



UNIVERSITAT  
POLITÈCNICA  
DE VALÈNCIA



UNIVERSITAT POLITÈCNICA DE VALÈNCIA

Escola Tècnica Superior d'Enginyeria Informàtica

Distribució òptima duna zona de cultiu mitjançant tècniques  
dintel·ligència artificial.

Treball Fi de Grau

Grau en Enginyeria Informàtica

AUTOR/A: Gasco Pau, Andrea

Tutor/a: Garrido Tejero, Antonio

CURS ACADÈMIC: 2022/2023



# Resum

L'objectiu principal d'aquest treball ha estat el disseny i la implementació d'una aplicació que optimitza la planificació agrícola, determinant la quantitat més òptima de cada cultiu i la seva assignació a la parcel·la de manera que s'obtinga un benefici econòmic elevat i al mateix temps, un consum hídric més baix. Per aconseguir aquest objectiu, s'han plantejat diversos subobjectius que inclouen la recopilació de dades rellevants sobre els cultius, la investigació de tècniques d'intel·ligència artificial aplicables, la modelització del problema i la personalització de l'aplicació segons les necessitats de l'usuari.

En el decurs del treball, s'ha aconseguit crear un algoritme genètic que s'adapta a les necessitats específiques de la planificació agrícola. Aquest algoritme no només considera variables com les necessitats hídriques i les condicions climàtiques, sinó que també ofereix una flexibilitat que permet als usuaris adaptar el model als seus requisits particulars, com ara les dimensions de la terra o el tipus de cultius que desitgen plantar. A més, s'ha utilitzat una arquitectura MVC (Model-Vista-Controlador) que facilita la interacció de l'usuari a través d'una interfície web. Des d'aquesta interfície, els usuaris poden modificar les dades i activar l'algoritme genètic per obtenir solucions òptimes.

Així, aquesta aplicació representa un solució a través de la intel·ligència artificial per a la resolució de problemes complexos en el sector agrícola. A més, també obri la porta a futures investigacions i aplicacions que poden enriquir encara més aquest camp.

**Paraules clau:** cultiu, terreny, intel·ligència artificial, benefici, restriccions, divisió, agràri

---

# Resumen

El objetivo principal de este trabajo ha sido el diseño y la implementación de una aplicación que optimiza la planificación agrícola, determinando la cantidad más óptima de cada cultivo y su asignación en la parcela para obtener un alto beneficio económico y, al mismo tiempo, un menor consumo de agua. Para lograr este objetivo, se han planteado varios subobjetivos que incluyen la recopilación de datos relevantes sobre los cultivos, la investigación de técnicas de inteligencia artificial aplicables, la modelización del problema y la personalización de la aplicación según las necesidades del usuario.

Durante el desarrollo del trabajo, se ha logrado crear un algoritmo genético que se adapta a las necesidades específicas de la planificación agrícola. Este algoritmo no solo considera variables como las necesidades hídricas y las condiciones climáticas, sino que también ofrece una flexibilidad que permite a los usuarios adaptar el modelo a sus requisitos particulares, como las dimensiones del terreno o el tipo de cultivos que desean plantar. Además, se ha utilizado una arquitectura MVC (Modelo-Vista-Controlador) que facilita la interacción del usuario a través de una interfaz web. Desde esta interfaz, los usuarios pueden modificar los datos y activar el algoritmo genético para obtener soluciones óptimas.

Así, esta aplicación representa una solución a través de la inteligencia artificial para la resolución de problemas complejos en el sector agrícola. Además, también abre la puerta a futuras investigaciones y aplicaciones que pueden enriquecer aún más este campo.

**Palabras clave:** cultivo, terreno, inteligencia artificial, beneficio, restricciones, división, agrario

---

# Abstract

The main objective of this work has been the design and implementation of an application that optimizes agricultural planning, determining the most optimal amount of each crop and its allocation in the field to achieve high economic benefits while minimizing water consumption. To achieve this goal, several sub-objectives have been set, including the collection of relevant data on crops, the investigation of applicable artificial intelligence techniques, problem modeling, and application customization according to user needs.

During the course of the work, a genetic algorithm has been successfully created that adapts to the specific needs of agricultural planning. This algorithm not only considers variables such as water needs and climatic conditions but also offers flexibility that allows users to adapt the model to their particular requirements, such as land dimensions or the type of crops they wish to plant. Additionally, an MVC (Model-View-Controller) architecture has been used that facilitates user interaction through a web interface. From this interface, users can modify the data and activate the genetic algorithm to obtain optimal solutions.

Thus, this application represents a solution through artificial intelligence for solving complex problems in the agricultural sector. Moreover, it also opens the door to future research and applications that can further enrich this field.

**Key words:** crop, terrain, artificial intelligence, benefit, restrictions, division, agrarian

---

# Índex

---

<b>Índex</b>	<b>v</b>
<b>Índex de figures</b>	<b>vii</b>
<b>Índex de taules</b>	<b>vii</b>

---

<b>1 Introducció</b>	<b>1</b>
1.1 Motivació	1
1.2 Objectius	2
1.3 Metodologia de treball	2
1.4 Estructura de la memòria	3
<b>2 Context tecnològic</b>	<b>5</b>
2.1 Intel·ligència artificial	5
2.1.1 Intel·ligència artificial aplicada al món de l'agricultura	6
2.1.2 Estudi de diverses tècniques per a la resolució del problema	6
2.1.3 Comparació i selecció de l'estratègia	16
<b>3 Tecnologies utilitzades</b>	<b>19</b>
3.1 Mongo DB	19
3.2 DEAP	20
3.3 Flask	20
3.4 Arquitectura	20
<b>4 Desenvolupament de la solució</b>	<b>23</b>
4.1 Plantejament del problema	25
4.1.1 Restriccions i condicionants	25
4.2 Disseny de l'algorisme genètic	27
4.3 Estructura de l'aplicació	31
4.3.1 Mòduls de la capa Model	31
4.3.2 Mòduls de la capa Vista	33
4.3.3 Mòduls de la capa Controlador	36
<b>5 Avaluació del sistema</b>	<b>37</b>
<b>6 Conclusions</b>	<b>39</b>
6.1 Treballs futurs	40
<b>Bibliografia</b>	<b>41</b>

---

<b>Apèndixs</b>	
<b>A Taula de cultius</b>	<b>45</b>
<b>B Extracte d'una instància de la base de dades</b>	<b>47</b>
B.1 Extracte de la Base de Dades	47
B.1.1 Registre de Cultiu	47
B.1.2 Registre Necessitats hídriques	47
B.1.3 Registre Temperatures	48
B.1.4 Registre Solucions	48
<b>C Objectius de Desenvolupament Sostenible</b>	<b>51</b>

C.1 Reflexió sobre la relació del TFG amb els ODS més relacionats . . . . .	51
---	----

## Índex de figures

---

2.1	Cerca en profunditat	8
2.2	Cerca en amplada	8
2.3	Esquema algorisme genètic.	11
2.4	Representació població i individu.	11
3.1	Diagrama arquitectura MVC.	21
4.1	Simplificació de la representació d'un camp.	25
4.2	Representació genotip-individu. Cromosoma i gen.	27
4.3	Representació fenotip. Diccionari cultiu-nombre plantes.	28
4.4	Esquema base de dades.	32
4.5	Document <i>solucio</i>	33
4.6	Imatge solució.	34
4.7	Imatge pàgina web. Obtenció dades del camp.	34
4.8	Imatge pàgina web. Obtenció informació.	34
4.9	Imatge pàgina web. Selecció cultius	35
4.10	Imatge pàgina web. Selecció cultius temporada	35
4.11	Imatge pàgina web. Missatge	36
4.12	Imatge pàgina web. Solució	36
5.1	Cas prova camp menut	37
5.2	Cas prova camp mitjà	38
5.3	Cas prova camp gran	38
5.4	Cas prova camp gran	38

## Índex de taules

---

2.1	Representació de Solucions. Tècniques intel·ligència artificial	16
2.2	Exploració. Tècniques intel·ligència artificial	16
2.3	Cerca. Tècniques intel·ligència artificial	17
2.4	Eficiència. Tècniques intel·ligència artificial	17
2.5	Garantia d'optimalitat. Tècniques intel·ligència artificial	17
2.6	Inicialització. Tècniques intel·ligència artificial	17
2.7	Coneixement. Tècniques intel·ligència artificial	17
4.1	Taula amb la classificació climàtica de Köpper-Geiger	24
A.1	Cultius i Estacions de l'Any	45

C.1 Objectius de Desenvolupament Sostenible . . . . . 51



---

---

# CAPÍTOL 1

## Introducció

---

L'agricultura ha sigut un pilar fonamental per al desenvolupament de la societat humana al llarg de la història. Des de les primeres civilitzacions fins als nostres dies, la capacitat d'obtenir aliments del terreny ha determinat la supervivència i el progrés de les comunitats. Amb el creixement exponencial de la població, ha augmentat considerablement la demanda d'aliments. Aquesta demanda, juntament al context de la societat en el qual ens trobem actualment, amb una consciència creixent sobre la sostenibilitat i la conservació dels recursos naturals, es torna indispensable optimitzar els recursos agrícoles per a satisfer aquesta creixent necessitat de producció d'aliments.

Davant aquest panorama, sorgix la problemàtica de com distribuir diferents cultius en un terreny definit. No es tracta només de produir aliments, sinó de fer-ho de manera que s'intente maximitzar el benefici econòmic, preservant l'entorn i minimitzant les necessitats hídriques del terreny. Obtenir les quantitats ideals de cada plantació és essencial per extraure el màxim rendiment de la terra i determinar quina organització és més beneficiosa en cada zona concreta de la parcel·la. Per açò, a més, s'han de considerar les característiques inherents tant del terreny com el de cadascun dels cultius. En l'àmbit d'aquest treball quan es parla de cultius, es farà referència a aquelles espècies declarades com hortícoles, és a dir, cultius anuals.

Per tant, el problema que s'intenta resoldre en aquest treball és el de trobar amb quina combinació de quantitats de determinats cultius es poden obtenir el major benefici econòmic amb el menor consum hídric. En altres paraules, es busca una forma eficient de cultivar que siga econòmicament rentable i al mateix temps sostenible amb l'ús de l'aigua.

En aquest context, es planteja la utilització de tècniques d'intel·ligència artificial [1] per a abordar el problema. Aquestes tècniques, que combinen l'anàlisi de dades amb algorismes avançats, poden oferir solucions òptimes o apropiades en funció de les restriccions y aportant una visió holística de la planificació agrícola.

### 1.1 Motivació

---

L'agricultura no és només una activitat que ha acompanyat a la humanitat des de les primeres civilitzacions, és la base fonamental per a la subsistència humana. Cada fruita o verdura que es consumeix prové d'un esforç agrícola. No obstant això, en un món en constant evolució, i amb desafiaments creixents com el canvi climàtic o la degradació dels sòls, incloure noves tecnologies pot jugar un paper important.

Per altra banda, la intel·ligència artificial ha irromput en quasi tots els sectors de la societat, canviant la manera en què operem, prenem decisions i resollem problemes.

La capacitat d'analitzar grans quantitats de dades i optimitzar processos fa que la intel·ligència artificial siga una eina molt demandada.

Així, la motivació d'aquest treball naix en la confluència d'aquests dos mons; l'agricultura, com a base de la humanitat, i la intel·ligència artificial, com a eina transformadora. La combinació d'aquests dos sectors pot conduir a solucions innovadores que aporten beneficis tant als agricultors, consumidors com al planeta que compartim.

## 1.2 Objectius

---

L'objectiu principal d'aquest treball és realitzar el disseny i la implementació d'una aplicació que trobe la quantitat més beneficiosa de cada cultiu, quina assignació d'aquests destinar a cada part de la parcel·la.

Per aplegar a obtenir la solució, s'han plantejat diferents subobjectius.

- Determinar tota la informació rellevant sobre els cultius que es van a utilitzar en el problema per a realitzar correctament la posterior modelització. Aquestes dades seran del tipus: quantitats d'aigua requerides per a cada cultiu, temps de sembra, temperatures de creixement, etc.
- La realització d'una investigació exhaustiva d'algunes de les diferents tècniques d'intel·ligència artificial amb les quals es podria resoldre el problema plantejat. S'avaluaran i compararan les diferents opcions amb l'objectiu d'escollir aquella més adequada al model.
- Durant el modelat del problema, es pretén tindre en molt alta consideració tots aquells elements que porten cap a un camí d'obtenir una solució més sostenible. Amb consideracions com podria ser la proximitat de cultius amb requisits hídrics semblants.
- S'intentarà modelar un problema que done pas a la màxima personalització per part de l'usuari. Per açò, es plantejaran diferents paràmetres i restriccions adaptables a cada cas concret. D'aquests podríem trobar, per exemple, les dimensions del terreny o el nombre de cultius que es volen cultivar.

A través d'aquest treball, a més, s'espera contribuir al desenvolupament i integració de la intel·ligència artificial en el sector agrícola amb la millora de la planificació i gestió de les plantacions.

## 1.3 Metodologia de treball

---

Per abordar la realització d'aquest treball, s'ha dissenyat un pla que contempla diverses fases per a assolir els objectius establerts:

**1. Entrevistes amb l'expert i recopilació de dades** per a detectar els problemes reals amb l'organització de cultius que es poden aplicar al treball de manera concreta, així com la definició de les diferents especificacions d'àmbit agronòmic.

**2. Definició del problema** per analitzar les principals necessitats i requeriments dels agricultors per a dissenyar el problema concret d'optimització a resoldre. A més d'establir les especificacions i restriccions que han de considerar-se en el disseny de la solució.

**3. Investigació de les relacions causa-efecte** dels diferents factors agrícoles i com aquests poden afectar a la planificació dels cultius.

**4. Anàlisi de les tècniques d'intel·ligència artificial** aplicades a problemes d'optimització per a poder seleccionar aquella que millor s'adapte al problema a resoldre.

**5. Implementació** de la tècnica seleccionada.

**6. Validació i avaluació** dels resultats amb la realització de diversos escenaris per a validar l'eficàcia de la solució obtinguda.

**7. Anàlisi dels resultats** i conclusió de la solució obtinguda.

Amb aquest pla de treball, es pretén aplegar a obtindre una solució vàlida al problema plantejat anteriorment.

## 1.4 Estructura de la memòria

---

La memòria es troba estructurada en diverses seccions que faciliten la comprensió del sistema del treball que s'ha realitzar en aquest projecte.

Començant amb el capítol 2, que proporciona una revisió de la investigació i l'anàlisi que s'ha realitzat, incloent una breu història de la intel·ligència artificial i l'agricultura, les tècniques que s'han estudiat i la comparació entre elles.

A continuació, al capítol 3, es descriuen de manera detalla les diferents tecnologies i ferramentes que s'han emprat per al desenvolupament de l'aplicació.

Al capítol 4, s'exposa amb detall tot el procés que s'ha seguit passant per la descripció de totes les dades i components que afecten al sistema, com el disseny proposat per a solucionar-lo i posteriorment, l'estructura que s'ha realitzat per a combinar totes les parts i crear una aplicació funcional.

Al següent capítol, el 5 es presenta l'avaluació de l'algorisme realitzat per a mesurar la seua escalabilitat amb diferents mètriques.

Per últim, al capítol 6 es fa un breu resum de tot el treball, valorant els objectius aconseguits i valorant treballs futurs relacionats amb aquest treball.



---

---

## CAPÍTOL 2

# Context tecnològic

---

En aquest capítol es parlarà d'una sèrie de temes rellevants que serviran tant com per a entendre l'entorn del problema, com per al correcte desenvolupament que es descriu al treball. En primer lloc es parlarà de la intel·ligència artificial [2], incloent algunes de les tècniques que es podrien aplicar al problema plantejat. Seguidament, es centrarà en estudiar l'impacte de la tecnologia i la intel·ligència artificial al món agrari els darrers anys.

### 2.1 Intel·ligència artificial

---

La primera vegada que es va escoltar el terme "intel·ligència artificial" va ser gràcies al científic informàtic John MacCarthy a la conferència de Dartmouth al 1956 [3].

La intel·ligència artificial, també nomenada IA, és un camp d'estudi de la informàtica que s'encarrega de crear programes informàtics amb la capacitat d'executar operacions que són comparables a les que pot realitzar la ment humana, com l'aprenentatge o el raonament lògic (Real Acadèmia Espanyola, 2023).

La idea de dissenyar màquines que puguen simular la intel·ligència humana ha sigut un tema de fascinació al llarg de la història, però va ser al segle XX quan aquesta es va convertir en un camp d'estudi més consolidat [4]. Des d'aleshores, aquest sector ha tingut un creixement exponencial, tant si es parla de coneixements com de les tecnologies emprades per als seus avanços.

Una xicoteta visió de com aquesta ha evolucionat des d'eixe moment es pot veure si fem un repàs a algunes de les etapes per les qual ha anat passant, on els temes d'estudi han anat canviat amb gran celeritat [5, 3].

En primer lloc, la intel·ligència artificial simbòlica, que abasta principalment des dels anys 50 fins als anys 80. En aquest període, la principal preocupació era la de representar el coneixement mitjançant símbols i regles lògiques. Açò va donar lloc a camps d'estudi com el raonament lògic i a que es definira el coneixement com la representació d'estructures lògiques, com bé poden ser els grafs o arbres que representen relacions i jerarquies. En aquests anys es desenvolupen també els llenguatges LISP [6] i Prolog [7], molt característics d'aquest tipus d'IA.

En aquesta etapa sorgeix també el desenvolupament dels sistemes experts, que com descriu Medina (1991) "És un sistema informàtic que incorpora en forma operativa el coneixement d'una persona experimentada de tal manera que és capaç tant de respondre com d'explicar i justificar les seues respostes. [...] La idea és que per a resoldre un problema intel·ligentment el que farem és imitar com un ésser humà resol eixe problema."

La IA simbòlica fou clau per establir la intel·ligència artificial com un camp d'investigació científica.

Paral·lelament, sorgeix altra corrent que destaca per intentar modelar la biologia del cervell compost per xarxes neuronals. Ací va sorgir el perceptró, unitat fonamental per a una gran quantitat de xarxes neuronals [8], però no és aproximadament fins al voltant dels anys 80 que aquesta vessant torna a prendre rellevància gràcies a l'algorisme de retropropagació [9], utilitzat per a l'entrenament de xarxes neuronals. És la època connexionista, i com s'ha dit, destaca per els avanços relacionats amb l'àrea de les xarxes neuronals.

També han hagut grans avanços respecte a la intel·ligència artificial estadística, com per exemple, el formalisme de les Xarxes de Bayes [10] per a representar el raonament. Que fins l'actualitat faciliten l'aprenentatge a partir de l'experiència.

Des d'els seus inicis fins ara, la intel·ligència artificial ha sigut un camp d'estudi ampli i en constant evolució i és fonamental entendre que tots aquests fets són només una breu pinzellada del que ha sigut i continua següent a dia de hui.

Pel que concerneix a aquest treball, es centra en el sector de l'agricultura, per a veure com la IA l'ha influenciat i en quin punt la qüestió que es planteja podria aportar alguna millora. Seguidament, s'estudiaran algunes de les tècniques que es podrien utilitzar per a abordar el problema.

### **2.1.1. Intel·ligència artificial aplicada al món de l'agricultura**

Tradicionalment, l'agricultura ha sigut una feina molt manual i pesada, però els avanços donats a les noves tecnologies, han anat introduint-se poc a poc en aquest sector.

Principalment, les tecnologies aplicades estan més encarades a la utilització de sensors, com els drons, que permeten un monitoratge real de per exemple, temperatures, humitat del sòl o nivells de nutrients. Totes aquestes dades són clau per a poder tindre informació de la situació del terreny i poder intervindre de manera ràpida en el moment en que es detecta alguna anomalia.

Més concretament, les aplicacions d'intel·ligència artificial que s'han donat han estat encarades a la nomenada agricultura de precisió [11] i han sigut per exemple, les de selecció de fruites i verdures en mal estat o de molt bona qualitat, amb la combinació de sensors de colors, grandària o forma, per a aplicar l'aprenentatge automàtic. També amb utilitats directes al camp, com l'aplicació de fertilitzants o la detecció de males herbes en temps real [12].

L'àrea en la que entra el treball que s'esta realitzant és la de la planificació agrícola. Ací la intel·ligència artificial s'ha utilitzat per a assignacions òptimes de recursos o la programació de tasques de manera òptima per a aplicar tractaments en el moment idoni [13]. Dins d'aquest context, aquest estudi es centra en trobar una organització del camp que aporte un major benefici econòmic mentre que s'intenta reduir el consum d'aigua.

### **2.1.2. Estudi de diverses tècniques per a la resolució del problema**

Al camp de la intel·ligència artificial, els algorismes s'han enfrontat a problemes complexos i computacionalment pesats, que tenen enormes espais de cerca, i on buscar solucions òptimes, s'ha tornat una feina molt difícil. En aquest sentit, els mètodes heurístics i metaheurístics han jugat un paper fonamental en la resolució d'aquest tipus de problemes, proporcionant una visió pràctica cap a trobar solucions aproximades en un temps raonable.

Un mètode heurístic[1] és una tècnica d'indagació i descobriment amb un enfocament pràctic per a la resolució de problemes. Es basa en regles empíriques. És a dir, en l'experiència en compte d'en mètodes rigorosos, formules o teories matemàtiques. Aquests algorismes tenen velocitats de còmput més baixes respecte a altres clàssics de cerca. Per contra, no tenen garantia d'optimalitat; no sempre apleguen a la solució òptima.

Un mètode metaheurístic [14] és una tècnica d'optimització avançada que es guia en les heurístiques per a explotar el espai de cerca del problema. Estan dissenyats per resoldre problemes difícils que amb mètodes heurístics no es poden solucionar i es basa en procediments iteratius amb l'objectiu de millor les solucions de l'heurística subordinada.

Després d'entendre les bases del que són els mètodes heurístics i metaheurístics, hi ha una àmplia gamma de tècniques específiques que es poden aplicar dins d'aquests paradigmes generals, com ara els problemes de satisfacció restriccions amb CSP (més concretament amb els problemes d'optimització de restriccions COP [15]), els algorismes genètics, la cerca tabú, els algorismes de colònies de formigues o el refredament simulat [16].

### CSP. Problemes de satisfacció de restriccions

Un CSP (*Constraint Satisfaction Problems*)[17] és un tipus de problemes dins de la rama de la intel·ligència artificial amb l'objectiu de trobar una solució que satisfaci un conjunt de restriccions donades. De manera més concreta al que s'explica en aquest treball, els COP (*Constraint Optimization Problems*), que a més de complir totes les restriccions, busca també maximitzar o minimitzar una determinada funció objectiu [16]. En aquests tipus de problemes hi ha un conjunt de components claus que són: un conjunt de variables finit, un domini finit per a cada una de les variables i les restriccions que limiten les combinacions vàlides de valors per a aquestes variables [18]

Més detalladament, les variables tenen un paper central dins d'aquest tipus de problemes, ja que són els objectes als quals es pretén assignar un valor per a trobar una solució. Cada variable posseïx un nom únic i un conjunt de possibles valors dins del seu domini associat. Això significa que el domini d'una variable especifica els valors que aquesta pot agafar i el seu contingut depèn directament del problema que es vulga resoldre. Les restriccions en un problema de CSP o COP són les regles que defineixen les relacions i limitacions entre les variables i els valors que poden prendre. El tipus de restriccions és molt variat, poder haver restriccions que afecten a una única variable, a diverses, o a moltes. La modelització correcta de les variables amb els seus dominis i les restriccions és un pas fonamental per a una bona resolució posterior.

Les tècniques més comuns per a abordar la resolució d'un CSP o COP són tres: la cerca sistemàtica, tècniques inferencials i les tècniques híbrides [16].

### Cerca sistemàtica

Aquesta és una tècnica que es centra en l'exploració del espai. Pot ser completa, si s'explora tot l'espai, o incompleta si s'explora una part. La primera, al realitzar una cerca completa, garanteixen que si hi ha una solució la trobe, però té un cost computacional molt elevat.

Les dos estratègies de cerca més conegudes són la cerca en profunditat (DSF) i la cerca en amplària (BFS)[2].

En la cerca en profunditat es comença a explorar des de el node arrel i s'avança tant com siga possible dins d'una mateixa rama. Quan ja no es pot explorar més, retrocedeix per realitzar la cerca en una altra rama distinta, com es pot veure a la figura 2.1. Aquesta estratègia és útil per a explorar capes més profundes, però té l'inconvenient de que es pot quedar atrapat en profunditats infinites si no es controla adequadament.

Per contra, en la cerca en amplària es centra en explorar l'espai de manera gradual i sistemàtica. Es comença per el node arrel, i s'exploren tots els nodes en el mateix nivell abans de passar a explorar el nivell següent com es pot veure a la figura 2.2. Aquesta tècnica, a diferència de l'anterior, troba solucions més pròximes al node arrel i no aprofundeix en una rama concreta.

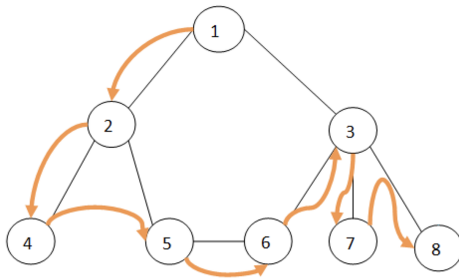


Figura 2.1: Cerca en profunditat

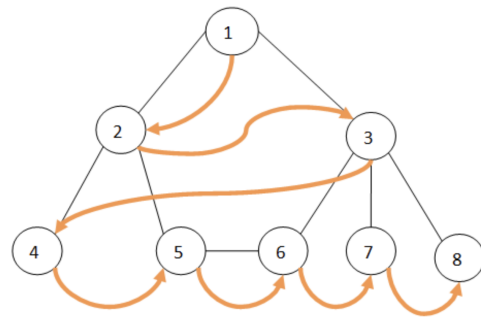


Figura 2.2: Cerca en amplada

## Tècniques d'inferència

Si bé les cerques sistemàtiques com la cerca en profunditat i en amplària proporcionen una cerca ordenada per a l'exploració, les tècniques inferencials afegien una capa d'intel·ligència i eficiència a aquest procés. Aquestes tècniques es centren en deduir informació addicional. És a dir, deduir noves restriccions a partir de les ja existents, de manera que s'eliminen opcions inconsistentes i es creen noves restriccions implícites. Aquest procés converteix el problema anterior amb un nou CSP equivalent, permetent acotar l'espai de cerca inicial.

En aquesta tècnica es pot parlar dels nivells de consistència, que es refereix al grau en el que s'ha aplicat la inferència per a eliminar valor invàlids del domini de les variables. Cada un d'ells implica un grau diferent de restriccions eliminades.

Alguns dels distints nivells de consistència que es poden trobar són: la consistència de node (1-consistència), la consistència d'arc (2-consistència), la consistència de camins (3-consistència), i la consistència global (k-consistència)[17].

- **Consistència de node (1-Consistència).** És la consistència de nivell més baix. A aquest nivell es verifica que per a cada variable i el seu domini, hi ha al menys una assignació de valors que complisca les restriccions unàries, si hi ha. Una restricció unària és aquella que involucra una sola variable.
- **Consistència d'arc (2-consistència).** Aquesta fa un pas més enllà de la 1-consistència i es centra en les relacions binàries, que són aquelles que involucren a dos variables. En aquest nivell es verifica que per a cada parell de variables rela-



cionades entre sí, hi haja al menys una combinació de valors que als seus dominis satisfacen amb aquesta restricció.

- **Consistència de camins (2-consistència).** Aquesta és un nivell encara més elevat que involucra les restriccions ternàries. És a dir, aquelles que involucren tres variables. Ací, es garanteix que totes les restriccions ternàries siguen coherents i es complisquen en totes les possibles combinacions.

- **Consistència global (k-consistència).** És el nivell més alt i general. Aquest s'assegura que totes les restriccions, unàries, binàries, ternàries, i de qualsevol ordre superior  $k$ , siguen coherents en totes les combinacions de valors entre totes les variables. Aquest és el nivell més fort de consistència, però també el més costós computacionalment.

Cal destacar que aquesta tècnica es centre en reduir significativament l'espai de cerca, però s'ha d'implementar d'igual manera alguna tècnica de cerca per a aplegar a trobar una solució.

## Tècniques híbrides

Als dos apartats anteriors, s'ha parlat de les tècniques de cerca, que en general, resulten exponencialment costoses amb la grandària del problema i de les tècniques per a eliminar valors inconsistents, permetent acotar l'espai de cerca. Amb açò, es pot realitzar un preprocès per a detectar i eliminar inconsistències locals per a realitzar una posterior cerca més acotada. L'acotació vindrà donada pel nivell de preprocès que s'haja realitzat. Les tècniques híbrides combinen aquests dos passos per a donar lloc a algorismes de cerca híbrids, que inclouen les tècniques inferencials al propi procés de cerca.

Algunes de les tècniques híbrides són els algorismes *Look-Backward* i *Look-Ahead* [17]. Els primers, es centren en retrocedir i explorar l'espai de cerca quan es troben situacions sense eixida, és a dir, retrocedir de manera intel·ligent quan es troben conflictes. I els segons, que es centren en realitzar comprovacions cap avant en cada instanciació per tal de reduir l'espai de cerca i detectar situacions sense eixida abans de que ocorreguen.

Aquestes tècniques en la resolució de CSP busquen aprofitar els avantatges dels diferents enfocaments de resolució, a fi d'abordar els problemes de manera més eficient i efectiva.

## Avantatges i desavantatges dels CSP

Una vegada vist en que consisteixen CSP en l'àmbit de l'optimització, en aquesta secció van a comentar-se els avantatges i desavantatges que tenen aquests tipus de problemes.

En primer lloc, els avantatges dels quals es pot parlar en aquesta secció són els següents.

- **Modelat efectiu.** Els CSP proporcionen un marc sòlid per a modelar els problemes. Es poden representar problemes reals amb variables, dominis i restriccions, el que pot fer que la comprensió del problema pugui ser major.

- **Abstracció.** Permeten l'abstracció de les restriccions i les seues relacions entre les variables. Açò simplifica la representació i resolució de problemes que involucren restriccions interconnectades.

- **Solucions òptimes i factibles.** Mitjançant tècniques de cerca i optimització, amb un CSP es poden trobar solucions òptimes i factibles que complisquen totes les restriccions establertes. Açò és molt útil si hi ha requisits molt estrictes.

Per contra, aquests tipus de problemes també presenten alguns desavantatges.

- **Complexitat en espais grans.** En problemes on hi ha un gran quantitat de variables i restriccions, l'espai de cerca es pot tornar massa complexe i necessitar una quantitat molt elevada de recursos computacionals.

- **Escalabilitat limitada.** L'escalabilitat pot ser complicada en alguns problemes ja que l'augment del nombre de variables pot augmentar exponencialment la grandària del problema.

- **Dependència de la modelització.** L'èxit o no del problema depèn en gran mesura del modelat adequat de les variables, el domini i les restriccions. S'ha de tindre un coneixement molt elevat de l'entorn del problema.

Aquests problemes ofereixen una metodologia efectiva per a abordar problemes d'optimització si es té una representació clara i concisa per a poder aplegar a solucions òptimes. No obstant això, poden tindre més dificultats si els espais de cerca són grans, hi ha explosió combinatòria de variables.

## Algorismes genètics

Un algoritme genètic és una tècnica d'intel·ligència artificial que es basa en la biologia evolutiva[19] i en la genètica per a resoldre problemes d'optimització [20] i recerca.

Amb les paraules del científic Goldberg, "els algorismes genètics són algorismes de cerca basats en la mecànica de la selecció natural i de la genètica natural. Combinen la supervivència del més apte entre estructures de seqüències amb un intercanvi d'informació estructurat, encara que aleatoritzat, per a constituir així un algorisme de cerca que tinga alguna cosa de les genialitats de les cerques humanes". [21]

Seguint aquest procés, es pot dir que l'esquema bàsic d'un algoritme genètic és el següent:

1. **Inicialització:** generar una població inicial d'individus (possibles solucions al problema).

2. **Avaluació:** avaluar l'aptitud (*fitness*) de cada individu de la població. La funció d'aptitud mesura que tan bo és un individu en comparació amb la resta.

3. **Selecció:** seleccionar els individus per a ser pares i participar en el creuament.

4. **Creuament:** a partir dels individus seleccionats, es crea una nova generació d'individus. Açò es realitza a partir de dos individus per a crear un o més descendents.

5. **Mutació:** amb un valor de probabilitat, es realitza una modificació aleatòria en els individus. Açò ajuda a introduir variabilitat en la població.

6. **Reemplaçament:** es reemplaça la població antiga per la nova generació d'individus.

7. **Acabament:** l'algorisme acaba quan es compleix algun criteri d'acabament, com un nombre màxim de generacions o una solució que aplega a un valor desitjat.

Els processos de selecció, creuament i mutació es realitzen fins que es compleix el criteri d'acabament [22].

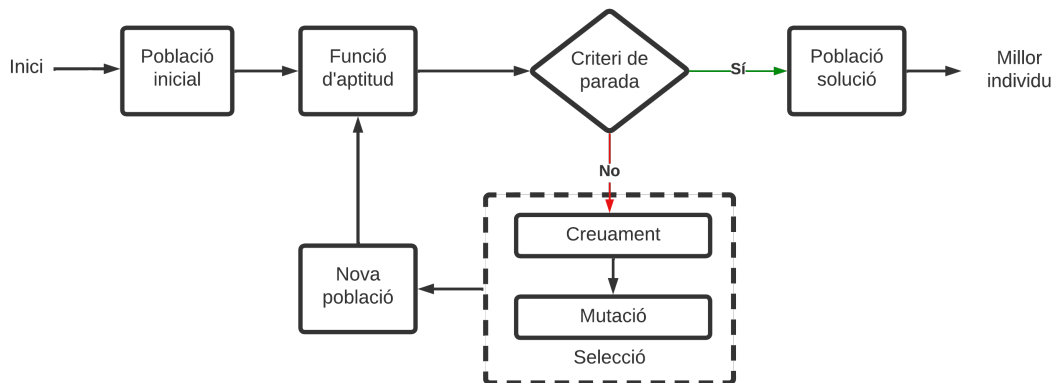


Figura 2.3: Esquema algorisme genètic.

D'aquesta manera, seguint l'esquema de la figura 2.3, per obtenir una solució al problema, es parteix d'un conjunt inicial d'individus, anomenats població, normalment generats de manera aleatòria. Cada un d'aquests individus són una potencial possible solució del problema. Aquests individus evolucionaran, es creuaran i mutaran seguint unes pautes establertes per aplegar a obtenir una solució lo suficientment bona al problema plantejat[20].

Al món de la biologia es troben els cromosomes, que són les estructures del nucli de la cèl·lula que contenen el material genètic (el que es coneix com ADN). Dins de cada cromosoma, existeixen segments específics denominats gens, que codifiquen instruccions determinades per a característiques biològiques. La totalitat de gens d'un organisme representant la configuració genètica es denomina genotip. Amb tot açò, el que és observable reflectint per exemple, les característiques físiques d'una persona, és el fenotip [23].

En analogia amb la biologia, els individus d'aquests algorismes representaran el genotip, és a dir, seran una estructura de dades que simbolitzarà la solució del problema. La descodificació d'un individu donarà al seu pas, el fenotip, que correspondrà amb la representació real de la solució.

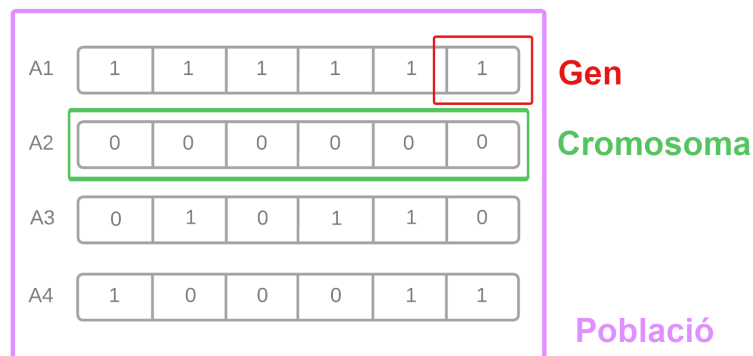


Figura 2.4: Representació població i individu.

És molt important escollir una bona representació dels individus de la població, per tal de tindre la certesa de poder representar totes les solucions del problema [23].

Una vegada obtinguda una població vàlida inicial, la funció d'aptitud o funció *fitness* avaluarà que tan bo és un individu o cromosoma de la població. Dit d'una altra manera, es mesurarà la qualitat d'una solució al problema. Aquesta funció rebrà com a entrada un individu i tornarà un valor numèric, que indicarà com de bona és eixa solució en relació a altres solucions de la població.

És important destacar que aquesta funció es dissenya per a cada problema concret, sent crucial per identificar la solució amb el valor més òptim, ja siga maximitzant-lo o minimitzant-lo.

A l'hora de realitzar el disseny de la funció d'aptitud, s'han de tindre en compte certes consideracions respecte a que l'individu que es va a avaluar no complisca les restriccions establertes per al problema.

Les tècniques utilitzades per corregir aquest problema són les següents:

- **Rebuig.** Simplement es descarten els individus que no compleixen les restriccions. És més comú utilitzar-la als problemes de factibilitat.
- **Penalització.** En lloc de descartar els individus no factibles, se'ls assigna una penalització al fer el càlcul de la seua aptitud. Fa que aquests individus siguen més difícils de seleccionar. La clau d'aquesta estratègia és saber determinar com de gran ha de ser la penalització per a cada restricció violada.
- **Reparació.** Quan es detecta un individu que no compleix les restriccions, es modifica per convertir-lo en un individu factible.
- **Preservació.** Aquesta estratègia implica mantenir una població separada d'individus factibles i utilitzar-la per a influir en la població principal.

El criteri d'acabament és el que defineix quan s'ha de parar d'executar l'algorisme[16]. És també essencial establir un criteri de parada adequat per garantir que l'algorisme no s'execute indefinidament. Els criteris d'acabament més usuals són tres: en primer lloc, quan els millors individus de la població representen solucions lo suficientment bones per al problema. En segon lloc, quan la població convergeix, el que significa que la major part de la població té el mateix valor per a un paràmetre determinat. I per últim, quan s'arriba al nombre de generacions màximes específiques.

Mentre no es complisca la condició de parada, per a passar a la nova població es realitzen un conjunt d'operadors genètics o evolutius [24] que utilitzen la generació anterior per crea una nova. En primer lloc, es fa la selecció dels individus que se reproduiran en el futur cicle. Després es faran els creuaments i mutacions seguint uns criteris establerts i per últim, es realitzarà el reemplaçament dels individus generats per els de la generació anterior. Al finalitzar açò, es tornarà a calcular l'aptitud de la nova població i es comprovarà de nou si es compleix el criteri d'acabament.

Com bé s'acaba de dir, els operadors genètics són mecanismes que permeten la manipulació i evolució de la població. Aquests són: la selecció, el creuament i la mutació.

## Selecció

Aquest operador simula el procés natural de la selecció, on els individus més aptes tenen més oportunitats de ser escollits per reproduir-se i passar les seues característiques

a les següents generacions. Així i tot, encara que predominen els millors individus, s'ha de donar també la oportunitat de reproducció als individus menys bons.

La selecció d'individus juga un paper molt important per determinar la direcció de l'evolució de la població. Per tant, és molt important mantindre un equilibri entre l'exploració i l'explotació.

L'exploració es refereix a la capacitat de l'algorisme per a buscar i descobrir noves àrees o regions dins del espai de solucions, sense quedar-se atrapat en solucions locals o subòptimes. Aquesta exploració permet al sistema descobrir noves i potencials millors solucions que no havien sigut considerades anteriorment. D'altra banda, la explotació es centra en aprofundir i optimitzar dins de les regions del espai de solucions que ja s'han identificat com a prometedores o amb altes aptituds. L'explotació busca millorar les solucions existents, afinant-les per a obtenir el màxim rendiment.

Alguns dels mètodes més utilitzats són:

- **Selecció per ruleta.** És un tipus de selecció proporcional, on els individus més aptes tenen més probabilitat de ser seleccionats, per no la certesa. A cada individu se li assigna una "porció" de una ruleta proporcional a la seua aptitud. Així, els millors individus rebran una porció de la ruleta major que la que rebran els pitjors. Per a seleccionar un individu es "gira" la ruleta i l'individu on es para és el seleccionat. És un mètode senzill però ineficient a mesura que la grandària de la població augmenta.
- **Selecció per torneig.** És un tipus de selecció jeràrquica, on els individus competeixen entre ells. Es seleccionen aleatòriament dos o més individus de la població i es compara la seua aptitud. L'individu amb major aptitud dins d'aquest grup menut és seleccionat. Aquest procés es repeteix fins que es tenen suficients pares per al creuament.
- **Selecció per *ranking*.** La població es classifica segons la seua aptitud, i es seleccionen els individus basant-se en la seua posició dins d'aquesta classificació. Pot fer-se de manera proporcional o assignant més probabilitat als individus amb millor *ranking*.
- **Selecció generacional.** La descendència d'una generació és la següent generació de pares. Per tant, no es conserven individus entre generacions.

En resum, la selecció és el mètode que guia l'evolució de la població cap a solucions més òptimes, assegurant-se que les bones solucions es propaguen i influeixen a les futures generacions.

## Creuament

Una vegada s'han seleccionat els pares, aquest operador genètic simula el procés natural de la reproducció, on els dos individus pares seleccionats anteriorment es combinen per produir un o més descendents.

El creuament serveix per introduir nova diversitat en la població. Si s'agafen dos individus amb una aptitud elevada, i s'obté una descendència que comparteix gens d'ambdós individus, és molt probable que els descendents tinguin una bona aptitud.

Existeixen també una multitud d'algorismes de creuament, però els més usuals són:

- **Creuament d'un punt.** És la tècnica més senzilla. Es selecciona un punt aleatori en els cromosomes dels pares per a generar dos segments diferenciats. Els segments d'abans d'aquest punt s'intercanvien entre ells per formar dos nous individus.
- **Creuament de dos punts.** Al igual que al creuament d'un punt, es seleccionen dos punts aleatoris en els cromosomes dels pares. Aquests segments s'intercanvien per a formar nous descendents.
- **Creuament uniforme.** Cada gen del descendent es selecciona de manera aleatòria d'un dels dos pares.

Respecte al mètode de creuament de certs punts, es poden afegir més punts per generar creuaments multipunt. Així i tot, existeixen estudis que desaconsellen esta tècnica, ja que quan més segments s'originen és més fàcil perdre les característiques que li caracteritzaven una aptitud elevada. El creuament de dos punts, aporta una millora considerable que el d'un punt.

En resum, el creuament és un mecanisme que permet la combinació de solucions existents per a explorar noves àrees del espai de solucions i, al mateix temps, explotar les àrees conegudes amb bones solucions.

## Mutació

La mutació és l'operador genètic que consisteix en simular el procés biològic on canvis aleatoris es produeixen en el material genètic d'un individu. Es sol realitzar després del creuament, a un o més dels descendents se'ls fa una mutació amb probabilitat  $P_m$ . Aquesta probabilitat sol ser molt baixa, normalment menor a l'u%.

Al igual que en els altres operadors evolutius, existeixen diferents tipus de mutacions. Les més utilitzades són:

- **Mutació d'intercanvi.** Dos gens dins del cromosoma s'intercanvien la posició en la qual es troben.
- **Mutació d'inserció.** Un gen és eliminat d'una posició i inserit en una altra.
- **Mutació de bit.** S'utilitza en representacions binaries. Un bit del cromosoma és invertit. De 0 a 1 o de 1 a 0.
- **Mutació gaussiana.** S'utilitza en representacions numèriques. Un valor generat a partir d'una distribució gaussiana es suma al valor del gen.

En resum, la mutació dona més possibilitats de que l'algorisme genètic tinga una diversitat suficient i evite la convergència prematura.

## Avantatges i desavantatges dels algorismes genètics

Després d'haver estudiat en que consisteixen els algorismes genètics en l'àmbit de l'optimització, en aquesta secció es van a veure quins avantatges i desavantatges posseeixen a l'hora de resoldre un problema [22].

Els avantatges que presenten aquest tipus d'algorismes són diversos.

- **Exploració ampla del espai de solucions.** Els algorismes genètics tenen la capacitat d'explorar un espai de cerca bastant gran degut a la generació diversa d'individus.
- **Escalabilitat i paral·lelisme.** Aquests algorismes es poden paral·lelitzar de manera ràpida, fent que siguin una bona opció per a problemes amb un alt grau de còmput.
- **Mínims locals.** Són algorismes per als quals pot ser més difícil, que no impossible, quedar atrapats en un mínim local, ja que poden eixir d'ells mitjançant operadors de creuament i mutació.
- **Adaptabilitat.** Poden ser aplicats a una gran varietat de problemes d'optimització sense tindre un coneixement detallat del domini.
- **Cerca multiobjectiu.** Són una bona opció per resoldre problemes amb diversos objectius, buscant solucions que representen un compromís entre diferents criteris.

Per contra, aquests algorismes també presenten alguns desavantatges.

- **Requereixen l'ajust de paràmetres.** Definir una representació del problema pot ser complex, i la robustesa i eficàcia del bon funcionament d'aquests algorismes depén en gran mesura d'una bona representació de les possibles solucions.
- **Convergència lenta.** Aquests algorismes poden requerir més generacions per aplegar o no a una convergència. Açò també dependrà dels paràmetres utilitzats. Com la grandària de la població o el nombre de generacions.
- **Convergència prematura.** En contra del punt anterior, també es pot donar el cas de que s'aplegue a una convergència prematura si un individu molt apte es reproduïx de manera abundant, disminuint la diversitat de la població. Açò es dona quan hi ha poblacions menudes.
- **No garanteix la solució òptima.** Aquests algorismes busquen solucions aproximades al que podria ser l'òptima, però no assegura que ho siga.
- **Representació i codificació.** L'elecció de la representació de les solucions poden influir, com s'ha dit abans, a l'eficàcia de l'algorisme, portant a una cerca ineficient.

En resum, els algorismes genètics són una ferramenta per a abordar problemes d'optimització on es pretén explorar espais de cerca molt amplis. Tenen avantatges significatius en termes d'adaptabilitat, però poden presentar desavantatges quan parlem de convergència i d'obtenir garanties respecte aplegar a solucions òptimes.

### Altres tècniques

Com s'ha dit abans, hi ha moltes més tècniques per resoldre problemes d'optimització. En aquesta secció es va a fer una breu menció a algunes d'aquestes, però cal mencionar que no són ni molt menys, totes les que existeixen.

Una d'aquestes tècniques, és la cerca tabú, que s'utilitza per realitzar la cerca d'una solució en espais de gran dimensió. La característica principal d'aquesta tècnica és que té una memòria a curt termini, en la que s'emmagatzemen les últimes solucions visitades. L'objectiu d'açò és evitar caure en òptims locals.

Seguidament, es troba la tècnica de colònia de formigues, inspirada en el comportament d'aquests insectes, especialment en la forma en que aquestes troben el camí més curt entre l'aliment que troben i el seu niu. Les formigues van deixant un rastre de feromones al sòl per marcar el seu camí, d'aquesta manera, amb analogia amb aquest comportament, aquesta tècnica es caracteritza per anar deixant una mena de feromona artificial, seleccionant-se el camí amb més quantitat d'aquestes. Açò es va actualitzant en cada iteració, donant pas a que les millors solucions tinguin més quantitat assignada.

Altra tècnica, anomenada refredament simulat, es basa en el comportament de l'aer en el moment en que se li apliquen altes temperatures i posteriorment, es refreda lentament eliminant defectes. El component clau d'aquesta tècnica és el camp de la temperatura, que controla la probabilitat de que s'accepte una solució o no. A mesura que s'avança en l'algorisme, la component temperatura va disminuint el seu valor. Aquest fet mesura com de ràpid es va a convergir cap a una solució.

Aquestes, com s'ha dit, són només una part molt menuda de la representació de tècniques metaheurístiques per a la resolució de problemes de cerca i optimització. La utilització d'una altra dependrà en gran mesura del problema que es vulga resoldre.

### 2.1.3. Comparació i selecció de l'estratègia

Una vegada s'han estudiat amb més profunditat tants el problemes d'optimització de restriccions i els algorismes genètics, i s'han vist per damunt algunes altres estratègies, s'ha realitzat un estudi comparatiu entre elles per poder seleccionar una que s'adeqüe al problema que s'intenta resoldre.

S'han comparat tant les representacions de les solucions, com la inicialització i el coneixement necessari per a modelar un problema, com l'exploració i cerca que utilitza cada una, l'eficiència i l'optimalitat.

Tècnica	Representació de Solucions
CSP/COP	Assignació de variables a valors del domini.
Algoritmes Genètics	Individus d'una població.
Cerca Tabú	Solucions en una llista tabú.
Colònia de Formigues	Camins amb feromones.
Refredament Simulat	Estats en un espai de cerca.

**Taula 2.1:** Representació de Solucions. Tècniques intel·ligència artificial

Tècnica	Exploració
CSP/COP	Determinista.
Algoritmes Genètics	Estocàstica.
Cerca Tabú	Estocàstica i determinista.
Colònia de Formigues	Estocàstica.
Refredament Simulat	Estocàstica.

**Taula 2.2:** Exploració. Tècniques intel·ligència artificial

Com es pot veure a les diferents taules [2.1](#) [2.2](#) [2.3](#) [2.4](#) [2.5](#) [2.6](#) [2.7](#), aquestes tècniques tenen un enfocament molt diferent per aplegar a obtenir una solució. Tant el modelat



<b>Tècnica</b>	<b>Cerca</b>
CSP/COP	Propagació i reducció de dominis.
Algoritmes Genètics	Selecció, creuament, mutació.
Cerca Tabú	Moviments locals amb memòria.
Colònia de Formigues	Actualització de feromones.
Refredament Simulat	Moviments aleatoris controlats por temperatura.

**Taula 2.3:** Cerca. Tècniques intel·ligència artificial

<b>Tècnica</b>	<b>Eficiència</b>
CSP/COP	Escalabilitat limitada.
Algoritmes Genètics	Alta escalabilitat.
Cerca Tabú	Moderada a alta.
Colònia de Formigues	Moderada.
Refredament Simulat	Moderada a alta.

**Taula 2.4:** Eficiència. Tècniques intel·ligència artificial

<b>Tècnica</b>	<b>Garantia d'optimalitat</b>
CSP/COP	Pot ser òptima.
Algoritmes Genètics	No garantida.
Cerca Tabú	No garantida.
Colònia de Formigues	No garantida.
Refredament Simulat	No garantida.

**Taula 2.5:** Garantia d'optimalitat. Tècniques intel·ligència artificial

<b>Tècnica</b>	<b>Inicialització</b>
CSP/COP	No sensible.
Algoritmes Genètics	Sensible.
Cerca Tabú	Moderadament sensible.
Colònia de Formigues	Sensible.
Refredament Simulat	Sensible.

**Taula 2.6:** Inicialització. Tècniques intel·ligència artificial

<b>Tècnica</b>	<b>Coneixement</b>
CSP/COP	Requereix detalls.
Algoritmes Genètics	Menys dependent.
Cerca Tabú	Requereix detalle.
Colònia de Formigues	Moderadament dependent.
Refredament Simulat	Menys dependent.

**Taula 2.7:** Coneixement. Tècniques intel·ligència artificial

del problema, com el tipus d'exploració i cerca són ben distintes i la utilització d'un o l'altre dependrà complement del problema al qual ens enfrontem.

Al cas que concerneix aquest treball, té més sentit la utilització d'un algorisme genètic, a causa de diversos motius.

En primer lloc, el nombre de variables i restriccions que posseeix el problema fan que tinga una grandària que amb un CSP podria dur a còmputos més elevats. A causa també de la quantitat de variables que componen el problema, el modelatge es pot dificultar si no es té un coneixement molt alt del domini.

Seguint amb la grandària de les dades, amb un algorisme genètic és molt fàcil l'adaptabilitat cap a la introducció de nous cultius o noves restriccions que afecten a tot el conjunt, mentre que açò podria ser una feina més costosa si es fera amb per exemple, CSP.

Per últim, altra cosa molt important a tenir en compte, és que per al problema que s'està intentant resoldre, no hi ha una única solució vàlida. Per tant, amb la cerca d'una solució apropada a l'optimalitat ja s'estaria en possessió d'una resolució correcta.

Per totes aquestes causes, s'ha pres la decisió de realitzar la implementació del sistema comentat al capítol 1 amb un algorisme genètic.

---

---

## CAPÍTOL 3

# Tecnologies utilitzades

---

A la actualitat, per a un correcte desenvolupament dels projectes, l'elecció de les tecnologies adequades es primordial per tal de garantir aplicacions robustes i eficients. Donada la complexitat i diversitat de les aplicacions actuals, es fa casi imprescindible treballar en més d'una eina. La unió de diferents tecnologies permet, a més, extraure el màxim potencial de cadascuna.

La base principal sobre la que s'ha construït aquest treball, és el llenguatge de programació Python per a la part *backend*, en la seua versió 3.11.4 [25]. Un llenguatge interpretat que destaca per la seua facilitat de llegibilitat del codi. La seua versatilitat i senzillesa permet realitzar la integració de diferents eines i biblioteques amb facilitat. Per a la part visual o *frontend* s'ha emprat HTML [26] i CSS [27], amb alguna pinzellada de JavaScript per a un poc d'interacció.

En aquest apartat, per tant, se centrarà en les ferramentes utilitzades per desenvolupar l'aplicació.

### 3.1 Mongo DB

---

MongoDB [28] és una base de dades NoSQL [29] de codi obert que es basa en un model orientat a documents. És a dir, en compte de treball en taules i esquemes fixes, emmagatzema les dades en documents BSON [30] que poden ser niades.

Aquesta característica mencionada, significa que amb MongoDB no s'ha de treballar, si no es vol, amb estructures rígides i uniformes, com les taules tradicionals amb files i columnes. En altres paraules, permet la recollida de dades no homogènies, com llistes o imatges, cosa que li proporciona una flexibilitat elevada.

Aquest tipus de dades, degut a la flexibilitat de format, permeten ser niades dades dins d'altres. Es pot utilitzar per tant, una estructura jerarquitzada, on un document continga subdocuments o elements secundaris.

A més, té una gran escalabilitat, ja que al treballar amb estructures heterogènies no es fa necessària modificar tot l'esquema de la base de dades. Aquestes s'adapten a nous conjunts de dades.

En l'àmbit del projecte que s'ha plantejat en aquest treball, s'ha utilitzat MongoDB per emmagatzemar les dades recollides i per guardar les solucions obtingudes, treballant amb facilitat. S'han guardat estructures com llistes, diccionaris de dades o imatges.

A més, s'han realitzat operacions bàsiques de tipus CRUD (*Create, Read, Update, Delete*) [31] que són operacions de creació, lectura, modificació i esborrament dels components que formen una base de dades persistent. Aquesta metodologia no és exclusiva de

MongoDB, sinó que es duen a terme en molts llenguatges de programació distints, així com en base de dades relacionals i no relacional, com és el cas de MonogDB.

## 3.2 DEAP

---

DEAP (Distributed Evolutionary Algorithms in Python)[32] és un *framework* [33] o marc de treball dissenyat per a Python en l'àmbit de la computació evolutiva. Proporciona un esquema de treball estructurat i coherent. La seua filosofia es centra en fer que els algorismes siguen explícits i les estructures en que el treballa siguen transparents, facilitant així la comprensió i adaptació a les necessitats concretes de cada treball.

Un dels elements a destacar de DEAP és que permet la creació ràpida d'individus de manera personalitzada. Aquesta característica és ideal per adaptar i modelar els individus de l'algorisme genètic segons les necessitats específiques de cada projecte, sense tindre la limitació de treballar en estructures predefinides.

També es podria destacar, encara que no ha sigut necessària la seua utilització al treball plantejat, la seua capacitat d'integració amb ferramentes de multiprocessament i SCOOP [34], aprofitant múltiples processadors per accelerar l'execució dels algorismes genètics.

## 3.3 Flask

---

Flask [35] és marc de treball de micro servicis que s'utilitza per al desenvolupament d'aplicacions web a Python. Es basa en Werkzeug [36], una biblioteca WSGI [37] i Jinja2 [38], un motor de plantilles també per a Python. Flask és característic per ser un *framework* senzill i minimalista, convertint-lo en una opció popular per a la realització de aplicacions web menudes.

Flask segueix el model MVC [39], però al mateix temps, no imposa cap estructura rígida, permetent que el desenvolupador agafe la llibertat, si vol, d'organitzar el seu projecte de la manera més beneficiosa.

A diferència d'altres marcs de treballs per a tecnologies web, com Django [40], Flask no imposa per defecte cap dependència a un model de base de dades específic. Açò permet una gran flexibilitat i control sobre l'aplicació que desitge implementar.

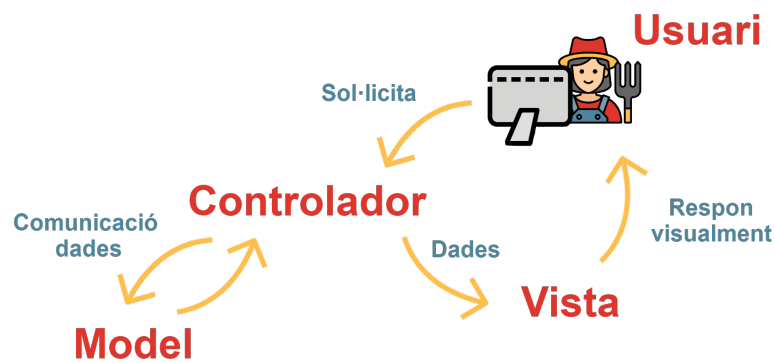
Al marc d'aquest treball, Flask s'ha utilitzat amb l'objectiu de realitzar la part visual de l'aplicació i per altra banda, per a la comunicació amb el motor de l'algorisme genètic.

## 3.4 Arquitectura

---

L'arquitectura seleccionada per a la realització d'aquest treball ha sigut la MVC (Model Vista Controlador) [39] degut a la seua disposició de treball que ofereix una estructura clara i modular per al desenvolupament d'aplicacions.

Aquesta arquitectura es caracteritza per tindre una clara separació de la responsabilitat que cada component realitza en l'aplicació. Primerament, la capa lògica o de negoci és la capa Model i és el motor de l'algorisme conjuntament amb la comunicació amb la base de dades. En segon lloc, la capa Vista on es troba part visual de la aplicació realitzada com bé s'ha dit, amb HTML i CSS. Per últim, la capa Controlador, que a aquest projecte equival a Flask, per realitzar la comunicació entre l'algorisme i l'usuari. Aquesta configuració es pot veure a la figura 3.1.



**Figura 3.1:** Diagrama arquitectura MVC.

Aquest model de treball on cada component pot actuar de manera independent permet l'escalabilitat i el manteniment del projecte fàcilment així com simplificar la implantació de proves unitàries i d'integració. Com a inconvenient a aquest model, es podria dir que en projectes molts grans, aquesta diferenciació pot provocar una lleugera disminució del rendiment de l'aplicació.

MVC no és l'única arquitectura i es poden trobar altres formes d'estructurar una aplicació com pot ser el model MVVM [41] que ofereix una vinculació de dades més directa, o inclús treballa a nivell de micro servicis [42] per a sistemes distribuïts.

La selecció d'utilitzar MVC per a aquest treball ha vingut donada sobretot per la facilitat d'integració dels diferents components que conformen l'aplicació de manera modular.



# Desenvolupament de la solució

---

Aquest capítol, se centrarà en el desenvolupament que s'ha realitzat per obtenir la solució al problema plantejat al capítol 1.

Com també s'ha mencionat amb anterioritat quan s'ha parlat de les tecnologies utilitzades al capítol 3, aquest treball és el fruit de la mescla de diferents ferramentes. En primer lloc, amb la creació d'una base de dades amb MongoDB per poder contindre totes les dades necessàries. Després, amb la creació del motor algorítmic per resoldre el problema. I per últim, amb Flask per la creació d'una interfície d'usuari senzilla. A través d'aquest capítol, es detallarà com s'han utilitzat i combinat per aplegar a obtenir una solució.

Gràcies a les entrevistes realitzades amb l'expert, es té a l'abast la informació rellevant per resoldre el problema plantejat. Es pot dir, aleshores, que es tenen dos tipus de dades diferents. Les del terreny o relacionades amb la zona i les dels cultius. Algunes d'aquestes, les del terreny, poden ser modificades per l'usuari, ja que han de ser lo més apropiades a la situació de cadascun d'ells. Mentre que les relacionades amb els cultiu no es poden modificar.

### Dades del terreny

- **Dimensions del terreny.** Es tindrà tant la longitud com l'amplada en metres. Per recomanació de l'expert i per simplicitat s'ha optat per treballar en zones de cultiu rectangulars.
- **Distància entre files.** Els camps es treballen per files de cultius paral·les entre elles. La distància entre cada una d'aquestes files pot ser canviada. Per defecte, és d'un metre.
- **Utilització de tractor.** Es tracta d'una variable booleana a tindre en compte si al terreny a treballar es farà ús d'aquesta ferramenta.
- **Organització aleatòria/per necessitat hídrica.** Una variable per poder decidir si d'esquerra a dreta els cultius s'ordenen de manera aleatòria controlada o de menys a més necessitat hídrica.
- **Tipus de recollida mensual/en diversos mesos.** Amb aquesta variable es pot escollir si la solució al problema tindrà una recollida dels cultius d'un sol mes, o de diversos mesos.

Si s'escollix la primera opció es considera que tot el cultiu es recol·lecta durant un mes, i el benefici serà el d'eixe sol mes. Si s'escollix la segona opció, es considera que vol mantindre el camp en producció durant un temps determinat. Açò fa que

la quantitat de cada cultiu es planteja escalonadament, per obtenir de igual manera una recol·lecció escalonada. Aquest paràmetre influenciarà directament la funció d'avaluació ja que si s'escollix la segona opció, el nombre de plantes estarà repartit entre els mesos desitjats.

- **Temperatures.** Es tracta d'una llista amb la mitjana de les temperatures, en graus, del any 2022. Per proximitat, s'ha treballat amb temperatures de la zona de la Comunitat Valenciana i no poden ser modificades[43]. L'usuari podrà escollir entre cinc zones climàtiques segons la classificació de Köppen-Geiger [44]. Aquestes seran Morella i Montanejos a la província de Castelló, València i Oliva a la província de València i Villena a la província d'Alacant. Açò es pot veure a la figura 4.1

Província	Població	Classificació	Detalls
Castelló	Morella	Cfb	Templat sense estació seca. Estiu templat.
	Montanejos	Csb	Templat. Estiu sec i templat.
València	València	Bsh	Estepa càlida.
	Oliva	Csa	Templat. Estiu sec i calorós.
Alacant	Villena	Bsk	Estepa gelada.

Taula 4.1: Taula amb la classificació climàtica de Köppen-Geiger

## Dades de cultius

Totes les dades que s'han obtingut relacionades a cada cultiu són estàtiques, és a dir, no es poden modificar per l'usuari [45, 46]. Aquestes es troben guardades a la base de dades del projecte. S'adjunta un apèndix amb la taula dels cultius utilitzats A.

- **Distància entre plantes.** Es tracta de la distància en metres que hi ha dins d'una fila entre cada una de les plantes que la componen.
- **Benefici mensual** És una llista amb la mitjana del benefici mensual per kilograms de cultiu en euros del any 2022, extrets de la Llotja de València.
- **Família.** Es tracta de la categoria taxonòmica que agrupa a diverses espècies que comparteixen característiques morfològiques semblants a la qual pertany el cultiu.
- **Kilograms per planta.** Aquest valor fa referència als kilograms que produeix un determinat cultiu.
- **Temperatura òptima màxima i mínima.** Els dos valor a les quals els cultius es desenvolupen de manera ideal.
- **Temperatura extrema màxima i mínima.** Són aquelles temperatures que si són sobrepasades, la planta no sobreviu o redueix dràsticament la seua productivitat.
- **Necessitats hídriques.** Es tracta d'una llista amb els valor de la demanda d'aigua per a cada mes del any, indicant la necessitat de la planta per a poder dur a terme la seua activitat. És un valor calculat en litres i depèn tant del tipus del cultiu com de la zona climàtica.
- **Varietat local.** Variable booleana que indica si es troba en estat *true* el cultiu es tracta d'una varietat local valenciana i en *false* si no ho és.



## 4.1 Plantejament del problema

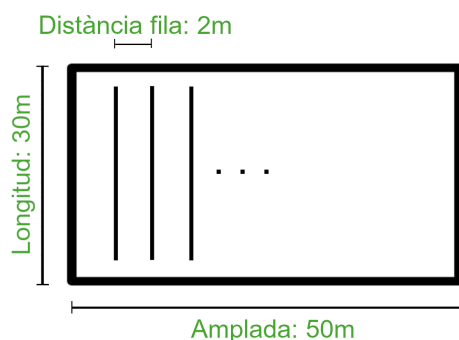
Tenint en compte les condicions i restriccions concretes de cada element, es desitja obtenir quina quantitat de cada hortalissa, en kilograms, en quina zona de la parcel·la s'ha de plantar concretament, així com el moment exacte de l'any. Tot seguint dos objectius principals:

- **Maximitzar el benefici econòmic** total de la zona cultivada, que ve donat pel benefici mensual que té atorgat cada hortalissa.
- **Minimitzar la necessitat hídrica** total del terreny, que ve donada per la necessitat hídrica. Dada que també es pot trobar a les variables de cada verdura.

### 4.1.1. Restriccions i condicionants

Aquesta secció va a centrar-se en les restriccions que s'han de complir a per poder resoldre el problema de manera factible i quin va a ser el seu efecte en la solució final donada. Algunes d'aquestes s'han resolt mitjançant l'algorisme genètic a en el moment de calcular la quantitat a obtenir, mentre que altres s'han contemplat al fer el preprocessament o postprocessament de les dades que ja es tenen gràcies a la base de dades, o al usuari que les ha introduït al programa.

Els camps que es destinen a la plantació agrícola s'organitzen per files de cultius. Per saber de quantes files, i per tant, quants metres són cultivables en el terreny, es necessita saber la grandària i la distància que hi ha entre aquestes. Com bé s'ha comentat al principi del capítol aquestes dades pertanyents al terreny i són modificables, cada problema concret tindrà unes característiques distintes. El càlcul dels metres de la fila és una informació fonamental per poder afrontar el problema. Es pot veure el càlcul amb un exemple d'un camp amb una longitud i una amplada de 30 i 50 metres respectivament, i una distància entre files de dos metres [4.1](#).



**Figura 4.1:** Simplificació de la representació d'un camp.

Amb aquesta simplificació de com es veuria un camp amb les característiques descrites, es pot veure com calcular els metres que realment es poden destinar per a la plantació. Per a calcular el nombre de files verticals que hi ha a la zona simplement s'ha de dividir l'amplada per la distància entre files. A este valor, se li resta de nou la distància entre files per assegurar que es planta a eixa distància del borde del camp. De manera simbòlica, pel mateix fet, es restarà de la longitud una distància de 0.5 metres per cada extrem. D'aquesta manera, per al exemple descrit, s'obtindria un camp compost per 23 files de 29 metres de llargària.

Per a facilitar la introducció dels cultius al problema, amb aquestes dades, es va a considerar que es treballa amb una única fila, *fila total*, que serà la suma de totes les files. Seguint amb l'exemple anterior, es treballaria amb una *fila total* de 667 metres de llargària. Sabent açò, la restricció a complir en el problema és que de cada cultiu  $i$  la suma de les distàncies entre cultius  $d_i$  ha de ser igual o menor a la longitud  $L$  de la *fila total*.

$$\sum_{i=1}^n d_i \leq L_{\text{fila total}} \quad (4.1)$$

Seguidament, perquè un cultiu pugui ser plantat en un terreny, s'han de donar unes condicions climàtiques que siguin beneficioses per al correcte desenvolupament vegetatiu i/o la correcta fructificació de la planta. Per tant, si la temperatura mitjana de la zona  $T_{m,zona}$  en la que es desitja plantar, en l'època seleccionada és menor o major a les temperatures mínimes o màximes extremes del cultiu,  $T_{\text{min-ext, cultiu}}$  i  $T_{\text{max-ext, cultiu}}$  respectivament, aquest no podrà ser plantat a eixe camp.

$$T_{\text{min-ext, cultivo}} \leq T_{m,zona} \leq T_{\text{max-ext, cultivo}} \quad (4.2)$$

Els melons han de complir una regla especial. Primerament, si es vol obtenir producció de melons sense llavors, per qüestions agronòmiques, s'han de plantar una tercera part de melons amb llavors. Açò vol dir, a més, que cada tres files que hi haja de melons sense llavors, hi d'haver-hi una amb melons amb llavors. Aquests últims, no es poden plantar a no ser que hi haja dels primers.

$$\text{Melons sense llavors} = M_{\text{sense}}, \quad \text{Melons amb llavors} = M_{\text{amb}} \quad (4.3)$$

$$M_{\text{amb}} = \frac{1}{3} \times M_{\text{sense}} \quad (\text{Si } M_{\text{sense}} > 3) \quad (4.4)$$

$$M_{\text{amb}} > 0 \Rightarrow M_{\text{sense}} > 3 \quad (4.5)$$

Si l'usuari selecciona que al camp a treballar s'utilitza el tractor com a ferrament agrícola, la grandària de la distància entre files  $D_{\text{files}}$  s'augmenta en un 20% respecte a la distància entre files original  $D_{\text{original}}$  introduïda per l'usuari.

$$D_{\text{files}} = \begin{cases} D_{\text{original}} & \text{si no hi ha tractor} \\ 1.2 \times D_{\text{original}} & \text{si hi ha tractor} \end{cases} \quad (4.6)$$

De la mateixa manera, l'usuari també ha de seleccionar com a mínim tres cultius distints per a poder obtenir una solució. A banda, l'usuari pot indicar quina quantitat mínima en kilograms de cada cultiu desitja obtenir de la solució a calcular. Aquest requisit pot ser zero.

$$N_{\text{cultius}} \geq 3 \quad (4.7)$$

$$Q_{\text{min},i} \geq 0, \quad i = 1, 2, \dots, N_{\text{cultius}} \quad (4.8)$$

$$Q_{\text{obtinguda},i} \geq Q_{\text{min},i}, \quad i = 1, 2, \dots, N_{\text{cultius}} \quad (4.9)$$

A l'hora de col·locar els cultius en el camp, s'ha de complir que totes les plantes d'un determinat cultiu han d'estar juntes. No es poden mesclar cultius.

També, si és possible, al moment d'assignar un ordre, es tindran en compte algunes consideracions respecte al cultiu de la tomaca. Són dos: la primera, si al camp hi ha tomaques i carabassins, aquests dos es trobaran junts. La segona, si al camp hi ha tomaques, albergínies i garrofó, aquests es trobaran separades.

Aquestes qüestions han sigut determinades per l'expert que ha proporcionat tota la informació rellevant per a poder obtindre una solució correcta al sistema.

## 4.2 Disseny de l'algorisme genètic

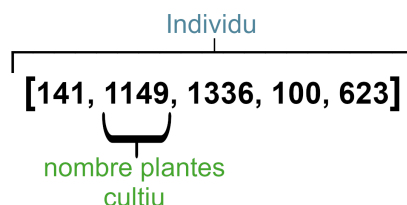
Per poder abordar un problema amb la utilització d'algorismes genètics, és fonamental plantejar un modelat inicial correcte, ja que aquest disseny establirà la base per al bon rendiment de l'execució de l'algorisme. L'esmentat modelat abasta diverses etapes, com el disseny de l'individu, la inicialització de la població i les decisions sobre les mutacions, creuaments i demés accions que marquen el camí pel quan ha d'anar l'algorisme per aplegar a obtindre una solució desitjada i consistent.

Per tant, en aquest apartat va a profunditzar-se un poc més en totes les decisions que s'han pres a l'hora de dissenyar les parts bàsiques de l'algorisme. Per fer això, es va a seguir l'esquema de l'apartat 2.1.2.

En primer lloc, és crucial establir com va a ser l'individu. Aquesta decisió determinarà que la resta d'operacions de l'algorisme es realitzen de manera correcta, pel fet de que tots els passos que es donen cap a trobar una solució es faran sobre l'individu. Al cas que s'està comentant en aquest treball, l'algorisme genètic tindrà la missió d'obtindre quina quantitat de cada cultiu s'ha de destinar al terreny.

Per facilitar el disseny i la implementació posterior, s'ha optat per que l'individu siga una llista d'enters. La llargària de la llista vindrà determinada pel nombre de cultius que es seleccionen i cada enter serà el nombre de plantes de cada un dels cultiu.

En analogia amb la figura 2.4, l'individu és l'estructura que representa la solució del problema i és equivalent al genotip, mentre que els valor, és a dir el nombre de plantes de cada cultiu, actuen com a gens. Un conjunts de molts d'aquests individus formaran la població. Així, al exemple de la figura 4.2, es troba un individu format per cinc cultius diferents, amb el valor de la quantitat de plantes corresponent a cada cultiu.



**Figura 4.2:** Representació genotip-individu. Cromosoma i gen.

A més d'açò, és també molt important la representació del fenotip, doncs serà el que es pot entendre del problema. En el context de la qüestió que s'està resolent, donarà la informació de a quin cultiu correspon cada un dels valors obtinguts al genotip. La correspondència genotip-fenotip es realitzarà amb un mètode que convertirà la llista en un diccionari amb clau el nom del cultiu i valor el nombre de plantes associades. A l'exemple de l'individu anterior, es troba el fenotip representat a la figura 4.3 on s'haurien de

cultivar 141 plantes de pebrera italiana, 1149 plantes de pebrera roja, i així successivament.

```
{
  "pebrera_italiana": 141,
  "pebrera_roja": 1149,
  "tomaca": 1336,
  "alberginia": 100,
  "cogombre": 623
}
```

**Figura 4.3:** Representació fenotip. Diccionari cultiu-nombre plantes.

A l'hora de fer la inicialització d'aquests individus s'ha optat per un mètode d'inicialització personalitzat. Com bé s'ha comentat abans, la longitud de l'individu ve determinada pel nombre de cultius que es marquen per entrar a formar part del problema. La funció d'inicialització crearà aleshores, llistes d'enters de valors aleatoris per a ser avaluats posteriorment. Els valors aleatoris, s'escolliran dins d'un rang determinat, per a no inicialitzar individus amb valors impossibles d'aconseguir.

Per definir aquest rang s'han tingut en compte alguns dels paràmetres que introdueix l'usuari. Es calcularà per a cada cas concret, i vindrà determinat per la *longitud de la fila* i per el *nombre de cultius*. Seguidament, per decidir quin valor s'inicialitzarà s'han tingut en compte alguna de les condicions explicades al apartat 4.1.1.

Primerament, si algun dels cultius dels introduïts té com a requisit mínim el valor zero, l'element corresponent a eixe cultiu es definirà amb "0", mentre que la resta es definiran aleatòriament dins del rang. Per altra banda, si tots els cultius tenen com a requisit mínim zero, tots s'inicialitzaran amb valor aleatoris. Açò es fa per que es dona per suposat que, si es donen tots els requisits amb valors baixos, tots han de tindre les mateixes opcions d'aparèixer a la solució mentre que si és sols un o diversos cultius, s'entén que no són tan rellevants com la resta. La solució serà tant vàlida si estan com si no.

Seguidament, a la inicialització es crearan individus que complisquen la restricció del nombre de melons amb llavors i sense llavors. D'aquesta manera, si els melons es troben entre els cultius seleccionats, quan es determine un valor aleatori per als melons amb llavor, s'afegiran a l'individu la tercera part de melons sense llavors.

Amb l'individu i el criteri d'inicialització definits, ja es podrà crear la població inicial per començar a aplicar els diferents operadors genètics. Els individus que formen part d'aquesta població seran ja potencials candidats per ser la solució del problema.

En el marc d'aquest treball, s'ha optat per començar l'algorisme amb una grandària de població elevada, amb una dimensió de 5000 individus. Açò s'ha determinat així per diverses raons: en primer lloc, per la diversitat genètica; crear una població inicial elevada proporciona una major diversitat genètica, el que es tradueix en que l'espai de solucions a explorar és més ampli. I en segon lloc, per la convergència cap a solucions més òptimes; amb un espai de cerca més gran, és més difícil que l'algorisme es quede estancat en solucions locals o subòptimes. Pel que fa al nombre de generacions es troba entre les 1500.

L'algorisme pot acabar la seua execució per dos vies distintes. La primera, si es compleixen el nombre de generacions establertes. La segona, si durant les últimes 100 generacions no ha hagut una millora significativa al llindar declarat. Al cas del treball, és de 0.05.

L'avaluació dels individus, ocorre a mans de la funció d'avaluació. Aquesta s'encarrega de mesurar que tan bo és un individu. En aquest treball, això significa obtindre un

benefici econòmic elevat mentre s'obté un demanda hídrica més baixa. En aquest punt és al que es contempen la major part de les restriccions dels sistema.

Com s'ha comentat al inici del capítol, l'usuari pot escollir el tipus de recol·lecció dels cultius. Aquesta dada influeix directament amb un dels paràmetres de la funció d'avaluació.

En quant al benefici econòmic, si s'ha escollit el tipus de recol·lecció en diversos mesos la funció d'avaluació actuarà de la següent manera: En primer lloc, per a cada cultiu representat a l'individu es buscaran els mesos de major benefici per a eixe cultiu dins dels mesos seleccionats. Posteriorment, el nombre de plantes que s'han associat a eixe cultiu es divideixen entre el nombre de mesos que hagen sorgit de la cerca. Açò es fa degut a que si la recol·lecció es manté durant una període de temps llarg es considera que la plantació i la recol·lecció es realitza escalonadament.

Per altra banda, si s'ha seleccionat el tipus de recol·lecció en un sol mes la funció d'avaluació actuarà primerament de la mateixa manera que l'altra, fent una cerca per a trobar els mesos de major benefici dins dels escollits. La diferència radica en que ací s'avaluara tot el nombre de plantes associat a cada cultiu amb un sol benefici.

Evidentment, açò es farà així si es compleixen la resta de restriccions del problema, que per als dos casos es realitzarà igual. En primer lloc, s'avaluarà la temperatura. La temperatura associada a cada cultiu per a la zona del camp seleccionada de manera que s'ha comentat a la secció 4.1.1 a més de tindre en compte els extrems en els quals es pot o no plantar, s'ha tingut en compte el rang de les temperatures per a modificar positivament o negativament el benefici econòmic. De manera que, si la temperatura de la zona  $T_{m,zona}$ , es troba dins del rang entre les dos temperatures òptimes del cultiu  $T_{opt-min, cultiu}$ ,  $T_{opt-max, cultiu}$ , s'augmenta el benefici en un 20%, mentre que si es troba en el rang entre les extremes i les òptimes del cultiu  $T_{min-ext, cultiu}$ ,  $T_{opt-min, cultiu}$ , per exemple, es penalitza el benefici econòmic en un 20%.

$$\begin{aligned}
 & \text{Si } T_{m,zona} < T_{min-ext, cultiu} \text{ o } T_{m,zona} > T_{max-ext, cultiu} : \\
 & \quad \text{No es pot plantar} \\
 & \text{Si } T_{opt-min, cultiu} \leq T_{m,zona} \leq T_{opt-max, cultiu} : \\
 & \quad \text{Benefici} = \text{Benefici} + \text{Plantes} \times \text{Benefici base} \times 1.2 \\
 & \quad \text{En altre cas :} \\
 & \quad \text{Benefici} = \text{Benefici} + \text{Plantes} \times \text{Benefici base} \times 0.8 \tag{4.10}
 \end{aligned}$$

Seguidament, s'avaluen els requeriments mínims establerts. Per a cada cultiu es calcula el nombre de plantes mínimes que es necessiten per garantir els kilograms demanats per l'usuari. De manera que si aquest requeriment mínim no es garanteix es redueix considerablement el benefici total. Aquesta reducció té en compte el nombre de cultius que es calculen per a la penalització. És a dir, la grandària de l'individu. De manera que quan més cultius, més penalització.

$$\begin{aligned}
 & \text{Si } \text{Plantes} < \text{Plantes mínimes} : \\
 & \text{Benefici total} = \text{Benefici total} - 2000 \times \text{longitud individu} \tag{4.11}
 \end{aligned}$$

Posteriorment s'avalua si es compleix el requisit associat als melons, que han d'estar en proporció tres a un. De manera que si no es compleix, el penalitza el benefici total.

$$\begin{aligned} &\text{Si } \text{individu}[\text{idx\_melo\_sense}] \neq 3 \times \text{individu}[\text{idx\_melo\_amb}] : \\ &\text{Benefici total} = \text{Benefici total} - 500 \times \text{longitud individu} \end{aligned} \quad (4.12)$$

Per últim, pel que respecta al càlcul del benefici econòmic s'avalua si la suma dels metres que ocupa cada un dels cultius és menor a la grandària de fila total que te el camp.

$$\begin{aligned} \text{Espai utilitzat} &= \sum_{\text{cultiu, plantes}} (\text{plantes} \times \text{distància entre plantes}) \\ &\text{Si } \text{Espai utilitzat} > \text{Longitud de fila} : \\ \text{Benefici total} &= \text{Benefici total} - 2000 \times \text{longitud individu} \end{aligned} \quad (4.13)$$

Pel que fa a l'avaluació de la necessitat hídrica, que és més senzilla, també és un poc diferent respecte al tipus de recol·lecció, però la basa és la mateixa. Es tenen en compte els mesos en els que es vol fer la collita i conjuntament amb les necessitats hídriques per a cada cultiu en cada zona climàtica que es tenen a la base de dades es realitza la suma de la necessitat pel nombre de plantes de cada cultiu. A més, s'afigen al càlcul els dos mesos posteriors no repetits, ja que es considera una aproximació del temps que estan els cultius plantats abans de tenir producció.

Per tant, amb aquestes dades, s'avalua a cada individu de manera que s'intenta obtenir un benefici econòmic elevat al mateix temps que es vol obtenir una necessitat hídrica més baixa.

Respecta al criteri de selecció s'ha utilitzat un selector multiobjectiu proporcionat per DEAP que s'anomena NSGA-II (*Non-dominated Sorting Genetic Algorithm II*) [47]. Aquest algoritme s'utilitza comunament per resoldre problemes que impliquen diversos objectius en conflicte, com és el cas del problema que es descriu. En primer lloc, es realitza una classificació no dominada de la població, on es divideixen els individus depenent de la seua dominància en diferents fronts. De manera que un individu pertany al primer front si cap altre el domina. Un altre individu pertany al segon si sols es superat per individus del primer, i així successivament. Després d'aquesta classificació, es realitza el càlcul de la "distància d'amuntegament" per a cada individu. Aquesta distància és una mesura que indica com de prop es troben els veïns d'un individu en l'espai. Quan major és la distància, més valor té l'individu. Finalment, la selecció es fa escollint tots els individus del primer front, segon front, fins que no es puga seleccionar més individus perquè ja s'excedeix de la grandària de la població. A l'últim front, els individus s'ordenen segons la seua distància d'amuntegament i es seleccionen els millors fins que s'aplegue a tindre de nou la grandària de la població.

Pel que fa al creuament s'ha realitzat un creuament de dos punts com el que s'ha explicat al capítol 2. De manera que en primer lloc, es seleccionen dos individus de la població actual, que fan el paper pares, i després es seleccionen dos punts de cada un d'aquests individus de manera aleatòria per dividir-lo en tres parts. Als individus fills, aquests fragments s'intercanvien.

Finalment, amb l'objectiu de realitzar la mutació dels individus s'ha seleccionat una mutació uniforme per a enters que funciona canviant alguns dels valors de l'individu de manera aleatòria. Aquest nou valor introduït be definit per uns valor que poder modificar-se, que en el cas del problema, s'han definit de igual manera que a la creació d'individus. Un valor mínim de 0 i un valor màxim definit per el nombre de cultius

que hi haja. També consta de l'índex de probabilitat, un paràmetre que determina la probabilitat de que un individu mute. Quan s'executa una mutació a l'algorisme, per a cada valor de l'individu es genera un valor aleatòria. Si eixe valor es troba per baix de l'índex de probabilitat es muta, sinó no. Així, valor alts provoquen que pràcticament tot l'individu canvie, mentre que valor més baixos no ho fan.

## 4.3 Estructura de l'aplicació

---

L'arquitectura de l'aplicació segueix un patró de model MVC (Model-Vista-Controlador)[39] per estructurar eficientment el flux de treball de manera modular. Açò vol dir, com s'ha mencionat al capítol 3, que es poden trobar tres capes ben diferenciades amb funcionalitats distintes però que es relacionen entre elles.

En primer lloc, es troba la capa Model, que al treball representa la base de dades i la lògica principal del programa. La primera, són la connexió i les operacions CRUD que es realitzen a la base de dades de MongoDB. S'han implementat diversos scripts per guardar i extraure la informació necessària. La segona, correspon al mòdul principal que crida a l'algorisme on es realitza el càlcul principal de l'aplicació i demés funcions auxiliars necessàries conjuntament amb el algorisme genètic pròpiament dit.

Seguidament, es té la capa Vista. Aquesta és la que pot visualitzar l'usuari i correspon amb la pàgina web creada amb el framework Flask. És per tant, on els usuaris poden interactuar amb l'aplicació desenvolupada seleccionant i introduint les dades per a les quals desitgen obtindre la resposta. Posteriorment, quan aquesta s'obté a la capa Model, la Vista presenta la solució trobada. En aquest cas, es mostra una imatge del terreny i una taula amb la informació detallada dels cultius a plantar.

Per últim, es distingeix la capa Controlador, que s'encarrega d'actuar com a intermediari entre el Model i la Vista. Ací es reben les interaccions que ha realitzat l'usuari a la Vista, es preparen les dades i són enviades al Model per a que s'execute el script principal de manera correcta. Quan el Model finalitza i torna la solució obtinguda al Controlador, aquest s'encarrega al seu torn d'enviar-les a la Vista per a que puguen ser visualitzades per l'usuari.

Amb aquesta estructura, es pot veure com els elements del *backend* i els del *frontend* es troben separats, podent funcionar la lògica de l'aplicació de manera independent. Això comporta que es puguen realitzar modificacions al codi o a la base de dades sense que afecte a les altres capes, facilitant, per exemple, afegir futures expansions o característiques.

A continuació, sabent com està estructurat el treball, es va a detallar els mòduls més rellevants que s'han implementat per a dur a terme el projecte que es planteja en aquest treball.

### 4.3.1. Mòduls de la capa Model

En aquest apartat, va a detallar-se el funcionament de la interacció amb la base de dades i de la creació i funcionament de l'algorisme genètic conjuntament amb el script que s'encarrega de cridar-lo i processar el resultat obtingut.

#### Base de dades

Les operacions que s'han realitzat a la base de dades han sigut les denominades com a tipus CRUD, és a dir, operacions d'inserció, lectura, modificació i esborrament. Per



tant, primerament es va crear la base de dades i un script per accedir a ella i introduir de manera automatitzada totes les dades relacionades amb les hortícoles a una col·lecció anomenada *cultius*, i les dades relacionades a les temperatures i les zones climàtiques una col·lecció anomenada *temperatures*. Aquestes dades sols són de tipus lectura i es pot veure un esquema a la figura 4.4.

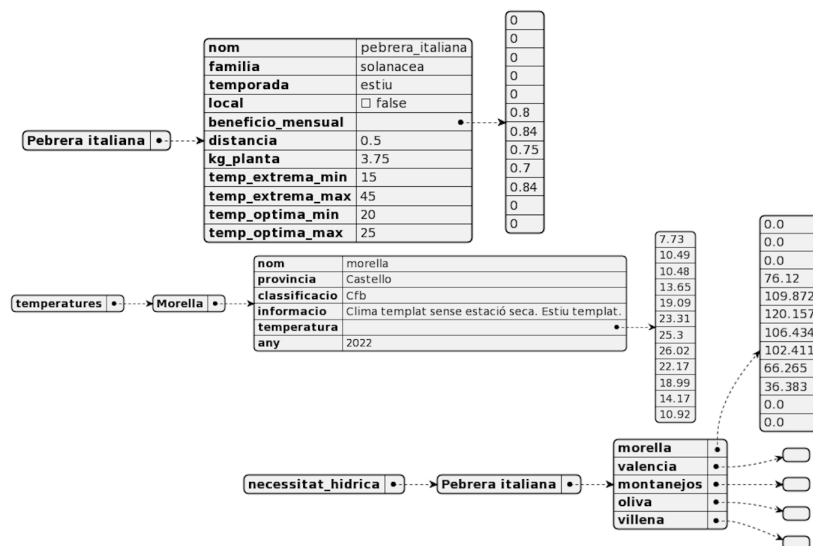


Figura 4.4: Esquema base de dades.

Per altra banda, es troba l'altre mòdul que, quan l'usuari introdueix les dades i aquestes són passades a través del Controlador, prepara les dades que es van a utilitzar per al càlcul d'eixe cas en concret. Amb noms dels cultius que han sigut seleccionats, es fa una consulta per obtenir de la base de dades aquells que coincidiscuen amb els elements de la col·lecció corresponent. Aquestes dades, per facilitar la lectura i processament dins de tot el projecte, s'afigen a un arxiu anomenat *config* amb format JSON. Al mateix temps, i també per petició del Controlador, s'afigen les dades relacionades amb la parcel·la que el usuari ha introduït a altre arxiu JSON anomenat *input* alhora que es fa una lectura a la base de dades per obtenir la temperatura de la zona del camp seleccionada. Aquests dos arxius són en els quals es treballa principalment. S'adjunta un extracte de la base de dades a l'apèndix B.

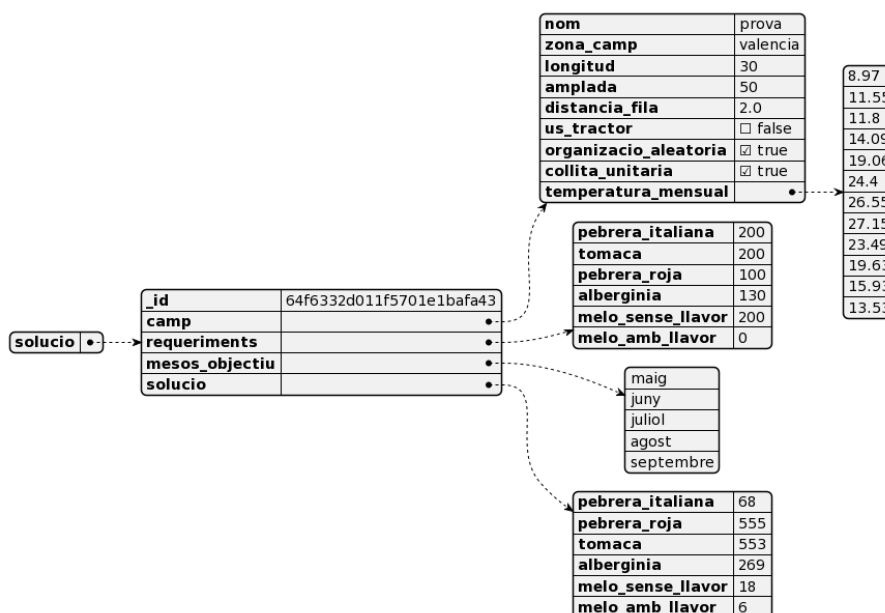
Per últim, quan s'ha obtingut una solució al problema, es realitza una altra inserció. Aquesta vegada, l'arxiu JSON creat anteriorment amb les dades de la parcel·la s'inclouen alhora amb la solució a una col·lecció anomenada *campes*. Si el nom del terreny coincideix amb una instància ja existent, s'actualitza la solució. Açò es pot veure a la figura 4.5

## Algorisme genètic

Tal com es comenta amb anterioritat, s'ha fet ús del marc de treball DEAP per a la implementació d'aquest. DEAP marca una estructura ben organitzada per al correcte funcionament i execució de l'algorisme. Facilitant la implementació del disseny explicat al apartat 4.2.

L'algorisme treballa amb els arxius JSON anomenats anteriorment conjuntament amb algunes dades processades calculades a la inicialització del mòdul.



Figura 4.5: Document *solucio*

DEAP treballa mitjançant la creació d'un Toolbox en el qual es van enregistrant tots els paràmetres i els selectors genètics escollits, permeten personalitzar-los per obtenir un funcionament diferent al proporcionat com a base pel framework.

L'execució d'aquest arxiu passa al script principal la solució. És a dir, una llista amb la quantitat de cada cultiu, el benefici econòmic i el consum hídric que s'obté d'eixa configuració extreta.

### Mòdul principal

Aquest és el mòdul que du el fil de la funcionalitat de l'aplicació. Quan es crida, totes les dades que s'han d'utilitzat es troben ja guardades als arxius JSON, de manera que aquest sols ha de fer el processament d'aquestes per cridar a l'algorisme genètic correctament.

Una vegada l'algorisme ha generat una solució, aquest transforma les dades perquè es pugen entendre. Es podria dir que en aquest mòdul es realitza la transformació genotip-fenotip del projecte. Aquesta transformació és el que aquest mòdul retorna a la Vista perquè pugui ser vista per l'usuari.

Aquesta transformació ens passa d'un individu com el vist a la figura 4.2 a representar la quantitat que hi ha de files completes i incompletes de cada cultiu. De manera que, com es pot veure a la figura 4.6, per exemple per al cultiu de pebreres italianes, hi haurà dos files completes amb 76 pebrers plantats i una única fila incompleta amb 31. Destacar que files incompletes no poden haver més d'una, ja que això passaria a ser una fila completa més.

#### 4.3.2. Mòduls de la capa Vista

Primerament, per realitzar la part de la Vista de l'aplicació, sent la pàgina web que es mostra al usuari, que ja s'ha comentat a la introducció del apartat ha sigut realitzada amb HTML, CSS i JavaScript, incorporat amb Jinja2.

Aquesta part és la més visual de tota l'aplicació i al projecte ha servit per obtenir les dades dinàmiques del problema. És a dir, s'ha utilitzat com a formulari.

Cultiu	Files completes	Plantes en cada fila completa	Plantes en fila incompleta
Pebrera italiana	2	76	31
Pebrera roja	7	108	11
Tomaca	34	95	90
Cogombre	9	76	27
Carabassi	2	47	30
Ceba seca	3	253	35

Figura 4.6: Imatge solució.



**eGultiu**

**Dades del camp**

Nom:   Utilització de tractor

Zona climàtica:

**Mesures**

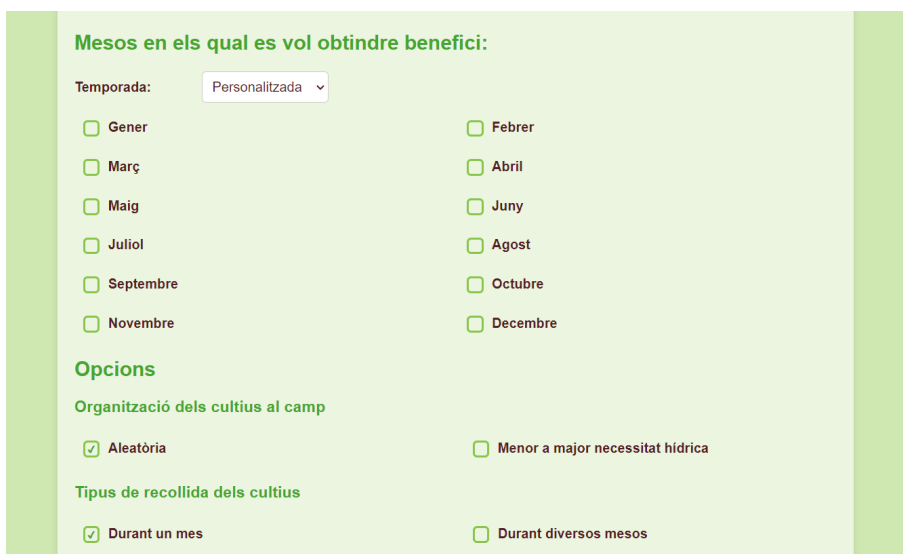
Longitud (m):

Amplada (m):

Distància files (m):

Figura 4.7: Imatge pàgina web. Obtenció dades del camp.

En aquesta figura 4.7 es pot veure com és la introducció de les dades del camp. En primer lloc es demana el nom que se li vol donar al camp, la zona climàtica a la que pertany i si es vol o no utilitzar tractor.



**Mesos en els qual es vol obtenir benefici:**

Temporada:

Gener  Febrer

Març  Abril

Maig  Juny

Juliol  Agost

Setembre  Octubre

Novembre  Decembre

**Opcions**

**Organització dels cultius al camp**

Aleatòria  Menor a major necessitat hídrica

**Tipus de recollida dels cultius**

Durant un mes  Durant diversos mesos

Figura 4.8: Imatge pàgina web. Obtenció informació.

Seguidament, en aquesta altra figura 4.8 es pot veure com és la selecció dels mesos per als quals es desitja obtenir una solució. Es poden escollir de manera personalitzada,

o escollint la temporada Estiu/Hivern. També, es pot seleccionar el tipus d'organització i el tipus de recollida. Com s'ha comentat amb anterioritat a l'inici del capítol 4, aquestes dades serviran per a garantir que l'algorisme siga aplicat a cada cas concret.

Cultiu	Requeriment mínim (Kg)
<input type="checkbox"/> Pebrera italiana	Requeriment mínim (Kg)
<input type="checkbox"/> Pebrera roja	Requeriment mínim (Kg)
<input type="checkbox"/> Albergínia	Requeriment mínim (Kg)
<input type="checkbox"/> Meló sense llavor	Requeriment mínim (Kg)
<input type="checkbox"/> Meló amb llavor	
<input type="checkbox"/> Meló tot any	Requeriment mínim (Kg)
<input type="checkbox"/> Cogombre	Requeriment mínim (Kg)
<input type="checkbox"/> Carabassi	Requeriment mínim (Kg)
<input type="checkbox"/> Garrofó	Requeriment mínim (Kg)
<input type="checkbox"/> Ceba seca	Requeriment mínim (Kg)
<input type="checkbox"/> Tomaca	Requeriment mínim (Kg)
<input type="checkbox"/> Creïlla	Requeriment mínim (Kg)
<input type="checkbox"/> Floricol	Requeriment mínim (Kg)
<input type="checkbox"/> Escarola	Requeriment mínim (Kg)
<input type="checkbox"/> Fava	Requeriment mínim (Kg)
<input type="checkbox"/> Pèsol	Requeriment mínim (Kg)
<input type="checkbox"/> Col	Requeriment mínim (Kg)
<input type="checkbox"/> Encisam	Requeriment mínim (Kg)
<input type="checkbox"/> Ceba tendra	Requeriment mínim (Kg)
<input type="checkbox"/> Safanòria	Requeriment mínim (Kg)

Figura 4.9: Imatge pàgina web. Selecció cultius

Cultiu	Màxim
<input type="checkbox"/> Pebrera italiana	1119007.5
<input type="checkbox"/> Pebrera roja	1053183.53
<input type="checkbox"/> Albergínia	767319.43
<input type="checkbox"/> Meló sense llavor	1641211
<input type="checkbox"/> Meló amb llavor	
<input type="checkbox"/> Meló tot any	1492010
<input type="checkbox"/> Cogombre	1193608
<input type="checkbox"/> Carabassi	746005
<input type="checkbox"/> Garrofó	62167.08
<input type="checkbox"/> Ceba seca	298402
<input type="checkbox"/> Tomaca	1492010
<input type="checkbox"/> Creïlla	298402
<input type="checkbox"/> Floricol	464180.89
<input type="checkbox"/> Escarola	422736.17
<input type="checkbox"/> Fava	128909.66
<input type="checkbox"/> Pèsol	14920.1
<input type="checkbox"/> Col	596804
<input type="checkbox"/> Encisam	298402
<input type="checkbox"/> Ceba tendra	223801.5
<input type="checkbox"/> Safanòria	186501.25

Figura 4.10: Imatge pàgina web. Selecció cultius temporada

Pel que fa a les dades que es poden introduir per a personalitzar el problema, a la figura 4.9 es troba la selecció dels cultius amb el seu corresponent valor de requisit mínim en kilogrames, de manera que l'usuari podrà introduir la quantitat mínima que desitja a la seua solució. Aquests valor, a més, van mostrant una cota del que podria introduir-se al camp partint de les mesures afegides a les dades del camp de la figura 4.7. A més, com es pot veure a la figura 4.10, si hi ha una temporada seleccionada, els cultius s'anul·len de manera automàtica si no pertanyen a eixa temporada concreta.

Destacar també, que la condició que determina que ha d'haver com a mínim tres cultius comentada a l'apartat 4.1.1 de les restriccions, s'ha realitzat a la part del *frontend* com es veu a la figura 4.11. Aquest missatge apareix si es seleccionen cap, un o dos cultius.

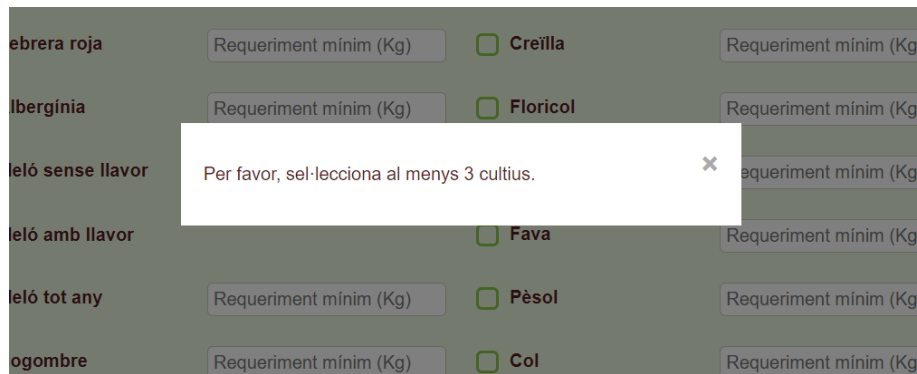


Figura 4.11: Imatge pàgina web. Missatge

Per últim, una vegada s'ha executat l'algorisme, Flask torna la solució en forma de taula, al mateix temps que mostra una imatge que representa percentualment els cultius al camp. Açò es pot veure a la figura 4.12.

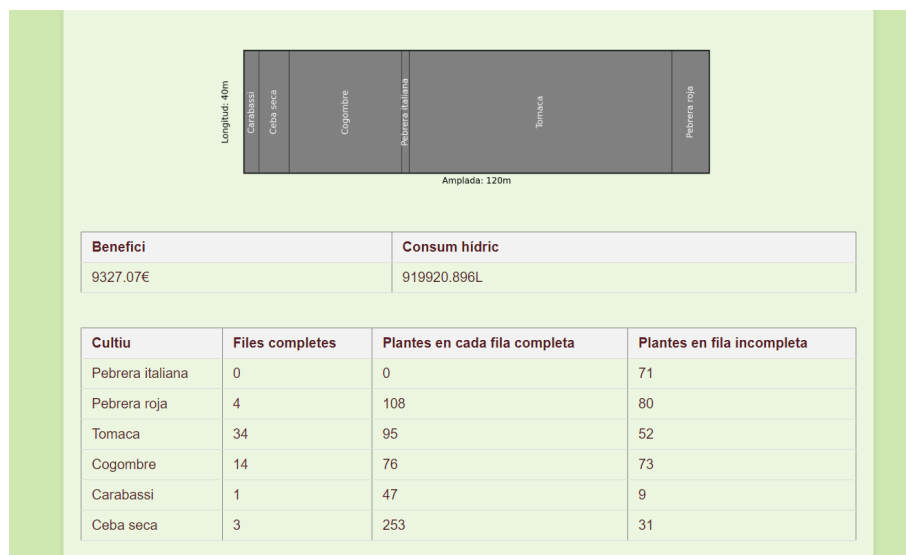


Figura 4.12: Imatge pàgina web. Solució

### 4.3.3. Mòduls de la capa Controlador

En aquesta secció s'aborda la interacció entre les dades introduïdes per l'usuari i el programa principal, que actua com a nucli controlador de l'aplicació. Aquesta tasca es realitza amb Flask.

Flask s'encarrega de crear un servidor web que acull la interfície d'usuari i controla les sol·licituds que es fan a través d'aquesta. Recopila totes les dades introduïdes i els envia al programa principal perquè puguin ser processades.

Una vegada que el algorisme s'ha executat i s'ha aplegat a una solució, Flask s'encarrega de nou, d'enviar aquesta informació al usuari a través de la pàgina de resultats que s'actualitza dinàmicament una vegada ha sigut generada una solució.

---

---

## CAPÍTOL 5

# Avaluació del sistema

---

Una vegada s'ha realitzat l'algorisme hi ha que comprovar que aquest realment obté els resultats que s'esperen. Un dels aspectes més crítics en el desenvolupament de qualsevol sistema d'optimització és la capacitat del algorisme per a escalar. Amb escalabilitat es fa referència a com el rendiment del sistema es veu afectat a mesura que augmenta la complexitat o el volum de dades en les qual s'ha de treballar.

Per abordar aquesta qüestió, s'han dut a terme una sèrie de proves utilitzant quatre conjunts de dades de diferents magnituds: un problema molt menut que es resol pràcticament de manera immediata, un altre de complexitat mitjana, altre molt gran, i finalment un enorme i costós en termes de temps i recursos computacionals. El que s'ha pretés ha sigut avaluar el rendiment de l'algorisme en cada un d'aquests escenaris i determinar si el benefici que s'obté justifica el cost que comporta. Al problema, les dades que s'han anat modificant han sigut les relacionades amb la grandària del terreny.

Aquesta avaluació es clau per entendre les limitacions a les que s'enfronta el sistema desenvolupat.

En aquestes proves es va a treballar modificant la grandària del camp, mantenint el nombre de cultius i la distància entre files. Els cultius que s'han considerat són: pebreres italianes: 200, tomaques: 100, albergínies: 130, cogombre: 80 i carabassins: 0. Per a cada prova s'ha augmentat exponencial també el nombre de cultius.

En primer lloc, s'ha realitzat l'avaluació amb un camp de 30x50 metres, equivalent a 0,15 hectàrees, obtenint una llargària de 667 metres de fila amb una distància de 2 metres entre files. En aquesta prova menuda s'ha aplegat a una solució de manera molt ràpida, amb menys de cinc minuts de còmput, i s'ha aplegat a un benefici de 2756.36 € i un consum hídric de 295775.97 litres. El carabassí ha tingut solució 0.

Cultiu	Files completes	Plantes en cada fila completa	Plantes en fila incompleta
Pebrera italiana	1	56	13
Tomaca	13	70	66
Albergínia	1	40	16
Cogombre	7	56	13

**Figura 5.1:** Cas prova camp menut

Seguidament, s'ha augmentat la grandària del camp fins als 30x500 metres, el que equival a 1.5 hectàrees, obtenint una llargària de 7192 metres de fila amb una distància de 2 metres entre files. Un camp d'hortícoles d'aquesta dimensió ja és un molt elevada i no és massa comú. S'ha obtingut després de quasi mitja hora amb uns resultats de 29570.2€ i 3199144.33 litres de consum.

Cultiu	Files completes	Plantes en cada fila completa	Plantes en fila incompleta
Pebrera italiana	13	56	13
Tomaca	153	70	59
Alberginia	32	40	5
Cogombre	57	56	33

**Figura 5.2:** Cas prova camp mitjà

A continuació s'han augmentat encara més les mesures del camp fins als 300x500 metres, que equival a 15 hectàrees, obtenint una llargària de 74152 metres de fila. Aquest problema ja ha estat realitzant-se sobre una hora i deu minuts de còmput i ha obtingut un valor de 305606.53 € i 32779673.978 litres.

Cultiu	Files completes	Plantes en cada fila completa	Plantes en fila incompleta
Pebrera italiana	11	596	565
Tomaca	144	745	314
Alberginia	7	425	56
Cogombre	81	596	391
Carabassi	3	372	251

**Figura 5.3:** Cas prova camp gran

Per últim, s'ha avaluat amb unes mesures de de 300x5000 metres, el que equival a 150 hectàrees, obtenint una llargària de fila de 746902 metres. Aquesta prova ha consumit un temps de còmput molt elevat, rondant les dues hores d'execució. S'ha obtingut un benefici de 3070230.97 € i un valor de 329477857.83 litres de consum hídic.

Cultiu	Files completes	Plantes en cada fila completa	Plantes en fila incompleta
Pebrera italiana	11	596	77
Tomaca	1258	745	246
Alberginia	238	425	291
Cogombre	941	596	578
Carabassi	56	372	160

**Figura 5.4:** Cas prova camp gran

Amb les proves realitzades, s'ha pogut constatar que l'algorisme desenvolupat és capaç d'escalar de manera efectiva en funció de la complexitat i el volum de dades. Aquesta escalabilitat és crucial per a la seva aplicabilitat en diferents contextos agrícoles, des de petites parcel·les fins a grans extensions de terra. Els resultats obtinguts en cada escenari han estat coherents amb les expectatives, tant en termes de benefici econòmic com de consum hídic.

Per descomptat, és important assenyalar que, tot i que aquesta avaluació s'ha centrat principalment en el benefici econòmic i el consum hídic en temps de resposta coherents amb l'aplicació d'un algorisme genètic a l'hora d'augmentar el problema, hi ha altres mètriques que podrien ser d'interès per a una avaluació més completa del sistema.

---

---

## CAPÍTOL 6

# Conclusions

---

L'objectiu principal d'aquest treball ha sigut el disseny i la implementació d'una aplicació per determinar les quantitat de que cultiu s'ha de destinar a una parcel·la. Per aconseguir-ho, s'han utilitzat tècniques d'intel·ligència artificial, més concretament, un algorisme genètic. Si es retrocedeix fins a l'inici d'aquesta investigació i revisitem els objectius que es varen esbossar en el capítol d'introducció 1), es pot afirmar que han estat assolits de manera satisfactòria.

Per arribar a obtindre l'objectiu principal, en primer lloc, s'ha realitzat una recopilació exhaustiva de la informació rellevant sobre els cultius, incloent factors com les quantitats d'aigua requerides, els temps de recol·lecció i les temperatures òptimes de creixement. Aquesta informació ha estat fonamental per a la modelització precisa del problema. Açò ha sigut possible gràcies a que en tot moments s'ha estat en contacte amb una persona experta en l'àmbit de l'agricultura, que ha realitzat un assessorament continu del que era viable o no.

En segon lloc, s'ha dut a terme una investigació de les tècniques d'intel·ligència artificial que s'ha cregut que podrien haver sigut convenients per a la resolució del problema. Després d'avaluar i comparar diverses opcions, s'ha escollit la que s'ha cregut més encertada per al model de problema que s'havia plantejat resoldre.

Un dels elements més destacats d'aquest treball ha estat, sens dubte, el modelat de l'algorisme genètic utilitzat. Aquesta tècnica d'intel·ligència artificial ha permès una cerca eficient en l'espai de solucions, adaptant-se als diferents paràmetres i restriccions del problema. La seva flexibilitat i capacitat per trobar solucions òptimes han validat la seva elecció com a mètode resolutiu del problema plantejat. Ací, a més, es pot afegir la consideració que s'ha tingut a l'hora de modelar el problema al incloure opcions que puguen afavorir el medi ambient.

Finalment, s'ha aconseguit un alt grau d'implicació amb i per a l'usuari. L'aplicació permet adaptar diferents paràmetres i restriccions, perquè cada problema puga ser adaptable a cada cas concret.

A través d'aquest treball, s'ha intentat donar una resolució al problema de la planificació agrícola, aconseguint donar un xicotet pas a que al sector agrícola s'incloguen més aplicacions tecnològiques. Aquesta iniciativa representa un xicotet, però significatiu pas cap a la modernització i la sostenibilitat en l'agricultura. A més, ha quedat reflectida la viabilitat de l'ús de la intel·ligència artificial per a la resolució de problemes aconseguint un model adaptable i sostenible que pot servir com a base per a futurs desenvolupaments en aquest camp.

---

## 6.1 Treballs futurs

---

Tot i que en aquest treball s'han contemplat diverses tipus de dades, hi ha diverses direccions en les quals es podria expandir la recerca i el desenvolupament futurs.

En primer lloc, una de les limitacions actuals de model és la falta de la consideració de la temporalitat i les rotacions de cultius. En treballs futurs, es podria partir d'una solució ja obtinguda i adaptar-la al llarg del temps, tenint en compte les necessitats de rotació de cultius per a la sostenibilitat del sòl.

També es podria realitzar una ampliació de la base de dades. L'expansió d'aquesta base de dades per incloure una gamma més àmplia de cultius i les seves respectives necessitats podria fer que l'aplicació fos més versàtil i aplicable a una major varietat de contextos agrícoles.

Per últim, encara que al model si es troba ja guardat si un cultiu és de la zona o no, no es realitza cap acció addicional, degut a la quantitat de cultius. Per això, una ampliació podria centrar-se en la incorporació de dades locals per a donar una preferència sobre d'altres que no ho són, permetent així solucions més adaptables a la zona geogràfica en la qual ens trobem.

Amb aquestes millores, es podria aconseguir un model més robust, flexible i adaptat a les complexitats inherents a la planificació agrícola, contribuint així de manera més efectiva al desenvolupament sostenible del sector.



# Bibliografía

---

- [1] J. C. Ponce Gallegos, A. Torres Soto, F. S. Quezada Aguilera, A. Silva Sprock, E. U. Martínez Flor, A. Casali, E. Scheihing, Y. J. Túpac Valdivia, M. D. Torres Soto, F. J. Ornelas Zapata *et al.*, *Inteligencia artificial*. Iniciativa Latinoamericana de Libros de Texto Abiertos (LATIn), 2014.
- [2] S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, 2nd ed. Pearson, 2022.
- [3] A. Abeliuk and C. Gutiérrez, "Historia y evolución de la inteligencia artificial," *Revista Bits de Ciencia*, no. 21, pp. 14–21, 2021.
- [4] L. E. Munera, "Inteligencia artificial y sistemas expertos," *Inteligencia artificial y sistemas expertos*, 1991.
- [5] L. Amador Hidalgo, *Inteligencia artificial y sistemas expertos*. Universidad de Córdoba, Servicio de Publicaciones, 1996.
- [6] (2023) Lisp lang. Accés el: 15 d'agost de 2023. [Online]. Available: <https://lisp-lang.org/>
- [7] (2023) Swi-prolog. Accés el: 15 d'agost de 2023. [Online]. Available: <https://www.swi-prolog.org/>
- [8] S. I. Gallant *et al.*, "Perceptron-based learning algorithms," *IEEE Transactions on neural networks*, vol. 1, no. 2, pp. 179–191, 1990.
- [9] S. E. T. Sánchez, M. O. Rodríguez, A. E. Jiménez, and H. J. P. Soberanes, "Implementación de algoritmos de inteligencia artificial para el entrenamiento de redes neuronales de segunda generación," *Jóvenes en la Ciencia*, vol. 2, pp. 6–10, 2016.
- [10] L. E. Sucar and M. Tonantzintla, "Redes bayesianas," *Aprendizaje Automático: conceptos básicos y avanzados*, vol. 77, p. 100, 2006.
- [11] E. García and F. Flego, "Agricultura de precisión," *Revista Ciencia y Tecnología*, vol. 8, pp. 99–116, 2008.
- [12] K. staff. (2023) Local robotics company using ai tech to eliminate weeds. Accessed: 2023-09-04. [Online]. Available: <https://mynorthwest.com/3908974/local-robotics-company-using-ai-tech-to-eliminate-weeds-laserweeder/>
- [13] A. Calvo. (2023) Planificación de cultivos: clave del éxito en explotaciones agrícolas. Consultado el 4 de septiembre de 2023. [Online]. Available: <https://www.agroptima.com/es/blog/planificacion-de-cultivos-clave-del-exito-en-explotaciones-agricolas/>

- [14] M. Abdel-Basset, L. Abdel-Fatah, and A. K. Sangaiah, "Metaheuristic algorithms: A comprehensive review," *Computational intelligence for multimedia big data on the cloud with engineering applications*, pp. 185–231, 2018.
- [15] F. S. Caparrini. (2023) Csp. Accés el: 15 d'agost de 2023. [Online]. Available: <https://www.cs.us.es/~fsancho/Cursos/SVRAI/CSP.md.html>
- [16] A. Garrido Tejero, "Apuntes de asignatura tècniques, entorns i aplicacions d'intel·ligència artificial." 2022, curs 2022.
- [17] F. Barber and M. A. Salido, "Introducción a la programación de restricciones," *Inteligencia Artificial. Revista Iberoamericana de Inteligencia Artificial*, vol. 7, no. 20, 2003.
- [18] F. Manyá and C. Gomes, "Técnicas de resolución de problemas de satisfacción de restricciones," *Inteligencia Artificial. Revista Iberoamericana de Inteligencia Artificial*, vol. 7, no. 19, p. 0, 2003.
- [19] SESBE, "¿qué es la biología evolutiva?" s.f., accés el: 01 de agosto de 2023. [Online]. Available: <https://sesbe.org/que-es-la-biologia-evolutiva/>
- [20] P. E. Valencia, "Optimización mediante algoritmos genéticos," *Anales del Instituto de Ingenieros de Chile*, vol. 109, no. 2, pp. 83–92, 1997.
- [21] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*. Reading, MA: Addison-Wesley, 1989.
- [22] J. Arranz de la Peña and A. Parra Truyol, "Algoritmos genéticos," *Universidad Carlos III*, 2007.
- [23] A. Kuri and J. Galaviz, *Algoritmos genéticos*. IPN, 2002, no. Sirsi i9789681663834.
- [24] Marcos, D. Rivero Gestal, J. R. Rabuñal, J. Dorado, A. Pazos, and M. Gestal, *Introducción a los algoritmos genéticos y la programación genética*. Coruña: Universidade da Coruña, 2010.
- [25] P. S. Foundation, "Python 3.11 documentation," 2023, accés el: 14 de juny de 2023. [Online]. Available: <https://docs.python.org/es/3.11/>
- [26] W. W. W. Consortium, "Html," World Wide Web Consortium, accés el: 14 d'agost de 2023. [Online]. Available: <https://www.w3.org/TR/html/>
- [27] —, "Css," World Wide Web Consortium, accés el: 14 d'agost de 2023. [Online]. Available: <https://www.w3.org/Style/CSS/>
- [28] MongoDB. (2023) MongoDB official documentation. [Online]. Available: <https://www.mongodb.com/docs/>
- [29] Oracle. (2023) Oracle nosql database - what is nosql? [Online]. Available: <https://www.oracle.com/es/database/nosql/what-is-nosql/>
- [30] MongoDB. (Año de acceso) Jjson and bson. [Online]. Available: <https://www.mongodb.com/json-and-bson>
- [31] C.-O. Truica, A. Boicea, and I. Trifan, "Crud operations in mongodb," in *2013 international conference on advanced computer science and electronics information (ICACSEI 2013)*. Atlantis Press, 2013, pp. 347–350.
- [32] D. Developers, "Deap documentation," 2023, accés el: 10 de juny de 2023. [Online]. Available: <https://deap.readthedocs.io/en/master/>

- [33] J. Olawanle, "What is a framework? software frameworks definition," 2023, accés el: 24 de juny de 2023. [Online]. Available: <https://www.freecodecamp.org/news/what-is-a-framework-software-frameworks-definition/>
- [34] S. Developers, "Scoop documentation," 2023, accés el: 10 de juny de 2023. [Online]. Available: <https://scoop.readthedocs.io/en/latest/>
- [35] Pallets. (2023) Welcome to flask. Accés el: 20 de juliol de 2023. [Online]. Available: <https://flask.palletsprojects.com/en/2.3.x/>
- [36] ——. (2023) Werkzeug. Accés el 21 de juny de 2023. [Online]. Available: <https://werkzeug.palletsprojects.com/en/2.3.x/>
- [37] P. S. Foundation. (2023) Wsgi utilities and reference implementation. Accés el: 21 de juny de 2023. [Online]. Available: <https://docs.python.org/es/3/library/wsgiref.html>
- [38] Pallets. (2023) Jinja documentation. Accés el: 21 de juny de 2023. [Online]. Available: <https://jinja.palletsprojects.com/en/3.1.x/>
- [39] GeeksforGeeks. Mvc framework - introduction. Accés el: 18 d'agost de 2023. [Online]. Available: <https://www.geeksforgeeks.org/mvc-framework-introduction/>
- [40] D. S. Foundation. (2022) Django documentation. Django Software Foundation. Accés el: 20 de juny de 2023. [Online]. Available: <https://docs.djangoproject.com/en/4.2/>
- [41] C. Anderson, "The model view viewmodel (mvvm) design pattern," in *Pro Business Applications with Silverlight 5*. Springer, 2012, pp. 461–499.
- [42] N. Alshuqayran, N. Ali, and R. Evans, "A systematic mapping study in microservice architecture," in *2016 IEEE 9th International Conference on Service-Oriented Computing and Applications (SOCA)*. IEEE, 2016, pp. 44–51.
- [43] (2023) Clima: Comunidad valenciana. Accés el: 10 d'agost de 2023. [Online]. Available: <https://es.climate-data.org/europe/espana/comunidad-valenciana-290/>
- [44] M. Kottek, J. Grieser, C. Beck, B. Rudolf, and F. Rubel, "World map of the köppen-geiger climate classification updated," 2006.
- [45] A. no especificados, *Cultivos hortícolas al aire libre*. Cajamar Caja Rural, 2016, iISBN-13: 978-84-95531-82-7, Depósito Legal: AL-94-2017. [Online]. Available: <http://www.publicacionescajamar.es>
- [46] C. per a la Investigació i l'Experimentació Forestal (CIEF). (2023) Catàleg valencià de varietats tradicionals d'interès agrari. Accés el: 2 de juny de 2023. [Online]. Available: <https://mediambient.gva.es/va/web/cief/catalog-valencia-de-varietats-tradicionals>
- [47] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: Nsga-ii," *IEEE transactions on evolutionary computation*, vol. 6, no. 2, pp. 182–197, 2002.
- [48] K. A. Dowsland and B. A. Díaz, "Diseño de heurística y fundamentos del recocido simulado," *Inteligencia Artificial. Revista Iberoamericana de Inteligencia Artificial*, vol. 7, no. 19, p. 0, 2003.
- [49] J.-G. Ganascia. (2018) Inteligencia artificial: entre el mito y la realidad. Última actualizació: 28 d'agost de 2018. [Online]. Available: <https://es.unesco.org/courier/2018-3/inteligencia-artificial-mito-y-realidad>



---



---

## APÈNDIX A

# Taula de cultius

---

En aquest apèndix, es presenta la taula de cultius utilitzada amb la seua respectiva família associada. L'estació de l'any definida principalment per estiu-hivern, excepte un cultiu anual, a sigut una simplificació per a l'execució de l'algorisme.

Família	Cultiu	Estació de l'Any
Solanàcies	Pebrera italiana	Estiu
	Pebrera roja	Estiu
	Albergínia	Estiu
	Tomaca	Estiu
	Creïlla	Hivern
Lleguminoses	Garrofó	Estiu
	Pèsol	Hivern
	Faves	Hivern
Cucurbitàcies	Carabassí	Estiu
	Cogombre	Estiu
	Meló alger amb llavors	Estiu
	Meló alger sense llavors	Estiu
	Meló de tot l'any	Estiu
Crucíferes	Col	Hivern
	Floricol	Hivern
Compostes	Escarola	Hivern
	Encisam	Hivern
	Floricol	Hivern
Liliàcies	Ceba tendra	Hivern
	Ceba seca	Estiu
Umbel·líferes	Safanòria	Anual

**Taula A.1:** Cultius i Estacions de l'Any



---

---

## APÈNDIX B

# Extracte d'una instància de la base de dades

---

## B.1 Extracte de la Base de Dades

---

En aquest apèndix, es presenta un extracte de la base de dades MongoDB que conté les diferents taules amb informació estàtica del sistema.

### B.1.1. Registre de Cultiu

L'extracte mostra un registre de cultiu amb diferents atributs, com el nom, la família, la temporada, el benefici mensual i altres dades rellevants.

```
1  {
2    "nom": "Pebrera italiana",
3    "familia": "solanacea",
4    "temporada": "estiu",
5    "local": false,
6    "beneficio_mensual": [0, 0, 1, 0, 2, 0, 3, 0, 4, 0, 5, 0.8,
7                          6, 0.84, 7, 0.75, 8, 0.7, 9, 0.84, 10, 0, 11, 0],
8    "distancia": 0.5,
9    "kg_planta": 3.75,
10   "temp_extrema_min": 15,
11   "temp_extrema_max": 45,
12   "temp_optima_min": 20,
13   "temp_optima_max": 25
14 }
```

### B.1.2. Registre Necessitats hídriques

A continuació, es mostra un exemple de l'extracte de la taula "Necessitats hídriques" del cultiu "Pebrera italiana", on es determinen els valor de consum d'aigua del cultiu a cada zona climàtica.

```
1  {
2    "Pebrera italiana": {
3      "morella": [0.0, 0.0, 0.0, 76.12, 109.872, 120.157, 106.434,
4                  102.411, 66.265, 36.383, 0.0, 0.0],
5      "valencia": [0.0, 0.0, 0.0, 66.229, 99.343, 120.574, 106.36,
6                   102.519, 66.697, 36.187, 0.0, 0.0],
```

```

5   "montanejos": [0.0, 0.0, 0.0, 65.52, 106.249, 123.763,
6     109.053, 101.912, 64.955, 39.812, 0.0, 0.0],
7   "oliva": [0.0, 0.0, 0.0, 68.087, 110.912, 133.292, 114.332,
8     106.204, 69.484, 36.227, 0.0, 0.0],
9   "villena": [0.0, 0.0, 0.0, 74.364, 122.771, 146.381, 126.846,
10    114.676, 72.954, 47.729, 0.0, 0.0]
11 },
12 }

```

### B.1.3. Registre Temperatures

Aquest extracte mostra un registre de la taula "Temperatures" on es poden visualitzar les dades associades a cada zona climàtica contemplada al problema.

```

1 {
2   "temperatures": {
3     "Morella": {
4       "nom": "morella",
5       "provincia": "Castello",
6       "classificacio": "Cfb",
7       "informacio": "Clima templat sense estacio seca. Estiu
8         templat.",
9       "temperatura": [7.73, 10.49, 10.48, 13.65, 19.09, 23.31,
10        25.3, 26.02, 22.17, 18.99, 14.17, 10.92],
11      "any": 2022
12    },
13  },
14 }

```

### B.1.4. Registre Solucions

Aquest registre conté una solució guardada. Conté la informació del camp, els requeriments mínims de cultius que s'han establert junt amb els mesos objectiu, la solució trobada per l'algorisme i la imatge que es genera.

```

1 {
2   "camp": {
3     "nombre": "prova",
4     "zona_camp": "valencia",
5     "longitud": 30,
6     "amplada": 50,
7     "distancia_fila": 2,
8     "us_tractor": false,
9     "organizacio_aleatoria": true,
10    "collita_unitaria": true,
11    "temperatura_mensual": [8.97, 11.55, 11.8, 14.09, 19.06,
12    24.4, 26.55, 27.15, 23.49, 19.63, 15.93, 13.53]
13  },
14  "requeriments": {
15    "pebrera_italiana": 200,
16    "tomaca": 200,
17    "pebrera_roja": 100,
18    "alberginia": 130,
19    "melo_sense_llavor": 200,
20    "melo_amb_llavor": 0
21  },
22  "mesos_objectiu": [

```



```
22     "maig",
23     "juny",
24     "juliol",
25     "agost",
26     "septembre"
27   ],
28   "solucio": {
29     "pebrera_italiana": 68,
30     "pebrera_roja": 555,
31     "tomaca": 553,
32     "alberginia": 269,
33     "melo_sense_llavor": 18,
34     "melo_amb_llavor": 6
35   },
36   "image_id": {
37     "$oid": "64f6332d011f5701e1bafa44"
38   }
39 }
```



---



---

## APÈNDIX C

# Objectius de Desenvolupament Sostenible

---

**Taula C.1:** Objectius de Desenvolupament Sostenible

	Alt	Mitjà	Baix	No procedeix
ODS 1. Fi de la pobresa				X
ODS 2. Fam zero	X			
ODS 3. Salut i benestar				X
ODS 4. Educació de qualitat				X
ODS 5. Igualtat de gènere				X
ODS 6. Aigua neta i sanejament				X
ODS 7. Energia assequible i no contaminant				X
ODS 8. Treball decent i creixement econòmic	X			
ODS 9. Indústria, innovació i infraestructures		X		
ODS 10. Reducció de les desigualtats				X
ODS 11. Ciutats i comunitats sostenibles	X			
ODS 12. Producció i consum responsables	X			
ODS 13. Acció pel clima	X			
ODS 14. Vida submarina				X
ODS 15. Vida d'ecosistemes terrestres				X
ODS 16. Pau, justícia i institucions sòlides				X
ODS 17. Aliances per aconseguir objectius	X			

### C.1 Reflexió sobre la relació del TFG amb els ODS més relacionats

---

Aquest treball està estretament relacionat amb diversos Objectius de Desenvolupament Sostenible (ODS) establerts per les Nacions Unides, reflectint així el seu compromís amb un futur més sostenible i equitatiu. En primer lloc, l'optimització de la planificació agrícola contribueix directament a l'ODS 2, "Fam Zero", ja que facilita una producció d'aliments més eficient, essencial per a l'erradicació de la fam i l'accés a una alimentació de qualitat.

Aquesta eficiència en la producció també té repercussions en l'ODS 8, "Treball Decent i Creixement Econòmic". Fomentant el creixement econòmic al maximitzar la rendibilitat de les terres cultivables. Això es lliga amb l'ODS 9, "Indústria, Innovació i Infraestructu-

res", ja que la implementació d'algoritmes genètics i tècniques d'intel·ligència artificial en l'agricultura representa una innovació significativa.

A més, el treball fa èmfasi en la sostenibilitat, alineant-se amb l'ODS 12 de "Producció i Consum Sostenibles". Al considerar factors com el consum d'aigua i la sostenibilitat dels cultius, es promouen pràctiques agrícoles més sostenibles. Aquest enfocament sostenible també té un impacte positiu en les Ciutats i Comunitats Sostenibles"(ODS 11) i en l'"Acció pel Clima"(ODS 13), ja que contribueix a la mitigació del canvi climàtic.

Finalment, aquest projecte fomenta les "Aliances per Aconseguir Objectius"(ODS 17), ja que convida a la col·laboració entre diferents sectors, incloent el tecnològic, l'agrícola i l'acadèmic, per assolir un objectiu comú de sostenibilitat.