Instituto Tecnológico de Monterrey
ESCUELA DE INGENIERÍA Y CIENCIAS

RETO: OPTIMIZACIÓN ESTOCÁSTICA

Etapa 3: Experimentación y resultados de la optimización de la distribución de plantación en la restauración ecológica

Juan José de Jesús Hernández Beltrán, Eliani González Laguna, David Alejandro Acuña Orozco, Alberto Rodríguez Reyes, Asgard Andrés Mendoza Flores.

September 8, 2024

---

1. INTRODUCCIÓN AL PROBLEMA

Se denomina reforestación al proceso de plantar árboles en áreas que previamente fueron bosques y que han sido degradadas o de alguna manera han perdido su cobertura forestal. Su objetivo es restaurar y recuperar los ecosistemas forestales que fueron dañados debido a actividades humanas como la tala de árboles, la agricultura, la minería, la urbanización, entre otras. [Econosus, 2023]

Existen distintos tipos de reforestación, a continuación se presentan algunos de ellos.

- *Reforestación urbana:* Tiene como objetivo principal plantar árboles en entornos urbanos para modificar el clima, mejorar la calidad del aire, aumentar las zonas de sombra o embellecer el entorno.
- *Reforestación rural:* Consiste en la plantación masiva de árboles en superficies forestales previamente deforestadas. Dentro de esta categoría, encontramos subtipos como:
  - Reforestación de conservación, protección y restauración: Orientada a preservar, proteger y recuperar los ecosistemas.
  - Reforestación agroforestal o productiva: Este tipo de reforestación busca la producción de madera, celulosa, fruta, fibras o combustibles.

Algunas consideraciones generales para realizar una reforestación de manera exitosa serían las siguientes:

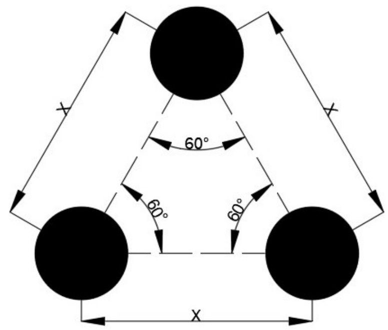
- *Planificación:* Definir objetivos claros y evaluar las condiciones del sitio antes de comenzar.
- *Selección de especies:* Escoger árboles adecuados considerando su adaptación al suelo, clima y ecosistema local.
- *Preparación del terreno:* Limpiar y acondicionar el área antes de la plantación.
- *Plantación de árboles:* Utilizar métodos apropiados para plantar las especies, asegurando una correcta profundidad y riego.
- *Mantenimiento inicial:* Proporcionar agua adecuada y cuidado a las plantas recién plantadas.

Sin embargo, realizar una reforestación de manera exitosa requiere de la consideración de muchos otros factores que deben de ser analizados por gente cualificada como las reservas y la dinámica de los cuerpos hídricos en la zona, la distribución espacial de las plantas, su carácter endémico a la zona y el comportamiento de simbiosis o daño que genera la introducción de cada planta con respecto a los demás especímenes de la reforestación y del ecosistema presenta en el lugar. [Econosus, 2023]

Un método de plantación diseñado para el óptimo cultivo en términos de distribución espacial de las plantas es el método "Tres Bolillos". Este método se basa en la idea de crear una malla en la que las plantas formen triángulos equiláteros. Esto causa que las plantas vecinas sean equidistantes, preservando la distancia mínima para un crecimiento adecuado. Al usar triángulos equiláteros, se minimiza el desaprovechamiento de espacio. También permite que no se tapen entre ellas para obtener luz del sol favoreciendo un 'microclima' apto para la retención de humedad y para prevenir la erosión (Iglesias, 2021). Para calcular el número de plantas en un sistema tres bolillos, se usa la siguiente fórmula:

$$n = \frac{S}{d^2 cos(30°)}$$

Donde n representa el número de plantas, s la superficie del terreno, y d la distancia óptima (deseada) entre plantas.



Una mala reforestación puede causar varios problemas ambientales, económicos y sociales. Algunos de los más destacados son:

- Pérdida de biodiversidad: Si se plantan especies no nativas o monocultivos en lugar de especies autóctonas, se puede alterar el equilibrio del ecosistema, lo que lleva a la pérdida de especies locales, tanto de flora como de fauna.
- Degradación del suelo: Plantar especies inapropiadas o realizar prácticas de reforestación sin considerar el tipo de suelo puede provocar erosión, pérdida de nutrientes y, a largo plazo, la degradación del suelo.
- Alteración del ciclo hidrológico: Una reforestación inadecuada puede afectar la absorción y retención de agua en el suelo, lo que puede alterar el ciclo hidrológico local, provocando inundaciones o sequías en áreas cercanas.
- Problemas con especies invasoras: La introducción de especies exóticas puede llevar a la proliferación de especies invasoras que compiten con las especies nativas, alterando los ecosistemas y causando desequilibrios ecológicos.
- Impacto en la fauna local: Una mala selección de especies puede resultar en la pérdida de hábitats para la fauna local, obligando a las especies a migrar o enfrentarse a la extinción.

Se conoce como plantas invasoras a las especies no nativas que se propagan rápidamente y desplazan a las autóctonas, afectando la biodiversidad y los ecosistemas. Un monocultivo es la plantación de una sola especie en grandes extensiones, lo que reduce la diversidad y puede tener consecuencias negativas en el suelo y la biodiversidad. En resumen, la reforestación es crucial para la supervivencia del planeta y debe realizarse de manera informada y sostenible para enfrentar los desafíos ambientales. [Azada Verde, 2024]

2. Situación Problema

El reto planteado para esta materia es el de encontrar la distribución óptima de las distintas especies de plantas utilizadas para reforestar un sector particular considerando la población de flora que existe en el sector y maximizando las relaciones simbióticas entre las especies reforestadas, de igual manera se busca reducir el número de interacciones negativas entre las plantas, siendo el objetivo final el aumentar la tasa de sobrevivencia de las especies sembradas en el sector.

2.1. **Justificación del problema.** El monitoreo de la supervivencia en las reforestaciones es un tema de gran interés para las personas que realizan este tipo de esfuerzos. En México ha habido un gran impulso y esfuerzo de distintas organizaciones gubernamentales y no gubernamentales por regenerar los ecosistemas mediante la reforestación. Se destaca el esfuerzo nacional realizado entre 1985 y 1998 con un promedio de 78,500 hectáreas reforestadas anualmente (Wightman, 2003).

Según las Reglas de Operación del Programa de Apoyos para el Desarrollo Forestal Sustentable una reforestación exitosa debería de contar con un porcentaje anual de sobrevivencia de al menos 80%, pero los esfuerzos realizados a nivel nacional a través de distintos tipos de ecosistema mantienen consistentemente una tasa de sobrevivencia anual de entre 30% y 53% con un promedio general de 43% (Prieto & Goche, 2016). Este gran desbalance ha llevado a que la forma de realizar reforestaciones siga siendo analizada de manera cuidadosa para buscar aumentar la tasa de sobrevivencia de las especies. El enfoque propuesto en este proyecto busca atacar esta problemática y aumentar la tasa de sobrevivencia de manera consistente en futuros esfuerzos de reforestación.

2.2. **Objetivo.** Generar un modelo o metaheurístico capaz de aumentar la tasa anual de sobrevivencia en los esfuerzos de reforestación considerando la permanencia de la flora nativa y la simbiosis entre los especímenes plantados.

3. Trabajo Relacionado

3.1. **Coloración de Arcos.** La coloración de arcos es un concepto en la teoría de grafos que es una extensión de la coloración de vértices y aristas. Mientras que la coloración de vértices asigna colores a los nodos de un grafo y la coloración de aristas asigna colores a las aristas, la coloración de arcos asigna colores a los arcos (es decir, las aristas orientadas) en un grafo dirigido.

Dado un grafo dirigido $G = (V, E)$, una coloración de arcos es una función que asigna un color a cada arco de tal manera que no hay dos arcos que compartan el mismo color y que parten del mismo vértice. Formalmente, si $f : E \to C$ es la función de coloración donde C es un conjunto de colores, entonces para cualquier par de arcos (u, v) y (u, w), si u es el vértice inicial de ambos arcos, entonces $f(u, v) \neq f(u, w)$.

3.1.1. *Aplicaciones.*

- Problemas de flujos de tráfico: En sistemas de tráfico, los caminos pueden ser modelados como arcos y la coloración de arcos puede utilizarse para asignar rutas o tiempos de cruce a diferentes vehículos o flujos de tráfico, asegurando que no haya congestión en un nodo particular (una intersección).
- Modelado de sistemas de producción: En sistemas de producción, donde las operaciones son modeladas como arcos en un grafo dirigido, la coloración de arcos puede ayudar a evitar que dos operaciones que deben realizarse en secuencia se realicen simultáneamente, asegurando así que el proceso de producción sea eficiente y libre de conflictos.
- Programación de tareas: Las tareas son representadas por arcos, y el objetivo es asignar tiempos (o colores) de inicio a las tareas de manera que no haya conflictos, es decir, que no haya dos tareas que necesiten realizarse al mismo tiempo por la misma máquina o individuo.

3.2. **Coloración de Aristas.** En este tipo de problemas se busca resolver la asignación de etiquetas llamadas colores o k *colores* a elementos del grafo a analizar. Siendo que una coloración de un conjunto de *aristas* E resulte en la asignación de n *numero* de colores a cada arista perteneciente del grafo se representa con la función $f : E \to \{1, 2, ..., k\}$ convirtiéndose esto en una $k - coloracion$, considerando que de tal forma, las aristas adyacentes no compartan el mismo color, como también, no puede haber dos aristas del mismo color teniendo vértices en común (Nekomath, 2022).

Por lo tanto, cuando se busca una generalización de un número específico de coloración existe el inconveniente de que, mientras menor sea este número mayor conflicto se creará en la propia coloración, dado que el índice cromático de una gráfica G es el menor entero k tal que G tiene una $k - coloracion$ propia que se denomina como $x'(G)$ donde se aplican los enteros no negativos en la selección de colores. Donde claro se busca acotar este índice cromático según su grado máximo permitido, presentándose de la siguiente manera, $x'(G) \geq \Delta(G)$ donde a su vez según el teorema de Vizing se puede considerar un grado más que el grado máximo demostrándose de la siguiente forma $Clase\ 1 : x_0(G) = \Delta(G)$ y $Clase\ 2 : x_0(G) = \Delta(G) + 1$ (Nahid, 2021).

3.2.1. *Aplicaciones.*

- En la programación: Especificando los casos en donde con base en la coloración de grafos resulta de gran utilidad en problemas donde es necesario la asignación de tareas o eventos para diferentes recursos o de diferencias horarias, donde se cuenta con un número cromático para el grafo que nos proporcionará el número mínimo de recursos o intervalos de tiempo necesarios para la resolución en el acomodo de todas las tareas o eventos sin conflicto alguno.
- Coloreado de mapas: Donde se busca la asignación de colores para todas las regiones de un mapa en específico de forma que las regiones adyacentes tengan colores diferentes para brindar una mejor comprensión para el usuario, aplicando el número cromático del grafo que proporciona un mínimo de colores necesarios para emplear en el mapa sin que las regiones adyacentes tengan el mismo color.

- Comunicación inalámbrica: Se destina en la asignación de las frecuencias a diferentes dispositivos de forma que no existan dos dispositivos adyacentes que utilicen la misma frecuencia inalámbrica, impidiendo posibles incomunicaciones, ya que se proporciona el número mínimo de frecuencias necesarias para evitar interferencias.
- Diseño VLSI: Este uso asigna los colores a utilizar en los módulos de un circuito de forma que no haya dos módulos adyacentes del mismo color, empleando un diseño de un circuito sin complicaciones y dificultades de identificación (Amithesh Gupta, 2023).

3.3. **Simulación de Montecarlo.** La Simulación de Montecarlo es una técnica matemática que emplea muestreo aleatorio para predecir resultados en sistemas complejos con incertidumbre. Utilizada en áreas como finanzas, ingeniería y gestión de proyectos, esta técnica evalúa riesgos y distribuciones de probabilidad, generando múltiples escenarios posibles. Aunado a esto, fue desarrollada durante la Segunda Guerra Mundial con la finalidad de analizar situaciones complejas mediante la repetición de cálculos con diferentes valores aleatorios. Actualmente, es altamente utilizada para comprender mejor los riesgos y tomar decisiones informadas bajo incertidumbre (IBM, 2019).

3.3.1. *¿Cuántas corridas se tienen que hacer para lograr cierto nivel de eficiencia?* Esta cantidad está vinculada directamente con el nivel de precisión deseada en los resultados aunado a la complejidad del modelo. Generalmente, lo que se busca optimizar en estas situaciones son los recursos computacionales y el tiempo en la vida real.

- Modelos sencillos: De 1,000 a 10,000 corridas para obtener resultados confiables.
- Modelos complejos: Más de 10,000 corridas para obtener resultados confiables.

3.3.2. *¿Qué ventajas y desventajas tiene este método?* Como ventajas encontramos la flexibilidad para modelar diferentes escenarios y variables, facilidad de implementación y útil en la evaluación de riesgos. Como desventajas, tiene un alto costo computacional; la precisión depende de la calidad de los datos de entrada y el número de simulaciones, y es sensible a suposiciones del modelo.

3.4. **Cadenas de Márkov.** Las cadenas de Márkov son usadas para modelar y predecir el comportamiento de fenómenos con un número finito de estados y de transiciones entre ellos en los que cualquier estado dado depende del estado anterior. En el libro “Introduction to Probability” (Ross, 2010) las cadenas de Márkov se definen de la siguiente manera: Sea $X_n, n \in \mathbb{N}$ un proceso estocástico cuyas acciones están expresadas por la serie de enteros naturales $n = \{0, 1, 2, \cdots\}$. Si $X_n = i$ se sabe que nos encontramos en el estado i en el tiempo n. En una cadena de Márkov se sabe que cuando el proceso se encuentra en el estado i existe una probabilidad P_{ij} de que en el próximo paso de tiempo el proceso se encuentre en el proceso j:

$$P\{X_{n+1} = j | X_n = i, X_{n-1} = i_{n-1}, \cdots, X_1 = i_1, X_0 = i_0\} = P_{ij}, \forall i_n, j_n \geq 0$$

En resumen, en una cadena de Márkov se cuenta con estados finitos con probabilidad de transición conocida en los que la probabilidad de transición de un estado a otro depende del estado actual del proceso. Las probabilidades de transición P_{ij} en una cadena de Márkov deben de cumplir con las siguientes condiciones:

$$\begin{gathered} P_{ij} \geq 0 \\ i, j \geq 0 \\ \sum_{j=0}^{\infty} P_{ij} = 1 \\ i \epsilon \mathbb{N} \end{gathered}$$

3.4.1. *Aplicaciones de cadenas de Márkov.* Las cadenas de Márkov son utilizadas para una gran cantidad de temas en los que se requiere una predicción sobre futuros eventos como las descritas a continuación.

- Física: Se utilizan para la modelación termodinámica y la mecánica estadística cuando las probabilidades se utilizan para conocer detalles desconocidos, o no modelados de un sistema.
- Teoría de filas: Esta teoría esta basada en las cadenas de Márkov. Su rol es crítico en optimizar el rendimiento de las redes de telecomunicación.
- Economía y finanzas: Las cadenas de Markov son utilizadas para la modelación de fenómenos como la distribución de ingresos, las fluctuaciones en bolsa y la predicción de los colapsos en los mercados.

3.5. **Distribuciones de Probabilidad.** Una distrbución de probabilidad es una función matemática que relaciona los estados de una variable de un *experimento aleatorio* con la probabilidad de que estos ocurran (Ash, 2008). Es importante distinguir entre las variables aleatorias clasificandolas como *continuas* o *discretas*, pues la forma de tratarlas a través de distribuciones de probabilidad es distinta.

3.5.1. *Variables Discretas.* En este caso basta con utilizar una función de masa de probabilidad, cuya salida al evaluar sobre un valor (para la variable aleatoria) es directamente su probabilidad de ocurrencia. Algunos ejemplos comunes de distribuciones discretas son:

- Distribución binomial: Expresa la probabilidad de obtener cierto número de éxitos en una serie de experimentos aleatorios e independientes entre sí, dada su probabilidad de éxito individual.
- Distribución de Poisson: Expresa la probabilidad de que ocurra un determinado número de eventos independientes en un intervalo de tiempo, dada la frecuencia media de los eventos.
- Distribución hipergeométrica: Expresa la probabilidad de obtener cierta cantidad de elementos de un tipo al realizar una muestra aleatoria y sin reemplazo de una población cuya distribución se conoce.

3.5.2. *Variables Continuas.* Por otro lado, dada la naturaleza de estas variables, sería absurdo asignar directamente una probabilidad a cada estado (pues son infinitos incontables), por lo que se utilizan funciones de densidad de probabilidad a través de las cuales puede calcularse la *masa de probabilidad* integrando sobre el intervalo cuya probabilidad desea obtenerse. Algunos ejemplos comunes de distribuciones continuas son:

- Distribución exponencial: Modela la probabilidad asociada al tiempo de espera para la ocurrencia de un evento, conociendo su frecuencia media de ocurrencia.
- Distribución normal: Modela datos que se distribuyen simétricamente al rededor de su media, volviéndose progresivamente menos comunes conforme se alejan de ella.

4. Definición del problema

Uno de los mayores desperdicios derivados de los procesos de restauración ecológica es el desaprovechamiento de las plantas durante la reforestación. La causa del desaprovechamiento de recursos radica en que se determina en base a estimaciones la cantidad de plantas necesarias para restaurar un área dada. Sin embargo, es común sobrestimar el número de plantas debido a la topografía del terreno y las plantas y elementos ya presentes en el área. Además, la sobrevivencia de las plantas disminuye si entre las plantas vecinas existen relaciones negativas. Considerando esto, el presente reto busca propooner un modelo matemático y computacional que permita plantear una mejor distribución de las especies a plantar en un polígono minimizando la competencia entre ellas y tomando en cuenta los elementos ya presentes en el mismo.

5. Modelo Matemático

En adelante se define la notación $[n] = \{i \mid i \in \mathbb{N}, i \leq n\}$ para facilitar la lectura.

5.1. **Conjuntos.** Sea $G = (N, M)$ un grafo no dirigido y completo, de nodos N y arcos M; donde $N = [n]$ es el conjunto de *espacios de plantación*. Se define el grafo $H = (S, A)$ como un grafo no dirigido y completo, de nodos S y arcos A, donde $S = [s]$ es el conjunto de las especies de plantas a considerar. Sea $L = \{l_i\} \forall i \in S$ el conjunto de requerimentos por especie.

5.2. **Parámetros.** Sean $n \in \mathbb{N}$ la cantidad de *espacios de plantación*. Sea $s \in \mathbb{N}$ el número de especies de plantas a considerar. Los parámetros $l_i \in L$ corresponden a la cantidad requerida de especímenes de la especie $i \in S$ por hectárea. El grafo G tiene ponderaciones binarias $g_m : m \in M$ que toman valor de 1 si los *espacios de plantación* representados por los nodos unidos por el arco m mantienen una relación de *adyacencia espacial* en la plantación. El grafo H tiene ponderaciones no-negativas $h_a : a \in A$ que corresponden a una penalización por la competencia entre las especies que representan los nodos conectados por el arco a.

5.3. **Variables.** Sea $p_{ia} : i \in N, a \in S$ una binaria que toma como valor 1 si en el *espacio de plantación* i está plantado un especimen de la especie k, de lo contrario toma el valor 0. Sean z_{ijab} tal que $i, j \in N$ y $a, b \in S$ variables auxiliares que toman como valor el resultado de la operación lógica $p_{ia} \wedge p_{jb}$.

5.4. **Problema de Programación Lineal.** El planteamiento es el siguiente:

5.4.1. *Función Objetivo.*

$$\text{Minimizar} \quad \sum_{i\in N}\sum_{j\in N}\sum_{a\in S}\sum_{b\in S}(g_{(i,j)} * z_{ijab}) * h_{(a,b)} \tag{1}$$

5.4.2. *Restricciones.*

$$\sum_{a\in S} p_{ia} = 1 \qquad \forall i \in N \tag{2}$$

$$\sum_{i\in N} p_{ia} = l_a \qquad \forall a \in S \tag{3}$$

$$z_{ijab} \geq p_{ia} + p_{jb} - 1 \qquad \forall i, j \in N \quad a, b \in S \tag{4}$$

$$z_{ijab} \leq p_{ia} \qquad \forall i, j \in N \quad a, b \in S \tag{5}$$

$$z_{ijab} \leq p_{jb} \qquad \forall i, j \in N \quad a, b \in S \tag{6}$$

$$z_{ijab} \in \{0, 1\} \qquad \forall i, j \in N \quad a, b \in S \tag{7}$$

$$p_{ia} \in \{0, 1\} \qquad \forall i \in N, \ a \in S \tag{8}$$

5.4.3. *Lógica tras las Ecuaciones.* La función objetivo (Ecuación 1) puede entenderse pensando en el primer producto como una operación que solo toma el valor 1 cuando se reunen a la vez las siguientes condiciones:

- Los *espacios de plantación* i y j son *espacialmente adyacentes*, por lo que compiten entre sí.
- La planta en el *espacio de plantación* i es, en efecto, de la especie a.
- La planta en el *espacio de plantación* j es, en efecto, de la especie b.

Y toma el valor 0 en cualquier otro caso. Es decir, que solo vale 1 cuando la competencia entre el par de plantas p_{ia} y p_{jb} debe ser considerada. Luego, este valor se multiplica por $h_{(a,b)}$ que es la penalización por la competencia entre dicho par de plantas.

La primera restricción (Ecuación 2) asegura que en cada *espacio de plantación* haya exactamente una planta; no espacios vacíos, no plantas superpuestas. La siguiente restricción (Ecuación 3) asegura que se cumplan los requerimentos de INECOL sobre cantidad de especímenes por especie en toda la plantación. Las siguientes tres restricciones (Ecuaciones 4, 5, 6) se utilizan para linealizar la operación lógica $p_{ia} \wedge p_{jb}$ asegurando que z_{ijab} tome su valor. Las últimas dos restricciones (Ecuaciones 7 y 8) describen la naturaleza de las variables como binarias.

El modelo puede facilmente adaptarse a través de restricciones para cumplir con los requerimentos de plantación establecidos por INECOL (Tabla 1), incluso considerando plantas nativas, también mediante restricciones.

6. Métodos Heurísticos Propuestos

Dado que el modelo matemático tiene complejidad exponencial respecto a la cantidad de espacios de plantación, se vuelve rápidamente inviable para escalarlo al problema real (o a otros más complejos) y aún menos útil para una Simulación de Montecarlo, que requiere un alto número de iteraciones para converger. Por ello se proponen los siguientes métodos heurísticos y metaheurísticos:

6.1. **Divide y Vencerás.** Se propone es una división del espacio total de plantación en dos medios, dividiendo por la mitad una de las dos dimensiones, obteniendo así dos subproblemas, cada uno con la mitad de nodos. Obteniendo las respectivas soluciones para cada uno de ellos —utilizando PL, el modelo matemático descrito en la sección 5.4, y considerando las restricciones de cantidad de especímenes por especie sobre ambos subproblemas en conjunto— solo resta unirlos y calcular las penalizaciones por competencia que se generan entre sus bordes colindantes.

Nótese que cada área plantada como solucion a los subproblemas tiene al menos dos orientaciones (pues puede *rotarse 180°*) que no afectan el costo individual, pero sí el costo adicional generado por *ensamblarla* con la de el otro subproblema (que también tiene al menos dos orientaciones), por lo que puede eligirse la combinación de orientaciones que resulte más conveniente apriori para reducir costos. Estas operaciones son viables, pues calcular los costos adicionales conlleva complejidad lineal. En caso de que las áreas sean cuadradas, se tienen entonces 4 orientaciones posibles por cada una, abriendo más posibilidades para reducir costos.

6.1.1. *Ventajas Apriori.* El método puede aplicarse de forma recursiva, pues los subproblemas generados en una iteración pueden a su vez dividirse para obtener su solución mediante el mismo algoritmo. Mientras más subdivisiones, más velocidad de computo, pero menor cercanía al óptimo. Es un parámetro que puede ajustarse.

6.1.2. *Desventajas Apriori.* Dado que mantiene cierta dependencia con la optimización mediante PL su efectividad para acercarse al óptimo en tiempos razonables puede verse disminuida en gran medida conforme se aplica a problemas más grandes. En este trabajo —que es relativamente pequeño en términos de nodos— resulta útil, pero no se puede garantizar que sea satisfactoriamente escalable.

6.2. **Recocido Simulado.** El algoritmo de recocido simulado busca encontrar soluciones óptimas o cercanas a óptimas en problemas complejos de optimización. Se basa en la idea de realizar cambios aleatorios a la solución actual, con la posibilidad de aceptar soluciones peores a medida que "explora" el espacio de soluciones, pero esta probabilidad de aceptar soluciones peores disminuye con el tiempo.

En el contexto del problema abordado de optimización de plantación, el recocido simulado ha sido implementado para encontrar una configuración de las plantas que minimizara el "odio" entre las especies, asegurando al mismo tiempo que todas las especies sean plantadas y optimizando su disposición.

El algoritmo empieza creando una instancia de un objeto llamado Plantacion, que representa la disposición inicial de las plantas en la cuadrícula. Esta disposición inicial también toma en cuenta posiciones fijas de ciertas especies, que están predefinidas. Durante la ejecución del recocido simulado, se generan soluciones vecinas alterando la disposición de un par de plantas en la cuadrícula. El algoritmo evalúa el costo (nivel de competencia o "odio" entre especies adyacentes) de la nueva solución. Si la nueva disposición tiene un menor costo, se acepta inmediatamente. Si el costo es mayor, la nueva solución aún puede ser aceptada, pero con una probabilidad que depende de la diferencia de costo y la temperatura actual, como es característico del recocido simulado. A medida que la temperatura disminuye, el algoritmo es menos propenso a aceptar soluciones peores, enfocándose en la mejora local.

La solución final del algoritmo se almacena y puede ser visualizada como una imagen de la cuadrícula con la disposición óptima o casi óptima de las plantas. También se guardan métricas importantes como la distribución de plantas por especie y el valor promedio de competencia (odio) entre las especies en la cuadrícula.

6.3. **Algoritmo Genético.** El algoritmo genético es una técnica de resolución de problemas que imita la evolución biológica como estrategia para resolver problemas englobándose dentro de lo que se denomina técnicas basadas en poblaciones. Algunas de sus caracetrísticas principales son las siguientes:

- Son algoritmos de optimización, búsqueda y aprendizaje inspirados en los procesos de evolución natural y genética de los individuos más aptos
- Puede ser utilizado para resolver problemas estocásticos.
- Cada solución se considera como un cromosoma codificado por un conjunto de genes (en este caso todo cromosoma debe de cumplir los criterios de factibilidad).
- Al conjunto de N soluciones (cromosomas) se les conoce como población.
- La función fitness se basa en la aptitud de sobrevivencia de un individuo en su entorno según la teoría de la selección. Siguiendo esta relación las mejores soluciones tendrán los mejores valores fitness.

- Existe un proceso de selección y cruce de padres en el que se combinan los cromosmoas para generar nuevos cromosomas que se denominan como hijos.
- Cada hijo tiene una probabilidad de mutación para cada gen en él. La mutación se define como un cambio en el gen que no proviene de ningún padre
- La población original se combina con los hijos y por cada generación que exista se realiza un criterio de reemplazo para disminuir la población existente.

El algoritmo genético fue diseñado de la siguiente manera:

(1) **Tipo de cromosoma:** Cada gen del cromosoma representa la planta que se encuentra en un surco.
(2) **Longitud del cromosoma:** La longitud del cromosoma es equivalente a la cantidad de surcos en el terreno en orden de izquierda a derecha, y de arriba a abajo.
(3) **Criterio de inicialización:** Los primeros cromosomas se generan en base a una distribución de plantación aleatoria que cumple con la distribución estipulada y con la posición de las plantas nativas.
(4) **Criterio de infactibilidad:** El cromosoma será infactible si la distribución de plantas no se cumple.
(5) **Criterio de paro:** Número de generaciones especificadas (100 generaciones).
(6) **Fitness:** El fitness es calculado como el inverso de la suma de la competencia según la distribución de plantas del cromosoma.
(7) **Tamaño de la población:** 50 individuos.
(8) **Probabilidad de cruce:** 100%. Los individuos tienen una probabilidad de selección proporcional al valor de su función fitness con respecto a la suma de las funciones fitness de todos los individuos en la población.
(9) **Punto de cruce:** Un único punto de cruce.
(10) **Lugar de cruce:** A la mitad del cromosoma.
(11) Probabilidad de mutación: 0.1
(12) **Criterio de reemplazo:** Eliminación de los individuos de peor fitness por generación

7. Simulación y Experimentación

Dados los métodos de solución para un problema particular mencionados en la sección anterior, el objetivo de esta sección es implementar el Método de Montecarlo para que, después de multiples interaciones, distintas posibles plantaciones aleatorias converjan a una cantidad concreta de especímenes nativos por especie, especímenes a suministrar por especie, y nivel de competencia esperado por hectárea. Con este fin, a continuación se mencionan los distintos parámetros, supuestos y métodos utilizados.

7.1. **Objetivos de Plantación.** De acuerles plantaciones do a los requerimentos de INECOL, las especies deben distribuirse en una hectárea ya sido intervenida según se indica en la Tabla 1. Una hectárea se define en nuestros modelos como 676 *espacios de plantación* distribuidos como un arreglo cuadrado de $26 * 26$. En cada uno de los espacios se colocará un espécimen; y ningún espécimen puede estar fuera de ellos.

Especie	**Especímenes por Ha.**
Agave lechuguilla	42,00
Agave salmiana	196,00
Agave scabra	42,00
Agave striata	42,00
Opuntia cantabrigiensis	49,00
Opuntia engelmannii	38,00
Opuntia robusta	73,00
Opuntia streptacantha	64,00
Prosopis laevigata	86,00
Yucca filifera	26,00
Total	**658,00**

Table 1. Requerimentos de INECOL por especie por hectárea.

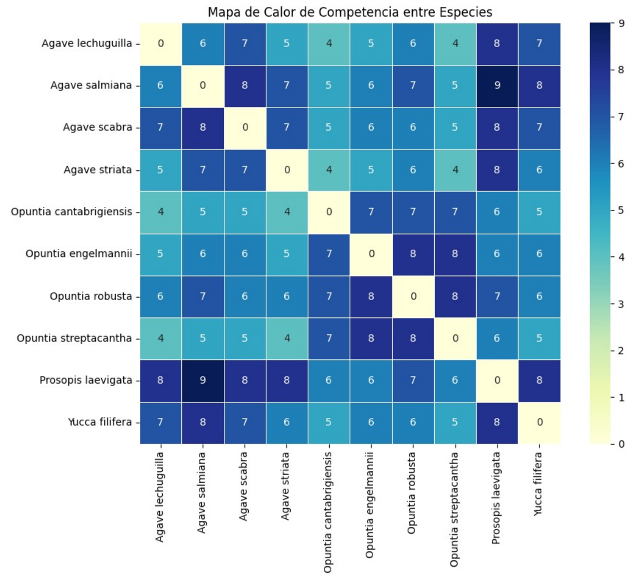


Figure 1. Mapa de Calor de Competencia entre Especies.

7.2. **Nivel de Competencia.** De acuerdo a la información proporcionada por el INECOL, en cuanto a la diversidad de especies de plantas a utilizar, se investigaron datos acorde a las necesidades principales de cada especie, como su región natural, la cantidad de agua que requieren, sus usos y datos adicionales, para crear los parámetros que determinaron el nivel de competencia entre especies mostrado en la Imagen 1.

Se consideraron valores en un rango de 0 y 9, donde el 0 es la mínima competencia y el 9 la máxima. Arreglando las especies y las competencias involucradas entre ellas dentro de una matriz, se creó un mapa de calor para visualizar el comportamiento de las espeies, y además haciendo uso de la librería de "Pandas" en Python, se calculó la media y desviación estándar de la competencia por especie, se hizo un DataFrame para visualizar los resultados, se calculó la matriz de correlación entre las especies y se analizaron los resultados obtenidos.

A partir de los cuales, posteriormente se evaluaron las medias de competencias, como también, sus desviaciones estándar entre las mismas, resultando en la Tabla 2.

Especie	**Media de Competencia**	**Desviación Estándar**
Agave lechuguilla	5.2	2.1354
Agave salmiana	6.1	2.3854
Agave scabra	5.9	2.2113
Agave striata	5.2	2.1354
Opuntia cantabrigiensis	5.0	2.0000
Opuntia engelmannii	5.7	2.1471
Opuntia robusta	6.1	2.1656
Opuntia streptacantha	5.2	2.2271
Prosopis laevigata	6.6	2.4166
Yucca filifera	5.8	2.1817

Table 2. Estadísticas descriptivas.

Se observa que Prosopis laevigata tiene la media de competencia más alta (6.6), lo que sugiere que es una especie altamente competitiva en comparación con las otras. En contraste, Opuntia cantabrigiensis tiene la media más baja (5.0), lo que indica una menor competitividad relativa.

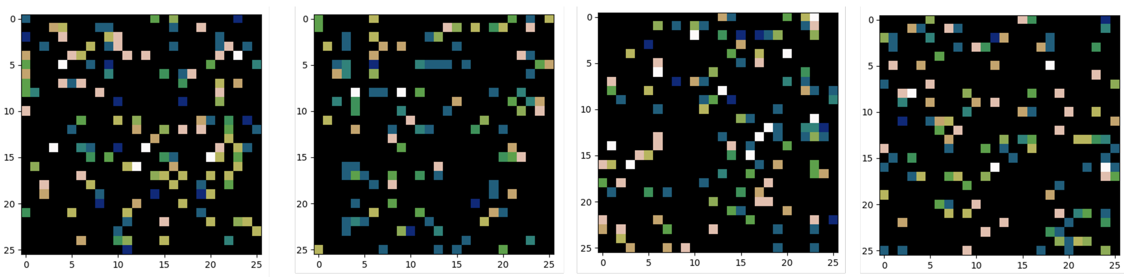


Figure 2. Hectareas nativas generadas aleatoriamente.

Las especies con una desviación estándar más alta, como Prosopis laevigata (2.416) y Agave salmiana (2.385), muestran más variabilidad en su nivel de competencia con otras especies. Esto podría sugerir que estas especies tienen interacciones más diversas dependiendo de la especie con la que compiten.

7.3. **Distribución de Probabilidad de Plantas por Hectárea.** El objetivo de una simulación es imitar la realidad, y en congruencia con ello, tomamos como base para nuestra generación aleatoria de *hectáreas nativas* (esta definición se concretará en la sección 7.4) datos reales proporcionados por INECOL (Ver Figura 4 en el anexo) que contienen 30 muestras sobre la cantidad de plantas nativas en un espacio determinado (denominado *polígono*). A partir de dichas muestras, escaladas a una hectárea, se calculó la media de aparición por especie, y se tomó esta como referencia para la distribución.

Se asumen las siguientes propiedades sobre la distribución de plantas nativas:

- Las plantas nativas son exclusivamente de alguna de las especies usadas para la reforestación.
- La cantidad de plantas nativas para cada especie corresponde a una distribución de Poisson cuya tasa de llegada (λ) es la media aritmética de los datos históricos de existencia nativa de esa especie.
- En cada *espacio de plantación* puede existir únicamente una planta nativa.
- Toda planta nativa que exista ha aparecido, convenientemente, en un espacio de plantación.
- Toda planta nativa debe de ser considerada dentro de los requerimiento de INECOL (Tabla 1).

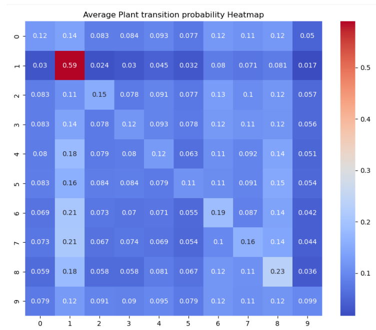
7.4. **Simulación de Montecarlo Aplicada.** Tal como se explicó en la Sección 3.3, el método de Montecarlo utliza el muestreo aleatorio para ofrecer conclusiones con las cuales se pueden tomar decisiones apropiadas en incertidumbre. En este caso las muestras consistirán en *hectáreas nativas*: matrices de $26 * 26$ con *espacios libres* (aquellos en los que INECOL puede intervenir y plantar la especie que más le convenga) y *espacios fijos* (aquellos donde ha aparecido una planta nativa de una especie determinada y no puede moverse, por lo que el resto de especies deben disponerse a su alrededor de la mejor forma posible). Como resultados, se espera obtener la cantidad esperada de especímenes nativos por especie, especímenes a suministrar por especie, y nivel de competencia esperado por hectárea.

7.4.1. *Generación de Hectáreas Nativas.* Para generar una hectárea nativa, para cada especie, se extraé un número aleatorio de su distribución (ver Sección 7.3), el cual será la cantidad de especímenes de dicha especie en esta hectárea. Posteriormente se seleccionan al azar (con probabilidad uniforme) tantos espacios de plantación en una hectárea vacía como especímenes nativos totales se han determinado en el paso anterior. Se asignan aleatoriamente uno a uno los especímenes nativos y los espacios seleccionados. En la Figura 2 se muestran algunas de estas configuraciones aleatorias, donde el color negro representa un *espacio libre* y el resto de colores son especies nativas en *espacios fijos*.

7.4.2. *Muestreo Iterativo.* Para este problema y en esta escala académica, se ha fijado a 1000 el número de iteraciones. Para cada iteración, primero se genera aleatoriamente una hectárea nativa, la cual se utiliza como entrada para ser optimizada mediante alguno de los métodos presentados en la Sección 6, y cuya salida será almacenada como parte del muestreo.

Una vez que se han completado todas las iteraciones y conseguido 1000 muestras se obtiene la convergencia estadística para cada una de las variables que se propuso conocer al inicio de esta sección, siendo estos los resultados fundamentales de la investigación.

Tras realizar 1,000 iteraciones del algoritmo genético en el que para cada iteración se simulaba también una distribución de plantas nativas distinta la convergencia de la matriz de transición entre especies fue la siguiente:



7.5. **Características de la PC y Software. Modelo de la PC:** OMEN by HP Laptop 15-dc1xxx
Sistema Operativo: Windows 10 Pro - 22H2
Capacidad de disco duro: 1 tb de SSD
Memoria RAM y capacidad: 24.0 GB (23.8 GB usable)
Tipo de procesador: Intel(R) Core(TM) i7-9750H CPU @ 2.60GHz
Número de núcleos: 6 núcleos
Software utilizado y versión: Jupyter Notebook: Ejecutado a través de Visual Studio Code (2024) con el kernel de Python 3.11.3

8. Resultados

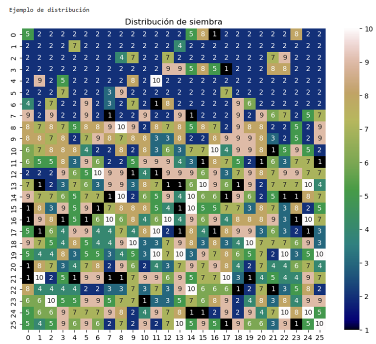
8.1. **Modelo Matemático + Divide y Vencerás.** La implantación del modelo matemático (Sección 5.4) se realizó en el software GAMS, donde una vez concluido nos provee los resultados obtenidos, siendo 58 el mejor posible con un total de 11,979 iteraciones y 366 nodos en un tiempo computacional de 1.478 segundos, sin embargo, este procedimiento no resulto viable para ser considerado, pues, es requerido dividir en numerosas ocasiones para que el modelo matemático resuelva de manera más rápida, lo que ocasiona que la optimización se vuelve casi nula, por lo que, este modelo no fue viable debido a su complejidad.

8.2. **Recocido Simulado.** Por parte de la simulación de recocido simulado podemos decir que se ejecutó satisfactoriamente, donde los tiempos de ejecución por cada iteración eran de un aproximado de 1.30 minutos, por lo que, el tiempo requerido para que se realicen las 1000 iteraciones es de un aproximado de 21.66 horas, por ende, a cuestiones de tiempo no se realizaron las 1,000 iteraciones. Donde a continuación se puede identificar el reporte de resultados obtenidos en la Tabla 3.

Especie	Plantas nativas	Plantas a suministrar	Competencia media por especimen
Agave lechuguilla	8.5	35.5	10.46
Agave salmiana	36.7	160.3	10.46
Agave scabra	8	36	10.46
Agave striata	7.6	36.3	10.46
Opuntia cantabrigiensis	10.3	40.6	10.46
Opuntia engelmannii	8.8	31.2	10.46
Opuntia robusta	14.7	60.3	10.46
Opuntia streptacantha	13	53	10.46
Prosopis laevigata	16.8	70.2	10.46
Yucca filifera	4.9	23	10.46

Table 3. Resultados obtenidos.

8.3. **Algoritmo Genético.** Tras haber realizado una simulación de Montecarlo de 1000 iteraciones el algoritmo genético obtuvo un valor de competencia promedio de 5974.611 según la distribución de plantas y la matriz de competencias proporcionadas. Dado que se consideró una competencia de 0 para plantas de la misma especie las soluciones propuestas tienen una fuerte tendencia a agrupar las plantas de la misma especie en base a la ubicación y especie de las plantas nativas. Siendo que existía una representación bastante grande de plantas de la especie 2: "Agave Salmiana". el algoritmo genético tiende a generar una gran isla de esta planta para cada solución propuesta. A continuación se presenta un ejemplo de distribución planteada por una iteración de la simulación de Montecarlo bajo los parámetros especificados en la sección 6.3.



9. Discusión

9.1. **Modelo Matemático + Divide y Vencerás.** Este enfoque se basa en dividir el problema en subproblemas más pequeños utilizando un enfoque de optimización matemática. A continuación, se enumeran sus características:

9.1.1. *Ventajas a Posteriori.*

- Permite descomponer el problema en piezas más manejables, lo cual puede facilitar el análisis de cada parte.
- Se puede aplicar de manera recursiva, generando nuevas subdivisiones que pueden mejorar la velocidad de cómputo.
- Aprovecha un enfoque estructurado (modelo matemático), que puede asegurar que las restricciones sean respetadas de manera exacta.

9.1.2. *Desventajas a Posteriori.*

- Aunque puede ser eficiente para problemas pequeños, su escalabilidad es limitada, especialmente cuando los nodos crecen en número.
- Su dependencia del enfoque lineal de optimización (Programación Lineal) podría generar soluciones subóptimas para problemas más complejos donde las interacciones entre nodos son no lineales.
- Aunque se considera la rotación de las áreas plantadas, no garantiza una solución globalmente óptima.

En resumen, el método es útil para descomponer el problema, pero su eficacia disminuye con problemas más grandes, pues es necesario subdividirlos en muchas partes para lograr tiempos de cómputo razonables, los cuales, en la misma línea, disminuyen a la par de la cercanía al óptimo, pues más subdivisiones crean un mayor márgen de mejora. No recomendamos que este método se utilice a gran escala.

9.2. **Recocido Simulado.** El recocido simulado es un método probabilístico que explora soluciones vecinas y decide aceptar soluciones subóptimas en función de una temperatura decreciente para evitar caer en óptimos locales.

9.2.1. *Ventajas a Posteriori.*

- Es flexible y puede evitar quedarse atrapado en óptimos locales debido a su capacidad para aceptar soluciones peor calificadas al inicio del proceso.
- Funciona bien en problemas con muchas restricciones y puede manejar la complejidad no lineal del problema de plantación.
- Es fácilmente adaptable a nuevas consideraciones que pueda requerir el socio formador.

- Se puede ajustar a diferentes parámetros (como la tasa de enfriamiento) para mejorar los resultados.
- Demostró ser útil para la simulación de Montecarlo en tiempos razonables.

9.2.2. *Desventajas a Posteriori.*

- El tiempo de cómputo puede ser mayor debido a la exploración de muchas soluciones vecinas.
- No siempre garantiza alcanzar la solución óptima, sino una aproximación buena.
- La calidad de la solución depende fuertemente de los parámetros de enfriamiento y la solución inicial, lo que podría hacer que no sea tan eficiente como el genético en problemas grandes.

En resumen, el recocido simulado es una herramienta poderosa para optimización no lineal, pero su eficiencia y exactitud pueden verse comprometidas cuando se trata de problemas muy grandes o complejos, especialmente si no se hace un correcto ajuste de hiperparámetros para cada instancia particular del problema. Es viable utilizar este algoritmo a gran escala, pero podría haber mejores opciones.

9.3. **Algoritmo Genético.** El algoritmo genético es un método inspirado en la evolución biológica, donde las soluciones se representan como "individuos" que evolucionan mediante operaciones como selección, cruza y mutación. Se evalúan las soluciones y las mejores se combinan para generar soluciones de mayor calidad.

9.3.1. *Ventajas a Posteriori.*

- Funciona muy bien en problemas de gran complejidad y con muchas restricciones, como el de plantación, donde la competencia entre especies es un factor clave.
- Es altamente paralelo, lo que significa que se pueden evaluar muchas soluciones simultáneamente, acelerando el proceso de optimización.
- Tiene menos tendencia a quedarse atrapado en óptimos locales en comparación con el recocido simulado, gracias a su capacidad para explorar grandes áreas del espacio de soluciones a través de la mutación y la recombinación de soluciones.

9.3.2. *Desventajas a Posteriori.*

- Como todos los algoritmos metaheurísticos, no garantiza la solución óptima global..
- Requiere una calibración cuidadosa de los parámetros (como la tasa de mutación y la selección), pero una vez ajustado, suele superar a otros métodos en términos de calidad de la solución.

Resultó muy útil para este problema, encontrando buenas soluciones aplicadas (minimizando la compentencia, pero sin llegar al monocultivo), y fue capaz de proporcionar muestras para una simulación de Montecarlo en tiempos razonables. Entre nuestras opciones, parece ser la más viable para utilizarse en un contexto práctico.

9.4. **Síntesis.** Aunque los tres métodos tienen sus propias ventajas y desventajas, el algoritmo genético demostró ser el más eficiente para resolver el problema de plantación optimizando las restricciones de competencia entre las especies y cumpliendo los requisitos de distribución de plantas.

El modelo matemático + divide y vencerás es útil para problemas más pequeños o altamente estructurados, pero no escala bien. El recocido simulado es flexible y robusto, pero puede ser lento y no siempre alcanza el óptimo global en problemas muy grandes. El algoritmo genético sobresale porque equilibra la exploración y explotación del espacio de soluciones, logrando mejores resultados en problemas complejos y de mayor tamaño, como el problema de plantación.

Conclusiones Generales

Este proyecto nos permitió explorar diversas metodologías para optimizar la distribución de plantas en un polígono de siembra, priorizando la reducción de la competencia entre especies. Tras aplicar un modelo matemático y comparar los resultados de diferentes algoritmos metaheurísticos, hemos concluido que el algoritmo genético fue el enfoque más efectivo para este tipo de problema. A continuación, se detallan los hallazgos principales y las recomendaciones basadas en los resultados obtenidos:

(1) **Análisis de Competencia entre Especies:** Utilizando la matriz de competencia, identificamos que especies como *Agave salmiana* y *Opuntia streptacantha* presentan altos niveles de competencia. Para mitigar estos efectos, recomendamos que las especies con alta probabilidad de competencia sean distribuidas más separadamente en el terreno. Este análisis fue clave para guiar las decisiones del algoritmo genético, que ajustó la distribución de las especies con el fin de minimizar estos conflictos.

(2) **Algoritmo Genético como la Solución Óptima:**
Eficacia: El algoritmo genético demostró ser el método más eficaz para minimizar la competencia entre especies. Esto se logró al analizar los picos negativos de competencia y ajustarlos a través de múltiples iteraciones. La deformación del espacio geométrico, antes de implementar los algoritmos de mis compañeros, mostró áreas de alta competencia que el algoritmo genético logró mitigar de manera eficiente.

Resultados: Con base en las 1000 iteraciones realizadas durante la simulación de Monte Carlo, se observó que el coeficiente de competencia convergió a 7071, con una competencia media de 10.46 por especie. Esto reafirma la efectividad del modelo y su capacidad para encontrar una distribución espacial óptima que minimice la competencia.

Importancia del Análisis Previal del Terreno: Es fundamental realizar un análisis topológico detallado del terreno antes de aplicar los algoritmos. La matriz de competencia y la proyección en el espacio geométrico proporcionaron una base sólida para las decisiones de siembra, ayudando a reducir los picos de competencia identificados. Este análisis guió la ejecución del algoritmo genético, asegurando una distribución eficiente y minimizando los conflictos entre especies.

Recomendaciones

- **Implementación del Algoritmo Genético:** Recomendamos que el socio formador implemente el algoritmo genético para la distribución de plantas, ya que ha demostrado ser la solución más eficaz. Es importante que las especies con alta probabilidad de competencia se ubiquen de manera más espaciosa, como se identificó en el análisis de la matriz de competencia y las visualizaciones topológicas.
- **Monitoreo y Ajustes:** Aunque el algoritmo genético es efectivo, sugerimos que se realice un monitoreo continuo de las plantaciones para ajustar la distribución según factores cambiantes del entorno, como las condiciones climáticas o la erosión del suelo.
- **Optimización del Proceso:** Si bien el modelo ha sido exitoso, es recomendable explorar formas de mejorar la eficiencia computacional, como la paralelización del código, para reducir los tiempos de cálculo en simulaciones más grandes.

9.5. **Posibles Mejoras.** Como siguientes pasos para la mejora de resultados en base a los algoritmos propuestos se proponen las siguientes mejoras:

- *Adaptación Dinámica de Parámetros:* Implementar un esquema de adaptación dinámica de los parámetros (tasa de mutación, cruza, etc.) durante la ejecución del algoritmo. Por ejemplo, aumentar la tasa de mutación si se observa estancamiento en las soluciones o reducirla a medida que las soluciones se acercan a la convergencia.
- *Diversificación del Pool Genético:* Se podría incorporar un mecanismo para garantizar la diversidad genética en la población, evitando que las soluciones converjan demasiado rápido hacia una única área del espacio de soluciones. Esto se puede hacer introduciendo variación en las selecciones o reforzando la mutación en momentos clave del proceso.
- *Mejorar la Función de Fitness:* Refinar la función de evaluación de las soluciones (fitness) para tener en cuenta más factores, como la estabilidad de las plantas a largo plazo o el impacto ecológico en otros aspectos del terreno. Esto podría ayudar a mejorar las soluciones y acercarse más al óptimo deseado.

9.6. **Recomendaciones para el Socio Formador.**

- Se recomienda implementar un sistema de monitoreo durante la ejecución del algoritmo, que permita visualizar cómo las soluciones van mejorando con el tiempo y si el algoritmo se está estancando en algún punto. Esto permitiría ajustar los parámetros en tiempo real y mejorar la eficiencia del proceso.
- Se sugiere realizar pruebas con diferentes configuraciones de parámetros (tamaño de la población, tasa de mutación, número de generaciones) para identificar qué configuración proporciona el mejor balance entre tiempo de ejecución y calidad de solución. Esto sería una fase experimental inicial antes de desplegar el modelo de forma completa.
- Es importante que el socio formador tenga en cuenta que a medida que el problema se vuelva más grande o más complejo (por ejemplo, al aumentar el número de plantas, especies, o el tamaño del terreno), el algoritmo genético puede requerir más recursos y tiempo para alcanzar buenas soluciones.

Referencias

Amithesh Gupta. (2023). *Graph coloring and its applications, chromatic number*. Medium Publisher: Medium. https://medium.com/@shadow-666/graph-coloring-its-applications-chromatic-number-9a930f1d0c75 Consultado el 17/08/2024.

Ash, R. B. (2008). *Basic probability theory*. Dover Publications.

IBM. (2019). *¿qué es la simulación monte carlo?* IBM. https://www.ibm.com/mx-es/topics/monte-carlo-simulation Consultado el 16/08/2024.

Iglesias, L. (2021). Siembra a Tresbolillo: Competencia Matemática, geometría plana aplicada en huertos y jardines. Día Mundial del Medio Ambiente. https://matematicas11235813.luismiglesias.es/2021/06/05/siembra-a-tresbolillo-competencia-matematica-geometria-plana-aplicada-en-huertos-y-jardines-dia-mundial-del-medio-ambiente/

Nahid, J. N. (2021). *Coloraciones consecutivas en gráficas y digráficas*. Miscelánea Matemática de la Sociedad Matemática Mexicana. https://doi.org/10.47234/mm.7105 Consultado el 17/08/2024.

Nekomath. (2022). *Coloración de aristas y teorema de ramsey — matemáticas discretas para ciencia de datos*. Nekomath. https://madi.nekomath.com/P2/Ramsey.html#:~:text=la%20siguiente%20definici%C3%B3n.-,Definici%C3%B3n.,izquierdo%20llegan%20dos%20aristas%20rojas Consultado el 17/08/2024.

Prieto, J., & Goche, J. (2016). *Las reforestaciones en Mexico* (1st ed.). UJED.

Ross, S. (2010). *Markov Chains* (10th ed.). ELSEVIER.

Wightman, B. S., Kevyn E. Cruz. (2003). La cadena de la reforestación y la importancia en la calidad de las plantas. *Foresta Veracruzana*. https://www.redalyc.org/articulo.oa?id=49750108

Anexos

Muestras de la Distribución de Especies Nativas. La información que se presenta en la Tabla 4 fue proporcionada por INECOL. Cada columna corresponde a una especie distinta (abreviada por motivos de espacio), excepto la primera, que contiene el número de muestra, y la última, que contiene el tamaño del espacio donde se obtuvo la muestra (medido en hectáreas).

#	A. le.	A. sa.	A. sc.	A. st.	O. ca.	O. en.	O. ro.	O. st.	P. la.	Y. fi.	Ha
1	8	46	16	16	11	14	18	12	15	9	1.28
2	58	263	47	49	60	44	111	95	98	39	6.64
3	66	236	51	50	71	56	100	78	123	28	6.76
4	10	52	15	9	15	10	18	20	23	11	1.38
5	65	280	66	61	92	54	124	114	106	41	8
6	67	306	63	61	81	62	118	91	133	45	7.82
7	41	209	43	49	48	52	87	60	97	27	5.53
8	32	252	46	40	73	39	86	79	91	26	5.64
9	58	269	58	53	57	58	94	91	117	37	7.11
10	47	209	47	41	66	43	96	71	108	29	6.11
11	48	233	41	37	48	40	94	65	94	27	5.64
12	36	182	43	44	51	41	68	72	67	19	4.92
13	50	189	49	28	43	33	73	76	97	25	5.05
14	35	186	38	43	39	44	73	61	74	23	4.75
15	40	311	52	62	68	63	132	89	149	53	7.97
16	70	290	55	54	84	50	105	77	121	40	7.34
17	56	233	39	46	71	45	83	81	97	32	5.98
18	45	204	34	33	56	46	82	48	95	17	5.4
19	56	223	51	48	50	48	93	90	120	33	6.28
20	69	285	57	75	77	74	107	103	104	25	7.6
21	54	287	70	62	89	58	126	109	139	44	8
22	75	292	60	68	104	62	109	95	125	43	8
23	44	288	69	51	69	60	122	100	106	39	7.67
24	10	59	15	11	17	18	17	10	39	8	1.47
25	44	155	28	31	39	39	65	50	67	25	4.19
26	72	291	73	50	66	62	125	101	122	31	7.52
27	74	342	52	55	70	76	100	100	129	44	8
28	67	309	67	66	86	54	120	103	141	33	8
29	60	280	61	53	76	68	109	85	135	36	7.56
30	34	195	56	58	55	46	91	63	86	23	5.4

Table 4. Muestras de plantas nativas en polígnos (INECOL).

9.7. **Códigos y Otros Materiales Digitales.** Se adjunta un repositorio de GitHub que contiene todos los códigos implementados para este trabajo con el fin de garantizar la transparencia de nuestros métodos.