 16/11/2020 - Término: 28/02/2021

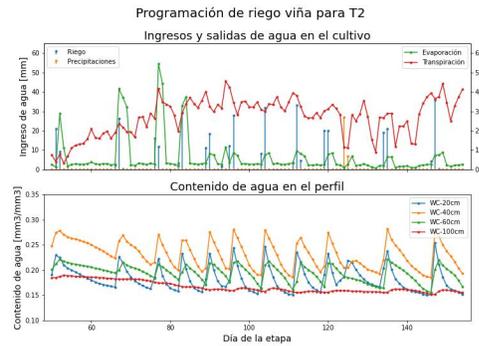
En las Figuras 8.11, se aprecia una secuencia de “vaciado y llenado” del suelo, donde el contenido de agua medido disminuye en pocos días y de forma casi homogénea para los sensores a los 20, 40 y 60 [cm], para luego volver a subir con cada riego aplicado. La dinámica que sigue el contenido de agua en las mediciones de 40 y 60 [cm] muestra que los riegos alcanzan las zonas más profundas del perfil, pero sin llegar al fondo en los 100 [cm].

Las curvas del contenido de agua muestran poca diferencia en los valores medidos por los sensores, lo que indica que pese a la existencia de diferentes tipos de suelo, las características hídricas de estos se parecen. Adicionalmente, dado que las pérdidas de agua son similares, se puede asumir que la distribución de las raíces es más homogénea que lo planteado en el modelamiento del sistema (ver Figura 7.5).



Riego en Viña para temporada 18-19

Riego en Viña para temporada 19-20



Riego en Viña para temporada 20-21

Figura 8.11: Programaciones de riego aplicadas por la Viña.

Con respecto a los riegos aplicados, en ocasiones se aprecian ingresos de agua en días consecutivos, sin embargo, esto corresponde a un único riego largo que se prolongo hasta el otro día, luego, en el conteo de riegos estos deben ser tomados como una única acción.

|                 | Información de riego aplicado en la Viña |           |              |                   |
|-----------------|--|-----------|--------------|-------------------|
| R. Convencional | R. total [mm]                            | Nº riegos | R. prom [mm] | Reposición $ET_o$ |
| Temporada 18-19 | 464.59                                   | 18        | 25.8105      | 0.9433            |
| Temporada 19-20 | 434.55                                   | 16        | 27.1593      | 0.8535            |
| Temporada 20-21 | 391.06                                   | 14        | 27.9328      | 0.7540            |

Tabla 8.11: Resumen del riego aplicado y reposición de la  $ET_o$  para 3 temporadas en la Viña.

La Tabla 8.11 muestra un resumen del riego aplicado, en primer lugar, la cantidad de riegos aplicados para un periodo de 105 días indica que estos se realizan espaciados de 6 a 9 días. Luego, se puede observar que la cantidad de agua aplicada es mucho mayor que en las programaciones de AR y estrategias convencionales.

Los valores de reposición se encuentran cercanos al rango de  $K_c (K_e + K_{cb})$  utilizado, luego los riegos aplicados logran suplir las pérdidas de agua por concepto de evapotranspiración. Sin embargo, se aprecia que en algunos casos se está regando más de lo necesario, por lo que, el método tiene aspectos de mejora.

## 8.3. Análisis de resultados

El modelo de simulación generado, el sistema de control, el entrenamiento del controlador con AR, los resultados obtenidos y la comparación con estrategias convencionales, son elementos que deben ser evaluados para analizar el desempeño del trabajo realizado.

### 8.3.1. Modelo de simulación

El modelo generado con *Phydrus*, logra integrar el software en HYDRUS-1D y ser configurado en base a la información del cultivo y parámetros agrícolas estándar. Luego, se logra simular la dinámica diaria del terreno de cultivo en base a la información meteorológica y las características del suelo y del cultivo.

Con esta implementación se pueden llevar a cabo las simulaciones de las primeras etapas, ver Figuras 8.1a y 8.2, con lo que se generan las condiciones iniciales para la siguiente fase y además sus resultados permiten ver efectos del cambio climático y su efecto en los cultivos, pues se aprecia que las temporadas actuales generan niveles de contenido de agua más bajos, puesto que aumenta la temperatura ambiente y esto a su vez aumenta el valor de evapotranspiración.

El estudio de una dinámica diaria del modelo permite una directa integración en el entorno *Gym* para su utilización en aprendizaje reforzado, dado que el objetivo del sistema de control es establecer un valor de riego diario, cuyos efectos pueden ser observados con la dinámica del entorno diseñado.

Los aspectos de mejora del modelo generado para el método 1, se centran en la utilización de más información para generar un modelo más preciso de una situación real, y permitir una comparación directa. Esto lleva a los cambios en el modelo del método 2, principalmente en la utilización de las 4 mediciones de contenido de agua que definen el estado hídrico del cultivo.

Uno de los principales cambios al modelo que beneficiaron su desempeño fue en la aplicación del riego, la cual se realizaba distribuyendo el ingreso de agua en las 24 horas del día, cuando en realidad los equipos de riego actúan en determinadas horas del día y bajo una tasa de  $\mathbf{x} \left[ \frac{mm}{hr} \right]$ , como se desconoce la tasa de los equipos en la Viña se opta por aplicar el riego en 6 horas, lo cual permitió la concentración del riego y su profundización del agua en el perfil. Este cambio podría ser extendido al resto de datos meteorológicos, de modo de concentrar lluvias, transpiración y evaporación en horas del día en específico.

Otro cambio relevante del modelo entre los métodos utilizados, fue la definición del perfil

en función de estudios de suelo e información del contenido de agua medido por los sensores de Viña. Esto permite replicar las condiciones de suelo que afectan el flujo de agua, puesto que un perfil compuesto de diferentes tipos de suelo presenta una dinámica completamente distinta a uno con suelo homogéneo. Sin embargo, la utilización de parámetros predefinidos de HYDRUS-1D para los tipos de suelo presenta una limitante para representar exactamente las condiciones del suelo, por lo que, un aspecto de mejora puede ser la estimaciones de estos parámetros para los diferentes tipos de suelo.

La utilización de la información del cultivo para configurar la temporada, el cambio en la distribución de las raíces y la separación en evaporación y transpiración de la evaporación de cultivo, con los coeficientes  $K_e$  y  $K_{cb}$  respectivamente, permiten obtener un modelo que representa la dinámica del flujo de agua de mejor manera. Sin embargo, los resultados obtenidos indican que se están subestimando las pérdidas de agua, reflejando en que el contenido de agua disminuye una tasa más lenta de lo observado en los datos reales de la Viña.

Finalmente, HYDRUS-1D posee capacidades aún no exploradas, como otros modelos de simulación de los movimientos de agua y las funciones de transporte de solutos y calor. Aplicar estos cambios podría mejorar la representación de la dinámica del flujo de agua en el perfil de suelo, lo que acercaría el modelo al sistema real.

### 8.3.2. Sistema de control

La configuración del sistema de control de sistema se basa en la acción del control, el criterio de riego y la recompensa para su entrenamiento, y son parte fundamental del proceso de aprendizaje del agente.

En cuanto a la acción de control, el espacio de acciones no es constante entre un paso y otro, puesto que el riego depende del contenido de agua medido para el cálculo del riego a capacidad. Esto genera que distintos valores de la acción posean el mismo efecto, en particular, cuando el valor de contenido de agua es superior a capacidad de campo, cualquier acción tomada resulta en un riego nulo.

Posibles soluciones a estos son el cambio de a un espacio constante de acciones, como por ejemplo, definir una cantidad máxima de riego en base a las capacidad del equipo de riego.

Respecto del criterio de riego, la Viña utiliza mediciones del potencial xilemático ( $\mathbf{Px}$ ) en la planta, por lo que, en base a la información de riego provista se realizó una interpretación de sus programaciones de riego previas, donde en el primer método se genera un criterio basado en una única medición del contenido de agua. Esto da poca flexibilidad al sistema y una poca representación del estado completo del perfil de suelo.

En el segundo método se busca mejorar esto y se utilizan más variables en la representación del estado hídrico del cultivo, lo que aumento la información con la que el agente tomaba las decisiones, luego las acciones tomadas fueron capaces de controlar el contenido de agua a distintas profundidades del perfil, asegurando así el cumplimiento de los requerimientos hídricos establecidos.

Otra alternativa que se podría abordar, sería establecer una relación entre las mediciones de contenido de agua en el suelo con las mediciones del estado hídrico de la planta. Una alternativa sería usar mediciones de  $P_x$  bajo las cuales se aplico el riego, luego, revisar el WC en el perfil en esos momentos, establecer una relación a modo de convertir el criterio basado en  $P_x$  en uno basado en WC.

En el primer método, las recompensas están ajustadas para favorecer ciertos estados por sobre otros y penalizar el uso de agua, sin embargo, estos costos poseen escalas diferentes, lo que perjudica la interpretación de la función de costos por parte del agente, ya que estar un día en un estado perjudicial puede equivaler a muchas acciones de riego.

El segundo método ofrece una solución a esto, restringiendo el valor de los costos por estado hídrico en el rango  $[-1,1]$ , de modo que la relevancia de cada costo esté determinado por sus pesos al definir el entrenamiento. De esta forma, la recompensa es interpretable para el agente, y el efecto de un correcto ajuste de los aumenta. Por ejemplo, al asignar un peso mayor a los riegos, el agente optaba por no disminuir los riegos aplicados puesto que estresar el cultivo penalizaba menos. Finalmente, se asigno un peso ligeramente superior al estado hídrico para que el agente se centrara en mantener el cultivo en su rango de estrés.

Otra alternativa, es reemplazar la función de los estados del suelo por una basada en el rendimiento del cultivo, pero para el caso de la uva vinífera, las variables de esa función pueden ser subjetivas o muy dependientes del criterio de la Viña. Sin embargo, en otros frutos esta puede ser la mejor alternativa puesto que el rendimiento posee una relación más directa con los requerimientos hídricos.

Todos estos elementos por otra parte se benefician de la realización de más pruebas y entrenamientos con diferentes parámetros, de modo de evaluar la reacción del sistema a dichos cambios.

### **8.3.3. Entrenamiento de los controladores**

#### **Observaciones del primer método**

De la mano con la formulación del sistema de control, el entrenamiento de los controladores para el primer método, visto en la Figura 8.3, se beneficiaría de un mayor número de pruebas y cambios en los parámetros y arquitecturas de las redes neuronales de los algoritmos.

Dado que los tiempos de entrenamiento pueden ser un problema, la mejor solución es seleccionar controladores basados en el algoritmo que genere un mejor desempeño. Y a partir de este realizar ajustes de parámetros y arquitecturas, para su posterior aplicación en el segundo método.

En función de esto, si se utiliza el criterio de costo computacional se ve que DQN es el peor algoritmo, debido a la necesidad de una gran número de iteraciones para llegar a un estado de aprendizaje que produzca resultados similares a TD3 y PPO.

Si la decisión se toma respecto del desempeño en el entrenamiento (valor de la recompensa

en el episodio), entonces el mejor algoritmo es PPO tanto continuo como discreto, luego DQN que posee un mejor desempeño que TD3 para las funciones de costos lineales y cuadrática.

Respecto de la convergencia y aprendizaje del algoritmo, se puede ver que una vez terminadas las fases exploratorias de los algoritmos se logra la convergencia, sin embargo, DQN es muy lento en el aprendizaje, TD3 detiene su aprendizaje requiriendo ruido en la acción y PPO aprende rápidamente y además continua su aprendizaje con el tiempo.

Sin embargo, ese rápido aprendizaje es producto de la poca variabilidad de casos de estudio, puesto que sólo se tienen 16 escenarios de información meteorológica, los algoritmos desarrollan una estrategia para cada caso y no siguen explorando después de eso, llegando a la convergencia observada.

Tomando en cuenta estos factores se hace claro que PPO representa la mejor alternativa de algoritmo, luego este es seleccionado para continuar explorando y ajustando su entrenamiento en el segundo método.

## Observaciones del segundo método

En el primer método, los algoritmos de mejor desempeño fueron los basados en PPO, por lo que, son seleccionados para la definición del agente en el método con la nueva definición del estado hídrico.

En la Tabla 8.6, se detallan los tiempos de entrenamiento de los agentes, los cuales están dentro de los esperados dada la cantidad de *timesteps* realizados.

Dado el comportamiento del aprendizaje en el método 1, se esperaba que aumento del tiempo de entrenamiento bastase para obtener una convergencia en la recompensa obtenida, sin embargo, el aumento de complejidad del sistema de control con 4 mediciones y la aleatoriedad de los datos meteorológicos generan un aprendizaje más lento y oscilatorio, lo que indica la necesidad de aumentar aún más el número de *timesteps*. Debido a los tiempos de ejecución, en lugar de realizar un nuevo entrenamiento desde cero, se aplicó un nuevo entrenamiento a los modelos previo a la obtención de los resultados. Lo anterior, permite esperar que si se aplica un entrenamiento más largo se puedan obtener resultados con un mejor desempeño que el ya alcanzado.

Respecto de la utilización de las versiones en continuo o discreto, se aprecia que el caso discreto posee tiempos de ejecución más bajos y los valores de recompensa alcanzados son mayores que en el caso continuo. Lo que indicaría que su desempeño y el costo computacional lo hacen una mejor alternativa para la generación de las programaciones de riego.

### 8.3.4. Resultados obtenidos

#### Resultados del primer método

Las programaciones de riego obtenidas para el primer método no permiten generar muchas conclusiones respecto de la calidad del sistema de control implementado. Sin embargo, se pueden realizar cambios que mejoren el desempeño y destacar algunos puntos positivos, a tomar en cuenta para las programaciones de riego del segundo método.

Las principales características que se buscan de los controladores es la optimización del riego y adelantarse a eventos como las precipitaciones. En las Figuras 8.5, los controladores que pueden manejar las predicciones de lluvia son aquellos basados en PPO, tanto continuo como discreto.

Las programaciones de riego obtenidas no representan un criterio de optimalidad en su ejecución, puesto que constantemente generan contenidos de agua superiores a capacidad de campo con sus riegos. En algunos casos se logra una secuencia de oscilaciones en torno a un contenido de agua, siguiendo cierta periodicidad, aún así no es tan claro para considerarlo una planificación esperada.

Se pueden analizar las recompensas utilizadas, los casos C2 y C3 donde el agente recibe una recompensa mejor por estar más cerca del estrés hídrico, efectivamente logran que los niveles de contenido de agua se acerquen al umbral. Esto se debe a que plantean una función de recompensa que indica claramente que hay estados mejores que otros, lo que es más interpretable por el agente en su aprendizaje.

Para lograr mejores resultados se deben aplicar los cambios al sistema de control y al proceso de entrenamiento discutidos previamente. De modo que el controlador implementado en el segundo método tenga un objetivo de aprendizaje que traduzca en la planificación obtenida.

#### Resultados del segundo método

Las programaciones de riego obtenidas responden a los cambios realizados en: la definición del estado hídrico, la configuración del modelo de simulación, el sistemas de control y las recompensas.

En primer lugar, si se mantuviera el objetivo de controlar únicamente el contenido de agua de un único sensor ubicado cercano a la superficie, las programaciones de riego resultantes logrando mantener las mediciones de WC dentro del rango esperado. En particular, se observa que no se llega a contenidos superiores a FC, logrando que no se desperdicie agua.

En cuanto al control del estado hídrico dado por las 4 mediciones, se aprecia que los controladores logran mantener el contenido de agua dentro de los rangos esperados, sin embargo, no es posible apreciar un patrón de riegos en particular que logre esto, por lo que, la planificación puede no ser intuitiva.

Las programaciones de riego obtenidas, no ejecutan riegos significativos hasta muy avanzada la etapa de desarrollo, lo cual puede ser un problema pues no se está interpretando la necesidad de regar pese a que han pasados meses sin un ingreso de agua. Esto se relaciona con la lenta disminución del contenido de agua en el perfil, lo que plantea mejoras en la estimación de la evapotranspiración.

De las dos funciones de costos utilizadas, se puede considerar que la opción cuadrática genera un comportamiento más estable para las 3 primeras mediciones, lo que puede estar relacionado al valor asociado a que el contenido de agua se encuentre cerca de EH.

Se considera que el controlador logro interpretar correctamente los requerimientos hídricos establecidos, sin embargo, se identifican aspectos de mejora para el modelo de simulación, en particular respecto del ajuste de las pérdidas de agua dadas por la evapotranspiración. También, se podría ajustar la aplicación del riego modificando el número de horas de riego y el momento del día en el cual este es aplicado.

### **8.3.5. Comparación con estrategias convencionales**

#### **Resultados riegos convencionales del primer método**

Los resultados obtenidos con la aplicación de estrategias de riego convencionales tenían como objetivo ser el punto donde comparar las mejoras de desempeño con aprendizaje reforzado, sin embargo, los resultados muestran la necesidad de aplicar mejoras en el trabajo realizado, tomando como base el desempeño obtenido por las técnicas convencionales.

Las Figuras 8.8 muestran que RCC al depender del estado actual genera un valor de riego particular para dicho estado, lo que le da una ventaja respecto al riego por monto fijo, que riega siempre los mismo, lo que según el estado puede ser más o menos agua de la necesaria, para llegar a capacidad en este caso.

Lo anterior, puede presentar una similitud con lo buscado con el controlador, puesto que se desea que para determinadas condiciones (estado) se aplique el mejor riego posible. Sin embargo, ya hemos discutido que para ciertos estados RCC siempre es cero, lo que puede arruinar la dinámica ya aprendida del sistema, ya que una acción que en el estado anterior valía 10 es 0 en el actual.

Plantear la alternativa de un espacio fijo de acción/riego podría permitir que para cada instante la acción signifique lo mismo, y es bajo este contexto que los resultados obtenidos con riego convencional permiten definir los límites del espacio de acción.

Otro dato importante es la periodicidad con la que cada método aplica los riegos, esto se obtiene a partir de la duración de la etapa y el total de riegos realizados, esto permite estimar cuanto tarda en estresarse la planta desde diferentes niveles de contenido de agua. Para los métodos que obtuvieron un buen desempeño, la periodicidad es entre 10 y 15 días.

## Resultados riegos convencionales del segundo método

Los resultados de las estrategias convencionales para el primer método no funcionaron cuando la cantidad aplicada era pequeña, por lo que, en el segundo método sólo se aplicaron los riegos: 100[%] de RCC, 75[%] de RCC, de 40 y 45 [mm], estos últimos se ajustaron dada las nuevas características del modelo.

Las programaciones nuevamente muestran la lentitud con la que el perfil de suelo pierde el agua, lo que genera que se realicen riegos muy espaciados, lo cual no se ajusta a los requerimientos establecidos por la Viña. En ese aspecto, las programaciones de riego obtenidas con AR muestran una ventaja, pues realizan riegos más pequeños, con lo que no pasa tanto tiempo entre uno y otro, además en lugar de apuntar a llegar exactamente a EH, se controla el contenido de agua manteniéndolo en un rango cercano a EH, que también cumple con el objetivo de estresar al cultivo.

### 8.3.6. Comparación con riegos aplicados en la Viña

Si bien los riegos convencionales permiten evaluar y comparar el desempeño de las programaciones obtenidas con AR, el objetivo de este trabajo es mejorar el riego actualmente efectuado en la Viña. Para poder realizar esto, en la segunda etapa se configura el modelo y el sistema de control para representar fielmente el caso real.

En primer lugar, los riegos aplicados por la Viña afectan el contenido de agua en las 3 profundidades medidas, aún cuando la cantidad aplicada no es tan grande. El riego aplicado en el modelo logra este efecto sólo con grandes cantidades de agua. Lo que plantea la necesidad de ajustar la aplicación del riego en el modelo.

Por otro lado, el flujo de agua en el perfil de suelo es tal que el agua aplicada se “pierde.”<sup>en</sup> una semana aproximadamente, donde se aplica el nuevo riego. Esto difiere completamente de lo logrado en el modelo pues los tiempos de “vaciado” del suelo son mayores, lo que hace que los riegos no tengan la periodicidad deseada. Luego, es necesario ajustar la evaporación y transpiración en el modelo, pues constituyen las pérdidas de agua del perfil.

Pese a las diferencias en las magnitudes de los valores de contenido de agua, en los efectos del riego y el “consumo” de agua, las programaciones de riego logran que el comportamiento del contenido de agua en el perfil se asemeje al observado en la Viña. Lo anterior es siempre valido a los 20 y 40 [cm], sin embargo, no es posible realizar un correcto control a los 60 [cm].

Si el agua no llega fácilmente hasta profundidades más bajas, se puede explicar por la presencia de una sección de suelo Franco Arcillosa (FA) a partir de los 35 a 50 [cm], la cual retiene el agua en sus poros pequeños para luego pasar a un suelo Franco Arcillo arenoso (FAa) que posee poros grandes que toman tiempo en ser llenados.

Luego, pese a las diferencias el control del riego con AR es satisfactorio y se asemeja a lo realizado por la Viña, logrando que el suelo pase por momentos de estrés hídricos y luego se recupere aplicando riego, lo que permitiría el desarrollo de la uva vinífera según los requerimientos para la producción de vino.

# Capítulo 9

## Conclusión y trabajo futuros

Se resume lo logrado con la ejecución de este trabajo de título, en términos del cumplimiento de los objetivos, resultados obtenidos, aprendizajes obtenidos y elementos de mejora. Es importante que destacar que la evaluación de estos puntos se realiza dentro de un contexto, en este caso el riego de uva vinífera en la sexta región. Además, se plantean cambios y complementos que abordar en el futuro, para profundizar lo realizado en este trabajo.

### 9.1. Conclusiones

En primer lugar, es importante destacar el trabajo realizado, comenzando por la utilización del software de simulación HYDRUS-1D para la generación de un modelo en Python, el cual toma en cuenta la información disponible y permite simular una dinámica del suelo cultivado.

En base a los resultados obtenidos, luego de aplicar la información disponible: composición del suelo, coeficiente de cultivo base  $K_{cb}$  y los umbrales de contenido de agua exactos, y realizar para configurar el modelo para que represente el caso real. Se puede concluir que los principales puntos de mejoras son la aplicación del riego y la salida de agua.

La aplicación del riego como una forma de precipitación requiere de mayores ajustes, ya que los sistemas de riego concentran el ingreso de agua en pequeña porción del suelo, mientras que las precipitaciones consideran un área mucho mayor, lo que hace difícil que el agua llegue a mayores profundidades. Al reducir el número de horas de riego en el método 2, se logra una pequeña mejora, sin embargo, aún no es del todo satisfactoria y puede seguir siendo ajustada.

Los resultados muestran que la salida de agua del perfil es más lenta que lo observado en la práctica, esto puede requerir de un ajuste en los valores transpiración y evaporación entregados al modelo. Un posible ajuste es considerar que al trabajar en un entorno de una dimensión, se puede pasar por alto el movimiento de agua horizontal que es tan relevante como el vertical que se estudia en HYDRUS-1D, luego, se podría aplicar un factor que represente estos flujos como transpiración o evaporación, logrando que la dinámica sea similar a la real.

Respecto del suelo también se puede mejorar la selección de parámetros del suelo, en el trabajo se utiliza los predeterminados de HYDRUS, sin embargo, estos podrían ser estimados para que las características del suelo sean las del caso real.

La integración del modelo con *Phydrus* da paso a su implementación en aprendizaje reforzado utilizando el entorno *Gym*, con la dinámica ya definida, el desafío está en la definición del sistema de control (agente, acciones de riego y recompensa) con los requerimientos particulares del cultivo de uva vinífera.

Respecto de las programaciones de riego generadas con los controladores del primer método, tras su análisis, se determina que la configuración del sistema de control no representa adecuadamente los criterios de riego, condicionando el aprendizaje y la toma de decisiones del controlador de riego. Debido a esto, en el segundo método se define el estado hídrico con 4 mediciones en lugar de 1, se ajustan las recompensas para que sean comparables e interpretables por el agente y se realizan ajustes al modelo para que se asemeje al caso real, de modo de comparar su desempeño con los datos reales.

Se tiene que las programaciones de riego para el control del contenido de agua a los 30 [cm], no presentan un seguimiento de los criterios establecidos, ni el comportamiento cercano a lo visto en las estrategias convencionales. Al utilizar el método con las mediciones de 20, 40, 60 y 100 [cm], las programaciones de riego si bien no replican exactamente lo realizado en la viña, principalmente debido a ajustes del modelo ya discutidos, el controlador logra generar episodios de estrés hídrico y manipular contenido de agua en el perfil por medio de las acciones implementadas.

La implementación de estrategias de riego convencional cumplió con entregar información para la mejora e identificación de problemas en el trabajo realizado. Para el primer método, dejó en evidencia la incapacidad del controlador para seguir los criterios impuestos y muestra una aproximación de comportamiento deseado. En el segundo método, su aplicación permitió observar y confirmar el problema referido a la salida de agua en el perfil.

Este análisis y resumen del trabajo realizado, permite definir el cumplimiento de los objetivos establecidos. Luego, el modelo de simulación diseñado con HYDRUS-1D permite representar la dinámica del terreno cultivado, y la implementación con *Phydrus* es capaz de ser integrada en el entorno *Gym* para su utilización con AR. En particular, en el segundo método se utiliza la información disponible para representar el cultivo real y permitir las comparaciones con los riegos realizados por la Viña.

Luego, en la proposición de sistemas de control con aprendizaje reforzado para dos definiciones del estado hídrico, que toman en consideración los requerimientos hídricos de la uva vinífera, mediante las recompensas y las configuración del entorno. Siendo el sistema de control del segundo método el que logra controlar el contenido de agua y optimizar el uso de agua con las acciones de riego tomadas.

Los esquemas tradicionales implementados fueron claves en la evaluación y mejora de los controladores propuestos, principalmente en el cumplimiento de los requerimientos y criterios de riego establecidos. Y por último, se realizó la comparación de las programaciones de riego obtenidas con AR con los riegos aplicados en la práctica, lo que fue un buen acercamiento a lo esperado en un caso real y dejando aspectos de mejora para trabajos futuros.

## 9.2. Trabajo Futuro

A partir de los resultados obtenidos, se puede comentar respecto de elementos que puedan complementar lo realizado en esta memoria en trabajos futuros, se tiene que la implementación de HYDRUS en un entorno *Gym* no está limitada al uso con aprendizaje reforzado, proponiéndose en particular la utilización de Control Predictivo basado en Modelos (MPC), con la implementación en *Phydurs* como modelo de simulación.

La librería *Phydurs* permite la utilización de HYDRUS-1D en Python, lo que limita la utilización de las capacidades avanzadas de la versión HYDRUS (2D/3D). Sin embargo, no se debe descartar explorar su uso y buscar la posibilidad de mejorar el modelo generado con la versión de pago de HYDRUS. En particular, debido a las limitaciones que presenta trabajar la dinámica tridimensional del flujo de agua en una sola dimensión.

El contenido de agua no es la única variable que puede servir para definir el estado hídrico de un cultivo, se puede proponer como alternativa relacionar las mediciones de la planta (ejemplo: potencial hídrico xilemático) con las condiciones del suelo, utilizando los datos y la información disponible, transformando los criterios de riego basados en mediciones en la planta pero unos que basados en la información del suelo.

Finalmente, el trabajo realizado se centró en el cultivo de uva vinífera, por lo que fue configurado para cumplir con los requerimientos de este tipo de cultivo. Sin embargo, con la información disponible que posee HYDRUS y la metodología para la generación del modelo, esta propuesta de solución se puede aplicar al problema de riego de cualquier otro cultivo.

# Bibliografía

- [1] Khadijeh Alibabaei, Pedro Gaspar, Eduardo Assunção, Saeid Alirezazadeh, and Tânia Lima. Irrigation optimization with a deep reinforcement learning model: Case study on a site in portugal. *Agricultural Water Management*, 263:107480, 01 2022.
- [2] R. Allen, L. Pereira, D. Raes, and M. Smith. Fao irrigation and drainage paper no. 56. *Rome: Food and Agriculture Organization of the United Nations*, 56:26–40, 01 1998.
- [3] J.-E. Bergez, M. Eigenraam, and F. Garcia. Comparison between dynamic programming and reinforcement learning: A case study on maize irrigation management. *European conference of the European Federation for Information technology in agriculture, Food and the Environment*, 01 2002.
- [4] J. Boonstra. Estimating peak runoff rates. In H. Ritzema, editor, *Drainage principles and application*. ILRI, 1994.
- [5] Greg Brockman, Vicki Cheung, Ludwig Pettersson, Jonas Schneider, John Schulman, Jie Tang, and Wojciech Zaremba. Openai gym. *OpenAI*, 06 2016.
- [6] M. Chen, Y. Cui, X. Wang, H. Xie, F. Liu, T. Luo, S. Zheng, and Y. Luo. A reinforcement learning approach to irrigation decision-making for rice using weather forecasts. *Agricultural Water Management*, 250:106838, 2021.
- [7] R.A. Collenteur, G. Brunetti, and M. Vremec. Phydus: Python implementation of hydrus-1d. <https://github.com/phydrus/phydrus>, 2021. Guía de instalación en <https://phydrus.readthedocs.io/en/latest/index.html>.
- [8] Dirección Meteorológica de Chile. Reporte anual de la evolución del clima en chile. Technical report, Dirección General de Aeronáutica Civil, 2020.
- [9] Oficina de estudios y políticas agrarias. *Agricultura Chilena, Reflexiones y Desafíos al 2030*. Ministerio de Agricultura, 2017.
- [10] Oficina de estudios y políticas agrarias. Panorama de la agricultura chilena. Technical report, Ministerio de Agricultura, 2019.
- [11] H. Dong, Z. Ding, S. Zhang, H. Yuan, H. Zhang, J. Zhang, Y. Huang, T. Yu, H. Zhang, and R. Huang. *Deep Reinforcement Learning: Fundamentals, Research, and Applications*. Springer Nature, 2020. <http://www.deeppreinforcementlearningbook.org>.

- [12] J. Doorenbos and A. Kassam. Fao irrigation and drainage paper no. 56. *Rome: Food and Agriculture Organization of the United Nations*, 09 2020. Yield Response to water-original 1979.
- [13] R. Ferreyra and G. Selle. Algunas consideraciones para manejar el riego en vides para vino. *Instituto de Investigaciones Agropecuarias. no. 39*, 05 2007.
- [14] Fertilab. Interpretación de resultados de textura y humedad del suelo. <https://www.fertilab.com.mx/>, 2016.
- [15] Scott Fujimoto, Herke Hoof, and Dave Meger. Addressing function approximation error in actor-critic methods. *ICML 2018*, 02 2018.
- [16] S. Geerts, E. Kipkorir, J. Wellens, and S. Ali. Simulation of yield decline as result of water stress with a robust soil water balance model. *Agricultural Water Management*, 81:335–357, 02 2006.
- [17] J. Girona. Manejo de riego de vides para vino en españa. *Curso Internacional: Manejo de riego y suelo en vides para vino y mesa*, 06 2005.
- [18] R. J. Hanks. Model for predicting plant yield as influenced by water use1. *Agronomy Journal*, 66(5):660–665, 1974.
- [19] M. Jensen. Water consumption by agricultural plants. *In: Kozlowski, T.T., Ed., Water Deficits in Plant Growth*, 02:1–22, 1986.
- [20] M. Jensen and R. Allen. *Evaporation, Evapotranspiration, and Irrigation Water Requirements*. American Society of Civil Engineers, second edition, 2016.
- [21] James Jones, Cheryl Porter, K.J Boote, William Batchelor, Leslie Hunt, P.W Wilkens, Upendra Singh, A.J Gijsman, and Joe Ritchie. Dssat cropping system model. *European Journal of Agronomy*, 18:235–265, 01 2003.
- [22] Matlab. Reinforcement learning for control systems applications, 2018. [www.mathworks.com/help/reinforcement-learning/ug/reinforcement-learning-for-control-systems-applications.html](http://www.mathworks.com/help/reinforcement-learning/ug/reinforcement-learning-for-control-systems-applications.html).
- [23] J. Merino. *Diseño e implementación de una estrategia de control para sistemas de riego utilizando aprendizaje reforzado*. Universidad de Chile, Santiago, 2022. Memoria para optar al título de ingeniero civil eléctrico.
- [24] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. Playing atari with deep reinforcement learning. *DeepMind Technologies*, 2013. cite arxiv:1312.5602Comment: NIPS Deep Learning Workshop 2013.
- [25] M. Muñoz. *Respuesta de la variación del diámetro de tronco y del potencial hídrico xilemático a diferentes regímenes de riego en uva de mesa, CV. Crimson Seedless*. Universidad de Chile, Santiago, 2005. Memoria de título.
- [26] OpenAI. Introduction to rl, 2018. <https://spinningup.openai.com/en/latest/index.html>.

- [27] Yakov Pachepsky, Dennis Timlin, and Walter Rawls. Generalized richards' equation to simulate water transport in unsaturated soils. *Journal of Hydrology*, 272:3–13, 03 2003.
- [28] H. L. Penman. “natural evaporation from open water, bare soil and grass”. *Proceedings of the Royal Society of London*, 193(1032):120–145, 1948.
- [29] T. Roje. Control predictivo aplicado al nexo “agua-energía-alimento” en comunidades rurales. Master’s thesis, Universidad de Chile, 2020.
- [30] M. Schaap. *Rosetta: A computer program for estimating soil hydraulic parameters with hierarchical pedotransfer functions*. Dept. of Soil, Water and Environmental Science College of Agriculture and Life Sciences, University of Arizona, 1.2 edition, 2002.
- [31] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *OpenAI*, 07 2017.
- [32] Soil Survey Staff. *Keys to Soil Taxonomy*. USDA-Natural Resources Conservation Service, 13th edition, 2022.
- [33] L. Sun, Y. Yang, J. Hu, D. Porter, T. Marek, and C. Hillyer. Reinforcement learning control for water-efficient agricultural irrigation. In *2017 IEEE International Symposium on Parallel and Distributed Processing with Applications and 2017 IEEE International Conference on Ubiquitous Computing and Communications (ISPA/IUCC)*, pages 1334–1341, 12 2017.
- [34] J. Šimůnek, M. Šejna, H. Saito, M. Sakai, and M. Th. van Genuchten. *The HYDRUS-1D Software Package for Simulating the One-Dimensional Movement of Water, Heat, and Multiple Solutes in Variably-Saturated Media*. Department of Environmental Sciences, University of California Riverside, 4.17 edition, Junio 2013.

# Anexos

# Anexo A

## Cálculo de las variables de la ecuación Penman-Monteith

La ecuación FAO Penman-Monteith del documento de la FAO [2], permite calcular la evapotranspiración de referencia  $ET_o$  para un cultivo según:

$$ET_o = \frac{0,408\Delta(R_n - G) + \gamma \frac{900}{T+273} u_2 (e_s - e_a)}{\Delta + \gamma(1 + 0,34u_2)} \quad (\text{A.1})$$

con  $ET_o$  la evapotranspiración de referencia en  $[mm]$ ,  $\Delta$  la pendiente de la curva de presión de vapor respecto a la temperatura  $[\frac{kPa}{^\circ C}]$ ,  $R_n$  la radiación solar diaria neta en la superficie del cultivo en  $[\frac{MJ}{m^2}]$ ,  $\gamma$  la constante psicrométrica en  $[\frac{kPa}{^\circ C}]$ ,  $T$  la temperatura media a dos metros de altura en  $[\text{^\circ C}]$ ,  $u_2$  la velocidad del viento a dos metros de altura en  $[\frac{m}{s}]$ ,  $e_s$  la presión de vapor de saturación en  $[kPa]$  y  $e_a$  la presión de vapor real en  $[kPa]$ .

El cálculo de cada una de las variables que influyen en la formulación de la ecuación A.1 se presenta conforme se plantea en [2], a partir de la información meteorológica, condiciones del ambiente y coordenadas geográficas para los siguientes parámetros:

- Temperatura mínima ( $T_{min}$ ) y máxima ( $T_{max}$ ) en  $[\text{^\circ C}]$ .
- Radiación solar media ( $R_s$  en  $[\frac{W}{m^2}]$ ).
- Velocidad del viento ( $u$ ) en  $[\frac{m}{s}]$ .
- Latitud ( $\varphi$ ) en  $[rad]$ .
- Altura sobre el nivel del mar ( $z$ ) en  $[m]$ .
- Altura de medición de velocidad del viento  $h$  en  $[m]$ .

Con mediciones para cada uno de estos factores y considerando el número del día como  $J \in [1, 365]$ , se pueden determinar las variables que influyen en la ecuación para la evapotranspiración de referencia (A.1).

## Temperatura media ( $T$ )

La temperatura media  $T$  se define como el promedio entre las temperaturas máxima y mínima, de la forma:

$$T = \frac{T_{max} + T_{min}}{2}. \quad (\text{A.2})$$

## Constante psicrométrica ( $\gamma$ )

El cálculo de la constante psicrométrica está sujeto a la determinación de la presión atmosférica ( $P_{atm}$ ) en  $[kPa]$ , que depende de la altura sobre el nivel del mar según la siguiente expresión:

$$P_{atm} = 101,3 \left( \frac{293 - 0,0065z}{293} \right)^{5,26}, \quad (\text{A.3})$$

con esto la constante psicrométrica se define como:

$$\gamma = 0,665 \cdot 10^{-3} \cdot P_{atm}. \quad (\text{A.4})$$

## Presión de vapor de saturación media ( $e_s$ )

La presión de vapor de saturación media se define como el promedio entre sus valores extremos, es decir:

$$e_s = \frac{e_{s,min} + e_{s,max}}{2}. \quad (\text{A.5})$$

donde  $e_{s,min}$  y  $e_{s,max}$  corresponden a las presiones de vapor de saturación mínima y máxima respectivamente, las cuales se determinan a partir de las siguientes expresiones:

$$e_{s,min} = 0,6108 \exp \left( 17,27 \frac{T_{min}}{T_{min} + 273,3} \right), \quad (\text{A.6})$$

$$e_{s,max} = 0,6108 \exp \left( 17,27 \frac{T_{max}}{T_{max} + 273,3} \right). \quad (\text{A.7})$$

## Presión de vapor real ( $e_a$ )

El valor de esta variable cuando no se posee información respecto de la humedad relativa, puede ser aproximado a la presión de vapor de saturación mínima, es decir:

$$e_a = e_{s,min}. \quad (\text{A.8})$$

## Pendiente de la curva de presión de vapor respecto a la temperatura ( $\Delta$ )

La expresión para la pendiente de la curva de presión de vapor respecto a la temperatura, se define a partir de la temperatura media calculada en A.2 como:

$$\Delta = 4098 \frac{0,6108 \exp\left(\frac{17,27T}{T+273,3}\right)}{(T+273,3)^2}. \quad (\text{A.9})$$

## Radiación solar diaria neta ( $R_n$ )

La radiación solar diaria neta está estrictamente ligada a la posición con respecto al sol del punto estudiado, es por ello que para su determinación es necesario calcular: la distancia inversa relativa de la tierra al sol ( $d_r$ ) y la declinación solar ( $\delta$ ) en [rad] como se muestra a continuación:

$$d_r = 1 + 0,033 \cos\left(2\pi \frac{J}{365}\right), \quad (\text{A.10})$$

$$\delta = 0,409 \sin\left(\left(2\pi \frac{J}{365} - 1,93\right)\right). \quad (\text{A.11})$$

Luego, se define el ángulo a la puesta de sol ( $w_s$ ) en [rad] como:

$$w_s = \arccos(-\tan(\varphi) \tan(\delta)). \quad (\text{A.12})$$

Con esto, se puede calcular la radiación solar extraterrestre diaria  $R_a$  en  $\left[\frac{MJ}{m^2 dia}\right]$  como:

$$R_a = \frac{1440}{\pi} G_{sc} d_r [w_s \sin(\varphi) \sin(\delta) + \cos(\varphi) \cos(\delta) \sin(w_s)], \quad (\text{A.13})$$

con  $G_{sc} = 0,082$  la constante solar en  $\left[\frac{MJ}{m^2 dia}\right]$ . A continuación, la radiación solar diaria de cielo despejado  $R_{so}$  se puede estimar en función de  $R_a$  y  $z$  como:

$$R_{so} = R_a (0,75 + 2 \cdot 10^{-5} z) \quad (\text{A.14})$$

A partir de este valor, se pueden calcular las radiaciones solares netas de onda corta ( $R_{ns}$ ) y onda larga ( $R_{nl}$ ) en  $\left[\frac{MJ}{m^2 dia}\right]$ , como se muestra a continuación:

$$R_{ns} = (1 - \alpha) R_s, \quad (\text{A.15})$$

$$R_{nl} = \sigma (0,34 - 0,14 \sqrt{e_a}) \left( \frac{(T_{min+273,16})^4 + (T_{max} + 273,16)^4}{2} \right) \left( 1,35 \frac{R_s}{R_{so}} - 0,35 \right), \quad (\text{A.16})$$

con  $\alpha = 0,23$  el albedo de referencia definido por la FAO en [2],  $\sigma = 4,903 \cdot 10^{-9} \left[\frac{MJ}{K^4 m^2 dia}\right]$  la constante de Stefan-Boltzmann y  $R_s$  la radiación solar convertida desde el promedio en  $\left[\frac{W}{m^2 dia}\right]$  a  $\left[\frac{MJ}{m^2 dia}\right]$  según:

$$R_s \left[ \frac{MJ}{m^2 dia} \right] = 0,0864 R_s \left[ \frac{W}{m^2 dia} \right] \quad (\text{A.17})$$

. Finalmente, se puede obtener una expresión para la radiación solar neta sobre la superficie del cultivo, a partir de  $R_{ns}$  y  $R_{nl}$ , dada por:

$$R_n = R_{ns} - R_{nl}. \quad (\text{A.18})$$

## Densidad de flujo de calor del suelo ( $G$ )

El valor para la densidad de flujo de calor del suelo, se estima como nula cuando se trata de periodos diarios, es decir:

$$G \approx 0. \quad (\text{A.19})$$

## Velocidad del viento a dos metros de altura ( $u_2$ )

La velocidad del viento a dos metros de altura se define en función de la velocidad del viento medida y la altura a la que fue realizada la medición, de la forma:

$$u_2 = \frac{4,87u}{\ln(67,8h - 5,42)}. \quad (\text{A.20})$$

# Anexo B

## Implementación de Phydrus

En el el capítulo 4 se detalla el proceso de configuración del programa de simulación HYDRUS-1D a través de la interfaz gráfica que este posee, lo cual permite la generación de proyectos que imiten casos de estudio reales, lo que hace ver las altas capacidades que tiene el programa, sin embargo, el programa base no es del todo combinable con otros elementos computacionales, por lo que, se debe realizar un ajuste o traducción de este a otros programas.

Es aquí donde surge la librería *Phydrus* [7] que permite la generación de proyectos de HYDRUS en Python, esta librería aprovecha el sistema de configuración del software base, transformando las ventanas de la interfaz de usuario a funciones que configuran al modelo. De esta manera, se pueden obtener los mismos resultados que con el software original, pero con todas los beneficios que puede prestar Python.

A continuación se describen brevemente los pasos necesarios para la utilización de *phydrus* y una explicación de las funciones que permiten generar los modelos.

### B.1. HYDRUS-1D en Python

La implementación de HYDRUS-1D en Python, requiere de un pequeño proceso de instalación, para luego pasar a identificar los pasos para generar el modelo y el análisis de los resultados.

#### B.1.1. Instalación de Phydrus

Como cualquier librería de Python esta debe ser instalada actualizando el resto de librerías necesarias para su uso, además de asegurar la instalación de la última versión de la librería en cuestión.

Luego, se debe obtener el archivo ejecutable de HYDRUS-1D en donde se llevarán a cabo los proyectos, esto puede lograrse generando el archivo siguiendo los pasos de la guía de

instalación, utilizando versiones ya compiladas de HYDRUS provenientes del repositorio o directamente el software original instalado.

## B.1.2. Utilizando Phydrus

Una vez que se tiene el ejecutable instalado y la librería instalada ya podemos utilizar *Phydrus*, tomando en cuenta que el objetivo de las funciones es replicar la interfaz gráfica del programa, luego al ejecutar el programa desde Python se busca escribir esta configuración en el programa, de modo que HYDRUS funcione de igual manera a como si estuviera realizando el proyecto en el programa original.

En primer lugar, se debe utilizar la función **Model** para generar el modelo en *Phydrus*, en este entrega la información básica del proyecto a realizar y elementos necesarios para la ejecución del mismo: ubicación del ejecutable *hydrus.exe*, el nombre y descripción del proyecto y las unidades de medida de concentración, tiempo y distancia (Replica lo visto en 4.3, 4.4 y 4.5).

Siguiendo con las configuraciones básicas, al modelo se le agrega la información temporal mediante el atributo **add\_time\_info**, es decir, la duración del proyecto, la discretización, la utilización de condiciones atmosféricas y los instantes a visualizar en los resultados (Replica lo visto en 4.5 y 4.6).

A continuación, se agregan los procesos mediante sus atributos propios, en este caso nos restringimos a la utilización del flujo de agua y la absorción de agua de la raíz, es decir, **add\_waterflow**, donde se define el modelo a utilizar, los criterios de iteraciones y las condiciones de borde el perfil (Replica lo visto en 4.7, 4.8 y 4.10) y **add\_root\_uptake**, donde se define el modelo y los parámetros asociados a este (Replica lo visto en 4.11, 4.12 y 4.13).

Una vez se tiene los procesos, se definen las características del suelo, donde lo primero es asociar el tipo de suelo o material que determina los parámetros del modelo de flujo de agua elegido, para ello se utilizan los atributos **get\_empty\_material\_df** para crear un DataFrame donde incluir los parámetros y **add\_material** para añadir el DataFrame creado al modelo (Replica lo visto en 4.9).

Siguiendo la estructura vista en 4, se deben entregar las condiciones atmosféricas que afectan la dinámica del suelo, para ello se utiliza el atributo **add\_atmospheric\_bc**, a la cual se entregan las variables en formato DataFrame o se definen directamente como parámetros (Replica lo visto en 4.14).

Finalmente, se debe configurar el perfil de suelo a semejanza de lo hecho en el editor gráfico de HYDRUS, para ello se recurre a la generación del perfil como DataFrame mediante la función **create\_profile** de *Phydrus*, a la cual se le entrega el tamaño del perfil, el tamaño de la grilla que divide a los nodos, la información de cada nodo (condiciones iniciales, material, raíces, entre otros). De manera alternativa, se puede cargar un perfil ya creado con la función **profile\_from\_file**. En cualquier caso, una vez definido el perfil a utilizar este es agregado al modelo mediante el atributo **add\_profile** (Replica lo visto en 4.15, 4.16 y 4.18).

Todo lo anterior, completa la configuración del proyecto y ya se puede ejecutar la simulación para la resolución de las ecuaciones, sin embargo, primero el se deben escribir los archivos que contienen dichas configuraciones para que puedan ser leídas al simular.

Al realizarse la simulación, se escriben archivos de salida que muestran una ejecución exitosa del programa, los resultados a visualizar dependerá de los instantes elegidos para imprimir y los puntos de observación agregados con el atributo **add\_obs\_nodes**. La obtención de los gráficos que muestran la dinámica del suelo, se puede obtener a partir de funciones de la librería que lo permiten o al cargar los archivos resultantes y manejar los datos de la forma que el usuario estime conveniente.

# Anexo C

## Algoritmos aprendizaje reforzado

En este anexo se exhiben pseudo-códigos con un mayor desarrollo provenientes de [11], a modo de complemento de los vistos en el capítulo 5 que están más orientados a la implementación. Estos algoritmos complementan los conceptos y explicaciones desarrolladas en el libro, por lo que están más detallados.

# DQN

---

**Algorithm 4** DQN

---

```
1: Hiper-parámetros: ( $N$ ): Capacidad del replay buffer, ( $\gamma$ ): factor de descuento de la
   recompensa, ( $C$ ): pasos para la actualización de función acción-valor objetivo, ( $\varepsilon$ ): factor
   del criterio  $\varepsilon$ -greedy;
2: Input: ( $\mathcal{D}$ ): replay buffer vacío, ( $\theta$ ): parámetros iniciales de la función acción-valor  $Q$ ;
3: Inicializar la función acción-valor objetivo  $\hat{Q}$  con parámetro  $\hat{\theta} = \theta$ 
4: for episodios = 0,1,2, ... do
5:   Inicializar el entorno y obtener la observación  $O_0$ 
6:   Inicializar secuencia  $S_0 = \{O_0\}$  y secuencia de preprocesamiento  $\phi_0 = \phi(S_0)$ 
7:   for t=0,1,2, ... do
8:     Con probabilidad  $\varepsilon$  aplicar una acción aleatoria  $A_t$ , de lo contrario elegir  $A_t =$ 
      $\arg \max_a Q(\phi(S_t), a; \theta)$ 
9:     Ejecutar la acción  $A_t$  y observar  $O_{t+1}$  y la recompensa  $R_t$ 
10:    if el episodio terminó then
11:      Asignar  $D_t = 1$ 
12:    else
13:      Asignar  $D_t = 0$ 
14:    end if
15:    Establecer  $S_{t+1} = \{S_t, A_t, O_{t+1}\}$  y preprocesamiento  $\phi_{t+1} = \phi(S_{t+1})$ 
16:    Guardar la transición  $(\phi_t, A_t, R_t, D_t, \phi_{t+1})$  en  $\mathcal{D}$ 
17:    Muestrear un minibatch aleatorio  $(\phi_i, A_i, R_i, D_i, \phi_{i+1})$  de  $\mathcal{D}$ 
18:    if  $D_i = 0$  then
19:      Asignar  $Y_i = R_i + \gamma \max_{a'} \hat{Q}(\phi'_i, a'; \hat{\theta})$ .
20:    else
21:      Aginar  $Y_i = R_i$ 
22:    end if
23:    Aplicar un paso de gradiente descendente en  $(Y_i - Q(\phi_i, A_i, \theta))^2$  con respecto a  $\theta$ 
24:    Sincronizar el objetivo  $\hat{Q}$  cada  $C$  pasos
25:    if el episodio terminó then
26:      romper el bucle
27:    end if
28:  end for
29: end for
```

---

## TD3

---

**Algorithm 5** TD3

---

- 1: **Hiper-parámetros:**  $(\rho)$ : factor de actualización suave,  $(\gamma)$ : factor de descuento de la recompensa,  $(c)$ : factor de recorte (*clip*);
  - 2: **Input:**  $(\mathcal{D})$ : *replay buffer* vacío,  $(\theta_1, \theta_2)$ : parámetros iniciales de los críticos  $Q_{\theta_1}, Q_{\theta_2}$ ,  $(\phi)$ : parámetros iniciales del actor  $\pi_\phi$ ;
  - 3: Inicializar las funciones objetivo  $\hat{\theta}_1 = \theta_1, \hat{\theta}_2 = \theta_2, \hat{\phi} = \phi$
  - 4: **for**  $t = 1$  a  $T$  **do**
  - 5:   Realizar acción con ruido de exploración  $A_t \sim \pi_\phi(S_t) + \varepsilon, \varepsilon \sim \mathcal{N}(0, \sigma)$
  - 6:   Observar recompensa  $R_t$  y nuevo estado  $S_{t+1}$
  - 7:   Guardar la tupla de transición  $(S_t, A_t, R_t, D_t, S_{t+1})$  en  $\mathcal{D}$
  - 8:   Muestrear un *minibatch* de  $N$  transiciones  $(S_t, A_t, R_t, D_t, S_{t+1})$  de  $\mathcal{D}$
  - 9:    $\tilde{a}_{t+1} = \pi_{\phi'}(S_{t+1}) + \varepsilon, \varepsilon \sim \text{clip}(\mathcal{N}((0, \sigma), -c, c))$
  - 10:    $y = r_t + \gamma(1 - D_t) \min_{i=1,2} Q_{\theta'_i}(S_{t+1}, \tilde{a}_{t+1})$
  - 11:   Actualizar el crítico  $\theta_i = \arg \min_{\theta_i} N^{-1} \sum (y - Q_{\theta_i}(S_t, A_t))^2$
  - 12:   **if**  $t$  módulo  $d$  **then**
  - 13:     Actualizar  $\phi$  según gradiente determinístico de la política:
  - 14:      $\nabla_\phi J(\phi) = N^{-1} \sum \nabla_a Q_{\theta_1}(S_t, A_t)|_{A_t=\pi_\phi(S_t)} \nabla_\phi \pi_\phi(S_t)$
  - 15:     Actualizar redes objetivo:
  - 16:      $\hat{\theta}_i = \rho \theta_i + (1 - \rho) \hat{\theta}_i$
  - 17:      $\hat{\phi} = \rho \phi + (1 - \rho) \hat{\phi}$
  - 18:   **end if**
  - 19: **end for**
-

# PPO

---

**Algorithm 6** PPO-Penalty

---

1: **Hiper-parámetros:**  $(\lambda)$ : coeficiente de penalización KL,  $(\gamma)$ : factor de descuento de la recompensa,  $(a = 1,5, b = 2)$ : parámetros adaptivos,  $(M, B)$ : número de sub-iteraciones  
2: **Input:**  $(\theta)$ : parámetros iniciales de la política,  $(\phi)$ : parámetros iniciales de la función valor  
3: **for**  $k = 0, 1, 2, \dots$  **do**  
4:   Ejecutar política  $\pi_\theta$  por  $T$  pasos, recolectar  $\{S_t, A_t, R_t\}$   
5:   Estimar los beneficios  $\hat{A}_t = \sum_{t'>t} \gamma^{t'-t} R_{t'} - V_\phi(S_t)$   
6:    $\pi_{old} = \pi_\theta$   
7:   **for**  $m \in \{1, \dots, M\}$  **do**  
8:      $J_{PPO}(\theta) = \sum_{t=1}^T \frac{\pi_{theta}(A_t|S_t)}{\pi_{old}(A_t|S_t)} \hat{A}_t - \lambda \hat{\mathbb{E}}_t[D_{KL}(\pi_{old}(\cdot|S_t) \parallel \pi_\theta(\cdot|S_t))]$   
9:     Actualizar  $\theta$  con el método del gradiente con respecto a  $J_{PPO}(\theta)$   
10:   **end for**  
11:   **for**  $b \in \{1, \dots, B\}$  **do**  
12:      $L_{bl}(\phi) = - \sum_{t=1}^T (\sum_{t'>t} \gamma^{t'-t} R_{t'} - V_\phi(S_t))^2$   
13:   **end for**  
14:   Calcular  $d = \hat{\mathbb{E}}_t[D_{KL}(\pi_{old}(\cdot|S_t) \parallel \pi_\theta(\cdot|S_t))]$   
15:   **if**  $d < \frac{d_{target}}{a}$  **then**  
16:      $\lambda = \frac{\lambda}{b}$   
17:   **else**  
18:     **if**  $d > d_{target} \times a$  **then**  
19:        $\lambda = \lambda \times b$   
20:     **end if**  
21:   **end if**  
22: **end for**

---

---

**Algorithm 7** PPO-Clip

---

- 1: **Hiper-parámetros:**  $(\varepsilon)$ : factor de recorte *clip*,  $(M, B)$ : número de sub-iteraciones
- 2: **Input:**  $(\theta)$ : parámetros iniciales de la política,  $(\phi)$ : parámetros iniciales de la función valor
- 3: **for**  $k = 0, 1, 2, \dots$  **do**
- 4:     Recoger conjuntos de trayectorias  $\mathcal{D}_k = \{\tau_i\}$  ejecutando la política  $\pi_{\theta_k}$  en el entorno.
- 5:     Calcular las recompensas  $\hat{G}_t$
- 6:     Calcular los beneficios estimados,  $\hat{A}_t$  en base a la función valor actual  $V_{\phi_k}$
- 7:     **for**  $m \in \{1, \dots, M\}$  **do**
- 8:         Definir:

$$l_t(\theta') = \frac{\pi_{\theta}(A_t|S_t)}{\pi_{\theta_{old}}(S_t|A_t)}$$

- 9:         Actualizar la política maximizando la función objetivo de PPO-Clip:

$$\theta_{k+1} = \arg \max_{\theta} \frac{1}{|\mathcal{D}_k|T} \sum_{\tau \in \mathcal{D}_k} \sum_{t=0}^T \min(l_t(\theta') A^{\pi_{\theta_{old}}}(S_t, A_t), \\ clip(l_t(\theta'), 1 - \varepsilon, 1 + \varepsilon) A^{\pi_{\theta_{old}}}(S_t, A_t)),$$

usualmente mediante el método del gradiente ascendente con Adam

- 10:     **end for**
- 11:     **for**  $b \in \{1, \dots, B\}$  **do**
- 12:         Ajustar la función valor mediante una regresión del error cuadrático medio:

$$\phi_{k+1} = \arg \min_{\phi} \frac{1}{|\mathcal{D}_k|T} \sum_{\tau \in \mathcal{D}_k} \sum_{t=0}^T (V_{\phi}(S_t) - \hat{G}_t)^2,$$

usualmente mediante algún algoritmo de gradiente descendente

- 13:     **end for**
  - 14: **end for**
-

# Anexo D

## Gym Environment

*Gym* es un paquete de Python para la aplicación de aprendizaje reforzado [5], particularmente para la generación de los modelos de simulación, la librería contiene modelos básicos para experimentar y también permite la generación de modelos propios, para esto último se debe seguir la siguiente estructura básica:

```
import gym
from gym import spaces
import numpy as np

class CustomEnv(gym.Env):
    """Entorno configurado en base a gym"""
    def __init__(self, arg1, arg2):
        super(CustomEnv, self).__init__()
        # Definir los parametros del modelo de simulacion
        self.arg1 = arg1
        self.arg2 = arg2
        # Definir espacio de acciones:
        # Acciones Discretas
        self.action_space = spaces.Discrete(num_actions)
        # Acciones Continuas
        self.action_space = spaces.Box(low = action_min ,
                                       high = action_max ,
                                       dtype = np.float32)

        # Definir espacio de observaciones (estado)
        self.observation_space = spaces.Box(low = state_min ,
                                             high = state_max ,
                                             dtype = np.float32)

    def step(self, action):
        # Aplicacion de la accion y desarrollo de la dinamica
        return observation, reward, done, info
```

```
def reset(self):  
    # Configuración del estado inicial del sistema  
    return observation  
  
def render(self, mode='human'):  
    # Visualización de la simulación
```

La generación de los elementos claves de aprendizaje reforzado es intuitiva al utilizar *Gym*, comienza recibiendo todos los parámetros y variables del modelo, establece los límites para la acción y el estado, define la condición inicial del sistema y finalmente define el entorno y su interacción con el agente, dinámica que genera las recompensas y nuevos estados.

*Gym* no es suficiente para implementar el aprendizaje reforzado, para eso hacen falta otras librerías como lo puede ser *Stable Baselines*, que aprovecha los entornos generados para la aplicación de los algoritmos de AR.

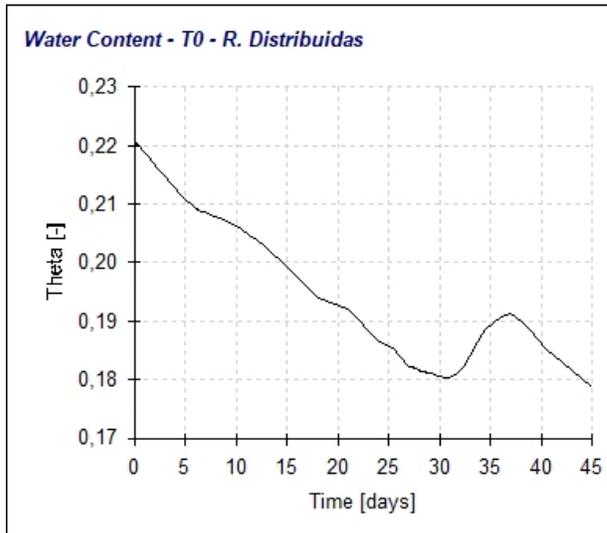
# Anexo E

## Otros resultados

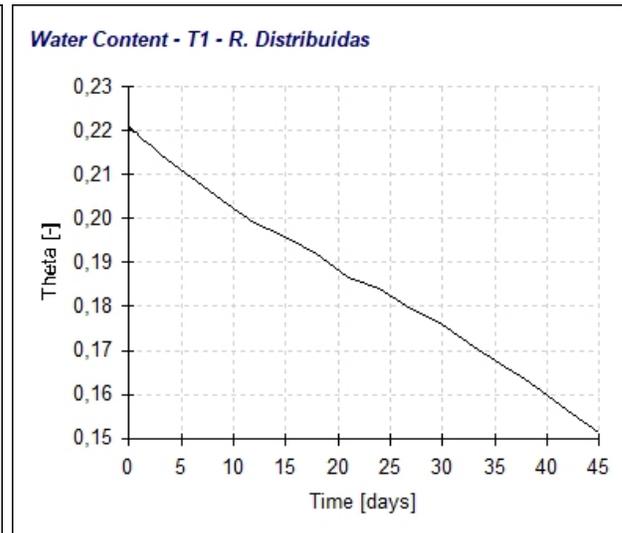
A continuación, se presentan gráficos asociados a diferentes etapas del desarrollo del trabajo, pero que no se presentan en el documento principal. Su estudio es opcional, pero puede servir de guía para reconocer elementos diferentes del estudio y comprender de mejor manera el trabajo realizado.

### E.1. Comparación HYDRUS-1D y *Phydrus*

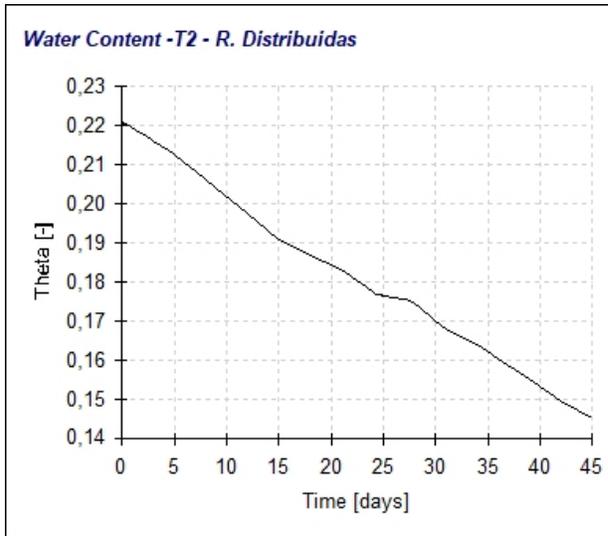
Simulación de la etapa 1 para el modelo con medición a los 30 [cm]



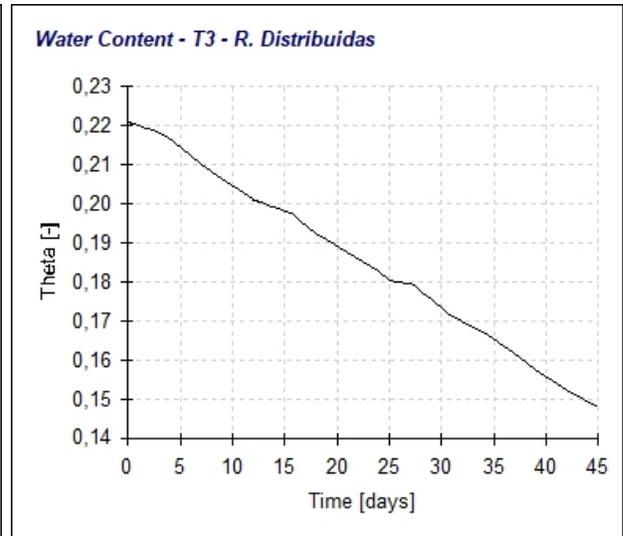
(a) Temporada 18-19



(b) Temporada 19-20

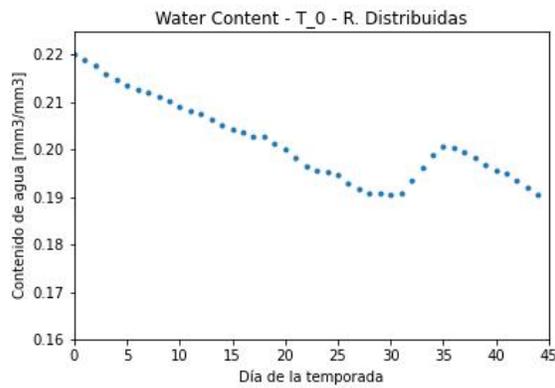


(a) Temporada 20-21

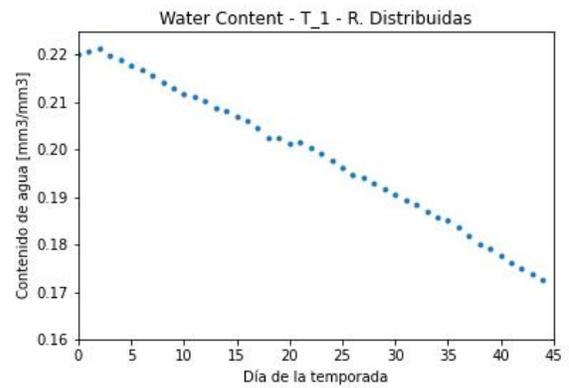


(b) Temporada 21-22

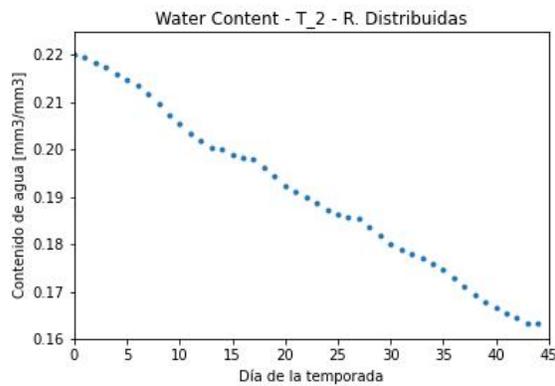
Figura E.2: Simulaciones de la etapa 1 con datos de las diferentes temporadas en HYDRUS-1D, para el método con 1 medición.



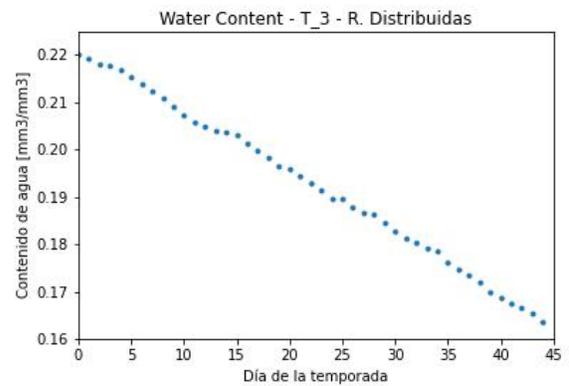
(a) Temporada 18-19



(b) Temporada 19-20



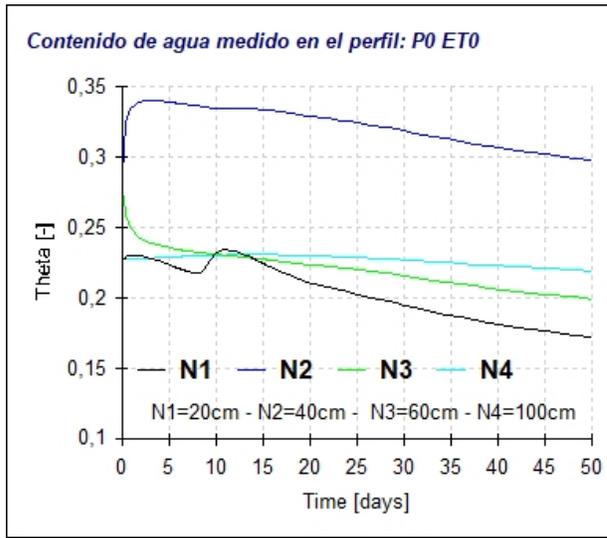
(c) Temporada 20-21



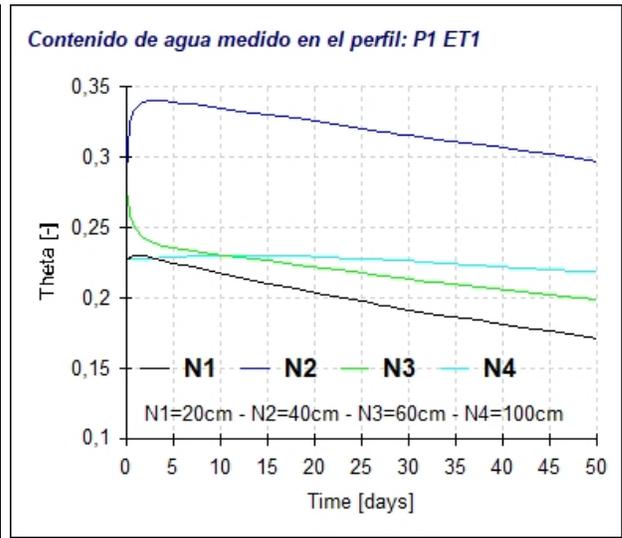
(d) Temporada 21-22

Figura E.3: Simulaciones de la etapa 1 con datos de las diferentes temporadas en *Phydrus*, para el método con 1 medición.

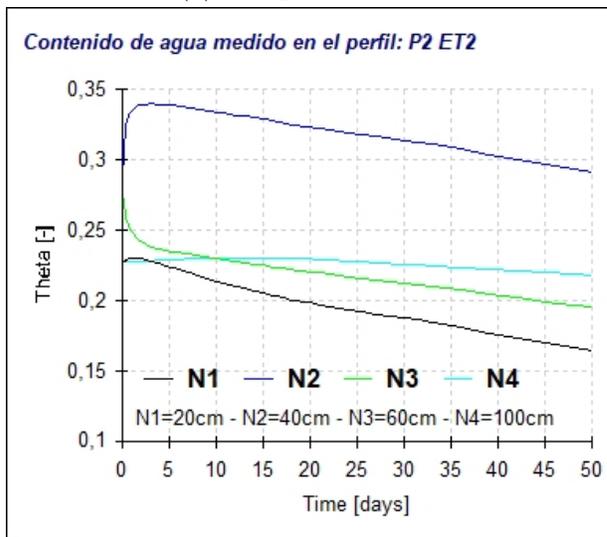
Simulación de la etapa 1 para el modelo con mediciones a los 20, 40, 60 y 100 [cm]



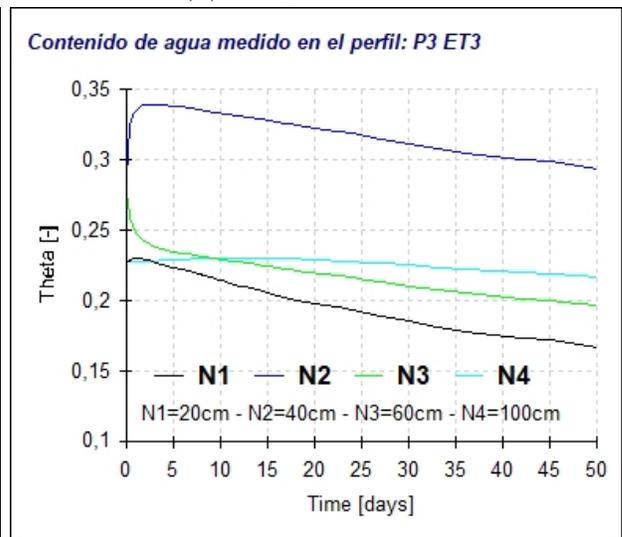
(a) Temporada 18-19



(b) Temporada 19-20

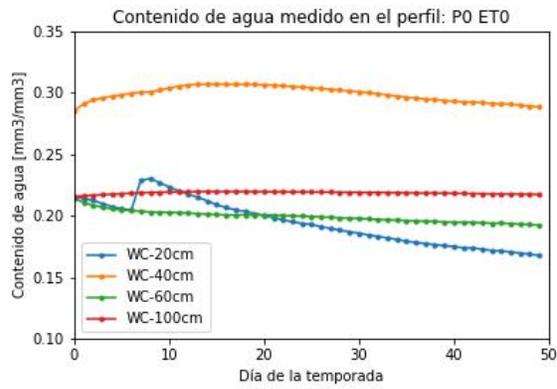


(c) Temporada 20-21

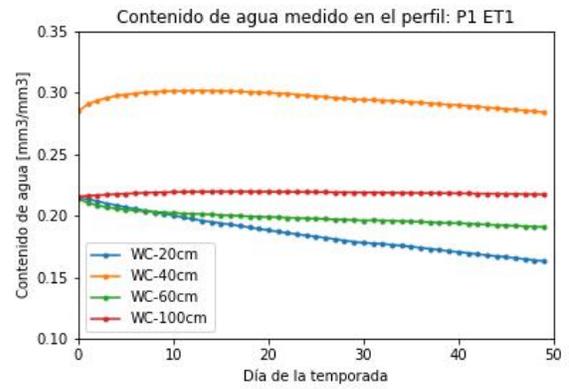


(d) Temporada 21-22

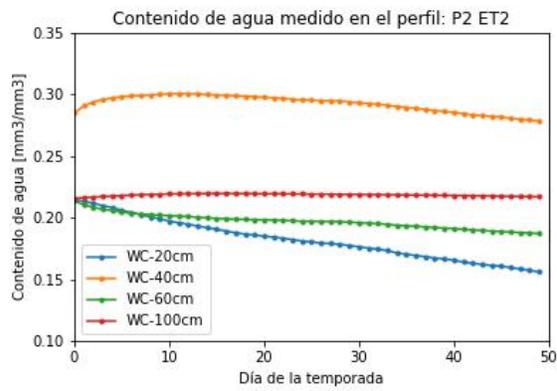
Figura E.4: Simulaciones de la etapa 1 con datos de las diferentes temporadas en HYDRUS-1D, con el método con 4 mediciones



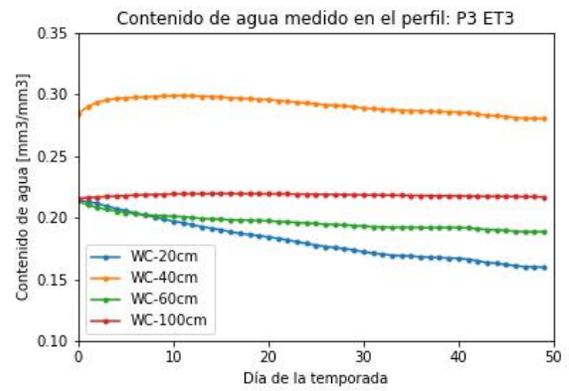
(a) Temporada 18-19



(b) Temporada 19-20



(c) Temporada 20-21

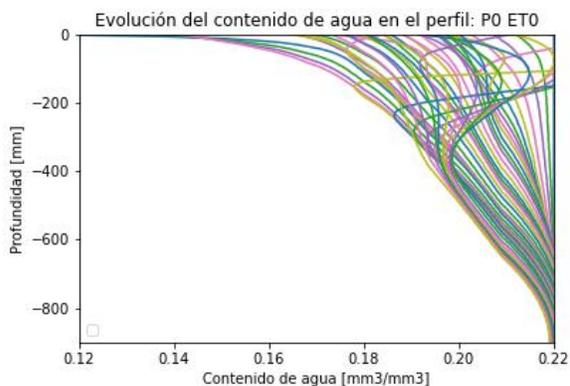


(d) Temporada 21-22

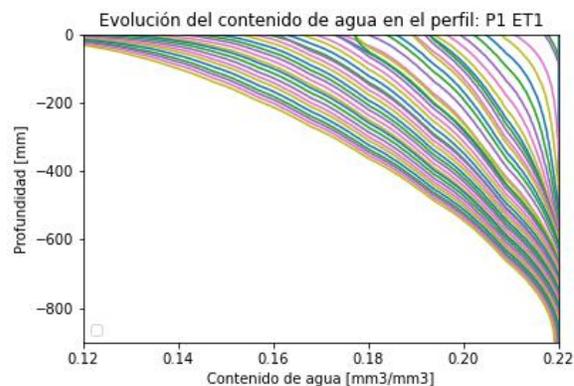
Figura E.5: Simulaciones de la etapa 1 con datos de las diferentes temporadas en *Phydrus*, para el método con 4 mediciones.

## E.2. Perfil de suelo en la primera etapa

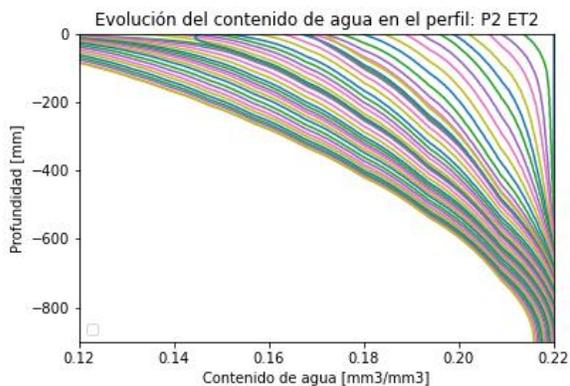
Perfiles de suelo para la etapa 1 con el modelo de medición a los 30 [cm]



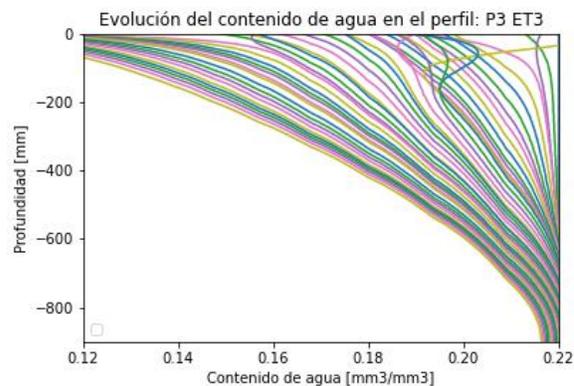
(a) Temporada 18-19



(b) Temporada 19-20



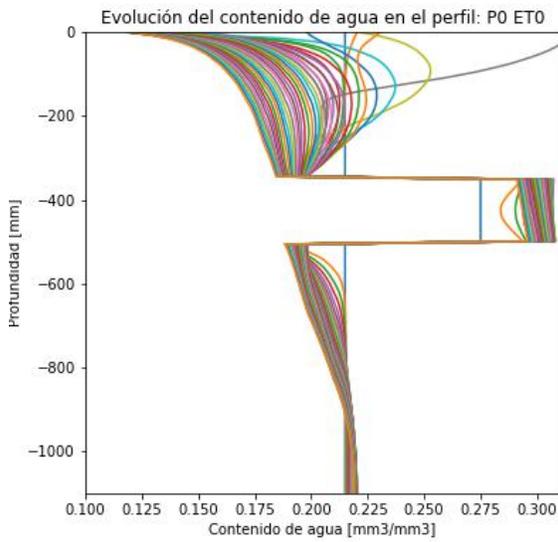
(c) Temporada 20-21



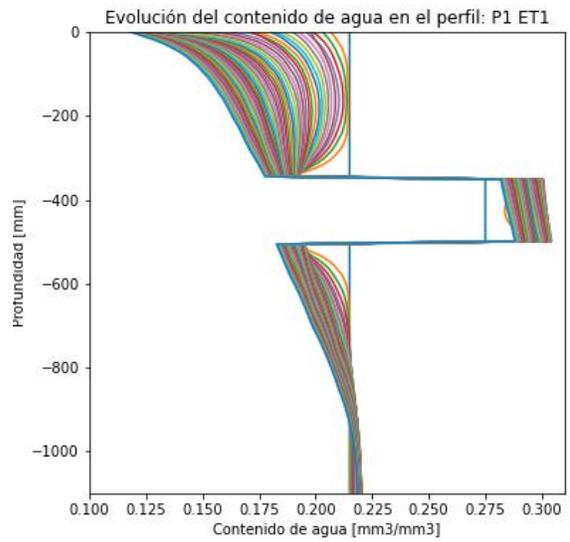
(d) Temporada 21-22

Figura E.6: Evolución del contenido de agua en el perfil de suelo para cada temporada, para el método con 1 medición.

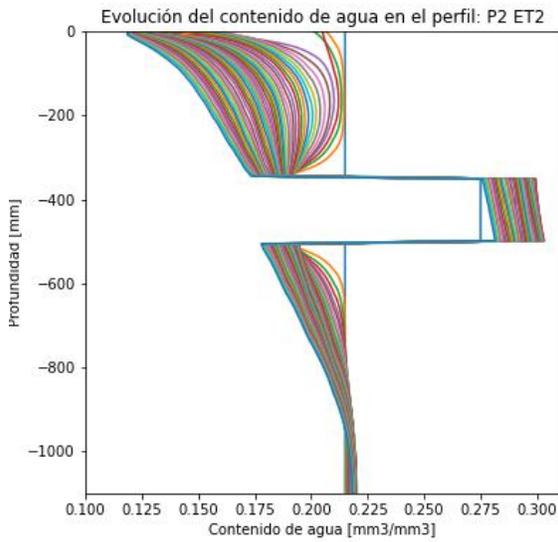
Perfiles de suelo para la etapa 1 con el modelo de mediciones a los 20, 40, 60 y 100 [cm]



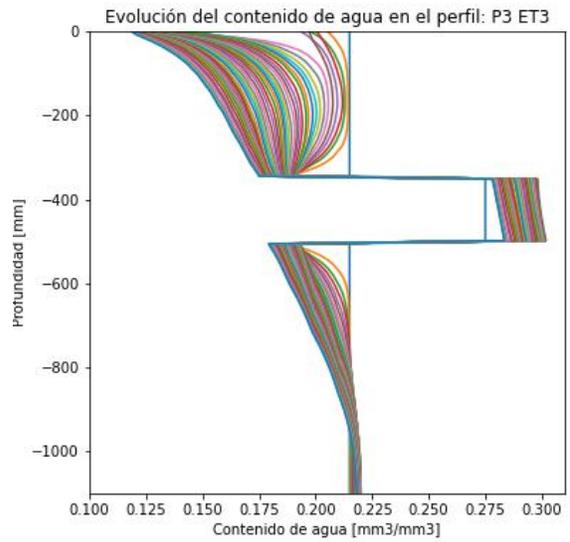
(a) Temporada 18-19



(b) Temporada 19-20



(c) Temporada 20-21

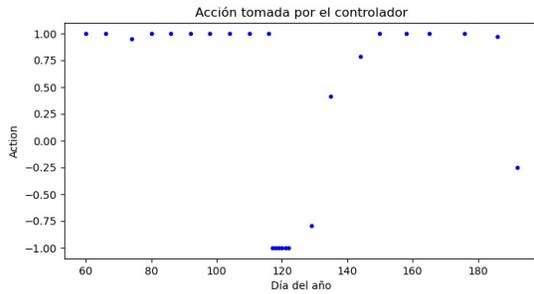


(d) Temporada 21-22

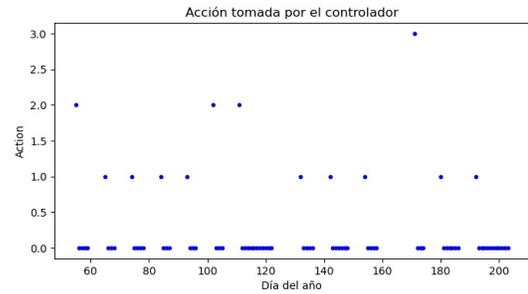
Figura E.7: Evolución del contenido de agua en el perfil de suelo para cada temporada, para el método con 4 mediciones.

### E.3. Acciones de riego y otras programaciones de riego

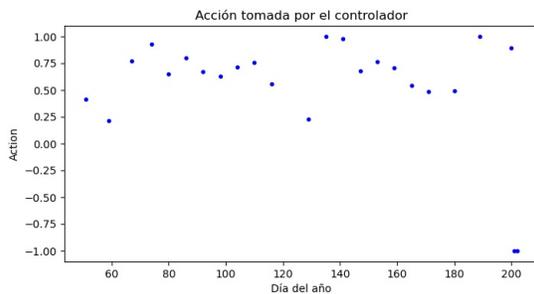
#### Método 1



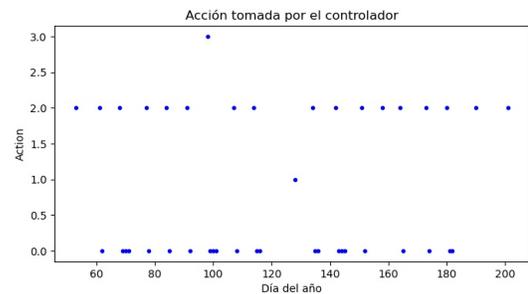
C1: Constante - PPO continuo



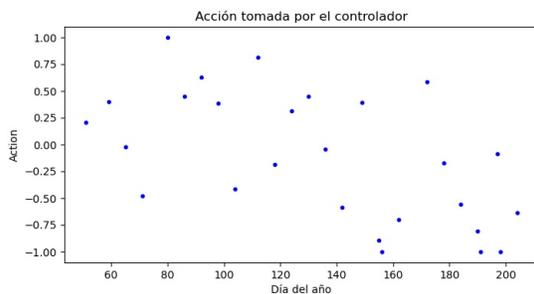
C1: Constante - DQN



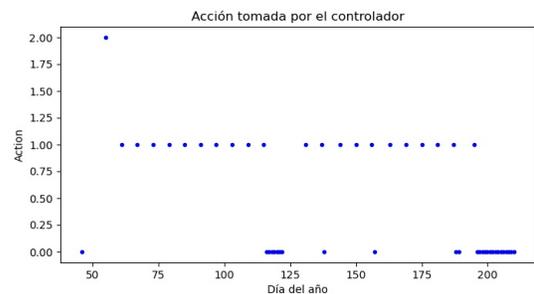
C1: Constante - TD3



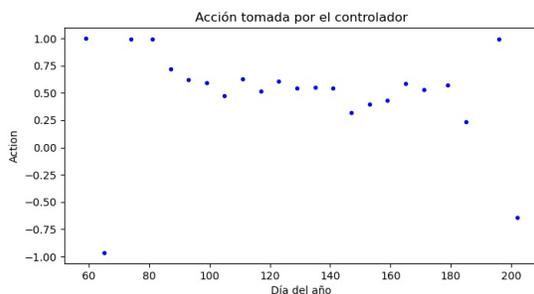
C1: Constante - PPO discreto



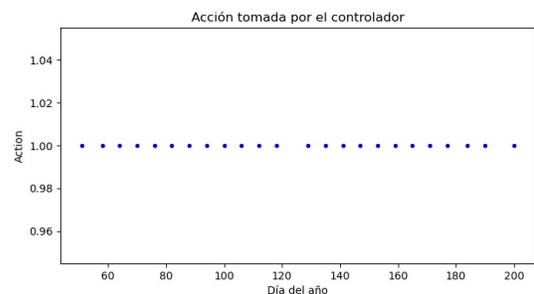
C2: Lineal - PPO continuo



C2: Lineal - DQN

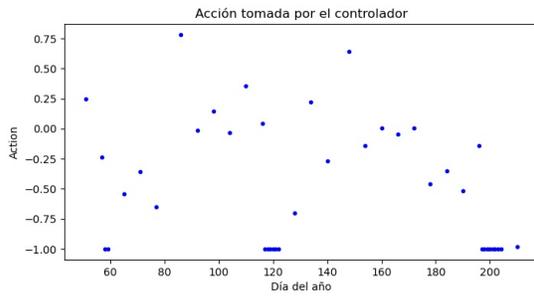


C2: Lineal - TD3

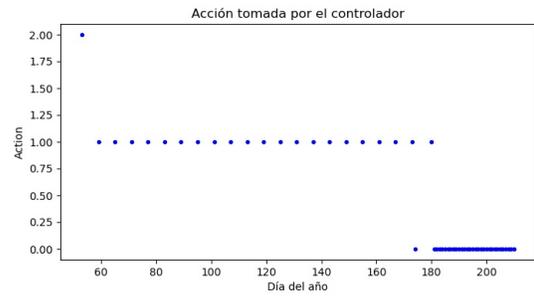


C2: Lineal - PPO discreto

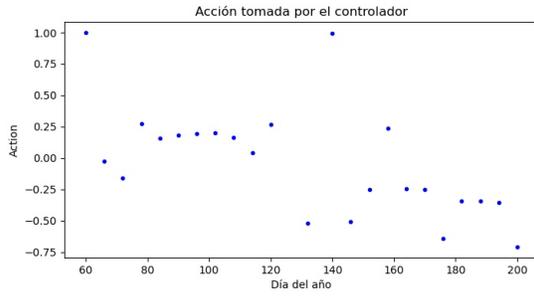
Figura E.8: Acciones del controlador para la segunda etapa, obtenidas con el agente entrenado (1/2).



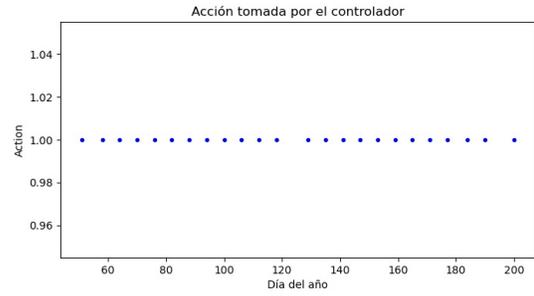
C3: Cuadrático - PPO continuo



C3: Cuadrático - DQN



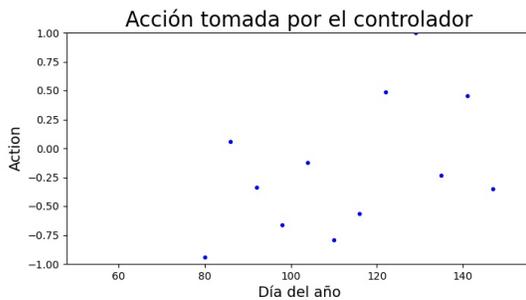
C3: Cuadrático - TD3



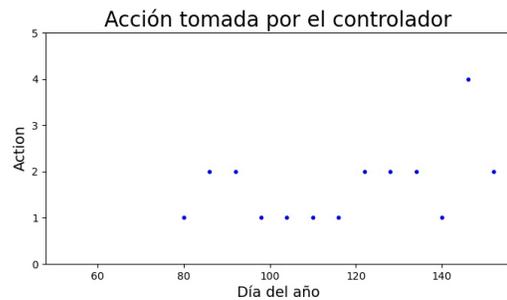
C3: Cuadrático - PPO discreto

Figura E.9: Acciones del controlador para la segunda etapa, obtenidas con el agente entrenado (2/2).

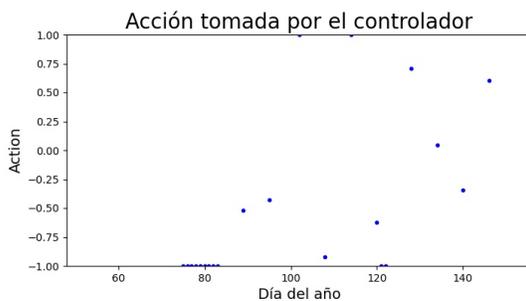
## Método 2



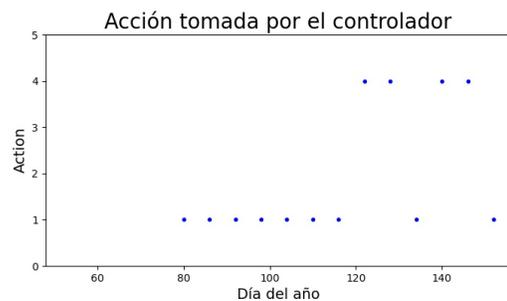
C2: Lineal - PPO continuo



C2: Lineal - PPO discreto



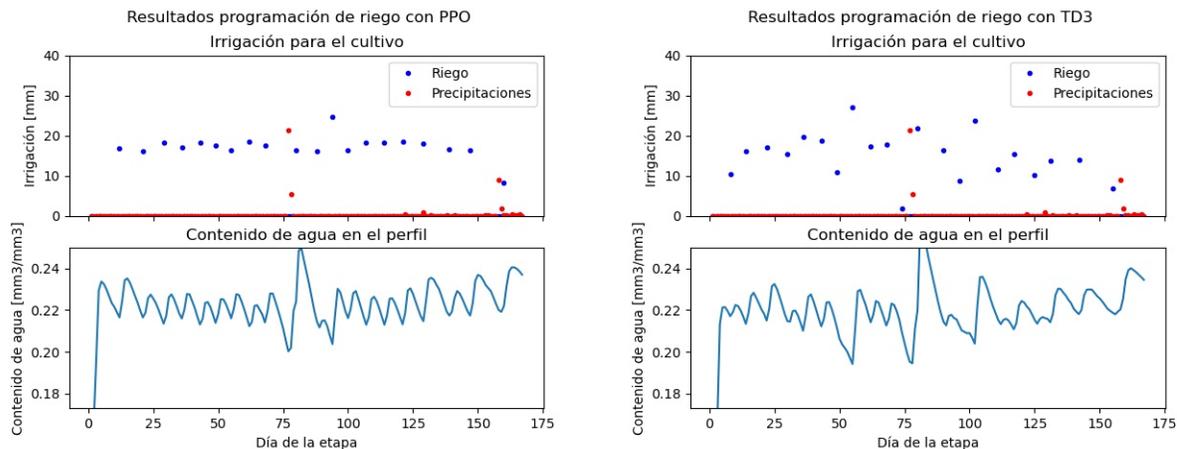
C3: Cuadrática - PPO continuo



C1: Cuadrática - PPO discreto

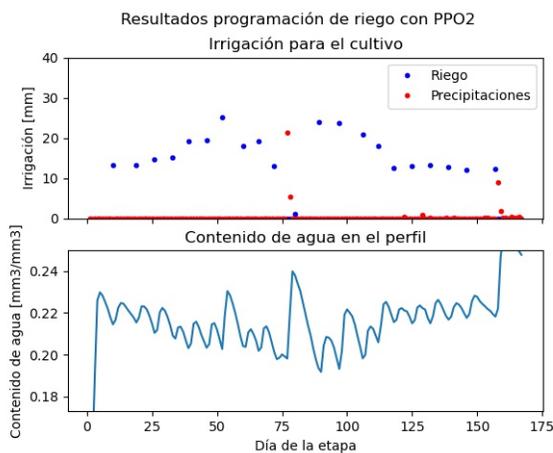
Figura E.10: Acciones del controlador para la segunda etapa, obtenidas con el agente entrenado para el segundo método.

## Programaciones de riego y sus acciones para raíces distribuidas homogéneamente



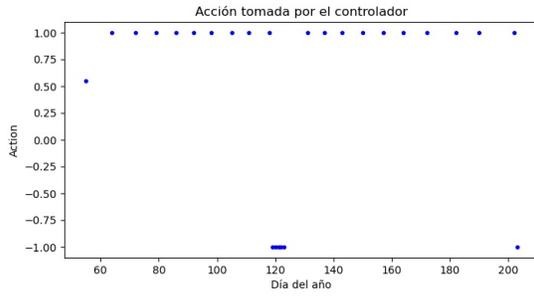
C1: Constante - PPO continuo

C1: Constante - TD3

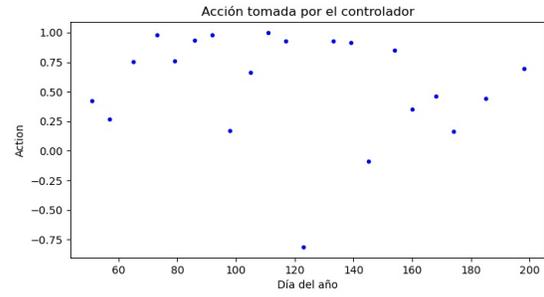


C1: Constante - PPO discreto

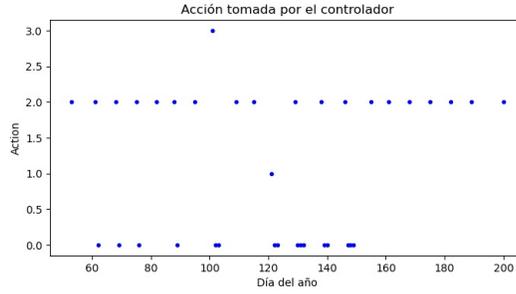
Figura E.11: Programaciones de riego para la segunda etapa, obtenidas con el agente entrenado, con raíces distribuidas homogéneamente.



C1: Constante - PPO continuo



C1: Constante - TD3



C1: Constante - PPO discreto

Figura E.12: Acciones del controlador para la segunda etapa, obtenidas con el agente entrenado, con raíces distribuidas homogéneamente

|       |           | Información de la programación de riego |           |              |               |
|-------|-----------|---|-----------|--------------|---------------|
| Costo | Algoritmo | R. total [mm]                           | Nº riegos | R. prom [mm] | Dist. prom EH |
| C1    | PPO       | 386.94                                  | 21        | 18.43        | 0.049         |
|       | TD3       | 356.64                                  | 22        | 16.21        | 0.047         |
|       | PPO2      | 384.03                                  | 22        | 17.45        | 0.044         |

Tabla E.1: Resumen de riego aplicado para cada programación de riego, y valor promedio de la diferencia diaria entre el contenido de agua y el umbral de estrés hídrico, con raíces distribuidas homogéneamente.

## E.4. Planificaciones de riego con estrategias convencionales

### E.4.1. Tablas de resumen para Método 1

|                 | Información de riego convencional (Temporada 19-20) |           |             |                |
|-----------------|---|-----------|-------------|----------------|
| R. Convencional | R. total[mm]  | N° riegos | R. prom[mm] | Dist. prom. EH |
| <b>100-RCC</b>  | 360.576   | 12        | 30.048      | 0.029          |
| <b>75-RCC</b>   | 362.934   | 16        | 22.683      | 0.018          |
| <b>50-RCC</b>   | 353.088   | 21        | 16.813      | 0.012          |
| <b>25-RCC</b>   | 357.966   | 27        | 13.258      | 0.030          |
| <b>30mm</b>     | 360   | 12        | 30          | 0.028          |
| <b>25mm</b>     | 350   | 14        | 25          | 0.021          |
| <b>20mm</b>     | 360   | 18        | 20          | 0.017          |
| <b>15mm</b>     | 360   | 24        | 15          | 0.040          |

Tabla E.2: Resumen de riego aplicado en estrategias convencionales y valor promedio de la diferencia entre el contenido de agua y el umbral de estrés hídrico, para la temporada 19-20.

|                 | Información de riego convencional (Temporada 20-21) |           |             |                |
|-----------------|---|-----------|-------------|----------------|
| R. Convencional | R. total[mm]  | N° riegos | R. prom[mm] | Dist. prom. EH |
| <b>100-RCC</b>  | 340.848   | 11        | 30.986      | 0.032          |
| <b>75-RCC</b>   | 324.432   | 14        | 23.173      | 0.022          |
| <b>50-RCC</b>   | 327.816   | 19        | 17.253      | 0.017          |
| <b>25-RCC</b>   | 326.196   | 25        | 13.047      | 0.030          |
| <b>30mm</b>     | 330   | 11        | 30          | 0.0235         |
| <b>25mm</b>     | 325   | 13        | 25          | 0.026          |
| <b>20mm</b>     | 320   | 16        | 20          | 0.020          |
| <b>15mm</b>     | 330   | 22        | 15          | 0.023          |

Tabla E.3: Resumen de riego aplicado en estrategias convencionales y valor promedio de la diferencia entre el contenido de agua y el umbral de estrés hídrico, para la temporada 20-21.

|                        | <b>Información de riego convencional (Temporada 21-22)</b> |                  |                                 |                       |
|------------------------|--|------------------|---------------------------------|-----------------------|
| <b>R. Convencional</b> | <b>R. total<math>[mm]</math></b>                           | <b>N° riegos</b> | <b>R. prom<math>[mm]</math></b> | <b>Dist. prom. EH</b> |
| <b>100-RCC</b>         | 366.192  | 12               | 30.516                          | 0.032                 |
| <b>75-RCC</b>          | 341.604  | 15               | 22.773                          | 0.019                 |
| <b>50-RCC</b>          | 341.820  | 19               | 17.990                          | 0.016                 |
| <b>25-RCC</b>          | 346.590  | 25               | 13.863                          | 0.032                 |
| <b>30mm</b>            | 360  | 12               | 30                              | 0.034                 |
| <b>25mm</b>            | 350  | 14               | 25                              | 0.026                 |
| <b>20mm</b>            | 360  | 18               | 20                              | 0.018                 |
| <b>15mm</b>            | 345  | 23               | 15                              | 0.049                 |

Tabla E.4: Resumen de riego aplicado en estrategias convencionales y valor promedio de la diferencia entre el contenido de agua y el umbral de estrés hídrico, para la temporada 21-22.

## E.4.2. Programaciones de riego para Método 1

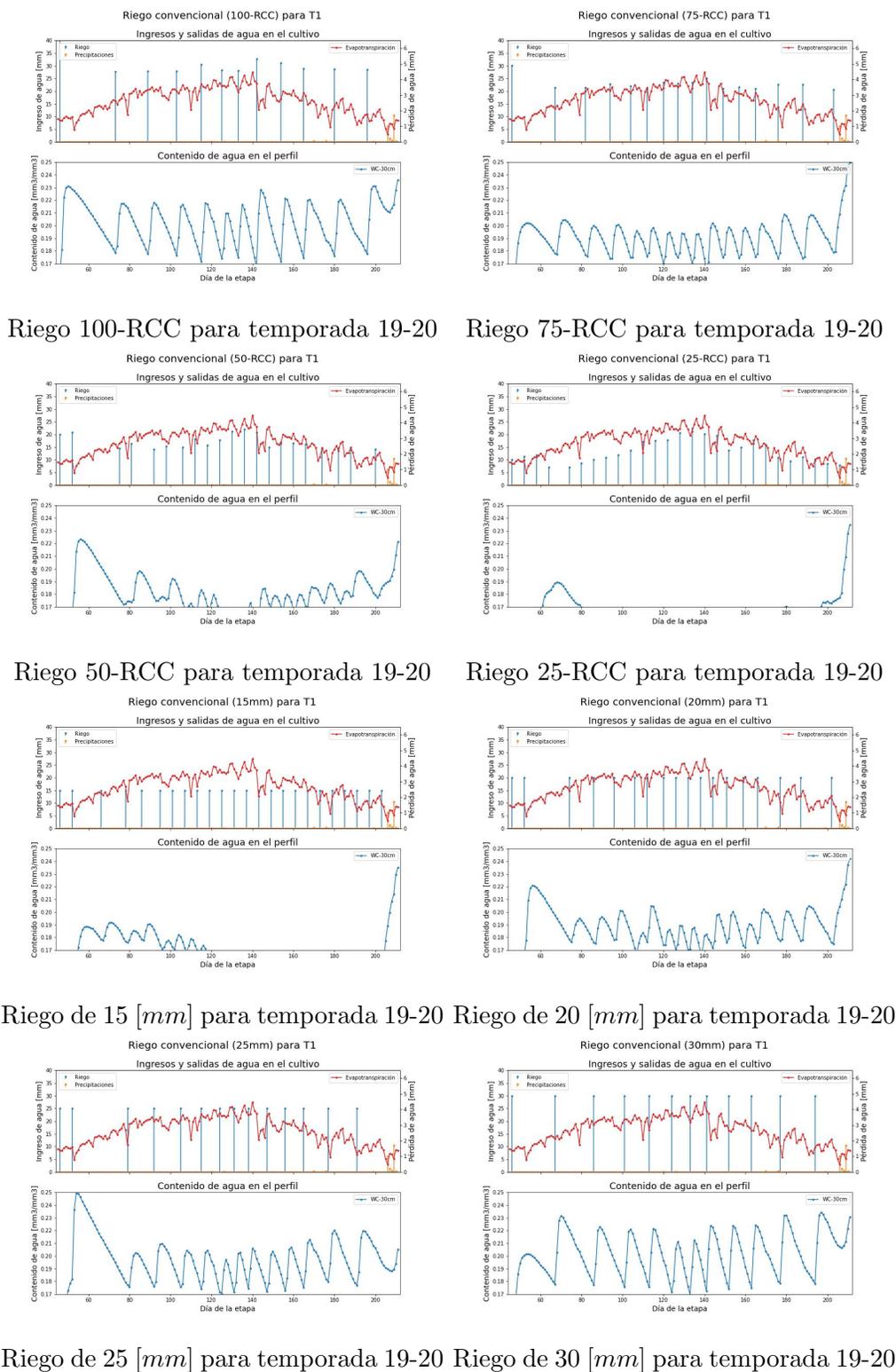
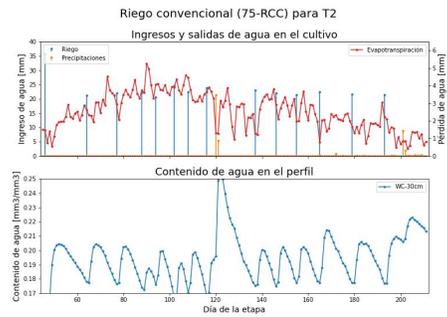
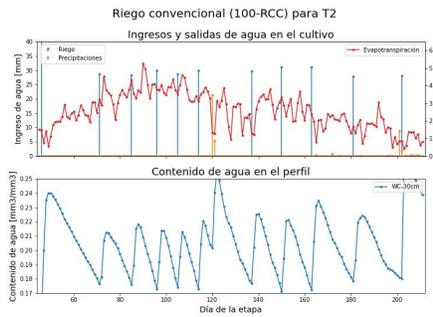
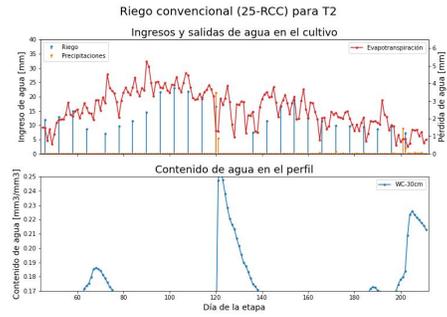
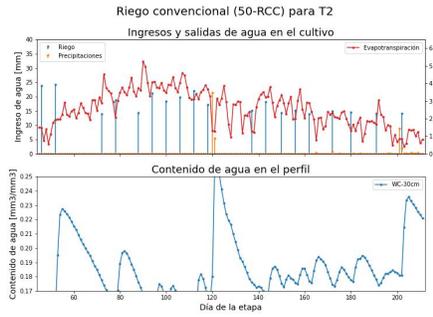


Figura E.13: Estrategias de riego convencional para la segunda etapa, en la temporada 19-20 para el método 1.



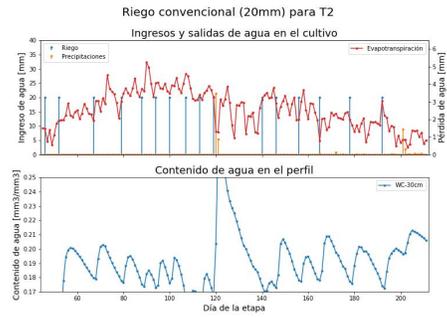
Riego 100-RCC para temporada 20-21

Riego 75-RCC para temporada 20-21



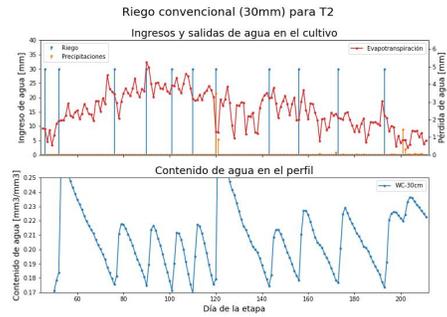
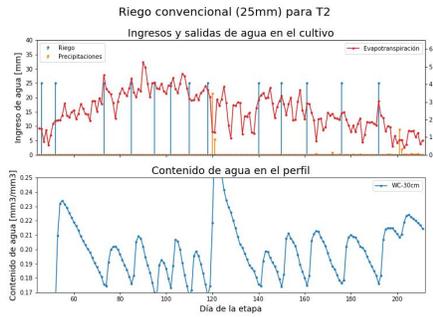
Riego 50-RCC para temporada 20-21

Riego 25-RCC para temporada 20-21



Riego de 15 [mm] para temporada 20-21

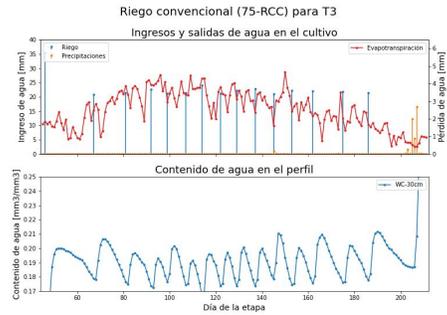
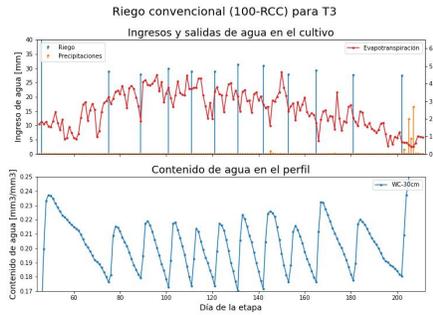
Riego de 20 [mm] para temporada 20-21



Riego de 25 [mm] para temporada 20-21

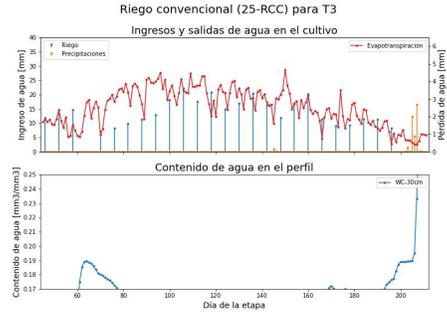
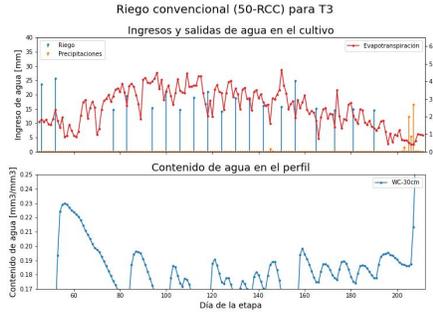
Riego de 30 [mm] para temporada 20-21

Figura E.14: Estrategias de riego convencional para la segunda etapa, en la temporada 20-21 para el método 1.



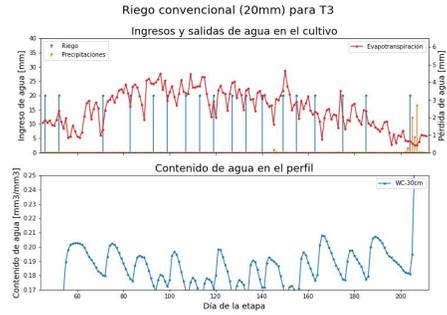
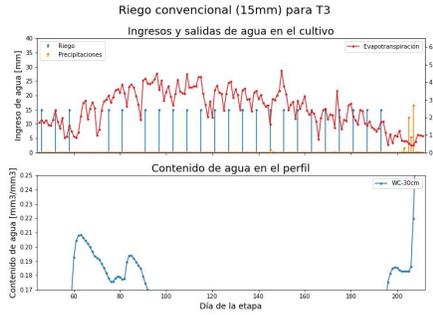
Riego 100-RCC para temporada 21-22

Riego 75-RCC para temporada 21-22



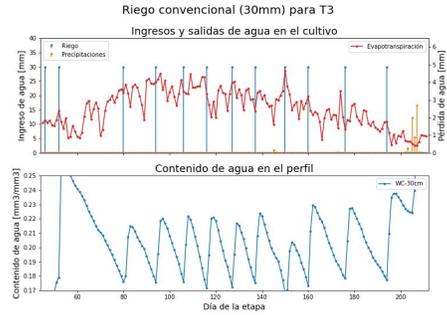
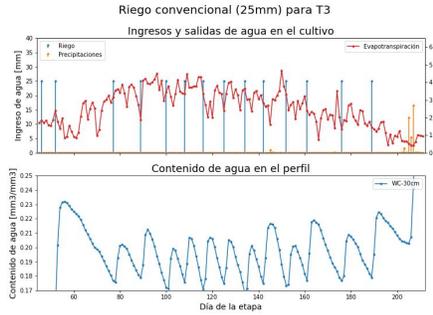
Riego 50-RCC para temporada 21-22

Riego 25-RCC para temporada 21-22



Riego de 15 [mm] para temporada 21-22

Riego de 20 [mm] para temporada 21-22



Riego de 25 [mm] para temporada 21-22

Riego de 30 [mm] para temporada 21-22

Figura E.15: Estrategias de riego convencional para la segunda etapa, en la temporada 21-22 para el método 1.

### E.4.3. Tablas de resumen para Método 2

|                        | <b>Información de riego convencional (Temporada 19-20)</b> |                  |                     |                                     |
|------------------------|--|------------------|---------------------|-------------------------------------|
| <b>R. Convencional</b> | <b>R. total [mm]</b>                                       | <b>N° riegos</b> | <b>R. prom [mm]</b> | <b>Reposición <math>ET_o</math></b> |
| <b>100-RCC</b>         | 262.0035   | 5                | 52.4007             | 0.5195                              |
| <b>75-RCC</b>          | 230.9985   | 5                | 46.1997             | 0.4580                              |
| <b>45mm</b>            | 225  | 5                | 45                  | 0.4461                              |
| <b>40mm</b>            | 240  | 6                | 40                  | 0.4759                              |

Tabla E.5: Resumen del riego aplicado para cada estrategia convencional y reposición de la  $ET_o$ , para la temporada 19-20.

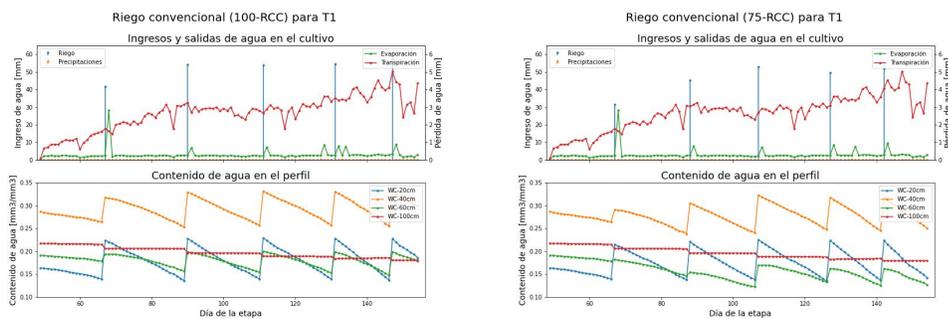
|                        | <b>Información de riego convencional (Temporada 20-21)</b> |                  |                     |                                     |
|------------------------|--|------------------|---------------------|-------------------------------------|
| <b>R. Convencional</b> | <b>R. total [mm]</b>                                       | <b>N° riegos</b> | <b>R. prom [mm]</b> | <b>Reposición <math>ET_o</math></b> |
| <b>100-RCC</b>         | 271.764  | 5                | 54.3528             | 0.5211                              |
| <b>75-RCC</b>          | 244.0563   | 5                | 448.8112            | 0.4680                              |
| <b>45mm</b>            | 225  | 5                | 45                  | 0.4314                              |
| <b>40mm</b>            | 200  | 5                | 40                  | 0.4602                              |

Tabla E.6: Resumen del riego aplicado para cada estrategia convencional y reposición de la  $ET_o$ , para la temporada 20-21.

|                        | <b>Información de riego convencional (Temporada 21-22)</b> |                  |                     |                                     |
|------------------------|--|------------------|---------------------|-------------------------------------|
| <b>R. Convencional</b> | <b>R. total [mm]</b>                                       | <b>N° riegos</b> | <b>R. prom [mm]</b> | <b>Reposición <math>ET_o</math></b> |
| <b>100-RCC</b>         | 259.452  | 5                | 51.8904             | 0.5076                              |
| <b>75-RCC</b>          | 278.019  | 6                | 46.3365             | 0.5439                              |
| <b>45mm</b>            | 270  | 6                | 45                  | 0.5282                              |
| <b>40mm</b>            | 240  | 6                | 40                  | 0.4695                              |

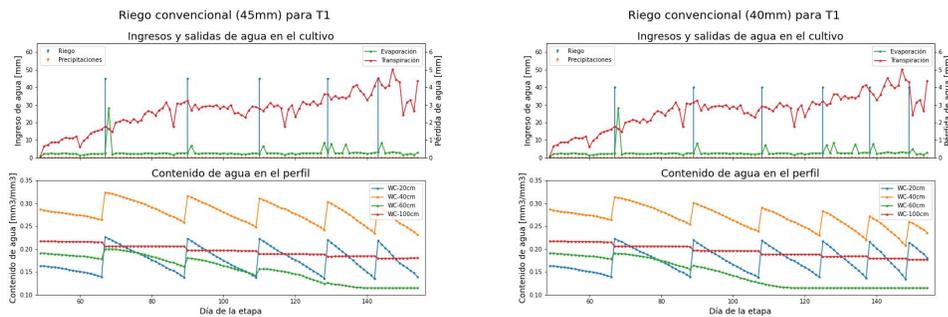
Tabla E.7: Resumen del riego aplicado para cada estrategia convencional y reposición de la  $ET_o$ , para la temporada 21-22.

## E.4.4. Programaciones de riego para Método 2



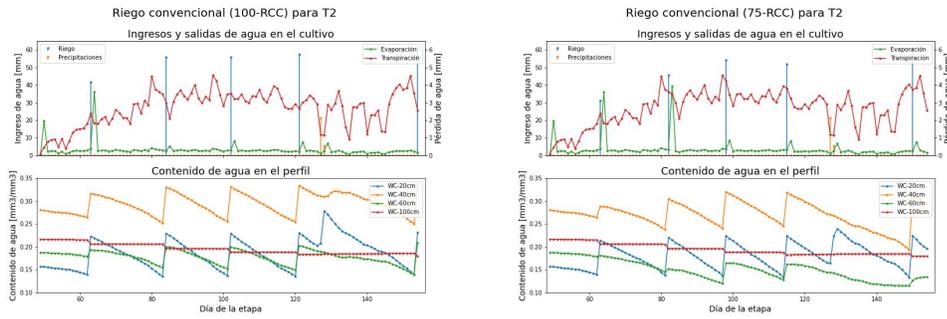
Riego 100-RCC para temporada 19-20

Riego 75-RCC para temporada 19-20

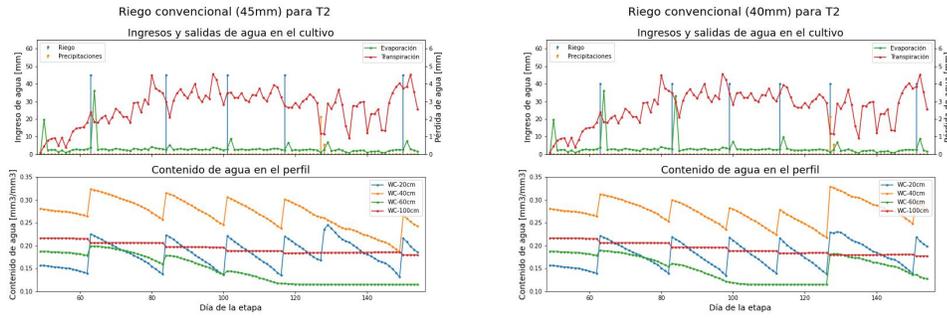


Riego de 45 [mm] para temporada 19-20 Riego de 40 [mm] para temporada 19-20

Figura E.16: Estrategias de riego convencional para la segunda etapa, en la temporada 19-20 para el método 2.

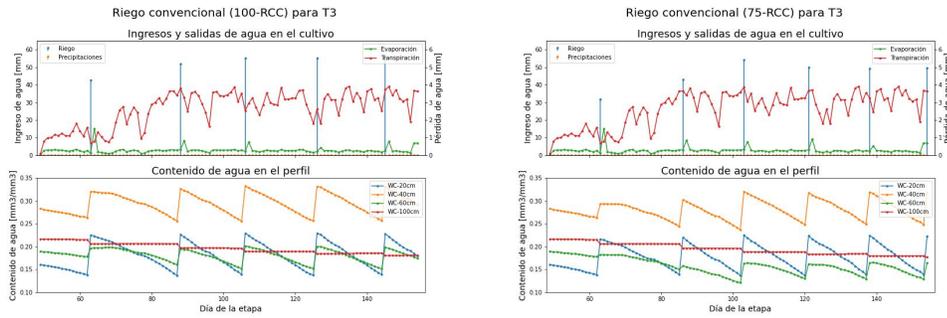


Riego 100-RCC para temporada 20-21 Riego 75-RCC para temporada 20-21

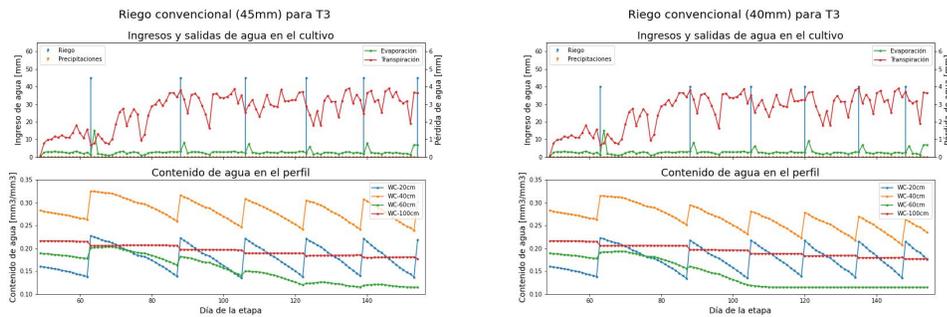


Riego de 45 [mm] para temporada 20-21 Riego de 40 [mm] para temporada 20-21

Figura E.17: Estrategias de riego convencional para la segunda etapa, en la temporada 20-21 para el método 2.



Riego 100-RCC para temporada 21-22 Riego 75-RCC para temporada 21-22



Riego de 45 [mm] para temporada 21-22 Riego de 40 [mm] para temporada 21-22

Figura E.18: Estrategias de riego convencional para la segunda etapa, en la temporada 21-22 para el método 2.