

En la búsqueda de establecer una velocidad de reacción para la formación de nanomateriales, se pueden emplear métodos algebraicos, matriciales y de ecuaciones diferenciales. Aunque los métodos algebraicos son útiles para obtener nanopartículas monodispersas, no son aplicables en nuestro caso debido a que el *mastersizer* muestra una distribución polidispersa de las fases.

Los métodos matriciales (*soft- and hard-modelling*) destacan por su capacidad para realizar regresiones completas de la absorbancia en función de la longitud de onda y el tiempo, proporcionando parámetros que se ajustan fielmente a los datos experimentales. Estos métodos requieren el coeficiente de extinción como función de la absorbancia, un dato que proporcionan los equipos UV-vis. Sin embargo, el equipo UV-vis utilizado en este trabajo no proporciona dicho coeficiente, lo que impide realizar una regresión completa sobre la absorbancia. En su lugar, se realizará una regresión en el tiempo, asumiendo un coeficiente de extinción constante, el cual se estimará a partir de los resultados de reactivos en exceso. Al monitorear solo la plata sólida con el UV-vis, se planteará un mecanismo de reacción y se harán las suposiciones necesarias para expresar todo en términos de condiciones iniciales y la concentración del producto de interés.

Uno de los objetivos del proyecto es la búsqueda de parámetros cinéticos en ecuaciones diferenciales-algebraicas (DAE), las cuales requieren métodos numéricos específicos para garantizar la convergencia. Según lo observado en artículos relacionados sobre ajustes de curvas asociados a DAE, a pesar de que estos ajustes tienen coeficientes de correlación (r^2) próximos a 1, los parámetros cinéticos pueden variar significativamente entre sí. Esto indica que, aunque los parámetros obtenidos representan excelentes correlaciones dentro del dominio de tiempo estudiado, su uso para extrapolación conlleva riesgos y debe hacerse con cautela.

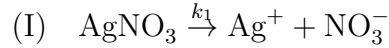
Por ello, aprovecharemos las condiciones de reacción para hacer suposiciones semejantes de modo que, por un lado, se deriven diferentes modelos con parámetros cinéticos estadísticamente iguales y, por otro, nos permitan la solución analítica de las ecuaciones diferenciales. Con la solución poder encontrar ecuaciones que relacionen los parámetros cinéticos. Finalmente, utilizaremos estos parámetros como valores semilla para el último método que involucra ED's, con el fin de hacer seguimiento tanto a reactivos como a productos.

Las reacciones fundamentales

La reacción de reducción de la Ag se debe a los grupos funcionales solubles en agua, sin embargo los polifenoles tienen una capacidad reductora superior debido a la estructura química que permite la donación de múltiples electrones y dado a que tienen múltiples grupos hidroxilos -OH su capacidad reductora es muy elevada en comparación a los otros compuestos. Por lo cual definimos a A como los reactivos de los polifenoles principales que están en mayor proporción y que reducen la plata hasta un estado de la plata hasta un estado de oxidación 0. Suponemos que $A.Ag^+$ es un complejo intermediario del cual se desprende Ag^0 . Todas las siguientes reacciones se llevan a cabo en presencia de NaOH y disueltas en H_2O :

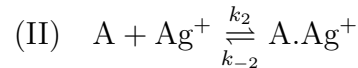
Mecanismo

1. Liberación de Ag^+ en solución:



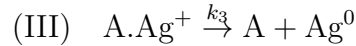
La disociación de AgNO_3 en agua produce iones de plata Ag^+ y iones nitrato NO_3^- . Esta reacción proporciona los iones de plata necesarios para la posterior reducción y formación de nanopartículas de plata. En presencia de NaOH , se crea un medio alcalino que favorece la estabilidad de los iones Ag^+ y prepara el entorno para la interacción con los polifenoles.

2. Creación del complejo intermedio de la interacción del Ag^+ y los polifenoles:



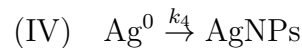
Los polifenoles presentes en la pepa de aguacate interactúan con los iones Ag^+ , formando un complejo intermedio $A.\text{Ag}^+$. Este paso es crucial porque los polifenoles actúan como agentes reductores naturales, donando electrones que facilitan la reducción de Ag^+ a Ag^0 . La formación del complejo $A.\text{Ag}^+$ es un paso transitorio que asegura que los iones Ag^+ estén adecuadamente preparados para la reducción.

3. Nucleación:



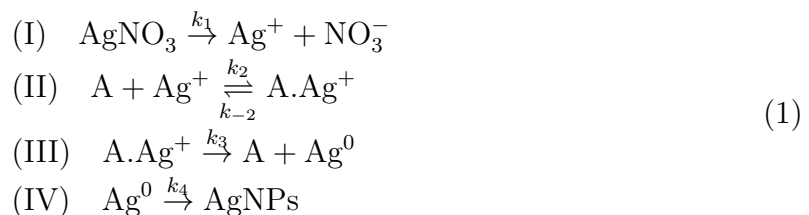
La ruptura del complejo $A.\text{Ag}^+$ resulta en la liberación de plata metálica Ag^0 y la regeneración de los polifenoles A . Este es el paso de nucleación, donde los átomos de Ag^0 comienzan a agregarse para formar núcleos. Este proceso es fundamental para la creación de nanopartículas de plata, ya que determina la cantidad inicial de núcleos que se formarán y, por lo tanto, influye en el tamaño y la distribución de las nanopartículas finales.

4. Formación de las nanopartículas:



Los átomos de plata metálica Ag^0 formados durante la nucleación se agrupan para formar nanopartículas de plata AgNPs . Este proceso de crecimiento de las nanopartículas es influenciado por factores como la concentración de reactivos, el tiempo de reacción y otros factores. La presencia de polifenoles no solo ayuda en la reducción inicial, sino que también actúa como un estabilizador que previene la aglomeración de las nanopartículas, asegurando una distribución de tamaño uniforme y mejorando las propiedades funcionales de las AgNPs .

por lo cual las reacciones principales son:



Para cada una de las reacciones supondremos que se cumple la ley de velocidad elemental, de este modo la velocidad de reacción III y IV vienen dadas respectivamente por:

$$r_3 = k_3[A.Ag^+]$$

$$r_4 = k_4[Ag^0]$$

La señal de plata sólida es detectada por el UV-Vis, tanto como Ag^0 y AgNPs por lo cual, definimos a P como $P := Ag^0 + AgNPs$, por lo tanto, la velocidad de formación de P es:

$$r_P = r_{Ag^0+AgNPs} = r_{Ag^0} + r_{AgNPs}$$

Pero, las velocidades de reacción del Ag^0 y la de AgNPs son:

$$r_{Ag^0} = r_3 - r_4$$

$$r_{AgNPs} = r_4$$

Por tanto:

$$r_P = r_3 - r_4 + r_4 = r_3 = k_3[A.Ag^+]$$

Es decir, la ecuación diferencial que determina la concentración de P es:

$$\frac{d[P]}{dt} = k_3[A.Ag^+] \quad (2)$$

Para resolver la ecuación (2) debemos hacer algunas suposiciones. Según el tipo de suposición, se derivarán varios métodos:

Suposición 1

La reacción 2 está en equilibrio, por tanto:

$$k_2[A][Ag^+] = k_{-2}[A.Ag^+]$$

Despejando $[A.Ag^+]$ y remplazando en la ecuación (2):

$$[A.Ag^+] = \frac{k_2[A][Ag^+]}{k_{-2}}$$

$$\frac{d[P]}{dt} = k_3[A.Ag^+] = k_3 \frac{k_2[A][Ag^+]}{k_{-2}}$$

Dado que la concentración del reactivo A está en exceso, se mantiene constante. Del mismo modo, los valores de k_2 , k_{-2} y k_3 también permanecen constantes. Por consiguiente, el producto de estas constantes se denota como k , y se obtiene la siguiente ecuación cinética:

$$\frac{d[P]}{dt} = k[Ag^+] \quad (3)$$

Método 1

Definiendo a Ag_i^+ como las moles iniciales de iones de plata que se convertirán en el producto, podemos expresar:

$$[\text{Ag}^+]_i = [\text{Ag}^+] + [P]$$

Por lo tanto, la ecuación (3) adquiere la forma:

$$\frac{d[P]}{dt} = k([\text{Ag}^+]_i - [P]) \quad (4)$$

Para expresar la ecuación diferencial de manera estándar como una ecuación diferencial lineal, reescribimos:

$$\frac{d[P]}{dt} + k[P] = k[\text{Ag}^+]_i$$

Usando un factor integrante, e^{kt} , obtenemos:

$$\frac{d}{dt} (e^{kt}[P]) = ke^{kt}[\text{Ag}^+]_i$$

Sean t el tiempo en minutos medido por el UV-Vis, $[P]$ la concentración asociada a la absorbancia en el tiempo t . Y t_d el tiempo muerto, que hace referencia al tiempo transcurrido hasta leer la señal de la muestra en el UV-Vis, Los límites de integración quedan de la siguiente forma:

$$\int_{[P]_{t_d}}^{e^{kt}[P]} d(e^{kt}[P]) = \int_0^t ke^{kt}[\text{Ag}^+]_i dt$$

Por lo cual, la integral queda de la forma:

$$e^{k(t)}[P] - [P]_{t_d} = [\text{Ag}^+]_i (e^{kt} - 1)$$

Así, la concentración del producto de interés viene dada por:

$$[P] = e^{-kt}[P]_{t_d} + [\text{Ag}^+]_i (1 - e^{-kt}) \quad (5)$$

Como la reacción comenzó t_d minutos antes de la primera lectura de señal, se espera que en $t = -t_d$ no haya producto P , es decir, $P = 0$. Por lo tanto, a partir de la ecuación (5), podemos determinar el tiempo muerto:

$$\begin{aligned} 0 &= e^{kt_d}[P]_{t_d} + [\text{Ag}^+]_i (1 - e^{kt_d}) \\ \Rightarrow t_d &= \frac{1}{k} \log \left(\frac{[\text{Ag}^+]_i}{[\text{Ag}^+]_i - [P]_{t_d}} \right) \end{aligned} \quad (6)$$

Si el tiempo $t \rightarrow \infty$, $[P] \rightarrow [\text{Ag}^+]_0$, lo cual sugiere que todo $[\text{Ag}^+]_i$ se convertirá en plata sólida.

Método 2

La concentración de Ag^+ decae de manera exponencial. Como $[\text{Ag}^+]$ decae de manera exponencial, existe un real positivo κ tal que:

$$[\text{Ag}^+] = [\text{Ag}^+]_0 e^{-\kappa t}$$

y sustituyendolo en la ecuación (3):

$$\frac{d[P]}{dt} = k[\text{Ag}^+] = k[\text{Ag}^+]_0 e^{-\kappa t} \quad (7)$$

Por cómo hemos definido t_d y t , los límites de integración quedan como:

$$\int_{[P]_{t_d}}^{[P]} d[P] = \int_0^t k[\text{Ag}^+]_0 e^{-\kappa t} dt$$

Por lo cual, la integral queda de la forma:

$$\begin{aligned} [P] - [P]_{t_d} &= k[\text{Ag}^+]_0 \left[\frac{e^{-\kappa t}}{-\kappa} \right]_0^t \\ [P] &= [P]_{t_d} + \frac{k[\text{Ag}^+]_0}{\kappa} (1 - e^{-\kappa t}) \end{aligned} \quad (8)$$

Como aún no se ha definido $[\text{Ag}^+]_0$, notemos que se puede escribir en términos de cosas ya definidas. Dado que las ecuaciones (7) y (4) surgen de las mismas suposiciones y, por tanto, representan el mismo fenómeno, al igualarlas se tiene:

$$\begin{aligned} \frac{d[P]}{dt} &= k[\text{Ag}^+]_0 e^{-\kappa t} = k([\text{Ag}^+]_i - [P]) \\ &\rightarrow [\text{Ag}^+]_0 e^{-\kappa t} = [\text{Ag}^+]_i - [P] \end{aligned}$$

Se cumple para todo tiempo, en particular para t en el inicio de la reacción, por lo que $t = -t_d$, y en este tiempo $[P] = 0$:

$$\begin{aligned} [\text{Ag}^+]_0 e^{\kappa t_d} &= [\text{Ag}^+]_i \\ [\text{Ag}^+]_0 &= [\text{Ag}^+]_i e^{-\kappa t_d} \end{aligned} \quad (9)$$

Así, al reemplazar (9) en (8), la concentración viene dada por:

$$[P] = [P]_{t_d} + \frac{k e^{-\kappa t_d}}{\kappa} [\text{Ag}^+]_i (1 - e^{-\kappa t}) \quad (10)$$

para obtener el t_d , se evalúa $t = -t_d$ en la ecuación (10):

$$\begin{aligned} 0 &= [P]_{t_d} + \frac{k e^{-\kappa t_d}}{\kappa} [\text{Ag}^+]_i (1 - e^{\kappa t_d}) \\ t_d &= \frac{1}{\kappa} \log \left(1 + \frac{[P]_{t_d}}{\theta} \right) \quad \text{con} \quad \theta = \frac{k e^{-\kappa t_d}}{\kappa} [\text{Ag}^+]_i \end{aligned} \quad (11)$$

Y si en (10) el tiempo $t \rightarrow \infty$, $[P] \rightarrow [P]_{t_d} + \theta$, es la concentración de plata sólida.

Suposición 2

La concentración del complejo intermediario $A.Ag^+$ es constante. Por lo tanto, del mecanismo propuesto se debe cumplir que:

$$k_2[A][Ag^+] - k_{-2}[A.Ag^+] - k_3[A.Ag^+] = 0 \quad (12)$$

Dado que el único compuesto que se está monitoreando con el UV-Vis es la plata sólida P , se supone que la concentración inicial de los polifenoles es la suma de la concentración de los polifenoles en algún tiempo más la concentración del complejo intermediario, es decir:

$$[A]_0 = [A] + [A.Ag^+]$$

Despejando $[A]$ y reemplazando en la ecuación (12), se llega a:

$$\begin{aligned} k_2[A][Ag^+] - k_{-2}[A.Ag^+] - k_3[A.Ag^+] &= k_2([A]_0 - [A.Ag^+])[Ag^+] - (k_{-2} + k_3)[A.Ag^+] \\ &= k_2[A]_0[Ag^+] - k_2[A.Ag^+][Ag^+] - (k_{-2} + k_3)[A.Ag^+] = 0 \\ &\Leftrightarrow k_2[A]_0[Ag^+] = k_2[A.Ag^+][Ag^+] + (k_{-2} + k_3)[A.Ag^+] \\ &\Leftrightarrow k_2[A]_0[Ag^+] = [A.Ag^+](k_2[Ag^+] + k_{-2} + k_3) \\ &\Leftrightarrow [A.Ag^+] = \frac{k_2[A]_0[Ag^+]}{k_2[Ag^+] + k_{-2} + k_3} \end{aligned}$$

Luego, al reemplazarlo en la ecuación (2):

$$\frac{d[P]}{dt} = k_3[A.Ag^+] = k_3 \frac{k_2[A]_0[Ag^+]}{k_2[Ag^+] + k_{-2} + k_3} = \frac{k_3[A]_0[Ag^+]}{[Ag^+] + \frac{k_{-2} + k_3}{k_2}} \quad (13)$$

Método 3

La concentración de Ag^+ decae de manera exponencial. Como $[Ag^+]$ decae de manera exponencial, existe un real positivo κ tal que:

$$[Ag^+] = [Ag^+]_0 e^{-\kappa t}$$

por lo cual la ecuación (13) queda de la forma:

$$\frac{d[P]}{dt} = \frac{k_3[A]_0[Ag^+]}{[Ag^+] + \frac{k_{-2} + k_3}{k_2}} = \frac{k_3[A]_0[Ag^+]_0 e^{-\kappa t}}{[Ag^+]_0 e^{-\kappa t} + \frac{k_{-2} + k_3}{k_2}} \quad (14)$$

Por cómo hemos definido t_d y t , los límites de integración quedan como:

$$\int_{[P]_{t_d}}^{[P]} d[P] = \int_0^t \frac{k_3[A]_0[Ag^+]_0 e^{-\kappa t}}{[Ag^+]_0 e^{-\kappa t} + \frac{k_{-2} + k_3}{k_2}} dt$$

Esta integral se resuelve por sustitución. Tomando u como el denominador, se llega a:

$$[P] - [P]_{t_d} = \frac{k_3[A]_0}{-\kappa} \left[\log \left| [Ag^+]_0 e^{-\kappa t} + \frac{k_{-2} + k_3}{k_2} \right| - \log \left| [Ag^+]_0 + \frac{k_{-2} + k_3}{k_2} \right| \right]$$

por propiedades de logaritmos:

$$[P] = [P]_{t_d} + \frac{k_3[A]_0}{\kappa} \log \left| \frac{[\text{Ag}^+]_0 + \frac{k_{-2}+k_3}{k_2}}{[\text{Ag}^+]_0 e^{-\kappa t} + \frac{k_{-2}+k_3}{k_2}} \right|$$

Sin pérdida de generalidad, podemos reemplazar $[\text{Ag}^+]_0$ de la ecuación (9) en la anterior ecuación, por lo que la concentración del producto es:

$$[P] = [P]_{t_d} + \frac{k_3[A]_0}{\kappa} \log \left| \frac{[\text{Ag}^+]_i e^{-\kappa t_d} + \frac{k_{-2}+k_3}{k_2}}{[\text{Ag}^+]_i e^{-\kappa(t+t_d)} + \frac{k_{-2}+k_3}{k_2}} \right| \quad (15)$$

Evalutando $t = -t_d$ y $[P] = 0$ en (15) para obtener el tiempo muerto t_d :

$$0 = [P]_{t_d} + \frac{k_3[A]_0}{\kappa} \log \left| \frac{[\text{Ag}^+]_i e^{-\kappa t_d} + \frac{k_{-2}+k_3}{k_2}}{[\text{Ag}^+]_i + \frac{k_{-2}+k_3}{k_2}} \right|$$

Pa simplificar el algebra tomemos $\alpha_\kappa := \frac{k_3[A]_0}{\kappa}$ y $\beta_i := \frac{k_{-2}+k_3}{k_2[\text{Ag}^+]_i e^{-\kappa t_d}}$, llegando a:

$$0 = [P]_{t_d} + \alpha_i \log \left| \frac{1 + \beta_i}{e^{\kappa t_d} + \beta_i} \right|$$

y despejando t_d :

$$t_d = \frac{1}{\kappa} \log \left| (1 + \beta_i) e^{\frac{[P]_{t_d}}{\alpha_\kappa}} - \beta_i \right| \quad (16)$$

Y si en (15) el tiempo $t \rightarrow \infty$, $[P]$ tiende a:

$$[P]_{t \rightarrow \infty} = [P]_{t_d} + \alpha_\kappa \log \left| \frac{1 + \beta_i}{\beta_i} \right|$$

Método 4

Por como definimos a Ag_i^+ , se sigue que:

$$[\text{Ag}^+]_i = [\text{Ag}^+] + [P]$$

reemplazando a $[\text{Ag}^+]$ en la ecuación (13) se llega a:

$$\frac{d[P]}{dt} = \frac{k_3[A]_0[\text{Ag}^+]}{[\text{Ag}^+] + \frac{k_{-2}+k_3}{k_2}} = \frac{k_3[A]_0 ([\text{Ag}^+]_i - [P])}{([\text{Ag}^+]_i - [P]) + \frac{k_{-2}+k_3}{k_2}} \quad (17)$$

es una ecuación diferencial autónoma, por lo tanto, es una ecuación diferencial de variables separables y al tener en cuenta el tiempo muerto t_d se llega a:

$$\begin{aligned} \int_0^t dt &= \int_{[P]_{t_d}}^{[P]} \frac{([\text{Ag}^+]_i - [P]) + \frac{k_{-2}+k_3}{k_2}}{k_3[A]_0 ([\text{Ag}^+]_i - [P])} d[P] \\ t &= \int_{[P]_{t_d}}^{[P]} \frac{1}{k_3[A]_0} d[P] + \int_{[P]_{t_d}}^{[P]} \frac{\frac{k_{-2}+k_3}{k_2}}{k_3[A]_0 ([\text{Ag}^+]_i - [P])} d[P] \end{aligned}$$

$$t = \frac{[P] - [P]_{t_d}}{k_3[A]_0} - \frac{\frac{k_{-2}+k_3}{k_2}}{k_3[A]_0} \log \left| \frac{[P] - [Ag^+]_i}{[P]_{t_d} - [Ag^+]_i} \right| \quad (18)$$

La ecuación (18) es una solución implícita. Más específicamente, (18) es la función inversa de $P(t)$. Evaluando el punto $(t = -t_d, [P] = 0)$, se obtiene de manera directa el tiempo muerto t_d :

$$t_d = \frac{[P]_{t_d}}{k_3[A]_0} + \frac{\frac{k_{-2}+k_3}{k_2}}{k_3[A]_0} \log \left| \frac{[Ag^+]_i}{[Ag^+]_i - [P]_{t_d}} \right| \quad (19)$$

De acuerdo con la teoría de ecuaciones diferenciales autónomas, se puede demostrar que cuando el tiempo $t \rightarrow \infty$, $[P] \rightarrow [Ag^+]_i$, como se mostrará más adelante.

Dado que necesitamos un modelo para $[P]$ de manera explícita, es necesario despejarla de la ecuación (18) usando la función W de Lambert. Sin embargo, no procederemos a despejarla directamente. En su lugar, transformaremos la ecuación diferencial correspondiente a la solución implícita en la ecuación diferencial característica de la función W de Lambert.

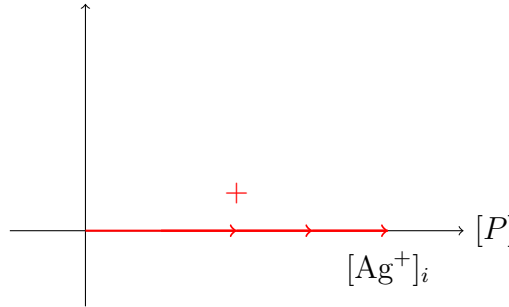
La ecuación diferencial (18) correspondiente a la solución implícita:

$$\frac{d[P]}{dt} = \frac{k_3[A]_0 ([Ag^+]_i - [P])}{([Ag^+]_i - [P]) + \frac{k_{-2}+k_3}{k_2}}$$

definiendo $\alpha := k_3[A]_0$ y $\beta := \frac{k_{-2}+k_3}{k_2}$ se llega a:

$$\frac{d[P]}{dt} = \frac{\alpha([Ag^+]_i - [P])}{[Ag^+]_i - [P] + \beta} \quad (20)$$

La ecuación (20) es una ecuación diferencial autónoma, donde el punto singular o de equilibrio se da en $\alpha([Ag^+]_i - [P]) = 0 \Leftrightarrow [Ag^+]_i - [P] = 0 \Leftrightarrow [P] = [Ag^+]_i$. Como $\forall t \in \mathbb{R}$, $[P] < [Ag^+]_i$, $\alpha > 0$, $\beta > 0$, entonces $\frac{d[P]}{dt} > 0$, y el diagrama de fase es:



por tanto, si $t \rightarrow \infty$, $[P] \rightarrow [Ag^+]_i$, como se esperaba.

Ahora exploremos las propiedades de W de Lambert y la ecuación diferencial característica. W de Lambert se define como la función inversa de $y = ze^z$, es decir que dada la ecuación anterior y un valor fijo para y , $W(y)$ devuelve el valor de z que satisface la ecuación. Note que:

$$\text{Si } ze^z = y \Leftrightarrow W(ze^z) = W(y) \Leftrightarrow z = W(y)$$

Para hallar $\frac{dW}{dy}$, partimos de $ze^z = y$, derivando implícitamente respecto a y :

$$\frac{dz}{dy} \cdot e^z + z \cdot e^z \frac{dz}{dy} = 1$$

$$\Leftrightarrow (1+z)e^z \frac{dz}{dy} = 1$$

Como $ze^z = y \Leftrightarrow e^z = \frac{y}{z}$, la ecuación diferencial toma la forma de:

$$(1+z) \frac{y}{z} \frac{dz}{dy} = 1$$

Así la ecuación diferencial característica de W -Lambert es:

$$\frac{1+z}{z} dz = \frac{dy}{y} \quad (21)$$

Notemos que la ecuación (21) es una E.D. lineal homogénea en y , pues:

$$\frac{1+z}{z} dz = \frac{dy}{y} \Leftrightarrow \frac{dy}{dz} = \frac{1+z}{z} y \Leftrightarrow \frac{dy}{dz} - \frac{1+z}{z} y = 0$$

Por lo cual si y_1 es una solución, entonces $c_1 y_1$ también, donde c_1 es una constante que se determina con las condiciones iniciales del P.V.I. Ya exploradas las propiedades de Lambert- W necesarias para convertir la ecuación (20) en la ecuación (21).

Reescribamos la ecuación (20):

$$\frac{d[P]}{dt} = \frac{\alpha ([Ag^+]_i - [P])}{[Ag^+]_i - [P] + \beta} = \alpha \frac{\frac{[Ag^+]_i - [P]}{\beta}}{\frac{[Ag^+]_i - [P]}{\beta} + 1}$$

Tomando a $z = \frac{[Ag^+]_i - [P]}{\beta} \Rightarrow dz = -\frac{d[P]}{\beta}$ y reemplazándolas en la E.D.:

$$-\beta \frac{dz}{dt} = \frac{d[P]}{dt} = \alpha \frac{\frac{[Ag^+]_i - [P]}{\beta}}{\frac{[Ag^+]_i - [P]}{\beta} + 1} = \alpha \frac{z}{z + 1}$$

Separando las variables:

$$\frac{z+1}{z} dz = -\frac{\alpha}{\beta} dt$$

Solo falta que el segundo miembro de la E.D. sea el mismo de la ecuación (21), por lo cual:

$$-\frac{\alpha}{\beta} dt = \frac{dy}{y} \Leftrightarrow -\frac{\alpha}{\beta} t = \ln y + c$$

Por lo tanto, la sustitución faltante $y = e^{-\frac{\alpha}{\beta} t}$ termina de convertir la ecuación (20) en la ecuación (21). Hemos visto que $z = W(y)$ es una solución de la ecuación (21), por lo que $z = W(c_1 y)$ es la solución que considera la condición inicial. Así, la solución de la ecuación (20) queda de la forma:

$$z = W(c_1 y)$$

Pero, dado que:

$$z = \frac{[Ag^+]_i - [P]}{\beta}$$

y

$$y = e^{-\frac{\alpha}{\beta} t}$$

se tiene:

$$\frac{[\text{Ag}^+]_i - [\text{P}]}{\beta} = W(c_1 e^{-\frac{\alpha}{\beta}t})$$

Como t es el tiempo en minutos medido por el UV-Vis, así $[\text{P}]$ es la concentración asociada a la absorbancia en ese instante. Por lo tanto, en $t = 0$, $[\text{P}] = [\text{P}]_{t_d}$, así:

$$\begin{aligned} \frac{[\text{Ag}^+]_i - [\text{P}]_{t_d}}{\beta} &= W(c_1) \\ \Leftrightarrow c_1 &= W^{-1}\left(\frac{[\text{Ag}^+]_i - [\text{P}]_{t_d}}{\beta}\right) = \frac{[\text{Ag}^+]_i - [\text{P}]_{t_d}}{\beta} e^{\frac{[\text{Ag}^+]_i - [\text{P}]_{t_d}}{\beta}} \end{aligned}$$

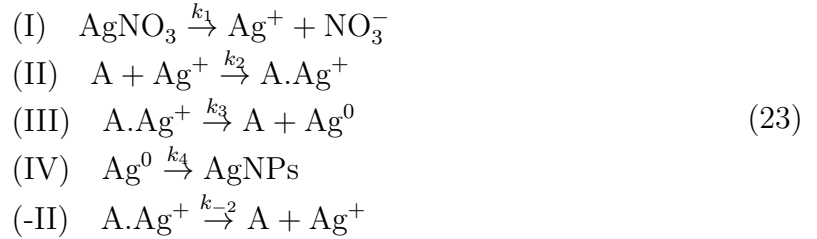
Por tanto, la concentración del producto de interés de manera explícita viene dada por:

$$[\text{P}] = [\text{Ag}^+]_i - \beta W\left(c_1 e^{-\frac{\alpha}{\beta}t}\right) \quad (22)$$

$$\text{siendo } \alpha = k_3[\text{A}]_0, \beta = \frac{k_{-2}+k_3}{k_2} \text{ y } c_1 = \frac{[\text{Ag}^+]_i - [\text{P}]_{t_d}}{\beta} e^{\frac{[\text{Ag}^+]_i - [\text{P}]_{t_d}}{\beta}}.$$

Método 5

Considerar todas las velocidades de reacción. Hemos visto que para los métodos anteriores, las ecuaciones diferenciales tenían solución analítica, pero al considerar todas las velocidades de reacción, la complejidad aumenta. Aunque esto nos permite obtener las concentraciones de todas las especies, separando los equilibrios químicos de las reacciones se llega a:



De manera ordenada, representemos en una tabla las especies y_j y los coeficientes que estos tienen en cada reacción, tomando como negativo cuando reaccionan y positivo cuando se producen. Se tiene:

Espece \ Reacción	I	II	III	IV	-II
$y_1 = \text{AgNO}_3$	-1	0	0	0	0
$y_2 = \text{Ag}^+$	1	-1	0	0	1
$y_3 = \text{NO}_3^-$	1	0	0	0	0
$y_4 = \text{A}$	0	-1	1	0	1
$y_5 = \text{A} \cdot \text{Ag}^+$	0	1	-1	0	-1
$y_6 = \text{Ag}^0$	0	0	1	-1	0
$y_7 = \text{AgNPs}$	0	0	0	1	0

Table 1: Arreglo ordenado de los coeficientes

Así definimos a \mathbf{r}_{ij} como la matriz de coeficientes de la especie j en la reacción i . Si para cada una de las reacciones se cumple la ley de velocidad elemental, las velocidades de reacción son:

$$\begin{aligned} r_1 &= k_1[\text{AgNO}_3] \\ r_2 &= k_2[\text{A}][\text{Ag}^+] \\ r_3 &= k_3[\text{A} \cdot \text{Ag}^+] \\ r_4 &= k_4[\text{Ag}^0] \\ r_{-2} &= k_{-2}[\text{A} \cdot \text{Ag}^+] \end{aligned}$$

Así definimos el vector columna $\mathbf{r}_i = (r_i)$. Y la matriz de coeficientes es:

$$\mathbf{r}_{ij} = \begin{pmatrix} -1 & 0 & 0 & 0 & 0 \\ 1 & -1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 1 \\ 0 & 1 & -1 & 0 & -1 \\ 0 & 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$

Si \mathbf{y} es el vector de concentración, entonces el sistema de ecuaciones diferenciales a resolver es:

$$\frac{d\mathbf{y}}{dt} = \mathbf{r}_{ij} \cdot \mathbf{r}_i \quad \text{ligado a} \quad \mathbf{y}(0) = \mathbf{y}_0 \quad (24)$$

Resumen de los modelos obtenidos en cada método:

Método	Modelo
1	$[P] = e^{-kt}[P]_{t_d} + [\text{Ag}^+]_i (1 - e^{-kt})$
2	$[P] = [P]_{t_d} + \frac{ke^{-\kappa t_d}}{\kappa} [\text{Ag}^+]_i (1 - e^{-\kappa t})$
3	$[P] = [P]_{t_d} + \frac{k_3[A]_0}{\kappa} \log \left \frac{[\text{Ag}^+]_i e^{-\kappa t_d} + \frac{k_{-2}+k_3}{k_2}}{[\text{Ag}^+]_i e^{-\kappa(t+t_d)} + \frac{k_{-2}+k_3}{k_2}} \right $
4	$[P] = [\text{Ag}^+]_i - \beta W \left(c_1 e^{-\frac{\alpha}{\beta} t} \right)$
5	$\begin{pmatrix} \frac{d[\text{AgNO}_3]}{dt} \\ \frac{d[\text{Ag}^+]}{dt} \\ \frac{d[\text{NO}_3^-]}{dt} \\ \frac{d[\text{A}]}{dt} \\ \frac{d[\text{A} \cdot \text{Ag}^+]}{dt} \\ \frac{d[\text{Ag}^0]}{dt} \\ \frac{d[\text{AgNPs}]}{dt} \end{pmatrix} = \begin{pmatrix} -1 & 0 & 0 & 0 & 0 \\ 1 & -1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 1 \\ 0 & 1 & -1 & 0 & -1 \\ 0 & 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} r_1 \\ r_2 \\ r_3 \\ r_4 \\ r_{-2} \end{pmatrix}$

Table 2: Tabla de los modelos

siendo $\alpha = k_3[\text{A}]_0$, $\beta = \frac{k_{-2}+k_3}{k_2}$ y $c_1 = \frac{[\text{Ag}^+]_i - [\text{P}]_{t_d}}{\beta} e^{\frac{[\text{Ag}^+]_i - [\text{P}]_{t_d}}{\beta}}$.

Implementación del algoritmo

En esta sección, mostraremos cómo y cuáles métodos numéricos usaremos para realizar el mejor ajuste utilizando mínimos cuadrados dados los datos de absorbancia sobre los cinco modelos obtenidos. Uno de los problemas que surge de la utilización de métodos numéricos radica en que los parámetros óptimos dependen de las condiciones iniciales y del número de iteraciones. Por tanto, probaremos que para los métodos 1 y 2, los parámetros del mejor ajuste existen y son únicos, es decir, que sin importar los valores iniciales y el método numérico utilizado, y teniendo suficientes iteraciones, estos deben llevar a los mismos parámetros óptimos. Mientras que para los métodos 3 y 4, solo podremos garantizar la existencia y unicidad al restringir uno de los parámetros; en la creación del algoritmo indicaremos cuál para cada método.

Debido a lo mencionado anteriormente, usaremos la herramienta de ajuste de curvas para estos cuatro métodos y, por conveniencia, para el modelo 5 realizaremos el ajuste en Python. Antes de comenzar con la implementación de los códigos, es necesario definir y encontrar cuáles son las cantidades conocidas. Recordemos que el UV-Vis nos reporta las absorbancias en función del tiempo de la plata sólida. Además, los modelos obtenidos son expresiones de concentración como función del tiempo. Dado que la concentración de [P] es lo suficientemente diluida como para aplicar la ley de Beer-Lambert, podemos obtener una buena estimación de la relación entre la absorbancia y la concentración. Así, llegamos a:

$$A = \epsilon l[P] + E \quad (25)$$

Siendo E el error o incertidumbre y donde l es la longitud de la trayectoria de 1 cm, mientras que ϵ se aproximará por medio de la ecuación (25) con los datos tomados de la primera parte del proyecto **Cita requerida**. Se establece que a una concentración de 0.001923 Molar, se tiene una absorbancia total de 3.75 u.a y que la del blanco es de 1.026 u.a, por tanto, $\epsilon \approx 1416.53 \text{ M cm}^{-1}$, el cual es un dato muy parecido al encontrado en la primera parte del proyecto.

Uno de las suposiciones fuertes de los modelos es que la concentración de los polifenoles está en exceso. Para probar que dicha suposición es válida, vamos a encontrar la concentración de los polifenoles. Como los polifenoles se refieren a una mezcla de sustancias, vamos a estimar una aproximación para $[A]_0$ haciendo una regla de mezcla simple, teniendo en cuenta solo las sustancias que son agentes reductores y que están en una proporción significativa.

Según la metodología explicada, se toman 300.73 g de pepa seca y se disuelven en 0.8 L de solución. El porcentaje de polifenoles está entre 2% y 4%, las proteínas entre 5% y 10%, los taninos entre 1% y 3%, los ácidos orgánicos entre 1% y 1.5% y los flavonoides entre el 1% y el 1.5%. Dado que todos estos compuestos actúan como agentes reductores, asumimos que la cantidad de masa reactiva es de 13.5%. Por lo tanto, la cantidad de masa reactiva es de 40.598 g.

Para convertir esto a moles, necesitamos la masa molecular de los polifenoles. Sin embargo, como son cuatro los compuestos que representan el mayor porcentaje de masa, haremos una estimación de la masa molecular mediante una regla de mezclado simple. Los cuatro compuestos son:

- Ácido gálico: concentración entre 0.002 y 0.005, peso molecular de 170.12 g/mol.
- Catequinas: concentración entre 0.001 y 0.003, peso molecular de 290.26 g/mol.

- Proantocianidinas: concentración entre 0.005 y 0.01, peso molecular de 516.127 g/mol.
- Epicatequinas: concentración entre 0.001 y 0.002, peso molecular de 290.271 g/mol.

La masa molecular de la mezcla varía entre 373.16 y 389.04 g/mol. Tomaremos el promedio, 381.10 g/mol, como valor característico. La solución se diluye en 0.8 L, se mezcla bien, y se toman 2 mL, que se diluyen en un beaker hasta completar 250 mL. Así, la concentración inicial de $[A]_0$ se calcula como:

$$[A]_0 = \frac{\text{Moles de } A_0}{\text{Volumen de la solución}} = \frac{40.598 \text{ g}}{0.8 \text{ L}} \cdot \frac{1 \text{ mol } A}{381.10 \text{ g}} \cdot \frac{2 \text{ mL}}{250 \text{ mL}} = 0.00106 \text{ M}$$

De un recipiente que contiene nitrato de plata con una concentración de 0.01 M, tomamos 5 mL y lo diluimos en un beaker hasta alcanzar un volumen de 26 mL. Por lo tanto, la concentración inicial de $[\text{AgNO}_3]_0$ en la solución es:

$$[\text{AgNO}_3]_0 = 0.01 \text{ M} \cdot \frac{5 \text{ mL}}{26 \text{ mL}} = 0.001923 \text{ M}$$

Las constantes de velocidad k_1 , k_2 , k_3 , k_4 y k_{-2} son todas positivas, al igual que κ .

Recordemos que t_d es el tiempo muerto, que hace referencia al tiempo transcurrido hasta la lectura de la señal de la muestra en el UV-Vis, t es el tiempo en minutos medido por el UV-Vis, y $[P]$ es la concentración asociada a la absorbancia en el tiempo t .

Dado que la reacción (I) es lo suficientemente rápida, la cantidad que habíamos definido como $[\text{Ag}^+]_i$ satisface:

$$0 \leq [\text{Ag}^+]_i \leq [\text{AgNO}_3]_0 = 0.001923 \text{ M}$$

Ya encontradas las constantes, se procede a definir los códigos. Para los cuatro primeros métodos, se usará la aplicación Curve Fitting.

Método 1 y 2

Para la implementación del mejor ajuste por mínimos cuadrados, veamos que para los métodos 1 y 2 se puede establecer la existencia y la unicidad de tales parámetros.

Escribiendo las ecuaciones (5) y (8) de manera genérica:

$$p(t) = x + y(1 - e^{-Kt}) \quad (26)$$

Sean n el número de datos de p_i medidos en el tiempo t_i . Se requiere encontrar x , y y K tales que:

$$\begin{aligned} p_1 &= x + y(1 - e^{-Kt_1}) \\ p_2 &= x + y(1 - e^{-Kt_2}) \\ &\vdots \\ p_n &= x + y(1 - e^{-Kt_n}) \end{aligned}$$

Es claro que es un problema no lineal; más aún, estos parámetros no tienen por qué satisfacer las igualdades.

Tomando K fijo pero arbitrario, el problema se linealiza y admite una factorización tipo:

$$\begin{pmatrix} p_1 \\ p_2 \\ \vdots \\ p_n \end{pmatrix} = \begin{pmatrix} 1 & 1 - e^{-Kt_1} \\ 1 & 1 - e^{-Kt_2} \\ \vdots & \vdots \\ 1 & 1 - e^{-Kt_n} \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}$$

Sean los vectores $\mathbf{a} = \begin{pmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{pmatrix}$, $\mathbf{b} = \begin{pmatrix} 1 - e^{-Kt_1} \\ 1 - e^{-Kt_2} \\ \vdots \\ 1 - e^{-Kt_n} \end{pmatrix}$ y $\mathbf{p} = \begin{pmatrix} p_1 \\ p_2 \\ \vdots \\ p_n \end{pmatrix}$. Entonces, definiendo a \mathbf{C} como $\mathbf{C} = (\mathbf{a} \ \mathbf{b})$, tenemos:

$$\mathbf{p} = \mathbf{C} \begin{pmatrix} x \\ y \end{pmatrix}$$

Por mínimos cuadrados se sabe que el mejor ajuste de la ecuación anterior debe satisfacer:

$$\mathbf{C}^T \mathbf{C} \begin{bmatrix} x \\ y \end{bmatrix} = \mathbf{C}^T \mathbf{p}$$

Es decir, x y y deben cumplir:

$$\begin{bmatrix} \mathbf{a} \cdot \mathbf{a} & \mathbf{a} \cdot \mathbf{b} \\ \mathbf{b} \cdot \mathbf{a} & \mathbf{b} \cdot \mathbf{b} \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \mathbf{a} \cdot \mathbf{p} \\ \mathbf{b} \cdot \mathbf{p} \end{bmatrix} \quad (27)$$

Como existen p_i no nulos, luego $\mathbf{a} \cdot \mathbf{p}$ es no nulo por tanto el sistema resultante es no homogéneo. Luego, (x, y) es solución única si y sólo si el determinante de $\mathbf{C}^T \mathbf{C}$ es no nulo.

$$\det \mathbf{C}^T \mathbf{C} = (\mathbf{a} \cdot \mathbf{a})(\mathbf{b} \cdot \mathbf{b}) - (\mathbf{b} \cdot \mathbf{a})(\mathbf{a} \cdot \mathbf{b})$$

$$\det \mathbf{C}^T \mathbf{C} = \mathbf{a}^2 \mathbf{b}^2 - (\mathbf{a} \cdot \mathbf{b})^2$$

La desigualdad de Cauchy-Schwarz establece que para todos los vectores \mathbf{a} y \mathbf{b} en un espacio con producto interno, se cumple la siguiente desigualdad:

$$\|\mathbf{a}\|^2 \|\mathbf{b}\|^2 \geq (\mathbf{a} \cdot \mathbf{b})^2$$

La igualdad se alcanza cuando \mathbf{a} y \mathbf{b} son L.D. Pero por la forma en que los definimos \mathbf{a} y \mathbf{b} son linealmente independientes, por lo que:

$$\det \mathbf{C}^T \mathbf{C} = \mathbf{a}^2 \mathbf{b}^2 - (\mathbf{a} \cdot \mathbf{b})^2 > 0$$

por tanto x , y existen y son únicos.

Ahora que sabemos que dado un K , x y y son únicos, es necesario probar que dicho K también existe y es único. Como estamos minimizando la suma de cuadrados, es decir:

$$\min S = \sum_{i=1}^n (p(t_i) - p_i)^2$$

Variando x , y , K y en caso de existir tal mínimo, la terna (x, y, K) debe volver nulas las derivadas parciales de S . En particular, la derivada parcial respecto a x :

$$\begin{aligned}\frac{\partial S}{\partial x} &= 2 \sum_{i=1}^n (p(t_i) - p_i) \cdot \frac{\partial p(t_i)}{\partial x} \\ &= 2 \sum_{i=1}^n (x + y(1 - e^{-Kt_i}) - p_i)\end{aligned}$$

Sea K fijo pero arbitrario, existen x y y que son únicos y así consideremos la siguiente función:

$$f(K) = \frac{\partial S}{\partial x}$$

f es continua en todo \mathbb{R} . Además, $t_i \geq 0$ veamos que f tiene cambio de signo $\Rightarrow f(-\infty) < 0$ y $f(\infty) > 0$. Como f es continua y presenta cambios de signo se tiene que por teorema de Bolzano f tiene una raíz, es decir, existe K tal que $f(K) = 0$. Además $f(K)$ es creciente, ya que:

$$\frac{\partial f}{\partial K} = 2y \sum_{i=1}^n t_i e^{-Kt_i} > 0$$

Como f es creciente y tiene una raíz, tal raíz K existe y es única. De esta forma hemos probado que los parámetros que realizan el mejor ajuste deben ser los mismos sin importar el método numérico que se use, eso sí, no hemos hablado de cuantas iteraciones se necesiten para encontrarlos.

La ecuación (5) representa la concentración del producto del método 1, y al reemplazarla en la ecuación (25), el modelo queda de la forma:

$$A = \epsilon l (e^{-kt}[P]_{t_d} + [\text{Ag}^+]_i (1 - e^{-kt})) + E \quad (28)$$

Dado que \mathbf{A} y \mathbf{t} representan los datos experimentales, y respetando la definición de $[P]$ y t dadas, es decir, que \mathbf{t} comienza a contar justo cuando se realiza la primera lectura de \mathbf{A} , que es la absorbancia reportada por el UV-Vis, y considerando que la ecuación (28) satisface las condiciones de existencia y unicidad para los parámetros k , P_{t_d} y $[\text{Ag}^+]_i$, estos son únicos. Por lo tanto, para asegurar que el número de iteraciones sean pocas, se agrega la condición de que cada una de estas variables sea no negativa.

Además, la regresión se realizará solo sobre los primeros n datos responsables de la nucleación, es decir, las absorbancias medidas para tiempos menores o iguales a 60 minutos, con la intención de comprobar si los modelos son predictivos. Posteriormente, se procederá a comparar la curva con todos los datos con el objetivo de validar la precisión predictiva. A continuación, se presenta el código:

Código MATLAB para Método 1

```
1 clear all
2 format long
3 % Datos experimentales
4 t = [0; 5; 10; 15; 20; 25; 30; 35; 50; 60; 90; 100; 140; 150; 160;
      180];
5 A = [0.12; 0.354; 0.624; 0.528; 0.645; 1.0105; 1.143; 1.2175; 1.506;
      1.4145; 1.806; 1.638; 1.953; 1.892; 1.913; 1.874];
6
```

```

7 % Llama a la funci n createFit para ajustar el modelo a los primeros "
  n" puntos
8 [fitresult, gof] = createFit(t, A);
9
10 % Muestra los par metros ajustados y la bondad del ajuste
11 disp('Par metros Ajustados:');
12 disp(fitresult);
13 disp('Bondad del Ajuste:');
14 disp(gof);
15
16 % Funci n para crear el ajuste
17
18 function [fitresult, gof] = createFit(t, A)
19     %CREATEFIT(T,A)
20     % Crea un ajuste.
21     % Datos para el ajuste 'Metodo1':
22     %     X de entrada : t
23     %     Y de salida  : A
24     % Salida:
25     %     fitresult : objeto de ajuste que representa el ajuste.
26     %     gof       : estructura con informaci n de bondad del ajuste.
27     %
28     % Ver tambi n FIT, CFIT, SFIT.
29
30     %% Prepara los datos para el ajuste utilizando solo los primeros "n
      " puntos
31     n = 10; %% n mero de datos utilizados "Datos de la nucleaci n"
32     xData = t(1:n);
33     yData = A(1:n);
34
35     %% Configura el tipo de ajuste y las opciones
36     ft = fittype('1416.536 * (p_td * exp(-k * t) + Ag * (1 - exp(-k * t
      )))', 'independent', 't', 'dependent', 'A');
37     opts = fitoptions('Method', 'NonlinearLeastSquares');
38     opts.Display = 'Off';
39     opts.Lower = [0, 0, 0]; % L mites inferiores para Ag, k, p_td
40     opts.Upper = [inf, inf, inf]; % L mites superiores para Ag, k,
      p_td
41     opts.StartPoint = [0.0018, 1, 0.00018]; % Puntos de inicio para Ag
      , k, p_td
42
43     %% Ajusta el modelo a los datos
44     [fitresult, gof] = fit(xData, yData, ft, opts);
45
46     %% Grafica el ajuste con todos los datos y los primeros n puntos en
      otro color
47     figure('Name', 'Metodo1');
48     plot(fitresult, t, A, 'r*');
49     hold on;
50     plot(t(1:n), A(1:n), 'b.', 'MarkerFaceColor', 'r'); % Primeros n
      puntos en rojo
51     hold off;
52     legend('Datos experimentales', 'Ajuste Modelo 1', 'Datos para la
      regresi n', 'Location', 'SouthEast', 'Interpreter', 'none');
53     xlabel('t [min]', 'Interpreter', 'none');
54     ylabel('A [u.a]', 'Interpreter', 'none');
55     grid on;
56 end

```


De manera análoga a lo que hicimos con el modelo 1, combinando las ecuaciones (10) y (25), se llega al modelo 2:

$$A = \epsilon l \left([P]_{t_d} + \frac{ke^{-\kappa t_d}}{\kappa} [\text{Ag}^+]_i (1 - e^{-\kappa t}) \right) + E \quad (29)$$

Tomando a $\theta := \frac{ke^{-\kappa t_d}}{\kappa} [\text{Ag}^+]_i$ para ahorrar coste computacional. Una vez obtenido el parámetro θ , podemos encontrar las variables que lo conforman por medio de los valores conocidos y correlaciones. Así, el modelo sobre el cual se hará la regresión es:

$$A = \epsilon l ([P]_{t_d} + \theta(1 - e^{-\kappa t})) + E \quad (30)$$

Escogiendo x y y correctamente, esta ecuación (30) es equivalente a la ecuación (26), garantizando así la existencia de valores únicos para $[P]_{t_d}$, θ y κ . De igual forma, se impondrá la condición de que estas variables deben ser no negativas. El código asociado a este método es el siguiente:

Código MATLAB para Método 2

```

1 clear all
2 format long
3 % Datos experimentales
4 t = [0; 5; 10; 15; 20; 25; 30; 35; 50; 60; 90; 100; 140; 150; 160;
5     180];
6 A = [0.12; 0.354; 0.624; 0.528; 0.645; 1.0105; 1.143; 1.2175; 1.506;
7     1.4145; 1.806; 1.638; 1.953; 1.892; 1.913; 1.874];
8
9 % Llama a la función createFit para ajustar el modelo a los primeros "
10 % n" puntos
11 [fitresult, gof] = createFit(t, A);
12
13 % Muestra los parámetros ajustados y la bondad del ajuste
14 disp('Parámetros Ajustados:');
15 disp(fitresult);
16 disp('Bondad del Ajuste:');
17 disp(gof);
18
19 % Función para crear el ajuste
20
21 function [fitresult, gof] = createFit(t, A)
22     %CREATEFIT(T,A)
23     % Crea un ajuste.
24     %
25     % Datos para el ajuste 'Metodo2':
26     % X de entrada : t
27     % Y de salida : A
28     % Salida:
29     % fitresult : objeto de ajuste que representa el ajuste.
30     % gof : estructura con información de bondad del ajuste.
31     %
32     % Ver también FIT, CFIT, SFIT.
33
34     %% Prepara los datos para el ajuste utilizando solo los primeros "n
35     % puntos

```

```

33 n = 10; %% n mero de datos utilizados "Datos de la nucleaci n"
34 xData = t(1:n);
35 yData = A(1:n);
36
37 %% Configura el tipo de ajuste y las opciones
38 ft = fitttype('1416.536 * (p_td + theta * (1 - exp(-k * t)))', '
    independent', 't', 'dependent', 'A');
39 opts = fitoptions('Method', 'NonlinearLeastSquares');
40 opts.Display = 'Off';
41 opts.Lower = [0, 0, 0]; % L mites inferiores para k, p_td, theta
42 opts.Upper = [inf, inf, inf]; % L mites superiores para k, p_td,
    theta
43 opts.StartPoint = [0.0018, 1, 0.00018]; % Puntos de inicio para k,
    p_td, theta
44
45 %% Ajusta el modelo a los datos
46 [fitresult, gof] = fit(xData, yData, ft, opts);
47
48 %% Grafica el ajuste con todos los datos y los primeros n puntos en
    otro color
49 figure('Name', 'Metodo2');
50 plot(fitresult, t, A, 'r*');
51 hold on;
52 plot(t(1:n), A(1:n), 'b.', 'MarkerFaceColor', 'r'); % Primeros n
    puntos en rojo
53 hold off;
54 legend('Datos experimentales', 'Ajuste Modelo 2', 'Datos para la
    regresi n', 'Location', 'SouthEast', 'Interpreter', 'none');
55 xlabel('t [min]', 'Interpreter', 'none');
56 ylabel('A [u.a]', 'Interpreter', 'none');
57 grid on;
58 end

```

Método 3

De manera análoga a las pruebas de existencia y unicidad, se puede establecer que para los métodos 3 y 4 existe unicidad de ciertas variables.

Para el método 3 se cumple la ecuación (15):

$$[P] = [P]_{t_d} + \frac{k_3[A]_0}{\kappa} \log \left| \frac{[Ag^+]_i e^{-\kappa t_d} + \frac{k_{-2}+k_3}{k_2}}{[Ag^+]_i e^{-\kappa(t+t_d)} + \frac{k_{-2}+k_3}{k_2}} \right|$$

Tomemos $x = [P]_{t_d}$, $y = \frac{k_3[A]_0}{\kappa}$, $K = \kappa$ y $\beta_j := \frac{k_{-2}+k_3}{k_2[Ag^+]_i e^{-\kappa t_d}}$, llegando a:

$$[P] = x + y \log \left| \frac{1 + \beta_j}{e^{-Kt} + \beta_j} \right| \quad (31)$$

y reemplazando la ecuación (31) en la (25) se tiene:

$$A = \epsilon l \left(x + y \log \left| \frac{1 + \beta_j}{e^{-Kt} + \beta_j} \right| \right) + E \quad (32)$$

Tomando a K , β_j fijos pero arbitrarios y a los vectores:

$$\mathbf{a} = \begin{pmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{pmatrix}, \quad \mathbf{b} = \begin{pmatrix} \log \left| \frac{1+\beta_j}{e^{-Kt_1} + \beta_j} \right| \\ \log \left| \frac{1+\beta_j}{e^{-Kt_2} + \beta_j} \right| \\ \vdots \\ \log \left| \frac{1+\beta_j}{e^{-Kt_n} + \beta_j} \right| \end{pmatrix}, \quad \mathbf{p} = \begin{pmatrix} p_1 \\ p_2 \\ \vdots \\ p_n \end{pmatrix}$$

y definiendo \mathbf{C} como $\mathbf{C} = (\mathbf{a} \ \mathbf{b})$, se ha logrado construir la ecuacion (27) y por consiguiente, se sigue que para K , β_j existen únicos $x = [P]_{t_d}$, $y = \frac{k_3[A]_0}{\kappa}$.

Variando K y β_j , se determinan x y y con la ecuación (32):

$$\mathbf{C}^T \mathbf{C} = \begin{bmatrix} \mathbf{a} \cdot \mathbf{a} & \mathbf{a} \cdot \mathbf{b} \\ \mathbf{b} \cdot \mathbf{a} & \mathbf{b} \cdot \mathbf{b} \end{bmatrix}$$

$$\begin{bmatrix} x \\ y \end{bmatrix} = (\mathbf{C}^T \mathbf{C})^{-1} \mathbf{C}^T \mathbf{p} \quad (33)$$

Basta definir las variables K y β_j positivas, y $K < 0.03$ por lo que el código del modelo 3 es:

Código MATLAB para Método 3

```

1 format long
2 % Datos experimentales
3 t = [0; 5; 10; 15; 20; 25; 30; 35; 50; 60; 90; 100; 140; 150; 160;
4     180];
5 A = [0.12; 0.354; 0.624; 0.528; 0.645; 1.0105; 1.143; 1.2175; 1.506;
6     1.4145; 1.806; 1.638; 1.953; 1.892; 1.913; 1.874];
7
8 % Llama a la función createFit para ajustar el modelo a los primeros "
9     n" puntos
10 [fitresult, gof] = createFit(t, A);
11
12 % Muestra los parámetros ajustados y la bondad del ajuste
13 disp('Parámetros Ajustados:');
14 disp(fitresult);
15 disp('Bondad del Ajuste:');
16 disp(gof);
17
18 % Función para crear el ajuste
19
20 function [fitresult, gof] = createFit(t, A)
21     %CREATEFIT(T,A)
22     % Crea un ajuste.
23     %
24     % Datos para el ajuste 'Metodo3':
25     %     X de entrada : t
26     %     Y de salida  : A
27     % Salida:
28     %     fitresult : objeto de ajuste que representa el ajuste.
29     %     gof       : estructura con información de bondad del ajuste.

```

```

28 %
29 % Ver tambi n FIT, CFIT, SFIT.
30
31 %% Prepara los datos para el ajuste utilizando solo los primeros "n
   " puntos
32 n = 10; %% n mero de datos utilizados "Datos de la nucleaci n"
33 xData = t(1:n);
34 yData = A(1:n);
35
36 %% Configura el tipo de ajuste y las opciones
37 epsilon = 1416.536;
38 ft = fitttype(sprintf('%f * (%s)', epsilon, ...
39     'x1 + y1 * log(abs(1 + beta) / ( exp(-k * t) + beta))'), ...
40     'independent', 't', 'dependent', 'A');
41 opts = fitoptions('Method', 'NonlinearLeastSquares');
42 opts.Display = 'Off';
43 opts.Lower = [2, 0.02, 0, 0]; % L mites inferiores para beta, k,
   x1, y1
44 opts.Upper = [3, 0.02725, 9.0226E-05, 0.00416]; % L mites
   superiores para beta, k, x1, y1
45 opts.StartPoint = [2.81547, 0.02725, 9.0226E-05, 0.00416]; % Puntos
   de inicio para beta, k, x1, y1
46
47 %% Ajusta el modelo a los datos
48 [fitresult, gof] = fit(xData, yData, ft, opts);
49
50 %% Grafica el ajuste con todos los datos y los primeros n puntos en
   otro color
51 figure('Name', 'Metodo3');
52 plot(fitresult, t, A, 'r*');
53 hold on;
54 plot(t(1:n), A(1:n), 'b.', 'MarkerFaceColor', 'r'); % Primeros n
   puntos en rojo
55 hold off;
56 legend('Datos experimentales', 'Ajuste Modelo 3', 'Datos para la
   regresi n', 'Location', 'SouthEast', 'Interpreter', 'none');
57 xlabel('t [min]', 'Interpreter', 'none');
58 ylabel('A [u.a]', 'Interpreter', 'none');
59 grid on;
60 end

```

Método 4

Se sigue que para K y c_1 existen únicos $x = [Ag^+]_0$ y $y = \beta$.

Para el método 4 se cumple la ecuación (22):

$$[P] = [Ag^+]_i - \beta W \left(c_1 e^{-\frac{\alpha}{\beta} t} \right)$$

Tomemos $x = [Ag^+]_i$, $y = \beta$, $K = \frac{\alpha}{\beta}$, llegando a:

$$[P] = x - y W \left(c_1 e^{-Kt} \right) \quad (34)$$

y reemplazando la ecuación (31) en la (25) se tiene:

$$A = \epsilon l \left(x + y \log \left| \frac{1 + \beta_j}{e^{-Kt} + \beta_j} \right| \right) + E \quad (35)$$

Tomando a K , c_1 fijos pero arbitrarios y a los vectores:

$$\mathbf{a} = \begin{pmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{pmatrix}, \quad \mathbf{b} = \begin{pmatrix} -W(c_1 e^{-Kt_1}) \\ -W(c_1 e^{-Kt_2}) \\ \vdots \\ -W(c_1 e^{-Kt_n}) \end{pmatrix}, \quad \mathbf{p} = \begin{pmatrix} p_1 \\ p_2 \\ \vdots \\ p_n \end{pmatrix}$$

y definiendo \mathbf{C} como $\mathbf{C} = (\mathbf{a} \ \mathbf{b})$, se logra construir la ecuación (27) y por consiguiente, se sigue que para K , c_1 existen únicos $x = [\text{Ag}^+]_i$, $y = \beta$.

Variando c_1 y β_j , se determinan x y y con la ecuación (38):

$$\mathbf{C}^T \mathbf{C} = \begin{bmatrix} \mathbf{a} \cdot \mathbf{a} & \mathbf{a} \cdot \mathbf{b} \\ \mathbf{b} \cdot \mathbf{a} & \mathbf{b} \cdot \mathbf{b} \end{bmatrix}$$

$$\begin{bmatrix} x \\ y \end{bmatrix} = (\mathbf{C}^T \mathbf{C})^{-1} \mathbf{C}^T \mathbf{p} \quad (36)$$

Basta definir las variables K y c_1 positivas, y $c_1 < 0.5$ por lo que el código del modelo 4 es:

Código MATLAB para Método 4

```

1 format long
2 % Datos experimentales
3 t = [0; 5; 10; 15; 20; 25; 30; 35; 50; 60; 90; 100; 140; 150; 160;
4     180];
5 A = [0.12; 0.354; 0.624; 0.528; 0.645; 1.0105; 1.143; 1.2175; 1.506;
6     1.4145; 1.806; 1.638; 1.953; 1.892; 1.913; 1.874];
7
8 % Llama a la función createFit para ajustar el modelo a los primeros "
9     n" puntos
10 [fitresult, gof] = createFit(t, A);
11
12 % Muestra los parámetros ajustados y la bondad del ajuste
13 disp('Parámetros Ajustados:');
14 disp(fitresult);
15 disp('Bondad del Ajuste:');
16 disp(gof);
17
18 % Función para crear el ajuste
19
20 function [fitresult, gof] = createFit(t, A)
21     %CREATEFIT(T,A)
22     % Crea un ajuste.
23     %
24     % Datos para el ajuste 'Metodo4':
25     %     X de entrada : t
26     %     Y de salida  : A
27     % Salida:
28     %     fitresult : objeto de ajuste que representa el ajuste.
29     %     gof       : estructura con información de bondad del ajuste.

```

```

30 % Ver tambi n FIT, CFIT, SFIT.
31
32 %% Prepara los datos para el ajuste utilizando solo los primeros "n
   " puntos
33 n = 10; %% n mero de datos utilizados "Datos de la nucleaci n"
34 xData = t(1:n);
35 yData = A(1:n);
36 %% Configura el tipo de ajuste y las opciones
37 epsilon = 1416.536;
38 ft = fitttype(sprintf('%f * (%s)', epsilon, ...
39     'x1 - y1 * lambertw(0, c1 * exp(-k * t))'), ...
40     'independent', 't', 'dependent', 'A');
41 opts = fitoptions('Method', 'NonlinearLeastSquares');
42 opts.Display = 'Off';
43 opts.Lower = [0, 0.026, 0.001356 , 0.00839]; % L mites inferiores
   para c1, k, x1, y1
44 opts.Upper = [0.3, 0.028, 0.001356, 0.00839]; % L mites superiores
   para c1, k, x1, y1
45 opts.StartPoint = [0.15, 0.027, 0.001356, 0.00839]; % Puntos de
   inicio para c1, k, x1, y1
46
47 %% Ajusta el modelo a los datos
48 [fitresult, gof] = fit(xData, yData, ft, opts);
49
50 %% Grafica el ajuste con todos los datos y los primeros n puntos en
   otro color
51 figure('Name', 'Metodo4');
52 plot(fitresult, t, A, 'r*');
53 hold on;
54 plot(t(1:n), A(1:n), 'b.', 'MarkerFaceColor', 'r'); % Primeros n
   puntos en rojo
55 hold off;
56 legend('Datos experimentales', 'Ajuste Modelo 4', 'Datos para la
   regresi n', 'Location', 'SouthEast', 'Interpreter', 'none');
57 xlabel('t [min]', 'Interpreter', 'none');
58 ylabel('A [u.a]', 'Interpreter', 'none');
59 grid on;
60 end

```

Método 5

En el siguiente código se presenta la implementación de un modelo matemático para describir la dinámica de un sistema químico mediante ecuaciones diferenciales. El modelo considera las siguientes reacciones:

Considerando todas las velocidades de reacción, la función dY_dt calcula la tasa de cambio de las concentraciones Y en función del tiempo t utilizando los parámetros k_1, k_2, k_3, k_4 y k_{-2} . Cada velocidad de reacción r_i se expresa como:

$$\begin{aligned}
 r_1 &= k_1[\text{AgNO}_3] \\
 r_2 &= k_2[\text{A}][\text{Ag}^+] \\
 r_3 &= k_3[\text{A} \cdot \text{Ag}^+] \\
 r_4 &= k_4[\text{Ag}^0] \\
 r_{-2} &= k_{-2}[\text{A} \cdot \text{Ag}^+]
 \end{aligned}$$

donde $Y = [\text{Ag}^+, \text{Ag}, \text{NPs}, \text{A}, \text{A} \cdot \text{Ag}, \text{A} \cdot \text{NPs}]$ son las concentraciones de las especies químicas en función del tiempo t .

Para resolver este modelo, se define la función `dY_dt`, que calcula la tasa de cambio de Y en función de los parámetros k_1, k_2, k_3, k_4 y k_{-2} . Esta función utiliza las velocidades de reacción determinadas por las leyes de velocidad elemental para cada especie química, como se muestra en la tabla 1.

Se utiliza la función `odeint` de SciPy para integrar las ecuaciones diferenciales y obtener las concentraciones Y en función del tiempo t . Además, se implementa la función `Y_modelo` para facilitar la resolución del modelo, devolviendo los valores calculados de Y en un intervalo de tiempo dado.

Además de la simulación del modelo, se realiza una optimización para ajustar los parámetros k_1, k_2, k_3 y k_{-2} , manteniendo k_4 constante con el valor de 0 debido a la falta de información del modelo, a los datos experimentales proporcionados. Se utiliza el método de mínimos cuadrados no lineales mediante la función `minimize` de SciPy para maximizar el coeficiente de determinación ajustado R^2 .

Finalmente, se generan gráficos que comparan los resultados del modelo con los datos experimentales, evaluando el ajuste mediante estadísticas como la suma de errores cuadráticos (SSE), la suma total de cuadrados (SST), R^2 y R^2 ajustado. Estas gráficas y estadísticas proporcionan una evaluación cuantitativa del rendimiento del modelo respecto a los datos experimentales.

Código Python para Modelo Cinético y Optimización

```

1 import numpy as np
2 from scipy.integrate import odeint
3 from scipy.optimize import minimize
4 import matplotlib.pyplot as plt
5
6 # Ecuación diferencial que describe el modelo
7 def dY_dt(Y, t, k1, k2, k3, k4, k_2):
8     # Definición de las velocidades de reacción
9     r1 = k1 * Y[0]
10    r2 = k2 * Y[1] * Y[3]
11    r3 = k3 * Y[4]
12    r4 = k4 * Y[5]
13    r_2 = k_2 * Y[4]
14
15    # Matriz de coeficientes de la especie j en la reacción i
16    r_ij = np.array([[-1, 0, 0, 0, 0],
17                    [1, -1, 0, 0, 1],
18                    [1, 0, 0, 0, 0],
19                    [0, -1, 1, 0, 1],
20                    [0, 1, -1, 0, -1],
21                    [0, 0, 1, -1, 0],
22                    [0, 0, 0, 1, 0]])
23
24    # Calcular las velocidades de reacción
25    ri = np.array([r1, r2, r3, r4, r_2])
26    rj = np.dot(r_ij, ri)
27
28    # Devolver las derivadas
29    dY_dt = [rj[0], rj[1], rj[2], rj[3], rj[4], rj[5]]
30    return dY_dt

```

```

31
32 # Modelo o soluci n de las ED's
33 def Y_modelo(t_i, t_f, Y0, k1, k2, k3, k4, k_2):
34     # Intervalo de tiempo
35     t_values = np.linspace(t_i, t_f, 1000)
36
37     # Resolver la ecuaci n diferencial
38     Y_values = odeint(dY_dt, Y0, t_values, args=(k1, k2, k3, k4, k_2))
39
40     return t_values, Y_values
41
42 # Funci n objetivo para la optimizaci n
43 def objetivo(params):
44     k1, k2, k3, k_2 = params
45     _, Y_values = Y_modelo(t_i, t_f, Y0, k1, k2, k3, k4, k_2)
46     A_prueba_optimo = np.interp(t_exp, _, epsilon * np.array(Y_values
47        [:, -2]) + epsilon * np.array(Y_values[:, -1])) # Comparando
48         con la [P]
49     _, _, _, R2_ajustado = evaluar_ajuste(A_exp, A_prueba_optimo)
50     return -R2_ajustado
51
52 # Optimizaci n para encontrar los mejores par metros k1, k2, k3 y k_2
53 resultado_optimizacion = minimize(objetivo, params_iniciales, method='
54     Nelder-Mead')
55
56 # Obtener los mejores par metros encontrados
57 k1_optimo, k2_optimo, k3_optimo, k_2_optimo = resultado_optimizacion.x
58
59 # Calcular A_prueba con los mejores par metros encontrados
60 t_values_optimo, Y_values_optimo = Y_modelo(t_i, t_f, Y0, k1_optimo,
61     k2_optimo, k3_optimo, k4, k_2_optimo)
62 A_prueba_optimo = np.interp(t_exp, t_values_optimo, epsilon * np.array(
63     Y_values_optimo[:, -2]) + epsilon * np.array(Y_values_optimo[:, -1])
64 )
65 A_values_optimo = epsilon * np.array(Y_values_optimo[:, -2]) + epsilon
66     * np.array(Y_values_optimo[:, -1])
67
68 # Graficar resultados
69 plt.figure(figsize=(10, 6))
70 plt.plot(t_exp, A_exp, 'ro', label='Datos experimentales')
71 plt.plot(t_values_optimo, A_values_optimo, 'b-', label=f'Mejor ajuste (
72     R2 ajustado = {R2_ajustado_optimo:.4f})')
73 plt.title('Comparaci n entre Datos Experimentales y Mejor Ajuste')
74 plt.xlabel('Tiempo min')
75 plt.ylabel('Absorbancia')
76 plt.legend()
77 plt.grid(True)
78 plt.show()

```