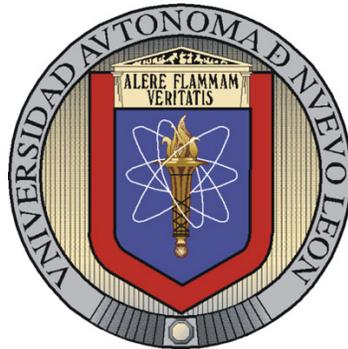


UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN

FACULTAD DE INGENIERÍA MECÁNICA Y ELÉCTRICA

DIVISIÓN DE ESTUDIOS DE POSTGRADO



**DESARROLLO Y ANÁLISIS DE UN SIMULADOR DE
NUCLEACIÓN Y CRECIMIENTO EN CONDICIONES DE
NUCLEACIÓN TÉRMICA**

POR

RAMÓN CANTÚ CUÉLLAR

TESIS

EN OPCIÓN AL GRADO DE DOCTOR EN INGENIERÍA DE
MATERIALES

SAN NICOLÁS DE LOS GARZA, N.L., AGOSTO DE 2009

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN

FACULTAD DE INGENIERÍA MECÁNICA Y ELÉCTRICA

DIVISIÓN DE ESTUDIOS DE POSTGRADO



**DESARROLLO Y ANÁLISIS DE UN SIMULADOR DE
NUCLEACIÓN Y CRECIMIENTO EN CONDICIONES DE
NUCLEACIÓN TÉRMICA**

POR

RAMÓN CANTÚ CUÉLLAR

TESIS

EN OPCIÓN AL GRADO DE DOCTOR EN INGENIERÍA DE
MATERIALES

SAN NICOLÁS DE LOS GARZA, N.L., AGOSTO DE 2009

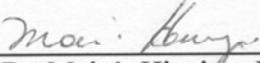
UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN
FACULTAD DE INGENIERÍA MECÁNICA Y ELÉCTRICA
DIVISIÓN DE ESTUDIOS DE POSTGRADO

Los miembros del Comité de Tesis recomendamos que la tesis “**Desarrollo y Análisis de un Simulador de Nucleación y Crecimiento en Condiciones de Nucleación Térmica**” realizada por el M.C. **Ramón Cantú Cuéllar** con número de matrícula **366536** sea aceptada para su defensa con opción al grado de Doctor en Ingeniería de Materiales.

El comité de tesis



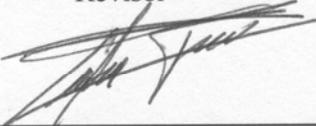
Dr. Virgilio A. González González
Asesor



Dr. Moisés Hinojosa Rivera
Revisor



Dr. Martín Edgar Reyes Melo
Revisor

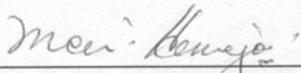


Dr. Alejandro Torres Castro
Revisor



Dr. Karím de Alba Romenus
Revisor

Vo. Bo.



Dr. Moisés Hinojosa Rivera
Subdirector de Estudios de
Postgrado

Ciudad Universitaria, San Nicolás de los Garza, N. L. a Agosto del 2009

Agradecimientos

Al Dr. Virgilio Ángel González González por introducirme en el fascinante mundo de la Cristalización de Polímeros y por toda su ayuda en todos estos años, él fué un gran asesor y un verdadero amigo de quién aprendí muchísimas cosas.

A los Doctores Moisés Hinojosa Rivera, Martín Édgar Reyes Melo y Alejandro Torres Castro, mis revisores de tesis quienes con sus observaciones y recomendaciones contribuyeron al enriquecimiento de este trabajo.

Al Dr. Karim de Alba Romenus por sus comentarios sobre programación en la revisión del presente trabajo y por honrarme con su amistad.

Al M.E.C. Rogelio G. Garza Rivera por su apoyo durante mis estudios de Postgrado.

A mi madre Argelia Cuéllar González y a la memoria de mi padre Rubén Cantú Cañamar y de mi tío César A. Cuéllar González.

Y finalmente a todos mis amigos quienes me apoyaron, motivaron e hicieron de mi permanencia en el DIMAT una gran experiencia en mi vida.

Índice

	Página
Resumen	1
1.-Introducción	3
2.- Antecedentes	7
2.1.- Cristalización	7
2.1.1.- Introducción	7
2.1.2.- Cristalización de polímeros	7
2.1.2.1. Modelos de cristalización molecular.	10
2.1.2.1.1. Teoría de superficie rugosa.	10
2.1.2.1.2. Teoría de nucleación secundaria.	11
2.1.2.2. Modelos de cristalización en masa.	12
2.1.2.2.1. Cristalización isotérmica	12
2.1.2.2.2. Cristalización dinámica	14
2.2.- Modelado y simulación de cristalización	15
3.- Hipótesis y Objetivos	23
3.1.- Hipótesis	23
3.2.- Objetivo general	23
4.- Metodología	24
4.1.- Deducciones	24
4.2.- Diagrama de bloques y programación	24
4.3.- Experimentos de la simulación	24
5.- El modelo	26
5.1.- Análisis de geometría de fronteras	26
5.2.- Determinación de los límites de las fronteras	33
5.3.- Calculo de áreas	35
6.- El simulador	39
6.1.- Aspectos generales de la simulación	39
6.2.- Procedimientos para la generación de núcleos	42
6.3.- Procedimiento para la determinación de fronteras entre núcleos	45
6.4.- Procedimiento para la validación por taponamiento	47
7.- Resultados de la simulación	49
7.1.- Rapidez de nucleación constante	49
7.2.- Función de nucleación asintótica	64
7.3.- Función de nucleación constante respecto al área no transformada	76
7.4.- El error en mediciones experimentales	88
8.- Conclusiones	93
9.- Trabajo futuro	94
10.- Bibliografía	95
Índice de figuras	101
Índice de esquemas	106
Índice de tablas	106
Apéndice A, Límites de fronteras	107
Apéndice B, Codificación del programa	118

Resumen

Basándose en el modelo de las fronteras hiperbólicas, se desarrolló un simulador mediante computadora que permite analizar estadísticamente los resultados cinéticos de fenómenos de nucleación y crecimiento en condiciones de nucleación térmica. Se utilizó el simulador para analizar tres condiciones de rapidez de nucleación (G_n), dos hipotéticas que consisten en considerar G_n en términos del número de núcleos respecto al área como constante o como una función asintótica y una más, considerando G_n como número de núcleos en función del área no cubierta.

En la introducción se describe el fenómeno a estudiar (nucleación y crecimiento), en el contexto de las ciencias naturales enfocando su uso en materiales al analizar la cristalización de polímeros y solidificación de materiales metálicos y cerámicos

En los antecedentes se establecen las bases teóricas, morfológicas y cinéticas, necesarias para hacer el planteamiento del modelo bidimensional para nucleación y crecimiento en la cristalización, se describen someramente los modelos de simulación existentes, todo esto como fundamento de la hipótesis y objetivo de la tesis, consistente en el desarrollo de un simulador de nucleación y crecimiento con nucleación térmica, basado en las fronteras intercristalinas hiperbólicas, resuelto analíticamente, así como su uso en el análisis del fenómeno.

Posterior al establecimiento de la metodología se reportan las demostraciones matemáticas más importantes, dejando otras para los apéndices, se describen mediante diagramas de bloques, el simulador en general y los algoritmos principales, cuya programación en lenguaje Pascal se reporta en un apéndice.

Los resultados de la simulación se reportan para tres casos, para G_n respecto al área de la muestra constante, donde se efectuaron trece simulaciones con cien repeticiones cada una, el segundo caso para G_n respecto al área de la muestra como una función asintótica del tiempo, donde se hicieron 34 simulaciones con diferente función y con 100 repeticiones cada una y el tercer caso, con G_n respecto al área residual constante en el que se hicieron 8 simulaciones de diferente G_n , también con cien repeticiones cada una. Dando así un total de 5,500 simulaciones. El exponente de Avrami obtenido para éstos tres casos tuvo un valor entre 2.52 y 2.92, teniendo una tendencia al valor de 3 a densidades de nucleación que tienden a infinito.

Los resultados para los dos primeros casos de simulación, permitieron comprender el efecto de las variables de entrada en los resultados del simulador, así como inspeccionar la reproducibilidad de las simulaciones, estas dos formas junto con la tercera mostraron la factibilidad de determinar G_n en todos los casos, así como también demostrar que es necesario analizar el exponente de Avrami en función del tiempo, para la selección del intervalo de datos del cual se podrá extraer este dato.

Se demostró también, utilizando los planteamientos teóricos y los resultados de simulación, que en las técnicas comúnmente usadas, como la metalografía, en lugar de reportar las estadísticas de tamaño de grano, reportan en realidad las relacionadas a áreas de grano descubiertas por el corte.

Posteriormente, se reportan dos métodos que pueden ser aplicados en el laboratorio para determinar G_n mostrando como es necesario utilizar muestras en forma de películas lo más delgada posible, reportando el error en función del espesor.

Por último se presentan los trabajos que podrían realizarse tanto en la aplicación del simulador, como su mejora al incluir la nucleación atérmica y su desarrollo a tres dimensiones.

Las dos principales contribuciones del presente trabajo es un método para la determinación de la función de nucleación a partir de la geometría de las fronteras intercrystalinas y la conveniencia de determinar el exponente de Avrami mediante la selección del intervalo de regresión mediante gráficas de la función de este exponente con el tiempo.

1.- Introducción

Innumerables fenómenos naturales involucran a los procesos de nucleación y crecimiento; desde condensados estelares [1], formación de aerosoles en la atmósfera [2], polimerización de hemoglobina celular [3], fusión de gases nobles [4], solidificación de metales [5] y cerámicos [6], cristalización de grasas [7] y aceites [8], cristalización de polímeros [9], hasta la formación de nanopartículas [10]. En los casos relacionados a materiales convencionales, la nucleación y crecimiento de cristales, determina las variables morfológicas como: tamaño, número y forma de los cristales, y en el caso particular de los polímeros define también el grado de cristalización, características que determinan a su vez la mayoría de sus propiedades (v.g. químicas, ópticas eléctricas, mecánicas).



Figura 1.1. a) Semillas de diente de león, b) esferulita de poli-cis-1,4-isopreno cristalizado durante 2 Hrs. a $-26\text{ }^{\circ}\text{C}$, observada mediante microscopía electrónica de transmisión TEM (25,000X).

Por lo general, durante los procesos de transformación, los materiales son sujetos a diversos procesos térmicos como templado o recocido, enfriamientos lentos o súbitos; y procesos mecánicos como estiramiento y compresión, en todos ellos ocurren cambios morfológicos relacionados a fenómenos de nucleación y crecimiento, razón por la que el

conocimiento de la fenomenología de nucleación y el posterior crecimiento de los núcleos formados es invaluable para el desarrollo y transformación de materiales.

Los polímeros cristalizan principalmente en forma de esferulitas, las cuales son estructuras en forma de esferas en que laminitas (o lamellas) crecen radialmente a partir del centro, si pudiésemos observarlas en tres dimensiones, estas se parecerían a las semillas de diente de león *Taraxacum officinale*, Fig. 1.1a, y observadas mediante microscopía electrónica de transmisión (TEM), se observan cómo se muestra en la Fig. 1.1b.

Las laminitas tienen una estructura en la que los ejes de las “cadenas” poliméricas se ordenan perpendicularmente al espesor de las laminitas (Fig. 1.2), estos segmentos se doblan y re-entran al cristal, generalmente en la celda unidad contigua, resultando en una fase cristalina y una fase semi-ordenada en la superficie embebidas en la fase amorfa.

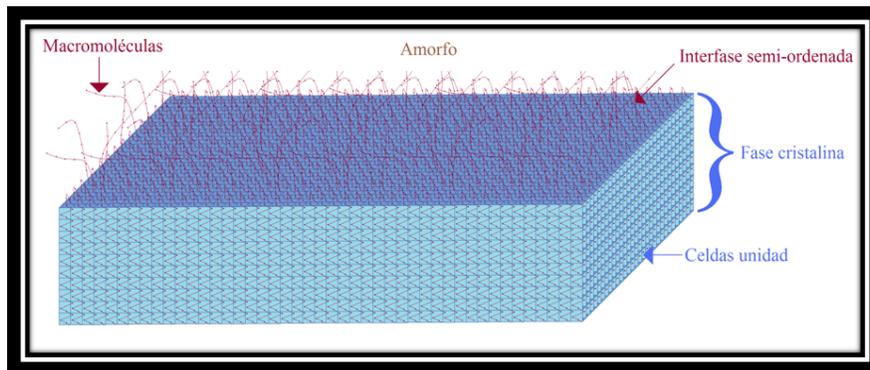


Figura 1.2.- Diagrama esquemático de la estructura de las laminitas

Por otra parte, además de las dendritas, una de las morfologías típicas de metales y cerámicos son los granos, estos son completamente llenos de material cristalino (con fallas estructurales como vacancias y dislocaciones), cuyas formas son de poliedros, entre los que se dice, bajo consideraciones energéticas y geométricas, que la forma de tetracaidecaedro es la más estable [11] y que además cubre todo el espacio, formando así las estructuras de Weaire-Phelan [12] (Fig. 1.3), en realidad esta estructura solo ha sido observada en espumas (aunque con las caras curvadas) y organizaciones de algunas zeolitas.

Como cualquier otro fenómeno natural, la nucleación y crecimiento puede estudiarse ya sea desde el punto de vista termodinámico, o bien cinético. En este trabajo se pretende elucidar desde el punto de vista cinético las formas y tamaños de las entidades cristalinas, ya sea granos (metales), o esferulitas (polímeros).



Figura 1.3.- Estructura Weaire-Phelan formada por tetracaidecaedros [12].

Considerando que el problema de relacionar la cinética de cristalización con la forma de los cristales, es un problema complejo, el enfoque que consideramos adecuado, es el del modelado y simulación.

Como es sabido, la matemática es la herramienta imprescindible para describir fenómenos físicos los cuales hayan sido sujetos de una descripción física [13]. La geometría, que se ocupa de las propiedades del espacio, puede considerarse como el inicio de las matemáticas. La llamada geometría euclidiana se establece en el año 300 A.C. y a pesar de que con el pasar de los años se han desarrollado las geometrías: cartesiana (1637) [14], de Lovachevsky (1829) [15], diferencial (1828) [16] y fractal (1975) [17], entre otras, la geometría euclidiana sigue siendo una de las más utilizadas.

La mayoría de los fenómenos naturales tienen características geométricas [13,18], así, se descubren planetas, estrellas y otros cuerpos siderales, de acuerdo a las formas de las órbitas y trayectorias (por ejemplo el descubrimiento de Neptuno por Leverrier en

1846), la factibilidad de las reacciones químicas[19], principalmente en el área de la fotoquímica orgánica es función de la geometría de las moléculas involucradas y, últimamente se caracterizan superficies [20-24], formadas durante la cristalización ó fractura de los materiales diversos como vidrios, metales, y plásticos, mediante la geometría fractal. Además, del empleo de la topología para analizar el crecimiento de granos en metales mediante modelos geométricos [25]. En esta tesis emplearemos la geometría analítica plana y del espacio, en el estudio de la cristalización de polímeros y solidificación de materiales metálicos y cerámicos mediante el modelado de los fenómenos de nucleación y crecimiento de cristales(granos y esferulitas).

2.- Antecedentes

2.1.- Cristalización

2.1.1.- Introducción

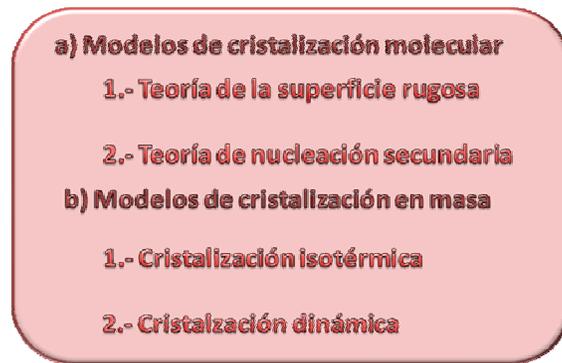
Aunque el planteamiento sobre la utilidad del modelado y simulación sujeto de este trabajo abarca diversos fenómenos de nucleación y crecimiento, la orientación principal es hacia el fenómeno de cristalización en materiales, principalmente a la formación de granos en la solidificación de cerámicos y metales, pero fundamentalmente en la cristalización de polímeros. Ya que los fenómenos de cristalización de materiales metálicos y cerámicos son muy semejantes desde el punto de vista geométrico a los de los polímeros, y considerando que la observación directa de la formación de esferulitas mediante microscopía óptica es una técnica relativamente fácil, mientras que la observación de formación de granos durante la solidificación es una tarea titánica que solo se podría obtener mediante técnicas como microscopía de rayos X. En seguida se discutirá el proceso de cristalización de polímeros, haciendo mención cuando sea necesario de su aplicación a materiales metálicos y cerámicos.

2.1.2.- Cristalización de polímeros

La cristalización de polímeros flexibles a partir de soluciones y fundidos es un problema fundamental en la física de polímeros, actualmente es aún tema de investigación [26]. En la introducción (Punto 1), se hizo una descripción de la morfología principal de los cristales poliméricos, llamada esferulita. Los modelos para describir su formación y crecimiento se dividen en a) modelos de cristalización molecular y b) modelos de cristalización en masa (ver el esquema 2.1), los cuales se pueden dividir a su vez en las teorías de la superficie rugosa, de nucleación secundaria, de la cristalización isotérmica dinámica.

Para que ocurra la cristalización [27], se forman inicialmente núcleos primarios en el medio fundido, la rapidez de formación de estos núcleos primarios (G_n), puede ser muy alta debido a diversos factores, entre los que destaca la presencia de impurezas o aditivos que sirven como agentes de nucleación o suficientemente lenta como para que

la nucleación siempre esté dirigida por las fluctuaciones termodinámicas en el medio, la única manera de que ocurra de esta última forma es que no haya agentes de nucleación y, a presión atmosférica, que la temperatura esté en el intervalo entre la transición vítrea (T_g) y la temperatura de fusión al equilibrio (T_m°). Debido a lo anterior, cuando todos los núcleos se forman al inicio de la cristalización, a este proceso se le denomina nucleación heterogénea, aunque es más correcto identificarla como nucleación atérmica. En contraste se dice que es nucleación homogénea cuando está ocurriendo durante el transcurso de la cristalización, y es preferible llamarla nucleación térmica.



Esquema 2.1. Los modelos de cristalización

Analizando la rapidez de cristalización en función de la temperatura, encontramos que a temperatura constante, la rapidez de crecimiento de los cristales es constante y obedece a la ecuación de Lauritzen-Hoffman, (Ec. 2.1):

$$Gr_{(T)} = G_0 \exp(-\Delta G_d / K_B T) \exp(-K_g / T \Delta T f) \quad [2.1]$$

donde $Gr_{(T)}$ es la rapidez de crecimiento de los cristales a la temperatura T , G_0 es un factor pre-exponencial, ΔG_d es la energía de activación de difusión de segmentos moleculares, K_g es la energía libre de formación del núcleo crítico, K_B es la constante de Boltzmann, ΔT es la diferencia de temperaturas entre la de equilibrio y la de cristalización y f es un factor de corrección por la disminución de la entalpía de fusión con la temperatura.

Como se aprecia en la ecuación 2.1, ésta se puede dividir en dos rapidezces de crecimiento, una asociada al primer factor exponencial del lado derecho y que

representa a la cristalización controlada por la difusión molecular, la cual se incrementa con la viscosidad y por ende con la temperatura, mientras que la rapidez de cristalización asociada con el segundo término exponencial de la derecha de la Ec. 2.1, representa a la cristalización controlada por la estabilización termodinámica de los núcleos en formación, la cual obviamente disminuye al incrementar la temperatura. Como resultado, la gráfica de Gr vs. temperatura resulta ser una curva en forma de campana (Fig. 2.1).

A temperaturas por debajo de la transición vítrea, no ocurre la cristalización, pues no hay movilidad molecular de largo alcance, a partir de esta temperatura, hasta llegar al máximo, la viscosidad es suficientemente baja y la estabilización termodinámica tan rápida, que la Gr está determinada por la difusión molecular, por lo tanto va aumentando cuando se incrementa la temperatura. A partir del máximo de la curva, que podría llamarse temperatura crítica, la cual es diferente para cada polímero y cambia con la presión a la que ocurre la cristalización, la difusión se hace muy rápida en comparación a una lenta estabilización de los núcleos en formación, por lo que esta última característica determina la rapidez de cristalización, disminuyendo así con la temperatura hasta llegar a la temperatura de fusión al equilibrio (T_{m_0}) en donde Gr se hace cero.

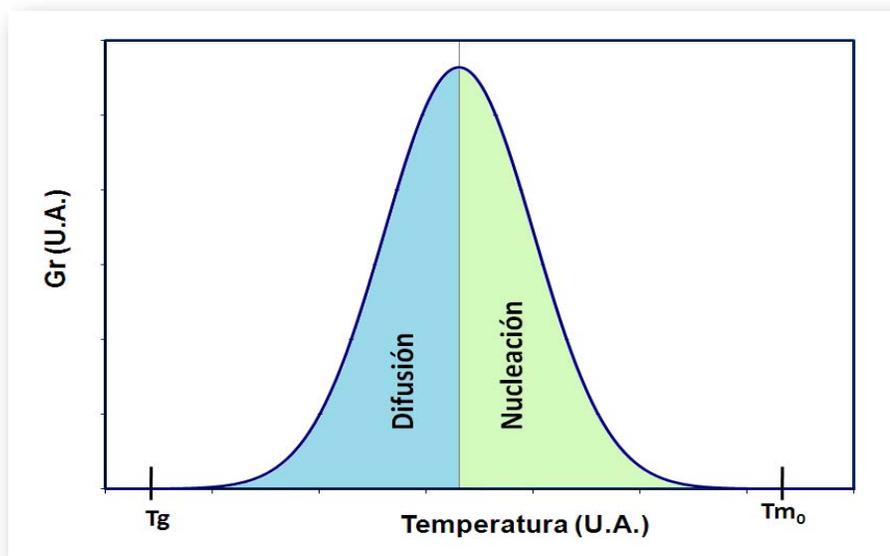


Figura 2.1. Rapidez de cristalización (Gr) en función de la temperatura, de acuerdo a la Ecuación de Lauritzen-Hoffman (Ec. 2.1).

Ya que es importante tener, al menos conceptualmente, bien entendido el proceso de formación de las esferulitas, en seguida se hace, muy sintéticamente la descripción de la fenomenología de acuerdo con las teorías mostradas en el esquema 2.1.

2.1.2.1. Modelos de cristalización molecular.

2.1.2.1.1. Teoría de superficie rugosa. [28]

Aquí la macromolécula se considera como una secuencia de eslabones (o segmentos), que están adhiriéndose y separándose continuamente del frente de crecimiento de las laminitas, (Fig. 2.2), esta dinámica va resultando en que cada vez son más grandes los segmentos moleculares adicionados hasta que se alcanzan conformaciones moleculares de mínima energía y tamaños de núcleos estables, incorporándose así definitivamente al cristal. El proceso se repite continuamente en el frente de crecimiento. Isotérmicamente se predice una rapidez de crecimiento constante.

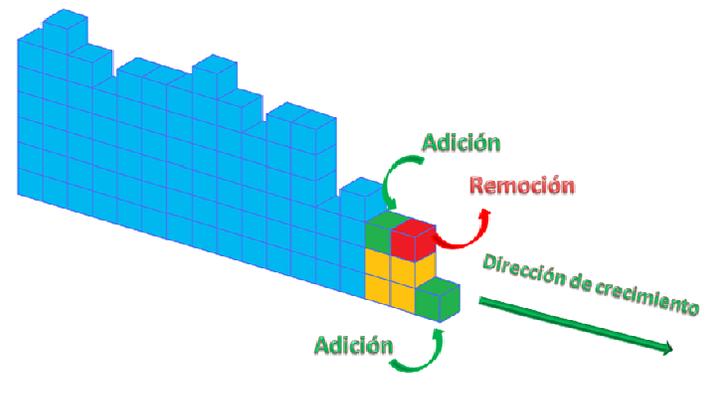


Figura 2.2. Representación esquemática del proceso de crecimiento de una laminita según la teoría de la superficie rugosa.

2.1.2.1.2. Teoría de nucleación secundaria. [29]

La Teoría de Nucleación Secundaria (TNS) describe el crecimiento del cristal basado en un proceso donde los segmentos moleculares nuclean continuamente sobre un núcleo primario formado previamente, la formación de un nuevo núcleo estable sobre la superficie de crecimiento está seguida por un crecimiento rápido lateral (Fig. 2.3). El modelo de la TNS describe a un núcleo creciendo en el frente de una laminita de espesor ξ , cuyas dimensiones son $\xi \times a \times b$. El núcleo inicia teniendo un espesor muy pequeño, por lo que, debido a la contribución de las interfaces, a la energía libre del núcleo (lateral, σ y basal σ_e), la energía libre de cristalización resulta positiva, hasta que alcanza un espesor crítico (ξ^*), la capa del frente de crecimiento se deposita a lo largo del cristal (L_0), con una rapidez g , la repetición de la formación de estos núcleos secundarios, resulta en una rapidez de crecimiento del cristal G_r .

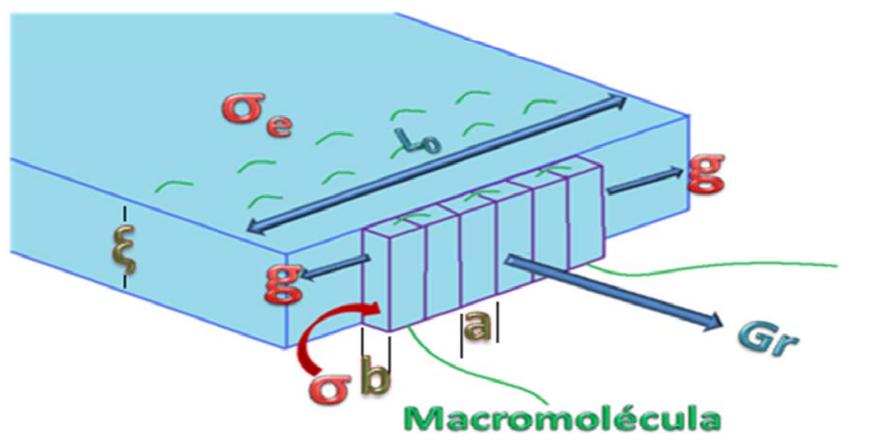


Figura 2.3. Conceptualización de la formación de un núcleo secundario en el crecimiento laminar de polímeros. L_0 es el ancho del cristal, ξ el espesor del cristal, a y b las dimensiones ortogonales del cristal, σ y σ_e las energías libres de superficie, g la rapidez de crecimiento lateral y G_r la rapidez de crecimiento laminar

Es interesante hacer notar que existen al menos tres regímenes de nucleación [30], (Fig. 2.4), considerando la rapidez con que se forman núcleos sobre la superficie de crecimiento “ i ” y la rapidez con que crecen los núcleos estables en la dirección del ancho de la laminita “ g ”, cuando $g \gg i$, siempre en el frente de crecimiento hay solo un núcleo en formación, y tenemos el régimen I, el régimen II se da cuando $g \approx i$, y en una

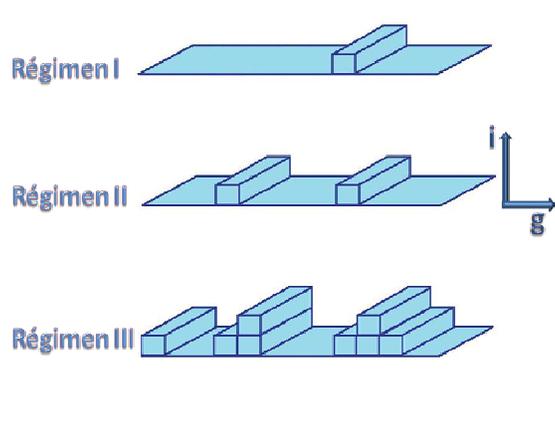


Figura 2.4. Los regímenes de nucleación. g: rapidez de crecimiento lateral, i: rapidez de formación de núcleos en la dirección de crecimiento laminar

misma superficie del frente de crecimiento puede haber más de dos núcleos en misma superficie del frente de crecimiento puede haber más de dos núcleos en formación, y por último cuando $g \ll i$, además de haber varios núcleos sobre la superficie en crecimiento, hay núcleos formándose sobre capas que no han sido completadas, presentándose así varias capas incompletas.

2.1.2.2. Modelos de cristalización en masa.

2.1.2.2.1. Cristalización isotérmica

El modelo aceptado como el que mejor describe la cristalización isotérmica, es el modelo de Johnson-Mehl-Avrami [31-33], en el que con base en consideraciones termodinámicas y cinéticas, además de un tratamiento estadístico relacionado a la dimensionalidad del crecimiento, se llega a la Ec. 2.2

$$\log(1 - \phi_{(t)}) = -kt^n \quad [2.2]$$

Donde $\Phi_{(t)}$ es el grado de cristalización al tiempo “t”, k es una constante, llamada de velocidad, relacionada con la geometría y rapidez de crecimiento y n, comúnmente

llamado el exponente de Avrami depende del tipo de nucleación primaria y la dimensionalidad del crecimiento (Tabla 2.1). Es conveniente aclarar que, a diferencia de los granos en materiales metálicos y cerámicos, en el caso de los polímeros, las esferulitas tienen un contenido relativamente elevado de material amorfo, por lo que nunca se alcanza el 100% de cristalización, entonces $\Phi_{(t)}$ para polímeros, representa un grado de cristalinidad relativo, es decir, considerando como 100% el máximo grado de cristalinidad alcanzable. En adelante, se sobreentenderá que el grado de cristalinidad para polímeros será el grado de cristalinidad relativo.

Como se aprecia en la tabla 2.1, el exponente de Avrami toma valores enteros entre 1 y 4 dependiendo tanto del tipo de nucleación como de la dimensionalidad del crecimiento. Según la Ec. 2.2, su valor se puede determinar de una gráfica entre $\ln(-\ln(1-\Phi(t)))$ vs. $\ln(t)$.

Tabla 2.1. Magnitudes de las constantes de Avrami (K y n) en función de dimensionalidad de crecimiento, tipo de nucleación y rapidez de crecimiento.

Dimensionalidad del crecimiento	Modo de nucleación	Exponente de Avrami	Constante de Velocidad (K)
1D (barras o cilindros) 	Atérmica	1	$(A)\rho_n G_l$
	Térmica	2	$(A/2)G_n G_l$
2D (discos) 	Atérmica	2	$(\pi E)\rho_n G_l^2$
	Térmica	3	$(\pi E/3)G_n G_l^2$
3D (Ej. Esferas) 	Atérmica	3	$(4\pi/3)\rho_n G_l^3$
	Térmica	4	$(\pi/3)G_n G_l^3$

Donde: A es el área transversal de la barra en crecimiento, E el espesor del disco, ρ_n la densidad de nucleación, G_n la rapidez de nucleación, G_l la rapidez de crecimiento lineal de la morfología.

Aunque como hemos visto, los fundamentos de la descripción de la cristalización mediante la teoría de Avrami datan de 1939 y nació considerando la solidificación de metales como principal aplicación [31-33], esta teoría se ha aplicado y se aplica a numerosos fenómenos de nucleación y crecimiento, por ejemplo: cristalización de polímeros [34], solidificación de metales [35] y cerámicos [36], cristalización de sustancias orgánicas de bajo peso molecular [37,38] fusión de argón [39] e inclusive en

cambios de “fase” magnéticos [40] entre otros. Actualmente sigue siendo muy utilizada al comparar sus predicciones con los resultados de simulación [41].

De los resultados de la utilización de la Ec. de Avrami en el estudio de la cristalización de polímeros, se pueden destacar que prácticamente ningún exponente de Avrami determinado experimentalmente coincide a plena satisfacción con números enteros, tampoco se reporta reproducibilidad para un mismo polímero, esto se puede corroborar consultando cualquiera de los *Handbook* especializados en polímeros y cristalización [27,30,42]. Lo anterior ha llevado a hacer modificaciones de la Ec. de Avrami, por ejemplo las planteadas por Lambrigger [43] quien establece tres tipos de modificaciones dependientes de la magnitud del exponente de Avrami.

2.1.2.2.2. Cristalización dinámica

La mayoría de los procesos de transformación de materiales incluyen el enfriamiento desde el estado fundido, resultando así en un proceso de cristalización dinámica [44]. Si observamos la Fig. 2.1, durante el enfriamiento, el material observa una transformación fundido – cristal variable, resultado de la integración de la rapidez de cristalización en función de la temperatura a lo largo de todo el intervalo de cristalización ($T_g < T < T_{m0}$). Se han efectuado diversas aproximaciones para caracterizar la cristalización dinámica, por lo general tomando como base la ecuación de Johnson-Mehl-Avrami o modificaciones de la misma.

En 1967 Ziabicki [45] estudiaba la cristalización dinámica del poly(butilen naftalen-2,6-dicarboxilato) proponiendo un comportamiento cinético semejante a una reacción de primer orden (Ec. 2.3)

$$\frac{d\theta}{dt} = K_{(r)}(1 - \theta) \quad [2.3]$$

donde θ es el grado de cristalinidad y $K(t)$ la constante de velocidad dependiente de la temperatura. En 1971, Ozawa [46] establece su modelo incluyendo la rapidez de enfriamiento (Ec. 2.4)

$$-\ln[1 - \theta_{(T)}] = \frac{K_{(T)}^*}{X^n} \quad [2.4]$$

donde $\theta_{(T)}$ es el grado de cristalinidad a la temperatura T, “X” es la rapidez de enfriamiento, $K_{(T)}^*$ es una función de cristalización por enfriamiento y n es el equivalente al exponente de Avrami. En 1972 Nakamura y Col. [47], proponen el modelo representado por la Ec. 2.5

$$\theta(t) = 1 - \exp \left[- \left(\int_{t'}^t K_{(T)} dt' \right)^n \right] \quad [2.5]$$

en esta ecuación la fracción de material cristalino (θ) es función del tiempo “t” relativo al tiempo de formación de un núcleo estable (t'). De la misma forma otros investigadores como Dietz [48] y Harnisch et. al. [49] han propuesto otros modelos, sin embargo el modelo más utilizado es el de Ozawa.

En este punto, la observación más importante es que en toda cristalización dinámica, la función de cambio del tamaño y cantidad de cristales formados es siempre una función creciente, la única manera de que esto no fuese así es que el tratamiento térmico incluyese un calentamiento.

2.2.- Modelado y simulación de cristalización

El modelado y simulación de la cristalización de polímeros generalmente se ha circunscrito a la aplicación de la ecuación de Avrami (Ec. 2.2), los modelos matemáticos han sido variados, desde modelos de Potts [50] donde se resuelve una función de partición exponencial, seguimiento de fronteras [51], de Montecarlo [52], de campo de fases [53] y autómatas celulares [54].

En 1999 Castro M y Col.[55] plantean un modelo muy sencillo y con poca fundamentación teórica partiendo de un entramado bidimensional y aplicando un proceso estocástico para el que se establecen reglas empíricas para la formación y crecimiento de los núcleos, como resultado final los cristales tienen formas geométricas caprichosas, la distribución de tamaños granulares coincide con lo predicho por un modelo basado en Potts [52], aunque se observan desviaciones de la teoría de Avrami.

En el año 2000, Hopkin y Bhadeshia [56] establecen un modelo basado en la Ec. de Avrami, modificándola para incluir el proceso de nucleación en las fronteras de cristales ya existentes, evitando tener como variable la característica térmica o atérmica de la nucleación, el modelo obtenido lo aplican a la recristalización de acero inoxidable 302, calculando exclusivamente tamaños de grano y fracción volumen de material cristalizado en función del tiempo. En dicho artículo no es posible apreciar la consistencia entre los resultados experimentales y las conclusiones del trabajo.

Holm y Battaile en el año 2001 [57] hacen simulaciones bi y tridimensionales de la evolución de los tamaños de grano utilizando un procedimiento que incluye el método de Montecarlo y el modelo de simulación de Potts, sin embargo se observa que desde el inicio las fronteras intercristalinas son de formas diversas (Fig. 2.5), no se reporta ninguna comparación con resultados experimentales.

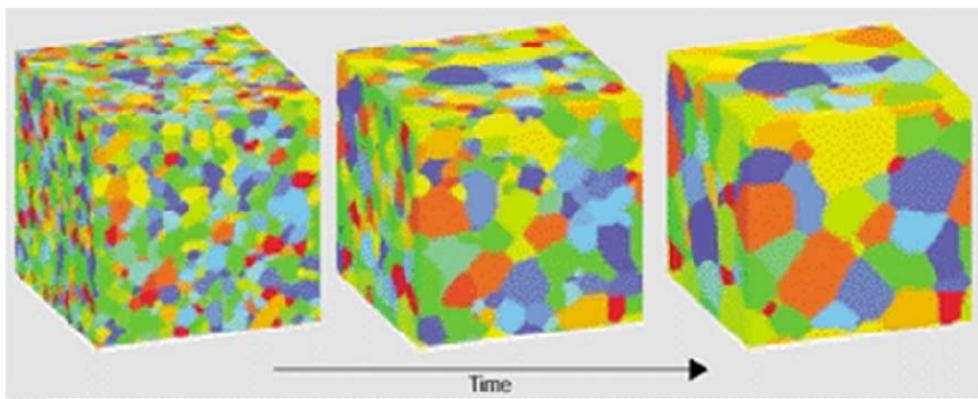


Figura 2.5. Simulación mediante el modelo de vértices del crecimiento policristalino isotrópico [57]

En el año 2002, Burger y Col. [58] desarrollan un modelo estocástico basado en la Ec. de Avrami, con marcaje de puntos, orientado fundamentalmente a analizar el efecto de las variaciones en temperatura, sobre la rapidez de cristalización. Utiliza resultados experimentales reportados por otros autores para la rapidez de crecimiento y nucleación en polipropileno, pero no compara los resultados de las curvas de grado de cristalinidad vs. tiempo obtenidas bajo este planteamiento.

En este mismo año, Gránásy y Col. [59], aplican la reciente teoría de campo de fases binaria a la solidificación de una aleación de Cu-Ni, concluyendo que el exponente de Avrami cambia con el grado de transformación (Fig. 2.6). Se denotará en esta figura el exponente de Avrami como p y n es el número de núcleos. Discute cualitativamente (mostrando imágenes, Fig. 2.7), las geometrías de los cristales en función de la anisotropía de la muestra. En esta última figura puede observarse que en las geometrías finales las fronteras intergranulares tienen cierta curvatura, característica no discutida en el documento.

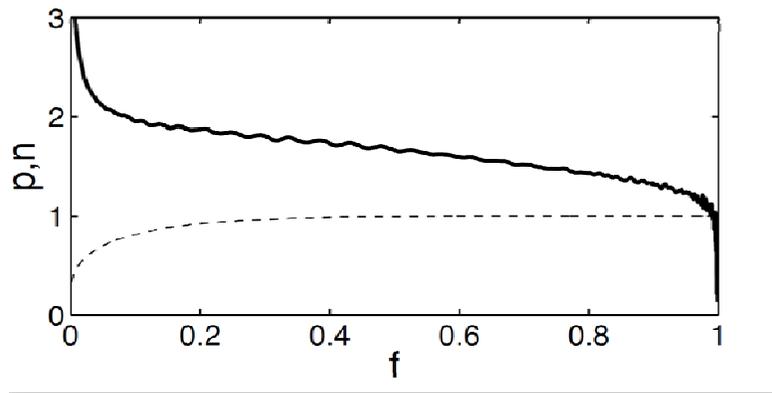


Figura 2.6. El exponente de Avrami (p) y el número de núcleos normalizados al máximo número obtenido ($n = N/N_{\max}$) según el modelo de campo de fases binario [59].

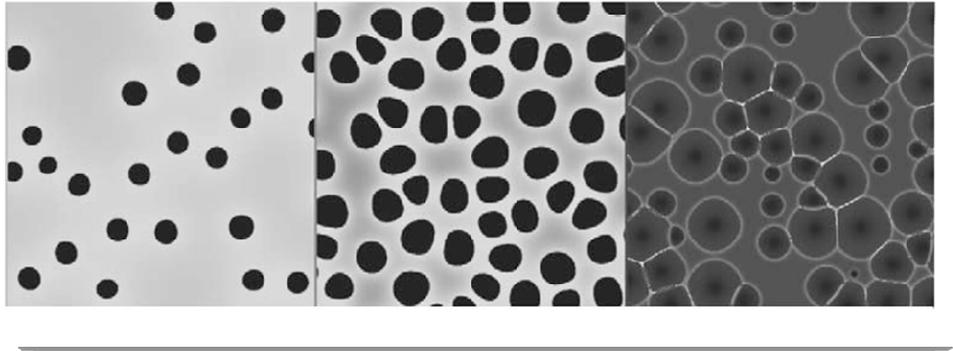


Figura 2.7. Variaciones en la forma del sólido en función del grado de conversión (X_c), según el modelo de campo de fases binario en condiciones isotrópicas, de izquierda a derecha $X_c=0.2, 0.5$ y 0.8 [59].

En el año 2005 Jun y Col. [60] reportan un modelo “generalizado” unidimensional basado en el modelo de Avrami, con la particularidad de considerar núcleos fantasma, es decir que se pueden sobreponer en el espacio, el objetivo era simular la réplica del DNA y reportan una buena concordancia con resultados experimentales.

Yang y Col. [61], ese mismo año, aplican el modelo de Avrami, empleando la aproximación de la capilaridad, la cual considera la suma de las energías libres de formación y de superficie de núcleos esféricos en el modelo cinético de cristalización de Gibbs – Thompson (ver referencias 22 y 28), para plantear un modelo que debido a su complejidad, se resuelve numéricamente, reinterpretando el exponente de Avrami como en función de un pre-factor de una función de nucleación, además compara los resultados de la simulación con Nylon-6, cuyas gráficas de Avrami parecen variar con el tiempo de cristalización. Aunque se reporta como satisfactorio, no incluye las geometrías de los cristales.

En el año 2005 Gránásy y Col. simularon la nucleación y crecimiento de esferulitas [62] usando el método de dinámicas de Langevin, el cual simula el efecto de un solvente individual en las moléculas del polímero, obteniendo morfologías esferulíticas, sin considerar en el modelo heterogeneidades dinámicas en una escala nanométrica.

Vanherpe y Col. en el año 2007 [63], reporta la simulación tridimensional aplicando la teoría de campo de fases, para resolverlo utiliza el método llamado “Box Bounding”. El

trabajo computacional requerido es suficientemente grande como para hacer simplificaciones importantes, como resultado, además de predecir las curvas de conversión –tiempo, muestran solo gráficamente las formas de los cristales (Fig. 2.8), apreciándose una alta irregularidad en las fronteras intergranulares.

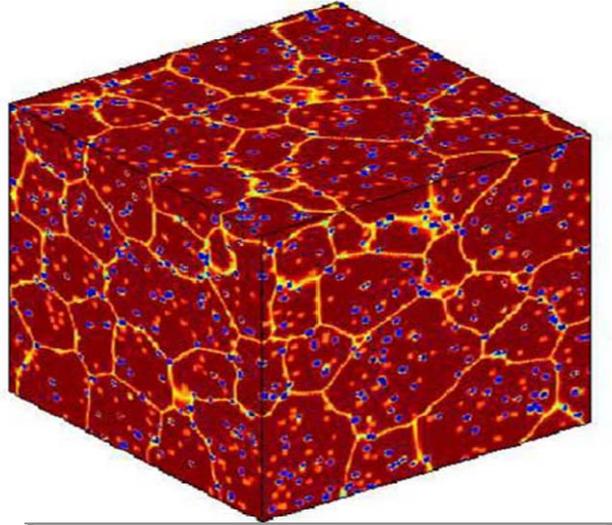


Figura 2.8. Resultado de la simulación tridimensional hecha por Vanherpe y Col.[63], mostrando la irregularidad de las fronteras.

Durante el año 2008, se reportan otras dos simulaciones utilizando la teoría de campo de fases [64,65]. En una de ellas [64], se aplica la teoría de Avrami a un modelo de campo de fases mediante un método de diferencias finitas llamado dinámica de celdas. Se predicen las isotermas de cristalización y su ajuste a datos experimentales es satisfactorio. La función de nucleación es constante y calcula los exponentes de Avrami (entre 2.74 y 3.44), e incluyen un parámetro para acotar la energía libre local llamado sobre-enfriamiento, del cual depende el exponente de Avrami. Resultando que el sobre-enfriamiento determina si “n” es directamente o inversamente proporcional a la temperatura de cristalización. Por su parte Moelans y Col. [65] emplean esta teoría tratando de acoplar diagramas de fases y termoquímica, las principales conclusiones son: lo gran consumidor de recursos computacionales que tiene el método, (memoria RAM y tiempo de cómputo) y la irregularidad de las fronteras intergranulares.

En el 2008, Canada y Col. [66], proponen una simulación mediante un autómata celular, modelando la formación de “cristales simples” en hierro. Es de destacarse que se simuló

la formación de granos (Fig. 2.9) y reportan buena concordancia con resultados experimentales de otros modelos, en cuanto a grados de cristalización en función del tiempo, las geometrías granulares parecen estar muy lejos de la realidad.

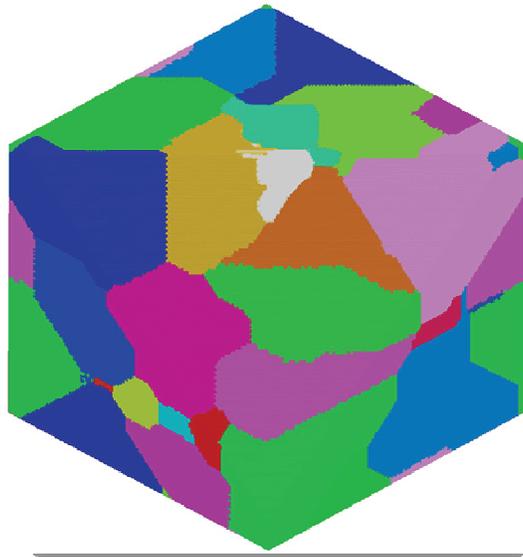


Figura 2.9. Microestructuras obtenidas mediante simulación utilizando un autómata celular para la recristalización del hierro [66].

En el año 2009, Dillon y Rohrer [67] proponen un modelo utilizando las topologías de las uniones entre granos, y el ángulo formado en puntos triples. Lo aplican a la solidificación de alúmina e itrio/aluminio dopados con magnesio y en realidad es un trabajo de correlación empírica de dichas variables, y consideraciones de energía de superficie.

En el 2009, Lehmann y Col.[68] obtienen mediante calorimetría diferencial de barrido(DSC) para el Tereftalato de Ciclobutileno (pCBT), bajo condiciones de cristalización isotérmica, el exponente de Avrami con valores entre 2.06 y 2.25 a temperaturas de cristalización entre 190 y 195 °C y para el Poli(Tereftalato de Butileno) (PTB) entre 3.69 y 4.03 a las mismas temperaturas de cristalización.

Por otra parte, el estudio de la geometría de fronteras intercristalinas de tipo hiperbólico en polímeros, analizado en dos dimensiones, tiene algunos antecedentes fundamentales que describiremos a continuación.

En los años 1980 y 1981, Galeski y Col. [69,70] describen la forma hiperbólica de las fronteras interesferulíticas y su relación con los parámetros cinéticos de rapidez de crecimiento, así como la diferencia entre tiempos de formación de dos núcleos contiguos y la curvatura de las hipérbolas. Reportan además los histogramas de la fracción de núcleos formados teniendo como referencia el producto ΔtGr , siendo éste un tiempo modificado, así como la distribución de tamaños de esferulitas. Concluyen un comportamiento relativamente acorde con Avrami con el exponente entre valores de 2.07 para nucleación atérmica y 2.55 para nucleación térmica. No se muestran dichas curvas que permiten obtener estos resultados en este artículo.

Siguiendo con esta línea perspectiva, a partir del año 2002, González y Col. [71,72] desarrollaron un modelo de nucleación y crecimiento unidimensional bajo condiciones de nucleación térmica y atérmica y parcialmente térmica, cuya solución analítica es relativamente sencilla, reportando exponentes de Avrami entre 1.7 y 2.29 al incrementar el número de núcleos formados térmicamente (Fig. 2.10).

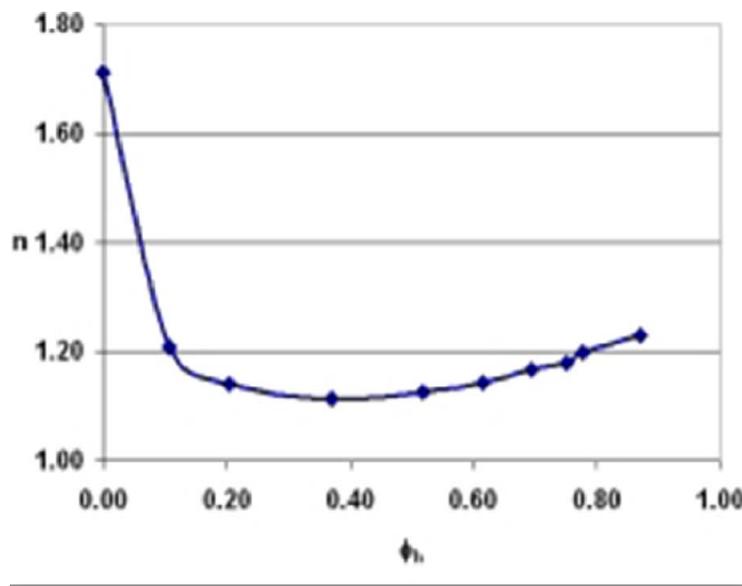


Figura 2.10. Exponente de Avrami calculado en función del número de núcleos formados atérmicamente, expresado en fracción de núcleos iniciales (Φ_h). [68]

Se reporta también la necesidad de hacer con el simulador una cantidad grande de experimentos para lograr una buena precisión, así como una relativa baja sensibilidad al número de núcleos formados atómicamente, lo cual se puede apreciar fácilmente en la gráfica de la figura 2.10.

En el año 2005, González y Col.[73] reportan la aplicación del modelo de las fronteras hiperbólicas al análisis de una muestra de polipropileno, mostrando su utilidad en la determinación de la relación entre la rapidez de nucleación y la de crecimiento.

Basándose en estos antecedentes, en este trabajo se propone desarrollar una simulación mediante computadora, acerca del modelo de las fronteras hiperbólicas, resuelto analíticamente y que permita la determinación de parámetros cinéticos de los fenómenos de nucleación y crecimiento, así como el análisis del exponente de Avrami.

3.- Hipótesis y Objetivos

3.1.- Hipótesis

Es posible desarrollar un simulador del fenómeno de nucleación y crecimiento de nucleación térmica, en base al modelo de fronteras hiperbólicas, que se resuelva analíticamente y sirva como base para la determinación de parámetros cinéticos y el análisis del exponente de Avrami.

3.2.- Objetivo general

Desarrollar un simulador de nucleación y crecimiento para nucleación térmica basado en el concepto de fronteras hiperbólicas mediante el análisis matemático-geométrico, que se resuelva analíticamente para hacer un análisis estadístico del fenómeno y determinar parámetros cinéticos.

4.- Metodología

4.1.- Deducciones

Como primer paso está la deducción del modelo matemático, que aunque ya se han reportado [65] las ecuaciones finales, es imprescindible asimilar los conceptos físicos que intervienen, así como facilitar las manipulaciones matemáticas requeridas en el desarrollo del simulador. En seguida se establecen las ecuaciones relativas a las intersecciones entre dos hipérbolas, entre una hipérbola y una recta, y el área de los núcleos en crecimiento.

4.2.- Diagrama de bloques y programación

Una vez teniendo todas las funciones matemáticas requeridas, se desarrolla el diagrama de bloques, incluyendo funciones de generación de posiciones y tiempo de núcleos, traslación y rotación de ejes coordenados, que permitan la presentación gráfica y numérica de los resultados. Incluyendo además la facilidad de efectuar corridas repetidas al menos 100 veces para poder hacer un análisis estadístico de los resultados, el cual se hace utilizando un software comercial. La formulación del programa es basado en el diagrama mencionado, lo que permitió el desarrollo del algoritmo y su posterior codificación. La programación del simulador se hace en Pascal de Borland.

4.3.- Experimentos de la simulación

Todas las simulaciones se hicieron con una rapidez de crecimiento constante e igual a 0.1, en relación con la rapidez de nucleación (G_n), se pueden dividir en tres conjuntos de simulaciones, siempre haciendo 100 repeticiones para cada conjunto de variables de entrada. La primera, donde se pretendió familiarizarse con los efectos que tiene exclusivamente G_n sobre los resultados de simulación, utilizándose así 13 magnitudes de G_n independientes del tiempo y del área residual. El segundo conjunto de simulaciones, se considero una G_n como función asintótica del tiempo (Ec. 6.2) donde

se hicieron 34 simulaciones (cada una con 100 repeticiones). En el tercer grupo se consideró G_n constante con respecto al área disponible para formarse, para este caso fueron 8 valores de G_n diferentes.

5.- El modelo

5.1.- Análisis de geometría de fronteras

Representamos dos núcleos como esferas en crecimiento con una función de rapidez ó de crecimiento del radio en términos del tiempo $F(t)$, colocadas sobre el eje x de las coordenadas rectangulares y equidistantes al origen (Fig. 5.1), le añadimos como una cuarta coordenada el tiempo de formación del núcleo, t_1 para el núcleo de la izquierda y t_2 para el de la derecha (núcleos 1 y 2 respectivamente), con $t_1 > t_2$.

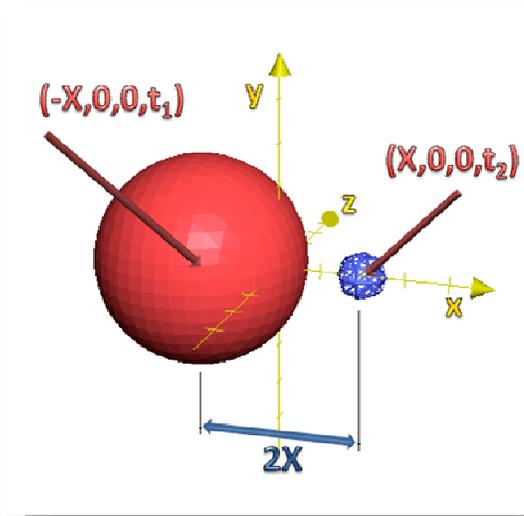


Figura 5.1. Coordenadas de los núcleos 1 y 2 en el espacio y tiempo.

Suponiendo que ambos núcleos crecen con la misma rapidez (G_r), para cada momento los radios están representados por las ecuaciones:

$$r_1 = F(t) - F(t_1) \quad [5.1]$$

$$r_2 = F(t) - F(t_2) \quad [5.2]$$

Las ecuaciones de los núcleos como esferas son:

$$(x + X)^2 + y^2 + z^2 - r_1^2 = 0 \quad [5.3]$$

$$(x - X)^2 + y^2 + z^2 - r_2^2 = 0 \quad [5.4]$$

Substituyendo [5.1] y [5.2] en [5.3] y [5.4] respectivamente, encontramos las ecuaciones [5.5] y [5.6] que representan las esferas en función del tiempo.

$$(x + X)^2 + y^2 + z^2 - [F(t) - F(t_1)]^2 = 0 \quad [5.5]$$

$$(x - X)^2 + y^2 + z^2 - [F(t) - F(t_2)]^2 = 0 \quad [5.6]$$

Para encontrar la intercepción de los núcleos en cada momento, hacemos simultáneas con las ecuaciones [5.5] y [5.6], así, restando la Ec. [6] de la [5] se obtiene la siguiente ecuación:

$$(x + X)^2 - (x - X)^2 - [F(t) - F(t_1)]^2 + [F(t) - F(t_2)]^2 = 0 \quad [5.7]$$

Haciendo operaciones algebraicas y eliminando términos semejantes, (x^2 y $[F(t)]^2$), resulta:

$$4Xx + 2F(t_1)F(t) - 2F(t_2)F(t) - [F(t_1)]^2 + [F(t_2)]^2 = 0 \quad [5.8]$$

Factorizando la ecuación anterior se obtiene:

$$4Xx - 2[F(t_2) - F(t_1)]F(t) + \{[F(t_2)]^2 - [F(t_1)]^2\} = 0 \quad [5.9]$$

Despejando "F(t)", en la ecuación [5.9] se obtiene la siguiente expresión matemática para F(t):

$$F(t) = \frac{4Xx + \{[F(t_2)]^2 - [F(t_1)]^2\}}{2[F(t_2) - F(t_1)]} \quad [5.10]$$

Considerando la diferencia de cuadrados, lo que simplifica la expresión y reorganizando términos se obtiene:

$$F(t) = \frac{2Xx}{F(t_2) - F(t_1)} + \frac{F(t_2) + F(t_1)}{2} \quad [5.11]$$

Substituyendo [5.11] en [5.5] para obtener la ecuación de la frontera inter-esferulítica resulta:

$$(x + X)^2 + y^2 + z^2 - \left[\frac{2Xx}{F(t_2) - F(t_1)} + \frac{F(t_2) + F(t_1)}{2} - F(t_1) \right]^2 = 0 \quad [5.12]$$

Haciendo operaciones llegamos a la siguiente expresión:

$$x^2 + 2Xx + X^2 + y^2 + z^2 - \left[\frac{2Xx}{F(t_2) - F(t_1)} + \frac{F(t_2) - F(t_1)}{2} \right]^2 = 0 \quad [5.13]$$

Desarrollando el binomio al cuadrado resulta:

$$x^2 + 2Xx + X^2 + y^2 + z^2 - \frac{4X^2x^2}{[F(t_2) - F(t_1)]^2} - \frac{4Xx[F(t_2) - F(t_1)]}{2[F(t_2) - F(t_1)]} - \frac{[F(t_2) - F(t_1)]^2}{4} = 0 \quad [5.14]$$

Haciendo operaciones en el séptimo término de [5.14], éste se cancela con el segundo de dicha ecuación, obteniéndose como resultado la siguiente ecuación:

$$x^2 - \frac{4X^2x^2}{[F(t_2) - F(t_1)]^2} + X^2 + y^2 + z^2 - \frac{[F(t_2) - F(t_1)]^2}{4} = 0 \quad [5.15]$$

Factorizando y pasando términos con solo constantes al lado derecho

$$\left\{ 1 - \frac{4X^2}{[F(t_2) - F(t_1)]^2} \right\} x^2 + y^2 + z^2 = \frac{[F(t_2) - F(t_1)]^2}{4} - X^2 \quad [5.16]$$

Dividiendo ambos lados por el término de la derecha

$$\left\{1 - \frac{4X^2}{[F(t_2) - F(t_1)]^2}\right\}x^2 + \frac{y^2}{\frac{[F(t_2) - F(t_1)]^2}{4} - X^2} + \frac{z^2}{\frac{[F(t_2) - F(t_1)]^2}{4} - X^2} = 1 \quad [5.17]$$

Si en la ecuación [5.17] hacemos que

$$V = \frac{F(t_2) - F(t_1)}{2} \quad [5.18]$$

Entonces, se obtiene el siguiente resultado

$$\left\{1 - \frac{X^2}{V^2}\right\}x^2 + \frac{y^2}{V^2 - X^2} + \frac{z^2}{V^2 - X^2} = 1 \quad [5.19]$$

Finalmente, simplificando, llegamos a la ecuación [5.20]

$$\frac{x^2}{V^2} - \frac{y^2}{X^2 - V^2} - \frac{z^2}{X^2 - V^2} = 1 \quad [5.20]$$

que es un hiperboloide con vértice en el punto $(V,0,0)$. Figura 5.2

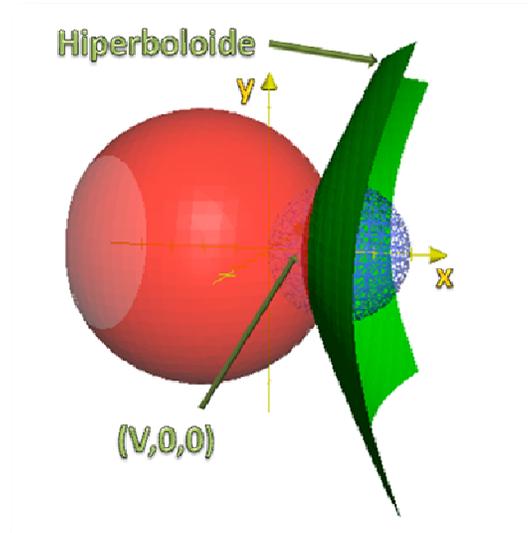


Figura 5.2. Se muestra la superficie representada por la Ec. 5.20, hiperboloide con vértice en $(V,0,0)$.

CASO PARTICULAR I. Nucleación térmica en condiciones isotérmicas

En este caso particular la función F es una constante que representa la rapidez de crecimiento (Gr), entonces:

$$V = \frac{Gr(t_2 - t_1)}{2} \quad [5.21]$$

La frontera sigue siendo un hiperboloide cuyo vértice es el punto $(V,0,0)$.

CASO PARTICULAR II (Atérmico, con función de crecimiento F)

Si ambos núcleos se forman al mismo tiempo, la Ec. 5.7 queda

$$(x + X)^2 - (x - X)^2 - [F(t) - F(t_1)]^2 + [F(t) - F(t_1)]^2 = 0 \quad [5.22]$$

Eliminando términos semejantes y haciendo operaciones queda:

$$x^2 + 2Xx + X^2 - x^2 + 2Xx - X^2 = 0 \quad [5.23]$$

Por lo que:

$$x = 0 \quad [5.24]$$

La frontera resulta ser el plano perpendicular al eje x y que pasa por el origen (Fig. 5.3).

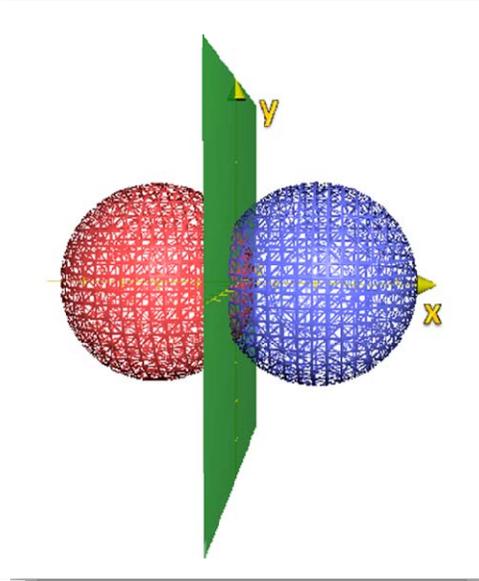


Figura 5.3. El plano $x=0$ como frontera en nucleación atómica

CASO PARTICULAR III. Nucleación térmica, en función de crecimiento (F), en dos dimensiones

En el caso de círculos que crecen equidistantes al origen sobre el eje x, toda la demostración desde la Ec. 5.1 hasta la 5.21 son válidas solo haciendo para el caso particular de $z = 0$, por lo que la ecuación que define la frontera es una hipérbola (Fig. 5.4), cuya ecuación es la Ec. 5.25.

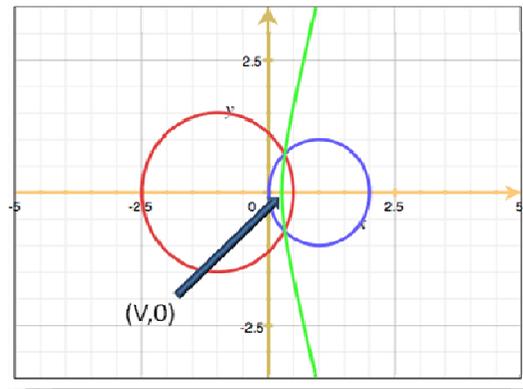


Figura 5.4. Hipérbola como frontera de dos núcleos formados en diferentes momentos (Ec. 5.25).

$$\frac{x^2}{V^2} - \frac{y^2}{X^2 - V^2} = 1 \quad [5.25]$$

donde V sigue estando definida por la ecuación [5.18]

CASO PARTICULAR III (Térmico, 2D, función Gr)

En el caso de que la función de crecimiento sea constante entonces V está definido por la Ec. [5.21].

CASO PARTICULAR IV (Atérmico, 2D, función F o Gr)

Cuando los núcleos se forman al mismo tiempo, la Ec. [5.24] es válida, por lo que la frontera es el eje y . (Fig. 5.5).

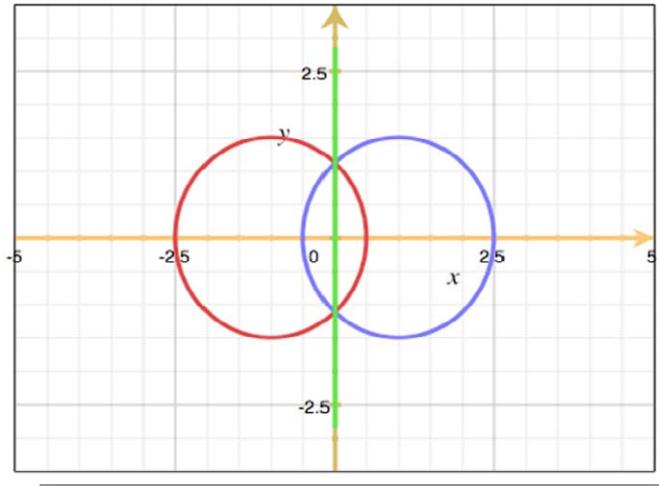


Fig. 5.5. El eje “y” como frontera de dos núcleos formados al mismo tiempo (nucleación atérmica).

5.2.- Determinación de los límites de las fronteras

Para definir los límites del cristal es necesario considerar dos casos (Fig. 5.6), primero cuando la frontera bajo estudio tiene como límites exclusivamente las intersecciones de las circunferencias de los núcleos y segundo cuando la frontera ha detenido su crecimiento al chocar con otra frontera vecina, considerando un tercer núcleo.

Para el primer caso, la intersección (x_i) al tiempo t utilizamos la Ec. 5.9 re-arreglándola para quedar como:

$$Xx_i - \frac{[F(t_2) - F(t_1)]F(t)}{2} + \frac{[F(t_2)]^2 - [F(t_1)]^2}{4} = 0 \quad [5.26]$$

Substituyendo la Ec. 5.18 en 5.26 nos da:

$$Xx_i - VF(t) + \frac{[F(t_2) + F(t_1)]V}{2} = 0 \quad [5.27]$$

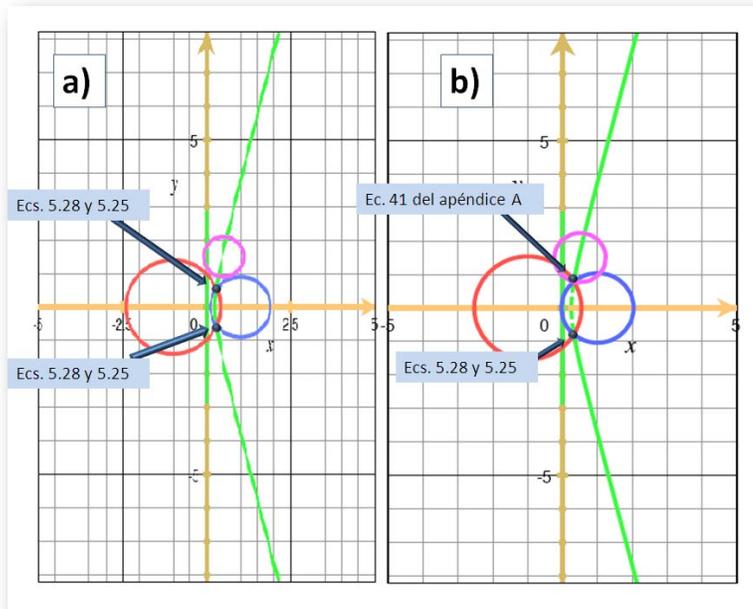


Figura 5.6.- Los dos casos para la determinación de límites de las fronteras, caso a) las fronteras quedan definidas por la intersección de dos círculos, caso b) una de las fronteras está definida como en el caso 1 y la otra por la intersección de tres círculos.

Reagrupando podemos fácilmente llegar a:

$$x_i = \frac{V}{2X} [F(t) - F(t_2) + F(t) - F(t_1)] \quad [5.28]$$

Comparando con las Ecs. 5.1 y 5.2 podemos escribir:

$$x_i = \frac{V}{2X} [r_1 + r_2] \quad [5.29]$$

Una vez calculado el valor de x_i de la intersección, los valores de y ($\pm y_i$) se calculan mediante la ecuación de la hipérbola (Ec. 5.25).

En el segundo caso (Fig. 5.6b), se tiene que obtener la intersección de la frontera con los tres círculos, considerando que ya tenemos la ecuación de la hipérbola, por ser este cálculo un procedimiento algebraico relativamente largo y tedioso, se presenta en el apéndice A al final de la tesis. Se usa la Ec. 41 de este apéndice para hacer los cálculos

del tiempo al que se intersectan los tres círculos, y de allí el cálculo de x_i , y_i mediante las Ecs. 5.28 y 5.29.

5.3.- Cálculo de áreas

El cálculo del área de un cristal antes de que choque con otro núcleo es simplemente el área del círculo, una vez que ha chocado el cristal tiene que dividirse desde su centro en ángulos que separen los conos (regiones en que la frontera es la circunferencia) y aquellas regiones donde la frontera es una hipérbola (Fig. 5.7), esta frontera puede poseer concavidad hacia el centro del núcleo o hacia afuera.

En el caso del cono, el área de éste será igual a la fracción de ángulo entre los límites del cono y los 360° multiplicado por el área del círculo (Ec. 5.30).

$$A_c = \pi r^2 \left(\frac{\phi}{\pi} \right) = \phi r^2 \quad [5.30]$$

Mientras que en el caso de las secciones donde la frontera es una hipérbola, cuando ésta tiene concavidad hacia afuera, (Fig. 5.7), el área sombreada en verde (A_1), será igual a la integral sobre el eje Y de la ecuación de la hipérbola (Ec. 5.25) más el valor de X, como resultado tenemos la Ec. 5.31.

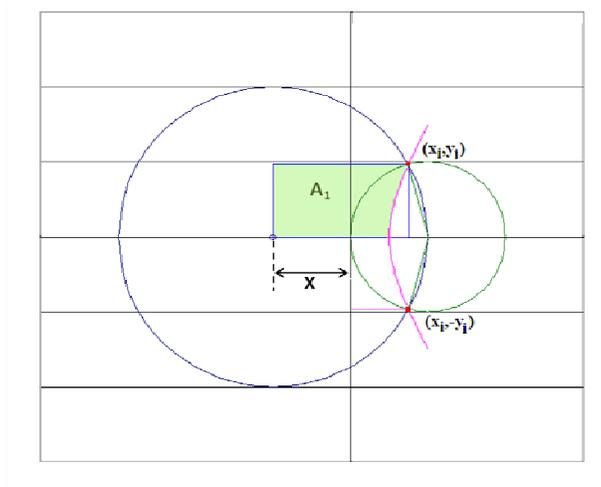


Figura 5.7. Delimitación del área A_1

Partimos de la Ec. 5.25, de la que despejamos x quedando:

$$A_1 = \int_{-y_i}^{y_j} \left[\sqrt{V^2 + \frac{V^2 y^2}{X^2 - V^2}} + X \right] dy \quad [5.31]$$

Integrando 5.31 mediante las técnicas de sustitución, fracciones parciales e integrandos racionales queda la Ec. 5.32

$$A_1 = Xy_i + \frac{1}{2} y_i \sqrt{V^2 + \frac{V^2 y_i^2}{X^2 - V^2}} + \frac{1}{2} V \sqrt{X^2 - V^2} \ln \left(y_i \sqrt{\frac{V^2}{X^2 - V^2}} + \sqrt{V^2 + \frac{V^2 y_i^2}{X^2 - V^2}} \right) - \frac{1}{2} V \sqrt{X^2 - V^2} \ln(V) \quad [5.32]$$

Simplificando resulta 5.33

$$A_1 = Xy_i + \frac{V}{2\sqrt{X^2 - V^2}} \left\{ y_i \sqrt{X^2 - V^2 + y_i^2} + \sqrt{X^2 - V^2} \left[\ln \left(\frac{1}{\sqrt{X^2 - V^2}} \left[y_i + \sqrt{X^2 - V^2 + y_i^2} \right] \right) \right] \right\} \quad [5.33]$$

definiendo:

$$c = X^2 - V^2 \text{ y } d = (c + Y_i^2)^{1/2} \quad [5.34]$$

la expresión 5.33 se transforma en la siguiente expresión matemática:

$$A_1 = Xy_i + \frac{V}{2c^{1/2}} \left\{ y_i d + c \ln \left| \frac{y_i + d}{c^{1/2}} \right| \right\} \quad [5.35]$$

restándole al área A_1 , la del triángulo indicado en la Fig. 5.8, calculamos el área definida como A_h , siendo la expresión la siguiente:

$$A_h = Xy_i + \frac{V}{2c^{1/2}} \left\{ y_i d + c \ln \left| \frac{y_i + d}{c^{1/2}} \right| \right\} - \frac{1}{2} (y_i x_i) \quad [5.36]$$

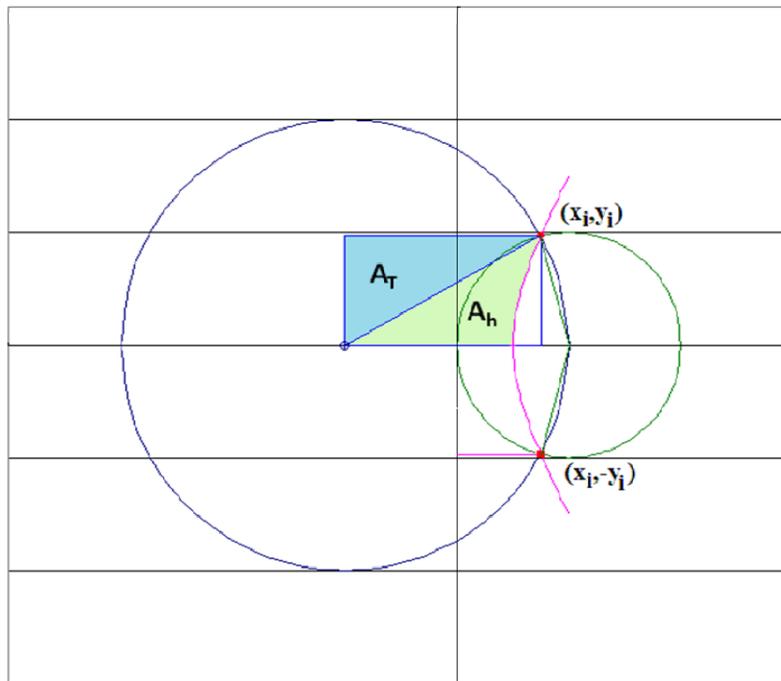


Figura 5.8. Área del semicono truncado por la hipérbola (A_h)

que reagrupando toma la forma:

$$A_h = \frac{V}{2c^{1/2}} \left\{ y_i d + c \ln \left| \frac{y_i + d}{c^{1/2}} \right| \right\} - \frac{1}{2} (x_i + 2X) y_i \quad [5.37]$$

Para calcular el área de la sección limitada por la hipérbola, pero con la concavidad hacia adentro, " A'_h " (Fig. 5.9), se resta al área del triángulo ABC, el área A_h , entonces esta área queda definida por la Ec. 5.38.

$$A'_h = X y_i - \frac{V}{2c^{1/2}} \left\{ y_i d + c \ln \left| \frac{y_i + d}{c^{1/2}} \right| \right\} - \frac{1}{2} (x_i + 2X) y_i \quad [5.38]$$

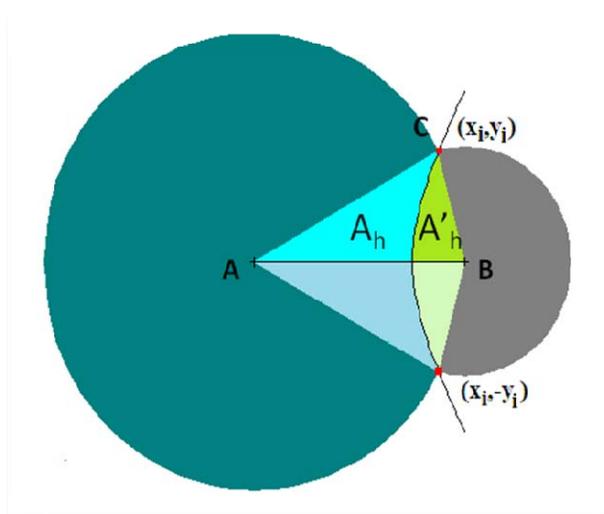


Figura 5.9 Definición del Área A'_h

que simplificando resulta:

$$A'_h = -\frac{V}{2c^{1/2}} \left\{ y_i d + c \ln \left| \frac{y_i + d}{c^{1/2}} \right| \right\} - \frac{1}{2} x_i y_i \quad [5.39]$$

De esta forma, el cálculo de las áreas entre el eje x, la hipérbola y la recta desde el centro del núcleo hasta el límite de la hipérbola, queda definido, ya sea por la Ec. 5.37 o 5.39 según la dirección de la concavidad

6.- El simulador

En el apéndice B se reporta el programa en Borland Pascal del simulador el cual consiste de 10,000 líneas aproximadamente, en seguida se hace una descripción de los aspectos más importantes del programa, considerando que procedimientos y funciones regularmente estándar como rotación y traslación, no son necesario explicitar.

6.1.- Aspectos generales de la simulación.

Se considera como muestra una región bidimensional cuadrada que mide una unidad de medida de longitud (arbitraria), por lado y que está localizada en el primer cuadrante del sistema cartesiano 2D. (Figura 1).

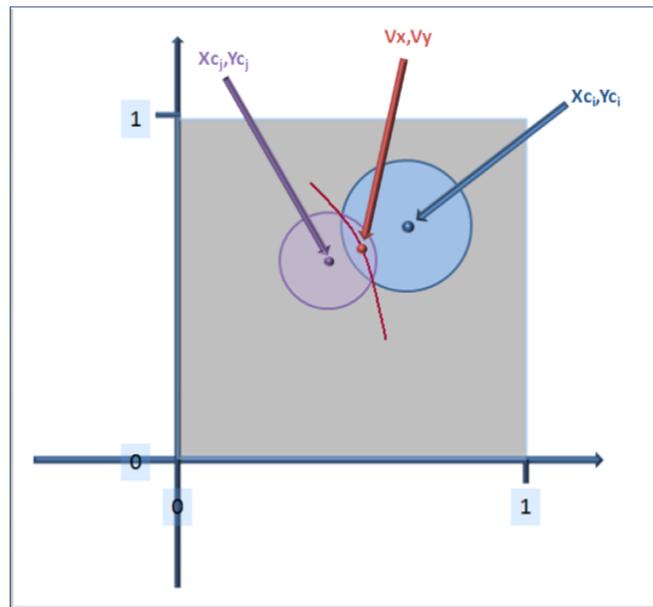


Figura 6.1.- Forma de la muestra virtual mostrando la posición de dos núcleos con centros en X_{c_i}, Y_{c_i} y X_{c_j}, Y_{c_j} , y una frontera hiperbólica con vértice en V_x, V_y

Entre los diferentes tipos de variables definidas en el simulador, destacan las correspondientes a los núcleos y a las fronteras o hipérbolas, estas se describen en seguida:

Núcleos:

Nu : Array[1..NuMax] of :

Ni	: Word	{Identificador del núcleo}
Pc	: Array [1..2] of Extended	{Coordenadas del centro}
t0	: Extended	{Tiempo de formación}
N	: Word	{Número de fronteras}
id	: Array[1..NFron] of Word	{ Identificador de fronteras asociadas}

Fronteras:

Hpp: Array[1..FrMax] of

idh	: Word	{Identificador de la hipérbola}
Nuc	: Array[1..2] of Byte	{Par de núcleos correspondientes}
V	: Extended	{Vértice sobre eje x}
Xm	: Extended	{Distancia media entre centros de núcleos}
A	: Extended	{Ángulo a girar para colocarla en muestra}
Tv	: Extended	{Tiempo al que se inicia su formación}
Pk	: Array[1..2] of Extended	{Constantes de hipérbola sobre eje X}
Po	: Array[1..2] of Extended	{Posición del centro (x,y)}
Vr	: Array[1..2] of Extended	{Posición del vértice (x,y)}
Lim	: Array[1..2,1..2] of Extended	{Límites de la hipérbola posicionada sobre eje x}
An1	: Array[1..3] of Extended	{Ángulos de los límites de hipérbola y el vértice posicionada sobre eje x}
HipRec:	Boolean	{Indica si es hipérbola o recta}
Split	: Boolean	{Indica si se dividió en dos partes}

Es conveniente aclarar que algunas de las variables del tipo Hpp son hasta cierto punto redundantes, por ejemplo el vértice “V” que es la distancia entre el núcleo que se formó

primero y el vértice, corresponde a las coordenadas de dicho vértice una vez posicionada la frontera en la muestra, es decir una vez girada y trasladada desde su posición sobre el eje x con distancias equidistantes a los núcleos hasta su posición en la muestra virtual. Otras variables como An_1 y Po solo sirven para efectuar operaciones que son muy repetitivas durante la programación.

Considerando que los datos duros de las fronteras son los correspondientes a cuando las hipérbolas están posicionadas sobre el eje x (Fig. 6.2), mientras que realmente están sobre la muestra virtual, y que por sencillez en cálculos es necesario en muchas ocasiones representar alguna hipérbola en los ejes coordenados de otra frontera posicionada sobre el eje x, hay dos procedimientos que es conveniente describir, estos son: **Posiciona** que coloca puntos de una hipérbola desde su posición sobre el eje x, hasta su posición en la muestra virtual y **Reposiciona** que coloca puntos de una frontera que está en la muestra virtual, en los ejes coordenados correspondientes a otra hipérbola colocada sobre el eje x.

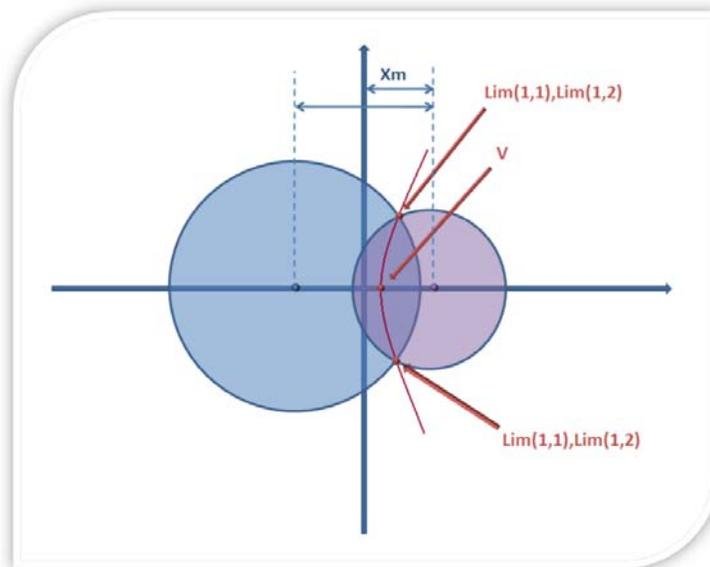
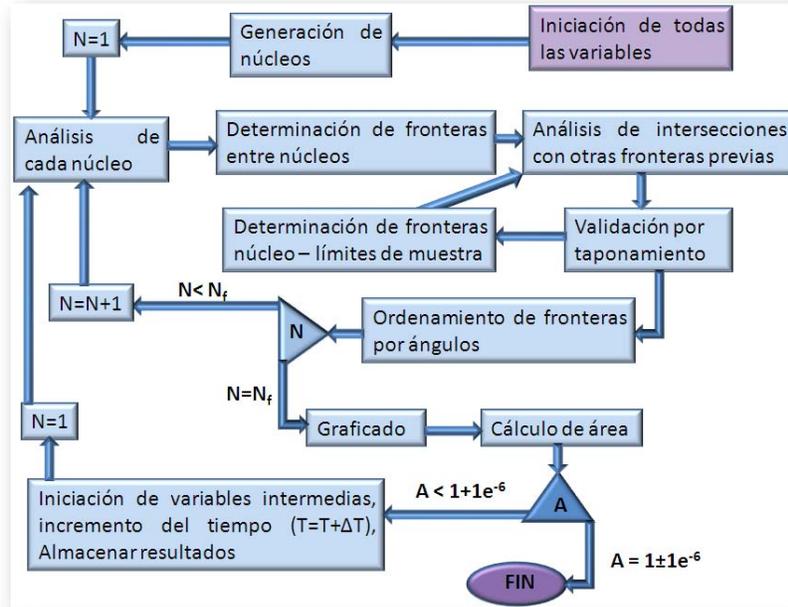


Figura 6.2.- Núcleos e hipérbola sobre el eje x mostrándose algunas variables de Hpp



Esquema 6.1.- Diagrama de bloques general

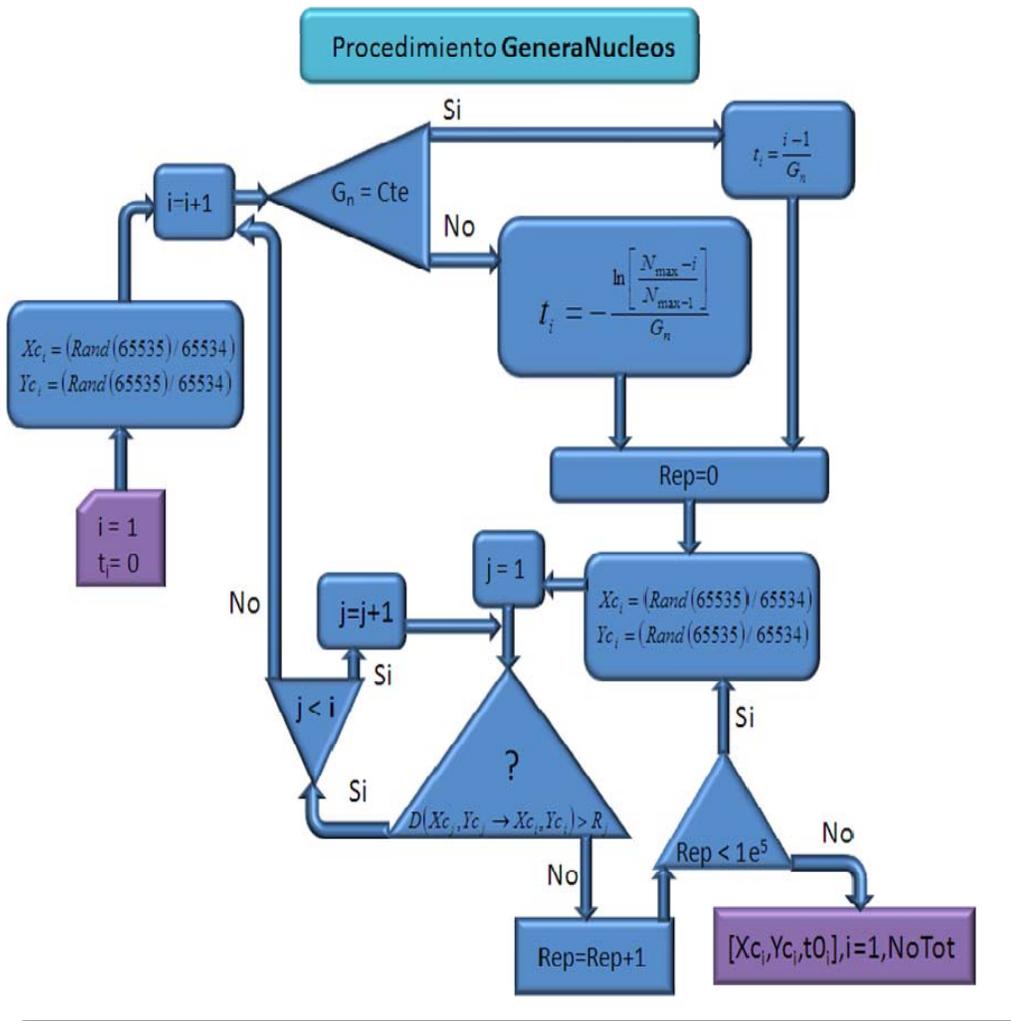
El esquema 6.1, muestra el diagrama de bloques general del modelo, en este se puede destacar la necesidad de explicitar el contenido de algunos bloques como los de generación de núcleos, determinación de fronteras, análisis de intersecciones y validación por taponamiento. Además es importante establecer que la simulación se da por terminada cuando el área cubierta por los núcleos es igual a 1 con un error de $\pm 1e^{-4}$.

6.2.- Procedimientos para la generación de núcleos

Habiendo seleccionado el tamaño de muestra de 1×1 , para cada coordenada de los centros de los núcleos se genera un número aleatorio entre 0 y 65534 (máximo valor para una variable tipo *word* y se divide entre 65534 redondeando el valor de las variables a 12 cifras significativas después del punto decimal. Como se verá, se desarrollaron tres algoritmos para la generación de núcleos, el primero y segundo (descritos en el esquema 6.2), consisten, el primero en la formación de núcleos con una rapidez de nucleación (G_n) constante e independiente del área descubierta por los núcleos (Ec. 6.1), el segundo mediante una función de rapidez de nucleación decreciente asintóticamente un número máximo de núcleos N_{max} , (Ec. 6.2).

$$t_i = \frac{i}{G_n} \quad [6.1]$$

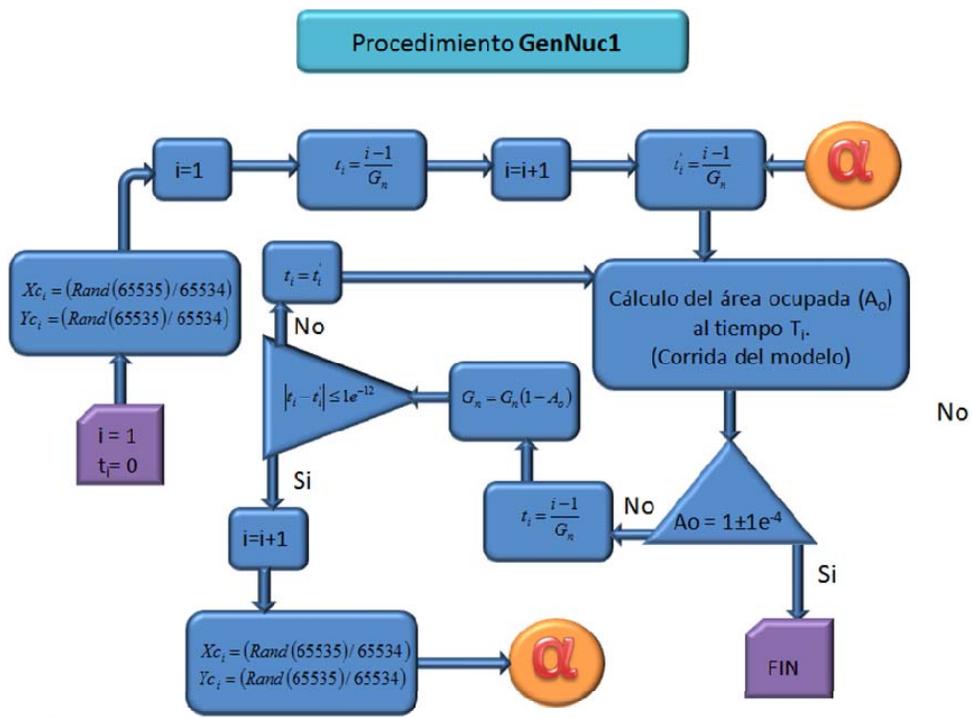
$$t_i = -\frac{\ln\left[\frac{N_{Max} - N_i}{N_{Max} - 1}\right]}{G_n} \quad [6.2]$$



Esquema 6.2.- Diagrama de bloques de la generación aleatoria de núcleos con rapidez constante y asínticos a N_{Max} .

Un tercer método empleado, fue considerar la rapidez de nucleación como constante respecto al área no cubierta por los núcleos previamente formados, para lo cual se utilizó un algoritmo iterativo como se describe en el esquema 6.3, aceptando la convergencia cuando el cálculo entre tiempos de formación del núcleo consecutivos difiere en menos de $1e^{-12}$.

Como se observa, el tiempo de formación de cada núcleo es definido ya sea por una función lineal (Ec. 1) o asintótica (Ec. 2), y la posición X_{c_i}, Y_{c_i} , mediante una función aleatoria donde se comprueba que al tiempo t_i la posición del núcleo no quede adentro de algún núcleo previamente formado. El número de núcleos generados queda definido por la ausencia de un área dentro de la muestra que no esté ocupada por otros núcleos

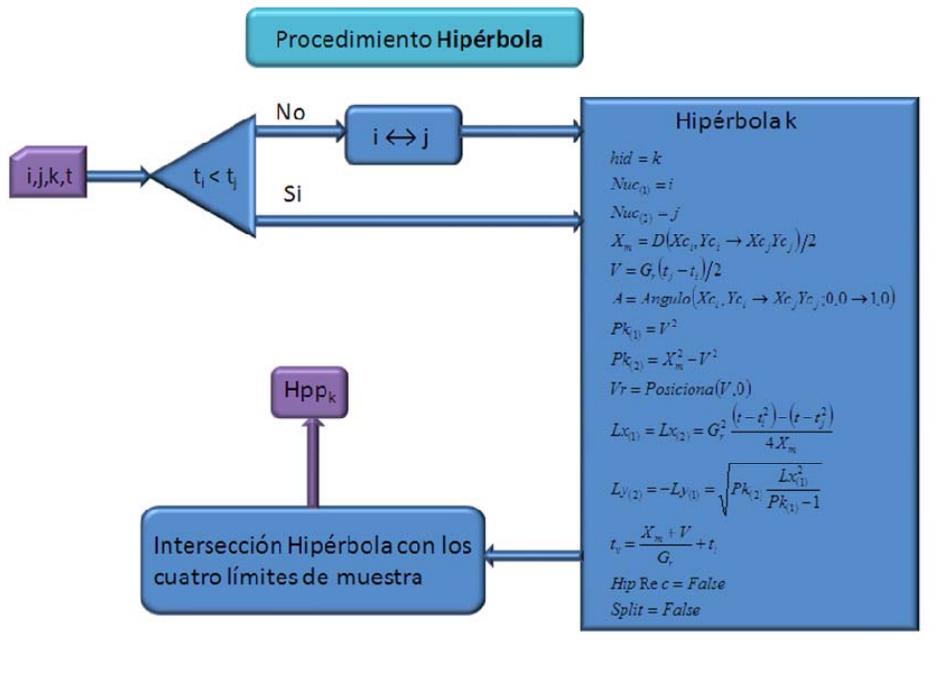


Esquema 6.3.- Diagrama de bloques de la generación aleatoria de núcleos con rapidez constante respecto al área no cubierta por los núcleos previos.

Como se observa en este esquema 6.3, para efectuar este procedimiento, es necesario tener terminado el programa principal del modelo, razón por la que éste se programó después de haber estructurado y probado el simulador con las formas anteriores de nucleación.

6.3.- Procedimiento para la determinación de fronteras entre núcleos

En el esquema 6.4 se muestra el diagrama de bloques al que se llamó *hipérbola*, en el cual se calculan las hipérbolas que formarían todas las parejas de núcleos entre sí, (todas las parejas i,j tal que $i \neq j$), para tal efecto se descartan aquellas fronteras cuyo tiempo de formación (o choque de núcleos), sea mayor al tiempo transcurrido. Las ecuaciones corresponden a las del modelo reportadas en el capítulo 5 ó en el apéndice A .

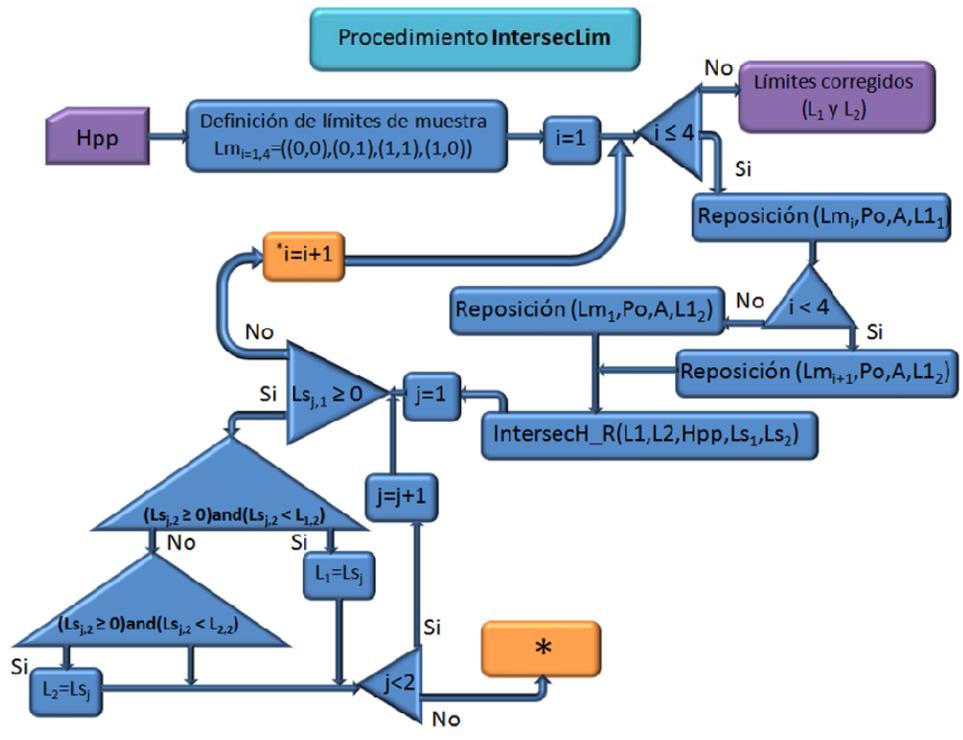


Esquema 6.4.- Diagrama de bloques de proceso de generación de fronteras.

El procedimiento presentado en este esquema como IntersecLim (intersección de la hipérbola con los límites de la muestra), se presenta en el esquema 6.5, este consiste en

la determinación de las intersecciones de las hipérbolas calculadas con las cuatro rectas que forman la muestra, el procedimiento identificado como REPOSICIÓN, consiste en rotaciones y traslaciones para colocar cada recta en los ejes coordenados correspondientes a la ecuación de la hipérbola bajo estudio. El procedimiento IntersecH_R precisamente determina la intersección entre la recta posicionada y la hipérbola.

El procedimiento para la determinación de las intersecciones entre dos hipérbolas, se definió siguiendo las ecuaciones presentadas en el punto 5.2, añadiendo además la validación por taponamiento.



Esquema 6.5.- Diagrama de bloques de proceso para la determinación de intersecciones de las fronteras con la muestra

6.4.- Procedimiento para la validación por taponamiento

Como un ejemplo del método para la validación por taponamiento entre fronteras, en la Figura 6.3 se muestra, con dos ejemplos, como se valida la existencia de una frontera. En dicha figura aparecen 5 fronteras esféricas, las hipérbolas 1 a 3 corresponden a fronteras que se habían generado previamente, si la frontera que estamos validando es la número 4, encontramos que no existe ninguna recta que parta del centro del núcleo, que llegue a la hipérbola bajo análisis y que no cruce alguna de las fronteras previamente formadas, caso en el cual no se valida la hipérbola, caso contrario es la frontera 5, donde hay un segmento (marcado en verde), en el que las rectas radiales no cruzan ninguna frontera previa, caso en el cual se valida la existencia de la hipérbola.

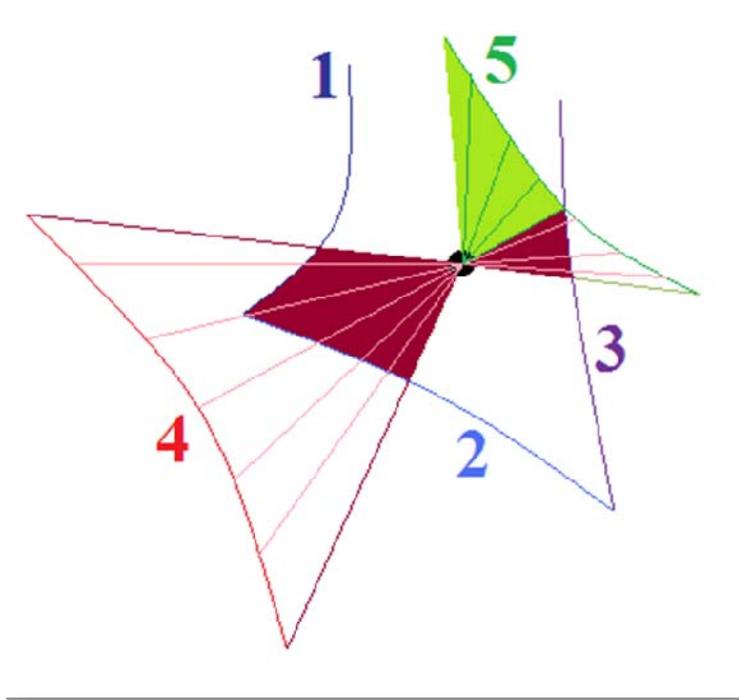


Figura 6.3.- Determinación de taponamiento de las fronteras 4 y 5 por las fronteras 1 a 3 previamente formadas. La frontera 4 se descarta, la frontera 5 se acepta.

Como con los algoritmos implementados se aceptan como válidas fronteras hiperbólicas que realmente no pueden ser visualizadas por encontrarse detrás de otros núcleos, por medio de la determinación de intersecciones de rectas entre el centro del núcleo y los límites de la hipérbola, a ser evaluada con las otras fronteras de dicho núcleo, se determina si la hipérbola está ocultada, en cuyo caso no es validada.

7.- Resultados de la simulación

7.1.- Rapidez de nucleación constante

La Figura 7.1 muestra seis pantallas de una simulación con rapidez de crecimiento esferulítico de $Gr = 0.1$ y de nucleación también constante de $Gn = 0.8$, los tiempos en que se tomaron dichas imágenes se reportan en la misma figura. Aquí se puede apreciar la aparición paulatina de núcleos y su crecimiento hasta llegar a cubrir el área total de la muestra. En lo que se alcanzan a apreciar, las fronteras hiperbólicas cuya están bien curvatura, siempre es con la concavidad hacia el núcleo que se formó más tardíamente. El número total de núcleos formados fue de $N = 6$.

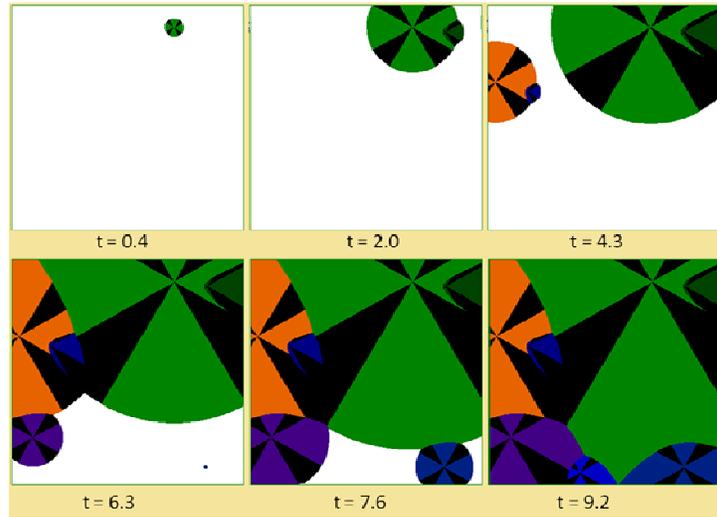


Figura 7.1. Seis imágenes de instantáneas correspondientes a diferentes tiempos en la simulación con $Gr = 0.1$ y $Gn = 0.8$

Ya que se hicieron 100 corridas en la figura 7.2 se muestran las gráficas de la fracción de área ocupada, a la que llamaremos grado de conversión o grado de cristalización, $(1 - \lambda)$ vs. tiempo; por analogía con el proceso de cristalización, les llamaremos isoterma de cristalización.

Como se puede apreciar, hay una gran variabilidad entre isoterma e isoterma, la gran mayoría de ellas tienen la forma típica de “S”, aunque algunas de ellas se ven seriamente deformadas. Sin embargo, la isoterma promedio (línea roja en la Fig. 7.2a),

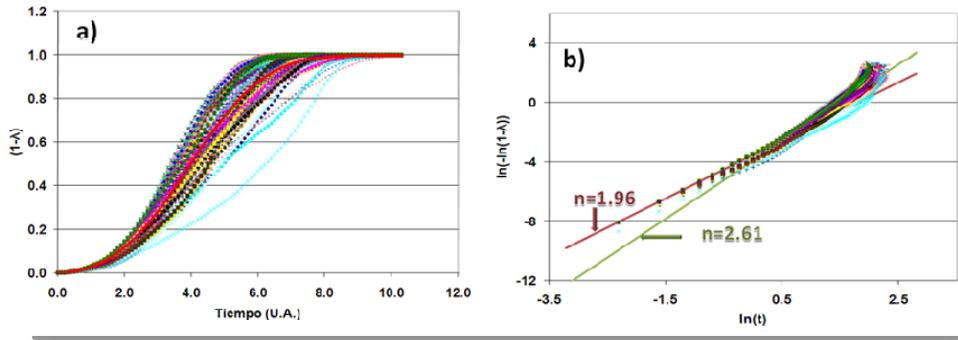


Figura 7.2. a) Gráfica de isotermas de cristalización en 100 simulaciones utilizando $Gr = 0.1$ y $Gn = 0.8$. La línea continua roja representa la isoterma promedio. b) Gráfica de Avrami, $n = 1.96$ utilizando el método visual y $n = 2.61$ el propuesto en este trabajo.

definitivamente tiene la forma típica de “S” y parece tener una uniformidad bastante grande.

La figura 7.2b muestra las gráficas del exponente de Avrami de las isotermas correspondientes a la figura 7.1a, encontrando que aunque podemos seleccionar los primeros datos para hacer la regresión, lo que nos da un exponente de Avrami de $n = 1.96$ podemos también pensar en tomar los datos intermedios, los que nos daría $n = 2.61$. Se obtuvieron los exponentes de Avrami de todas las isotermas y con ellas se construyó un histograma normalizado (Fig. 7.3), encontrando que la gran mayoría de los valores (66%), fluctúan entre 1.975 y 2.05, es decir alrededor de 2.01, lo que coincide con una nucleación atérmica con crecimiento bidimensional. Esto se puede explicar por la influencia que tienen los bordes de la muestra (Fig. 7.1), aquí las fronteras son líneas rectas, las cuales corresponden al caso particular IV explicitado en el punto 5.1 (análisis de la geometría de las fronteras).

A partir de las gráficas de Avrami (Fig. 7.2b), se consideró interesante obtener la gráfica de la pendiente de las gráficas de Avrami promedio, lo que resulta en curvas que nos representan el exponente de Avrami conforme evoluciona la nucleación y crecimiento. La figura 7.4 muestra esta gráfica, donde la magnitud del exponente de Avrami inicia como 2, pues al inicio solo hay un núcleo que crece como círculo, su magnitud desciende ligeramente debido al efecto de los límites de muestra que truncan el círculo. Al momento de formarse el segundo núcleo (indicado en la figura con una

recta vertical) n se incrementa rápidamente hasta aproximadamente 2.36, desde donde de nuevo comienza a disminuir lentamente hasta la formación del tercer núcleo, dando así lugar a una gráfica escalonada a partir de la formación del segundo núcleo, donde cada “escalón” aparece en cada intervalo de tiempo entre la formación de dos núcleos.

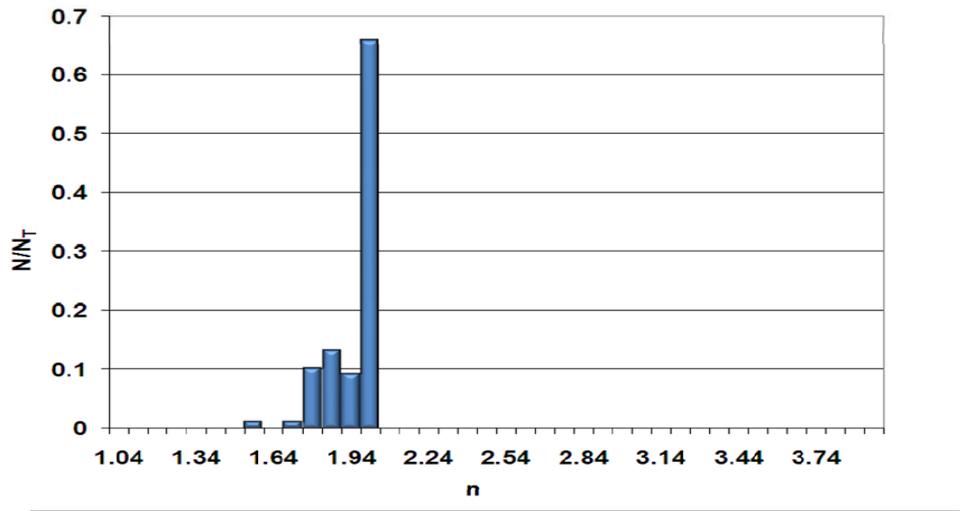


Figura 7.3. Histograma de valores de exponentes de Avrami, curva normalizada. ($Gr = 0.1$, $Gn = 0.8$).

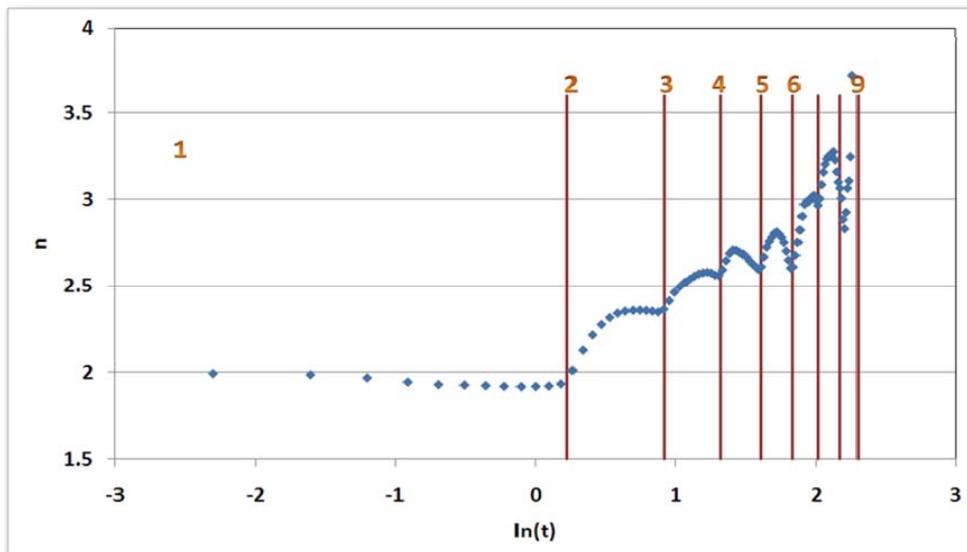


Figura 7.4 Gráfica del exponente de Avrami en función del logaritmo del tiempo para la simulación con $Gr = 0.1$ y $Gn = 0.8$. Las líneas rectas representan el tiempo de formación de núcleos.

En este caso, en las 100 repeticiones, se formaron entre 4 y 8 núcleos, con un promedio de 5.68 y desviación estándar de 0.71.

Como una forma de demostrar el uso del modelo para la determinación de la función de nucleación, se partió de los vértices de los núcleos vecinos (Tabla 7.1), y ya que estos obedecen a la Ec. 5.21, la cual se repite a continuación, entonces dándole como tiempo de formación igual a cero al primer núcleo, es posible calcular los productos $Gr(t_i)$ de los demás núcleos.

$$V = \frac{Gr(t_2 - t_1)}{2} \quad [5.21]$$

El vértice multiplicado por 2 es la quinta columna de la tabla 7.1, al darle el valor de cero al tiempo de formación del primer núcleo, simplemente mediante sumas y restas es posible obtener el producto $Gr(t_i)$, el cual lo podemos tomar como un tiempo modificado por la constante Gr , entonces podemos hacer una gráfica del número de núcleos (N) vs. $Gr(t_i)$. Ésta gráfica se muestra en la figura 7.5, donde encontramos que resulta en una recta con pendiente de 8.

Tabla 7.1.- Cálculo de la función de nucleación a partir de los vértices de las fronteras formadas mediante una simulación con $Gr = 0.1$ y $Gn = 8$.

No. De Frontera	Entre los núcleos		Vértice	$Gr(t_2)-Gr(t_1)$	No. De núcleos (N)	$G(t_i)$
	N1	N2				
3	1	2	0.0625	0.125	1	0.000
1	1	3	0.1250	0.250	2	0.125
2	1	4	0.1875	0.375	3	0.250
7	2	4	0.1250	0.250	4	0.375
8	2	4	0.1250	0.250	5	0.500
6	2	5	0.1875	0.375	6	0.625
11	3	4	0.0625	0.125		
14	4	5	0.0625	0.125		
15	4	6	0.1250	0.250		

Ya que la pendiente la podemos escribir como:

$$\frac{N}{Gr(t_i)} = \frac{Gn}{Gr} \quad [7.1]$$

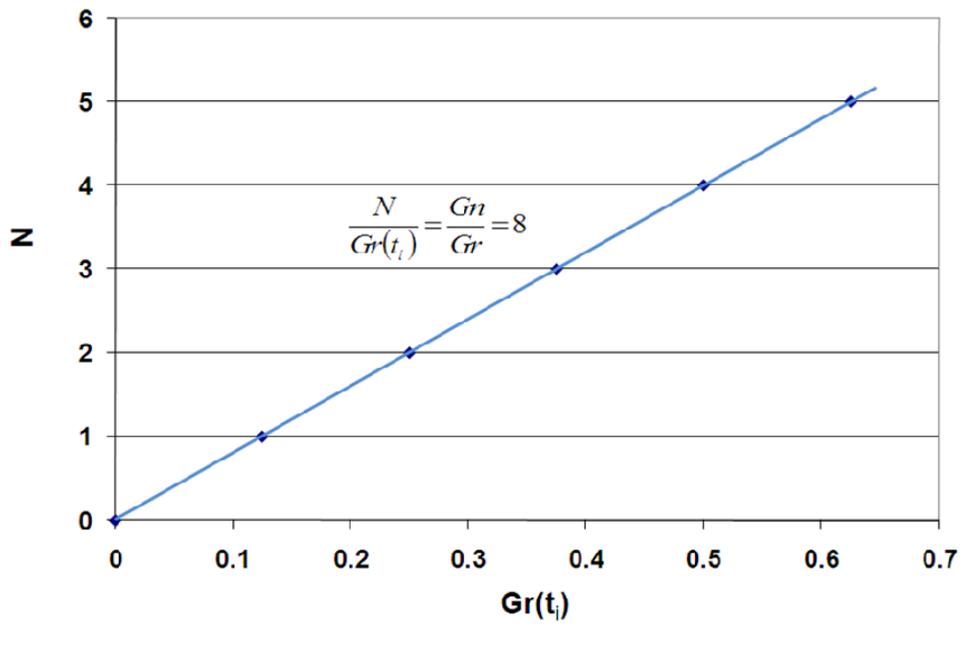


Figura 7.5. Gráfica de número de núcleos vs. el tiempo modificado por la rapidez de crecimiento.

Entonces, ya que $Gr = 0.1$, la magnitud de Gn calculada es exactamente 0.8. Esto además de demostrarnos que la programación está correcta, nos enseña que es posible calcular la función de nucleación, al menos cuando Gn es constante, a partir de los vértices de las hipérbolas, es decir de la imagen final de la simulación

Se hicieron simulaciones con 13 diferentes rapidezces de nucleación (Gn), en las figuras 7.6 y 7.7 se muestran las instantáneas del simulador con $Gn = 15$ y $Gn = 52$ respectivamente.

Además de observar que no hay errores en las limitaciones de las esferulitas, en las Figs. 7.1, 7.6 y 7.7 se aprecian el incremento del número total de núcleos y la disminución del tiempo requerido para cubrir toda la superficie, conforme aumenta la rapidez de nucleación. Todo esto es congruente con lo esperado.

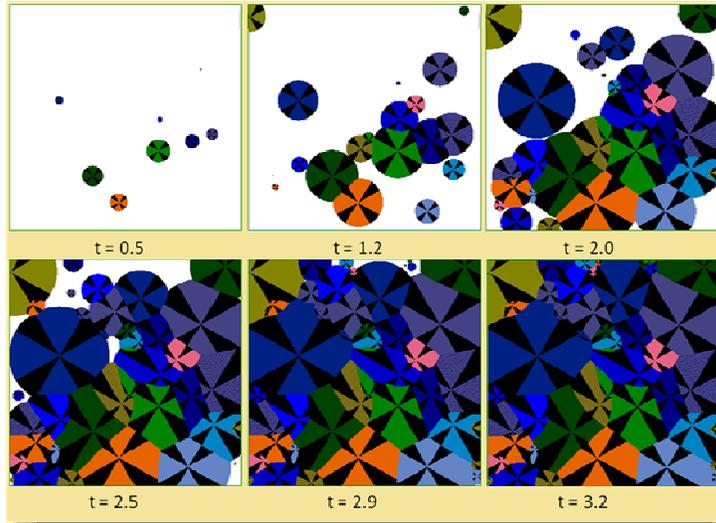


Figura 7.6. Seis imágenes correspondientes a diferentes tiempos en la simulación con $Gr = 0.1$ y $Gn = 15.0$.

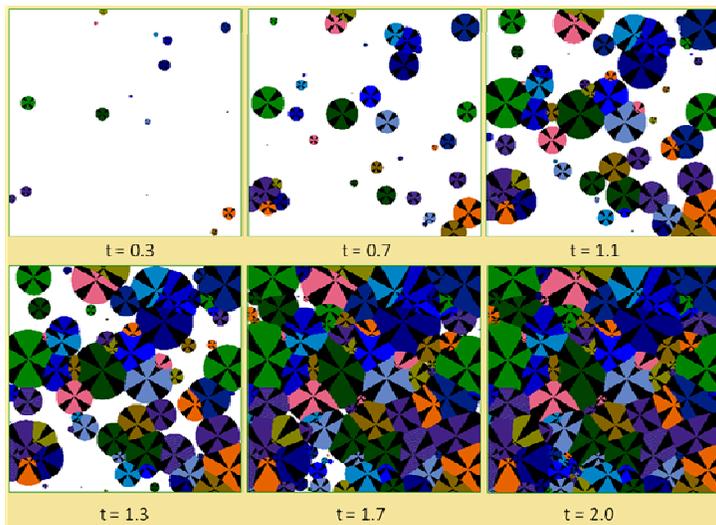


Figura 7.7. Seis imágenes correspondientes a diferentes tiempos en la simulación con $Gr = 0.1$ y $Gn = 52.0$.

En las figuras 7.8 a 7.10, se reportan las isotérmicas de cristalización de las simulaciones con diferentes valores de Gn , como en el caso de la Fig. 7.2a, cada conjunto de gráficas representan las 100 repeticiones (gráficas punteadas), y la curva promedio es la marcada como línea continua en rojo.

Es interesante notar que cuando la magnitud de G_n decrece, hay una mayor variabilidad en la forma de cada gráfica de simulación, desde $G_n = 0.8$ (Fig. 7.2a) hasta $G_n = 5.5$ (en Fig. 7.8), hay gráficas que distan de tener la forma de “S” uniforme, en otras palabras cuando G_n aumenta, disminuye la variabilidad de la forma de las curvas “S”. En todos los casos la curva promedio tiene la forma esperada.

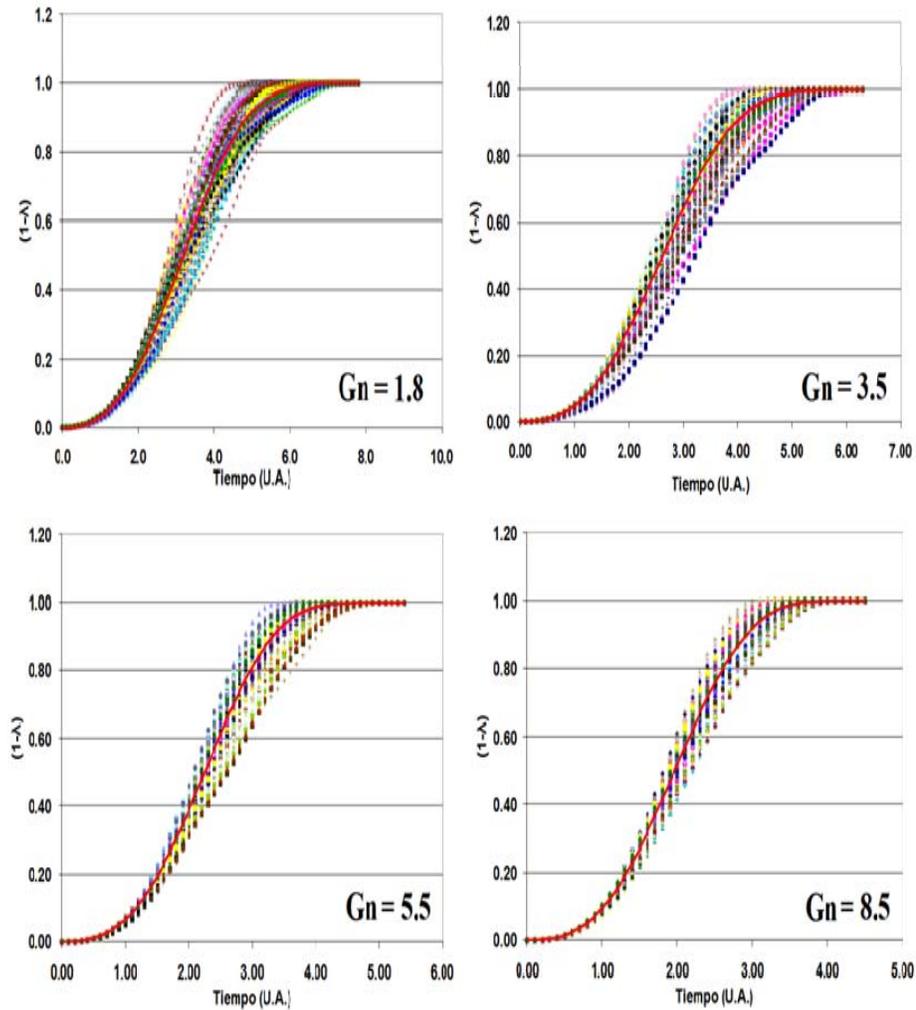


Figura 7.8. Isothermas de cristalización para las 100 repeticiones y curva promedio (línea continua en rojo), de las simulaciones con $Gr = 0.1$ y G_n entre 1.8 y 8.5 (se indica en las gráficas).

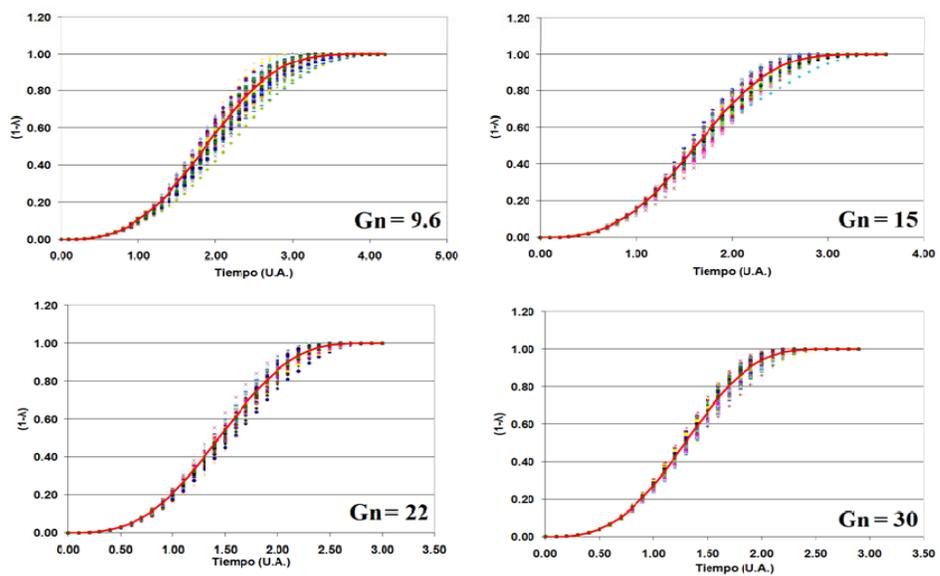


Figura 7.9. Isothermas de cristalización para las 100 repeticiones y curva promedio (línea continua en rojo), de las simulaciones con $Gr = 0.1$ y Gn entre 9.6 y 30 (se indica en las gráficas).

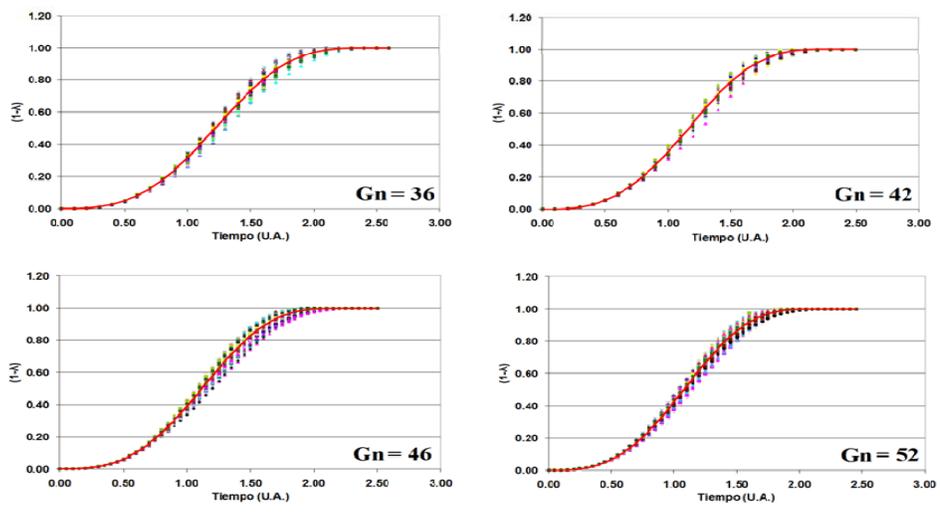


Figura 7.10. Isothermas de cristalización para las 100 repeticiones y curva promedio (línea continua en rojo), de las simulaciones con $Gr = 0.1$ y Gn entre 36 y 52 (se indica en las gráficas).

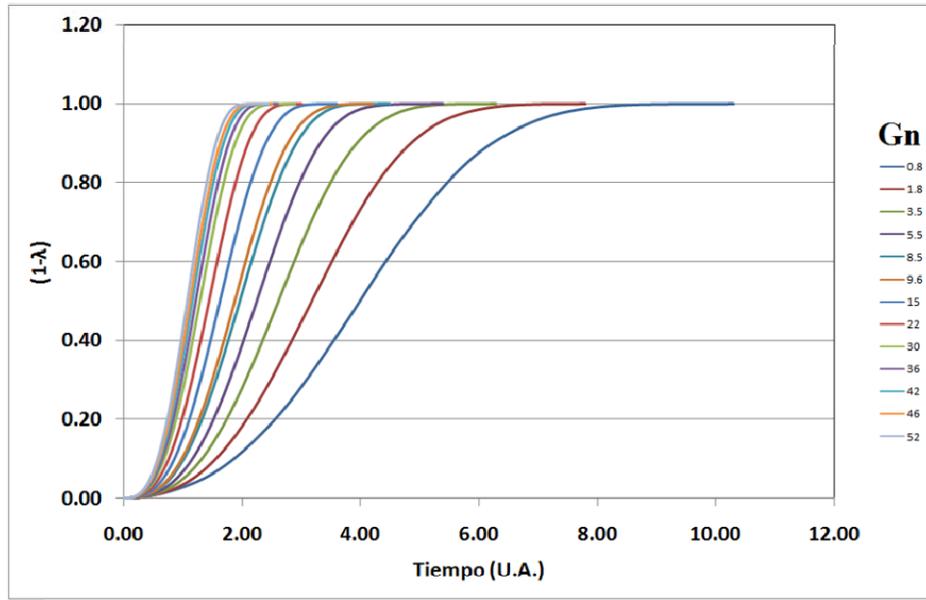


Figura 7.11. Isothermas de cristalización para las 13 condiciones de rapidez de nucleación, $Gr = 0.1$, G_n se indica en la figura.

La figura 7.11 muestra las isothermas de cristalización promedio, en ella observamos la disminución de la pendiente y el aumento del tiempo final de cristalización al incrementar G_n , esto también es un resultado esperado.

Como es sabido, el exponente de Avrami se calcula a partir de la pendiente de la gráfica logarítmica de la Ec. 2.2, sin embargo, al observar la gráfica de la pendiente de la curva de Avrami en función del logaritmo del tiempo (Fig. 7.4), primero se obtuvieron estas gráficas promedio para cada G_n , (Fig. 7.12). Aquí se puede apreciar que las gráficas con $G_n < 3.5$ muestran un comportamiento de tipo escalonamiento muy visiblemente, y que este va disminuyendo al aumentar G_n hasta prácticamente desaparecer cuando $G_n > 22$. Es probable que debido al aumento en número de núcleos formados y consecuentemente, la disminución de tiempo para completar la “cristalización”, el escalonamiento sólo sea detectable si se disminuye el intervalo de tiempo entre medición y medición, que en forma general para nuestro caso fue de $\Delta t = 0.1$.

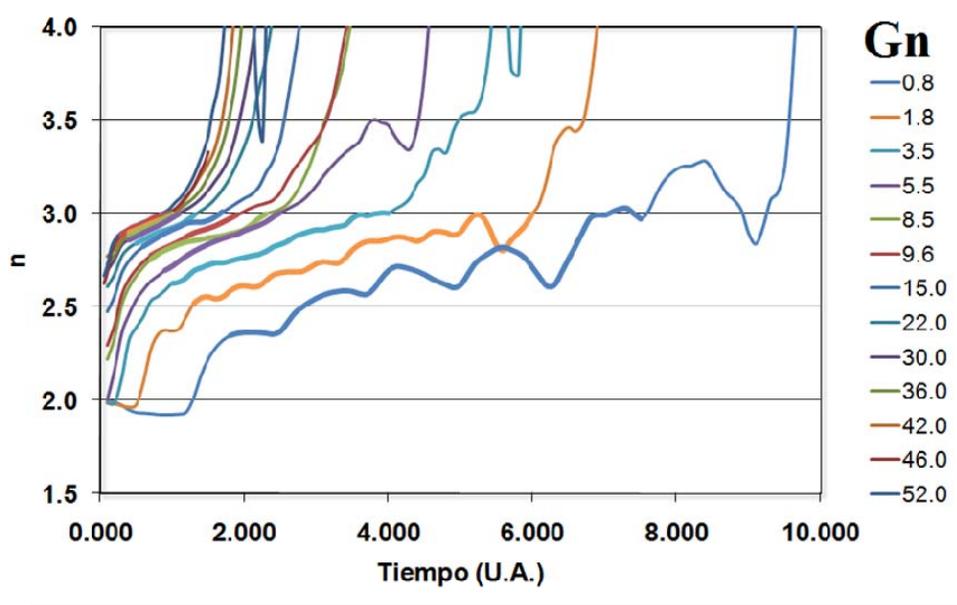


Figura 7.12. Derivadas de las gráficas de Avrami promedio para corridas con $Gr = 0.1$ y diferentes valores de Gn (indicadas en la figura)

De cualquier manera, las pendientes generales de las curvas son positivas, y no parecen disminuir al aumentar Gn . Si los datos se ajustaran a la Ec. de Avrami, deberíamos tener una pendiente de cero.

Por lo anterior, se decidió calcular el exponente de Avrami de dos formas, primero de la forma tradicional, es decir simplemente ajustando la sección en el intervalo que a “ojo” parece ser recta y al que llamaremos el “*método visual*”, y segundo (“*método propuesto*”), eliminando el primer escalón que, debido a que en un principio ha habido pocos o ningún choque entre núcleos (ver primeros recuadros de Figs. 7.6 y 7.7), se tiende a un exponente típico de nucleación atérmica ($n = 2$), así como también se elimina la región a altos tiempos, donde la pendiente crece muy rápidamente, posiblemente debido a que se hace importante el área cubierta por la formación y crecimiento de nuevos núcleos, en relación al área de crecimiento de los núcleos previamente formados y que ya han dejado de crecer por haber chocado con otros núcleos en todas las direcciones (ver penúltimos recuadros de Figs. 7.6 y 7.7). Los puntos escogidos para este último método se representan con una línea más gruesa en la Fig. 7.12.

Como se puede apreciar en las figuras 7.13 a 7.15, de las gráficas de Avrami para simulaciones con diferente G_n , muestran el mismo fenómeno de exactitud y precisión que el observado para $G_n = 0.8$, pero además se aprecia que entre mayor G_n , hay mayor precisión de los resultados (en las 100 repeticiones).

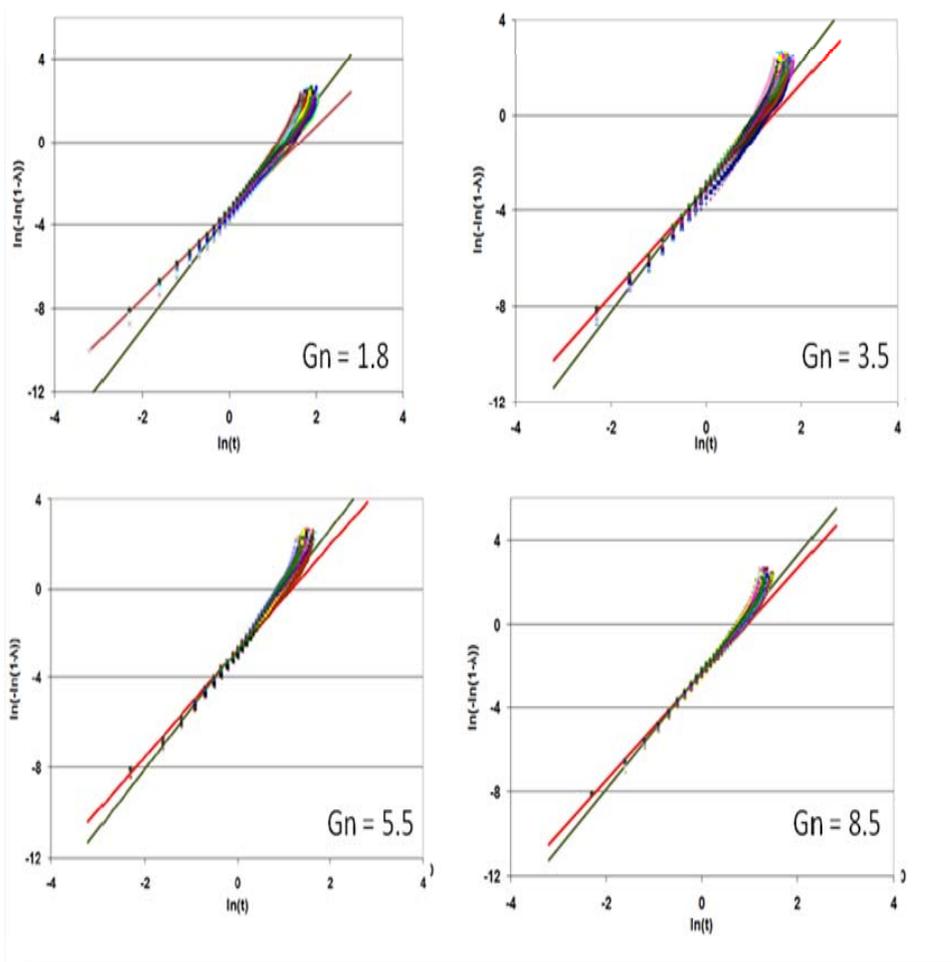


Figura 7.13. Gráficas de Avrami para las 100 repeticiones y regresión de la curva promedio (línea continua en rojo), de las simulaciones con $G_r = 0.1$ y G_n entre 1.8 y 8.5 (se indica en las gráficas). La línea verde es la regresión a la Ec. de Avrami de acuerdo al método aquí propuesto.

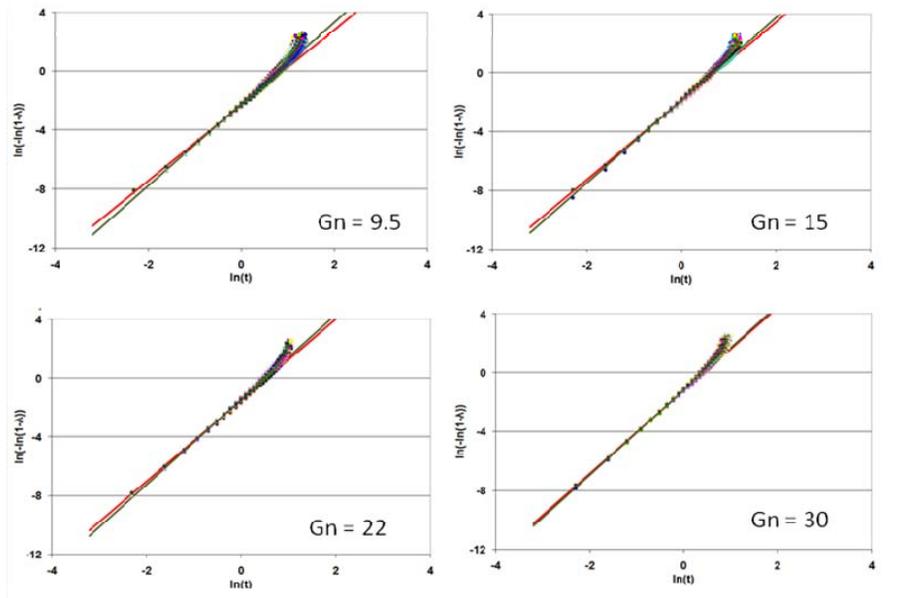


Figura 7.14. Gráficas de Avrami para las 100 repeticiones y regresión de la curva promedio (línea continua en rojo), de las simulaciones con $Gr = 0.1$ y G_n entre 9.5 y 30 (se indica en las gráficas). La línea verde es la regresión a la Ec. de Avrami de acuerdo al método aquí propuesto.

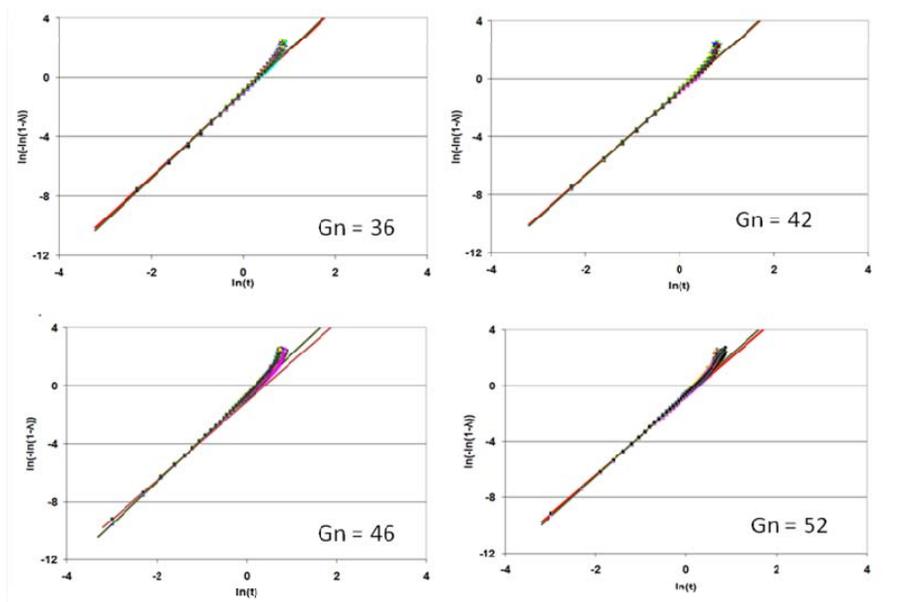


Figura 7.15. Gráficas de Avrami para las 100 repeticiones y regresión de la curva promedio (línea continua en rojo), de las simulaciones con $Gr = 0.1$ y G_n entre 36 y 52 (se indica en las gráficas). La línea verde es la regresión a la Ec. de Avrami de acuerdo al método aquí propuesto.

Por otra parte, el exponente de Avrami calculado mediante el *método visual* (líneas continuas rojas) siempre es menor que el calculado por el *método propuesto* (líneas continuas verdes), por otra parte, al incrementar G_n la diferencia entre ambas rectas es menor.

Para cada valor de G_n , se determinaron los números promedio de núcleos formados y su desviación estándar. En la tabla 7.2 se reportan estos resultados, junto con los exponentes de Avrami calculados mediante los dos métodos.

Observación interesante de la tabla 7.2, es que los exponentes de Avrami en ningún caso alcanzan el valor teórico para un fenómeno de nucleación térmica bidimensional, esta magnitud es de 3 (Tabla 2.1), aunque en ambos casos parecen aproximarse a este valor. Además como se había previsto al analizar las gráficas 7.13 a 7.15, los exponentes calculados por ambos métodos tienden a ser más próximos entre sí al aumentar G_n .

Tabla 7.2. Número de núcleos promedio N , su desviación estándar σ y exponentes de Avrami n calculados mediante los métodos denominados *visual* y *propuesto*, para cada conjunto de experimentos con diferente G_n .

Gn	N	σ	n	
			Visual	Propuesto
0.8	5.68	0.71	1.95	2.61
1.8	10.83	1.11	2.03	2.74
3.5	17.00	1.49	2.22	2.80
5.5	22.64	1.88	2.38	2.84
8.5	30.69	2.11	2.53	2.86
9.6	33.18	2.18	2.58	2.87
15.0	44.97	2.39	2.69	2.91
22.0	57.24	2.52	2.77	2.90
30.0	70.08	3.43	2.83	2.91
36.0	78.77	2.87	2.85	2.93
42.0	86.73	3.22	2.87	2.94
46.0	92.45	2.89	2.89	2.94
52.0	97.53	2.02	2.81	2.94

En la Figura 7.16, se muestra una gráfica del exponente de Avrami respecto al número de núcleos promedio generados en cada conjunto de simulaciones. Aquí mismo se presentan como líneas continuas las regresiones a ecuaciones exponenciales de segundo orden, que resultan ser las siguientes (Ecs. 7.2 y 7.3):

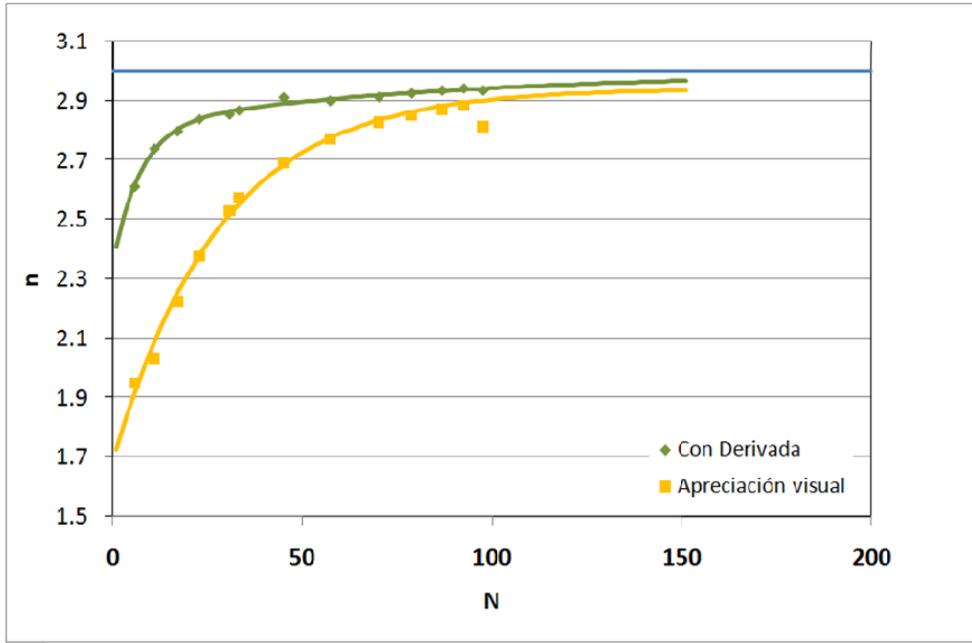


Figura 7.16. Exponentes de Avrami en función del número promedio de núcleos generados, tanto para el cálculo de apreciación visual como de la derivada.

$$n = 3 - 0.184 \exp\left\{-\frac{N}{83.9}\right\} - 0.468 \exp\left\{\frac{N}{7.23}\right\} \quad [7.2]$$

$$n = 2.94 - 2 \exp\left\{-\frac{N}{28.3}\right\} \quad [7.3]$$

La Ec. 7.2 corresponde a la regresión sobre los puntos seleccionados mediante el *método propuesto*, y es asintótica inequívocamente al valor teórico de 3, mientras que para los cálculos obtenidos mediante el *método visual*, parece ser asintótica a $n = 2.92$. Lo anterior sugiere que el *método propuesto*, es el más acertado.

En las figuras 7.17 y 7.18 se muestran los histogramas del exponente de Avrami para ambos métodos, encontramos que cuando se utiliza el *método visual*, la dispersión es relativamente baja en comparación al método propuesto. Esto es comprensible ya que al inicio de la nucleación existen pocos núcleos, además de que la tendencia es a tomar valores de $n = 2$, mientras que en el segundo caso, el rango de datos utilizados es el intermedio, y por lo tanto hay un mayor número de núcleos, lo que produce la mayor

variabilidad, y mayor número de choques, resultando en valores que se acercan cada vez más a $n = 3$.

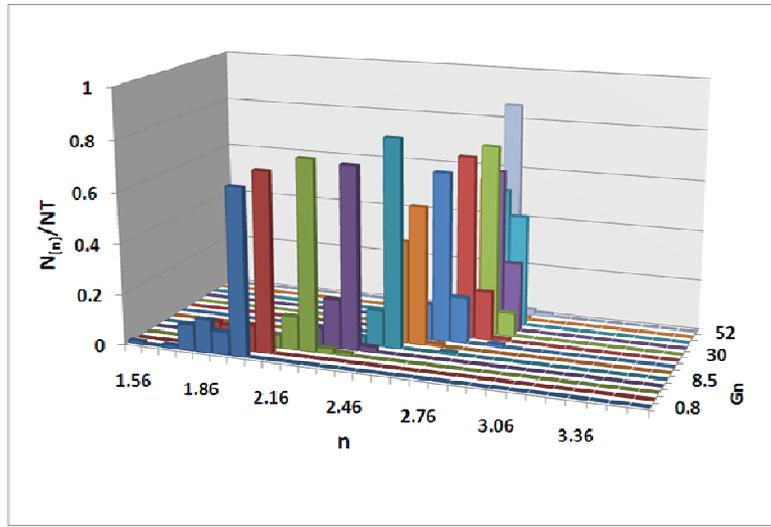


Figura 7.17. Histogramas normalizados de las magnitudes de n , cuando el rango de ajuste se determina de la forma tradicional. Las 13 curvas con diferentes G_n están en profundidad.

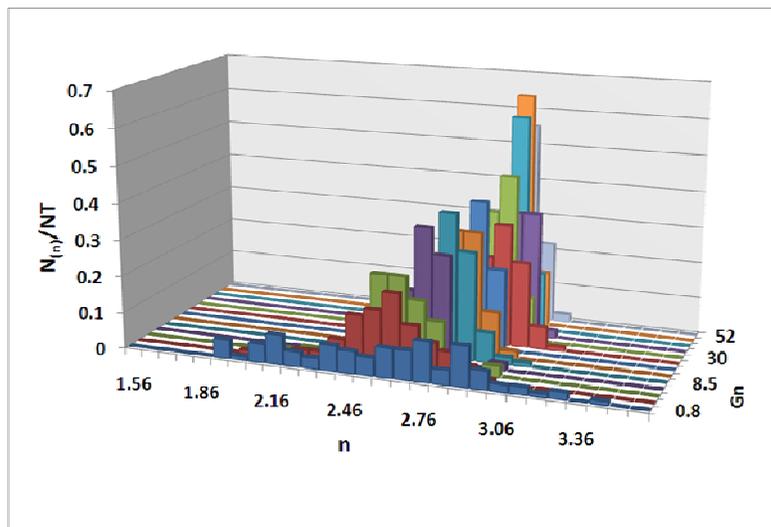


Figura 7.18. Histogramas normalizados de las magnitudes de n , cuando el rango de ajuste se determina mediante la derivada de la gráfica de Avrami. Las 13 curvas con diferentes G_n están en profundidad.

7.2.- Función de nucleación asintótica

Se utilizó la Ec. 6.2, para analizar el efecto sobre las isotermas de cristalización tanto de la rapidez de nucleación Gn como del número máximo N_{Max} de núcleos a formarse.

$$t_i = - \frac{\ln \left[\frac{N_{Max} - N_i}{N_{Max} - 1} \right]}{Gn} \quad [6.2]$$

En la tabla 7.3 se muestran las variables Gn y N_{Max} de las corridas de simulación llevadas a cabo, (con 100 repeticiones cada una).

Tabla 7.3. Variables Gn y N_{Max} de las simulaciones realizadas a Gn constante

Gn	N_{max}							
0.02	50	70	100	120	220	350	550	1100
0.20	10	25	50	70	100	150	250	
0.50	8	13	22	33	70	95		
1.00	8	18	28	43	70	100		
1.90	7	15	30	50	70	80	100	

Considerando el número tan grande de gráficas (34), que se tendrían que incluir para cubrir todas las experiencias de simulación llevadas a cabo, se documentarán más ampliamente las marcadas con negritas en la tabla 7.3, lo que reduce a 15 en número de gráficas de isotermas de cristalización y de Avrami, las cuales son suficientes para hacer las observaciones necesarias. En cuanto a las imágenes de salida de simulación, solo se mostrarán tres tiempos para dos diferentes N_{Max} para $Gn = 0.02$ y las gráficas finales de las demás corridas mostradas en la tabla 7.3.

Es interesante notar que las últimas imágenes de las simulaciones que corresponden a diferentes valores de Gn (última imagen de Fig. 7.19 y Fig. 7.20), no se puede inferir nada en relación a tamaños y formas de los cristales. Además, como las posiciones y tiempos de formación de los núcleos determinan que tan rápido se cubre el área del material transformado, el tiempo requerido para cubrir toda el área de la muestra y el número de núcleos formados será diferente para cada corrida con las mismas variables de entrada (Gn y N_{Max}).

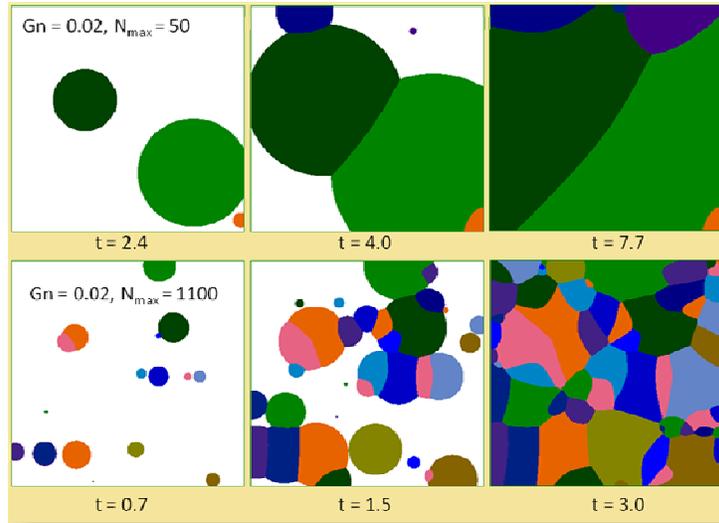


Figura 7.19. Imágenes de simulación para tres tiempos (se indican en figura), con $G_n = 0.02$, asintótica, con un número máximo de núcleos diferente en cada una de las dos series.

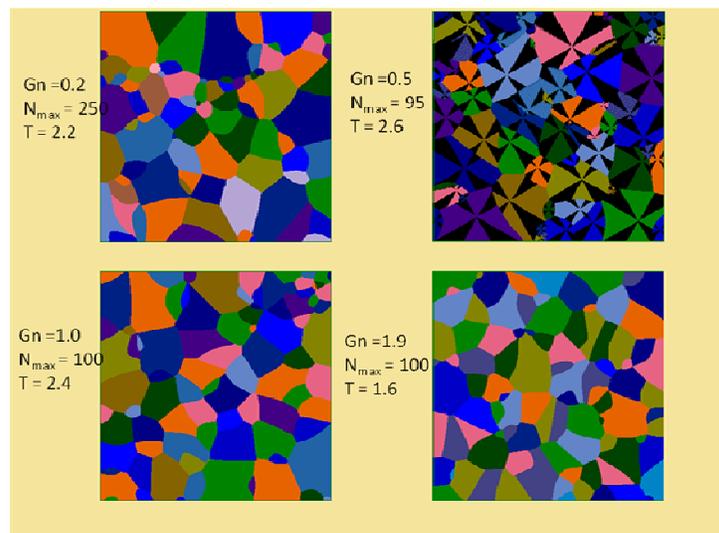


Figura 7.20. Imágenes finales de las simulaciones de diferente G_n y N_{Max} , valores que se indican en la figura

Igual que en el caso de las simulaciones con G_n constante (Figs. 7.8 a 7.10), en este caso, donde se consideró la rapidez de nucleación como una función asintótica, las corridas individuales muestran isotermas de cristalización (Figs. 7.21 a 7.25), cuya forma se acerca más a la forma de “S”, que es la observada experimentalmente mediante técnicas como dilatometría y DSC, así como menor variabilidad, al

incrementar el número de núcleos formados. Por supuesto que el incremento del número de núcleos obviamente trae consigo una disminución del tiempo final de la cristalización.

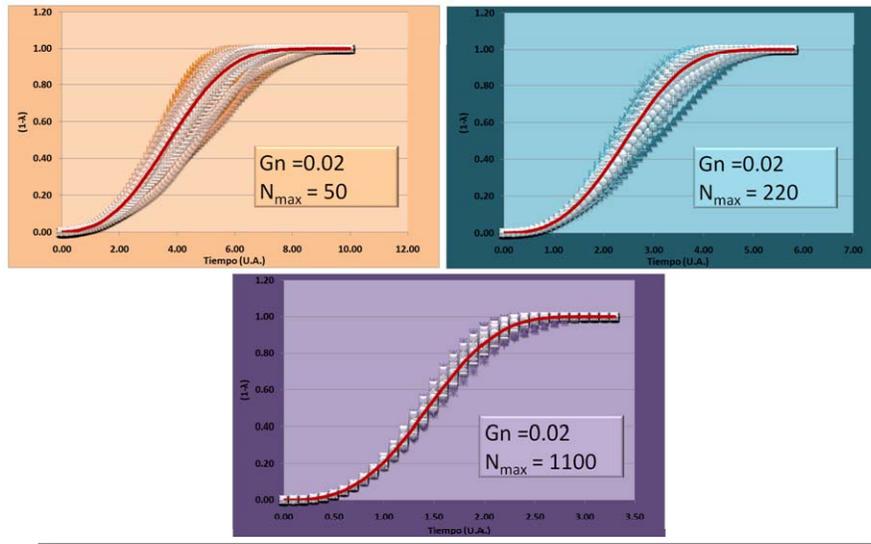


Figura 7.21. Isothermas de cristalización para las simulaciones con función asintótica con $G_n = 0.02$ y número variable de N_{Max} (Se indican en el inserto). La línea roja es la isoterma promedio de 100 repeticiones.

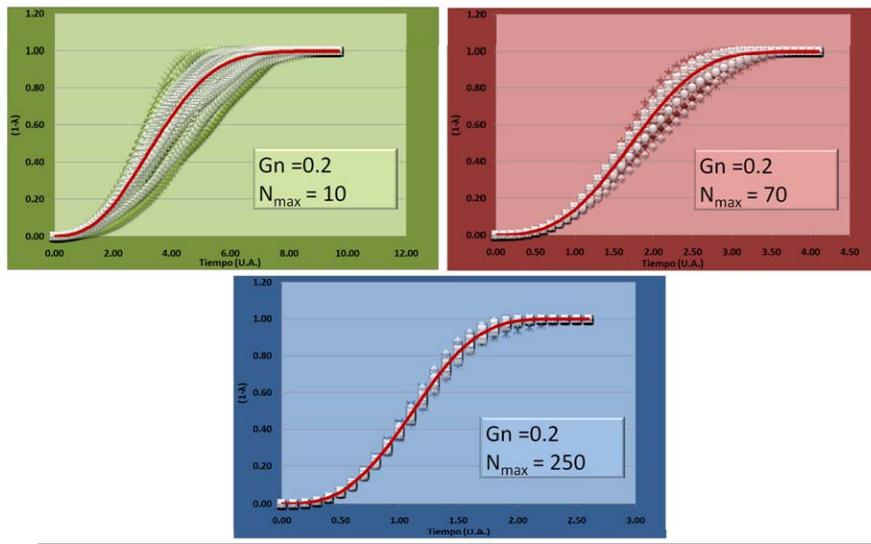


Figura 7.22. Isothermas de cristalización para las simulaciones con función asintótica con $G_n = 0.2$ y número variable de N_{Max} (Se indican en el inserto). La línea roja es la isoterma promedio de 100 repeticiones.

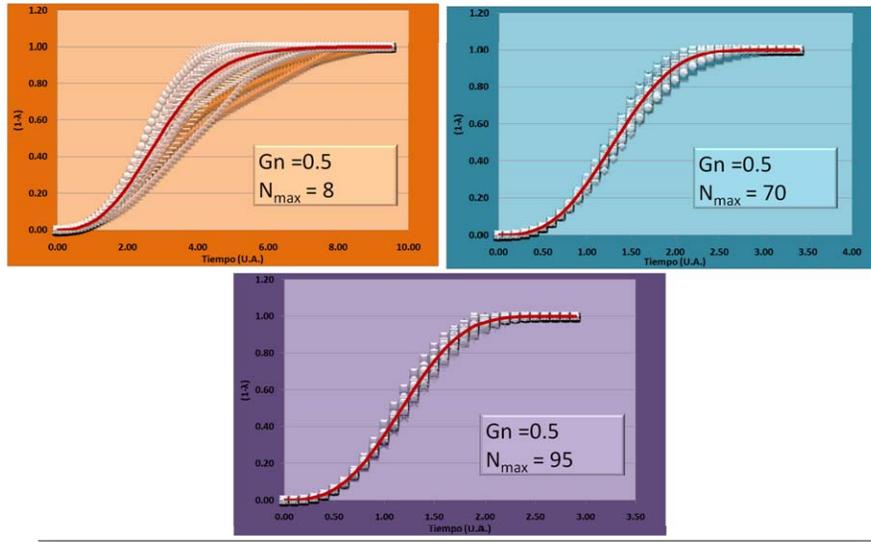


Figura 7.23. Isothermas de cristalización para las simulaciones con función asintótica con $G_n = 0.5$ y número variable de N_{\max} (Se indican en el inserto). La línea roja es la isoterma promedio de 100 repeticiones.

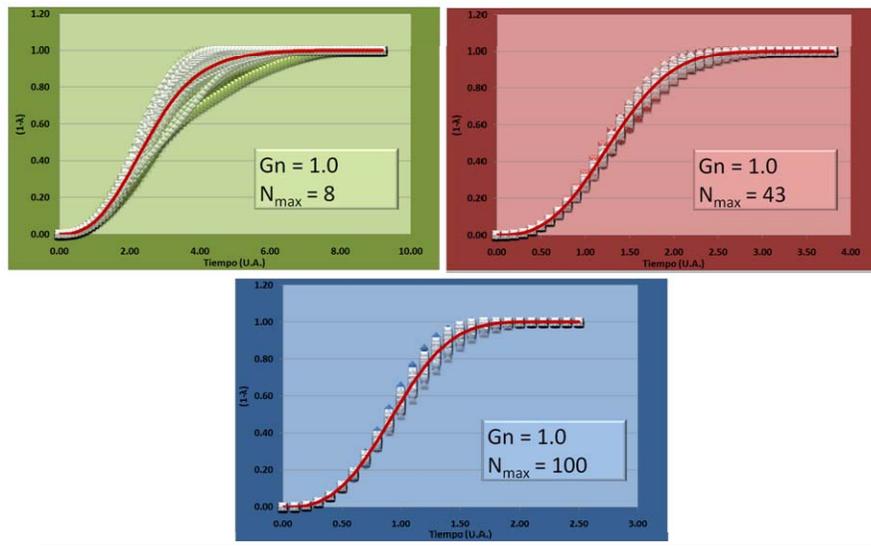


Figura 7.24. Isothermas de cristalización para las simulaciones con función asintótica con $G_n = 1.0$ y número variable de N_{\max} (Se indican en el inserto). La línea roja es la isoterma promedio de 100 repeticiones.

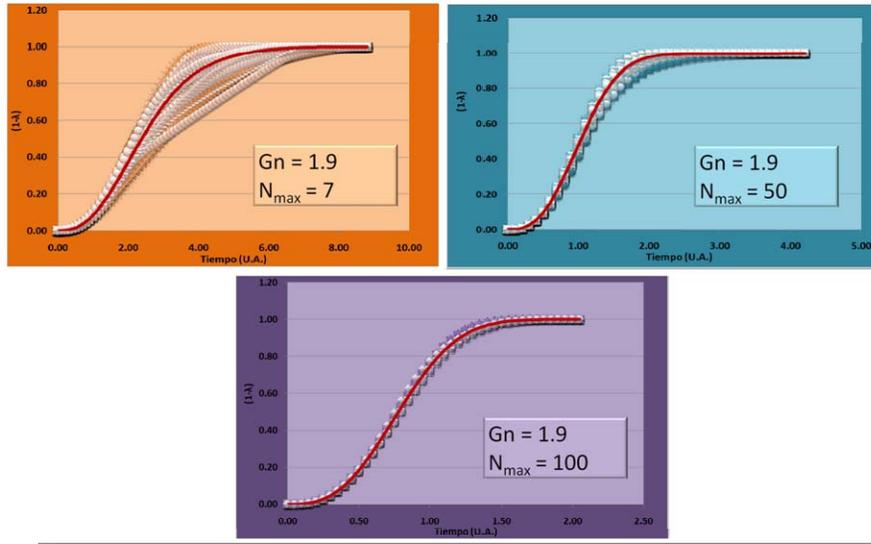


Figura 7.25. Isothermas de cristalización para las simulaciones con función asintótica con $Gn = 1.9$ y número variable de N_{Max} (Se indican en el inserto). La línea roja es la isoterma promedio de 100 repeticiones.

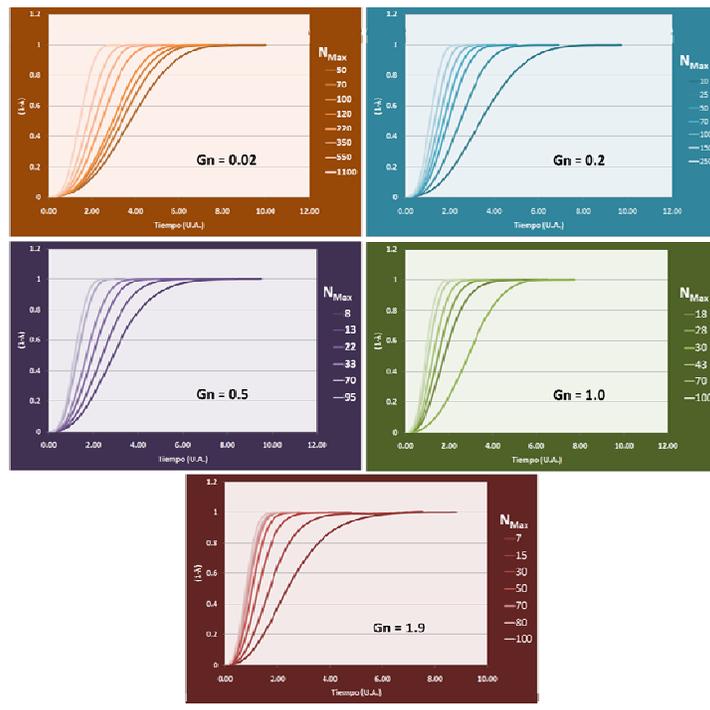


Figura 7.26. Efecto del número máximo de núcleos a formarse sobre las isótermas de nucleación promedio para las diversas rapidezces de nucleación (se indica en insertos)

Al comparar las gráficas que tienen diferente G_n pero el mismo N_{Max} , por ejemplo para $N_{Max} = 70$ con $G_n = 0.2$ (Fig. 7.22) y $G_n = 0.5$ (Fig. 7.23), o bien para $N_{Max} = 8$ con $G_n = 0.5$ (Fig. 7.23) y $G_n = 1$ (Fig. 7.24), encontramos que la dispersión de curvas en las 100 repeticiones va disminuyendo, además de con el incremento de N_{Max} a G_n constante, también con el incremento de G_n con N_{Max} constante. Esto puede ser indicativo de que la reproducibilidad de las curvas se incrementa, con el número de núcleos formados.

La Figura 7.27 muestra las gráficas de n vs. tiempo para las isothermas promedio de cada conjunto de experimentos.

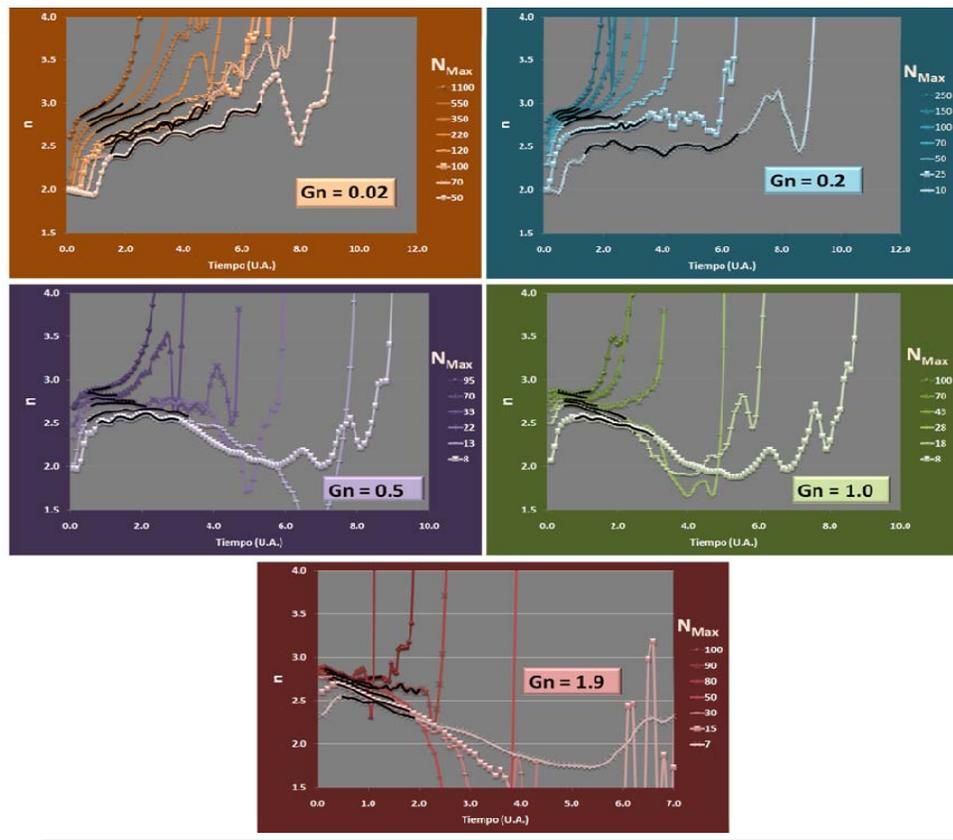


Fig. 7.27. Función del exponente de Avrami respecto al tiempo para las curvas promedio de todas las simulaciones. (se indica G_n y N_{Max} en los insertos). En negro se marca los datos utilizados para el *método propuesto*

En ellas se observa el mismo fenómeno de escalonamiento, solo que la pendiente de las gráficas disminuye con la rapidez de nucleación hasta tomar valores negativas a $G_n = 1$ y $G_n = 1.9$.

En las figuras de la 7.28 a 7.32 se muestran las gráficas de Avrami para cada conjunto de experimentos con G_n diferente. Se aprecia de nuevo mayor reproducibilidad en las cien repeticiones, conforme aumenta N_{Max} o G_n . Así mismo la pendiente calculada mediante el método propuesto, es siempre mayor para el método visual, y esta diferencia disminuye conforme aumentan G_n y N_{Max} .

De nuevo, podemos intuir que la exactitud y precisión, más que estar relacionada con G_n o N_{Max} , está relacionada con el número de núcleos formados durante la simulación.

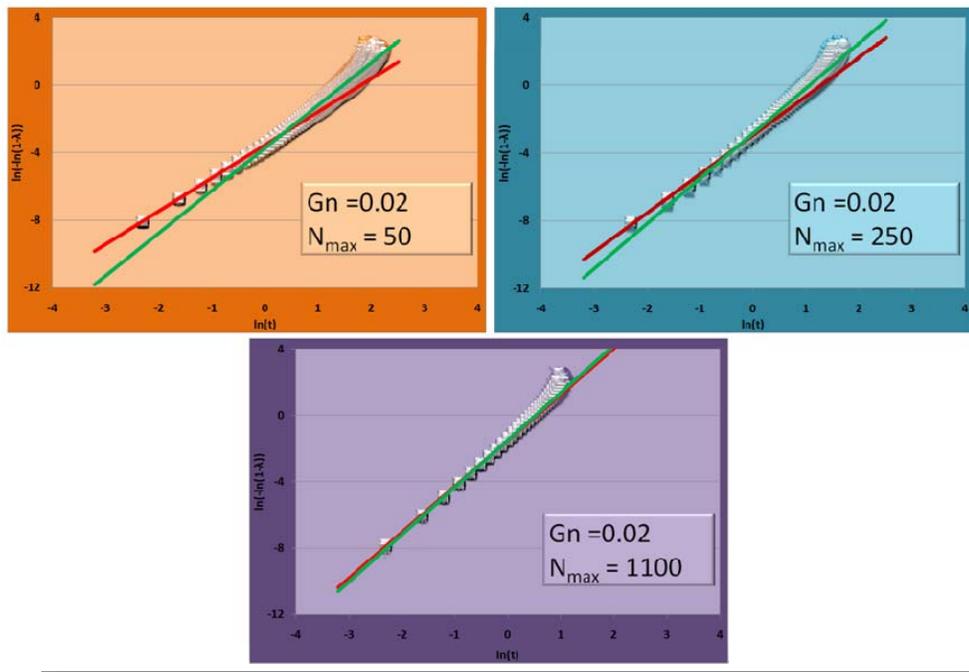


Figura 7.28. Gráficas de Avrami para las simulaciones de cien repeticiones de nucleación térmica con $Gr = 0.1$, $G_n = 0.02$ y N_{Max} variable (se indican en insertos)

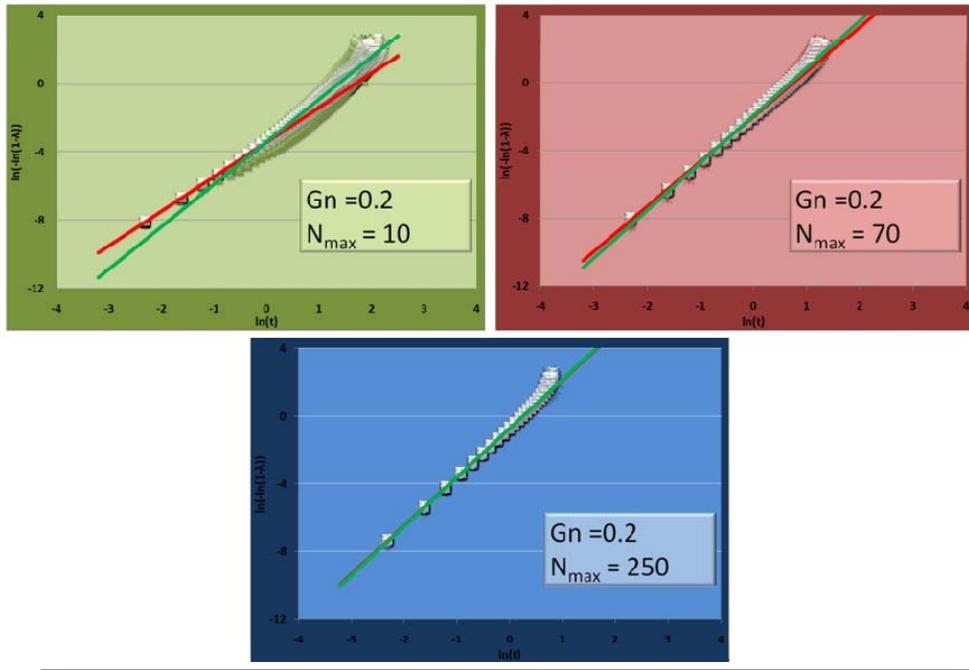


Figura 7.29. Gráficas de Avrami para las simulaciones cien repeticiones de nucleación térmica con $Gr = 0.1$, $G_n = 0.2$ y N_{Max} variable (se indican en insertos)

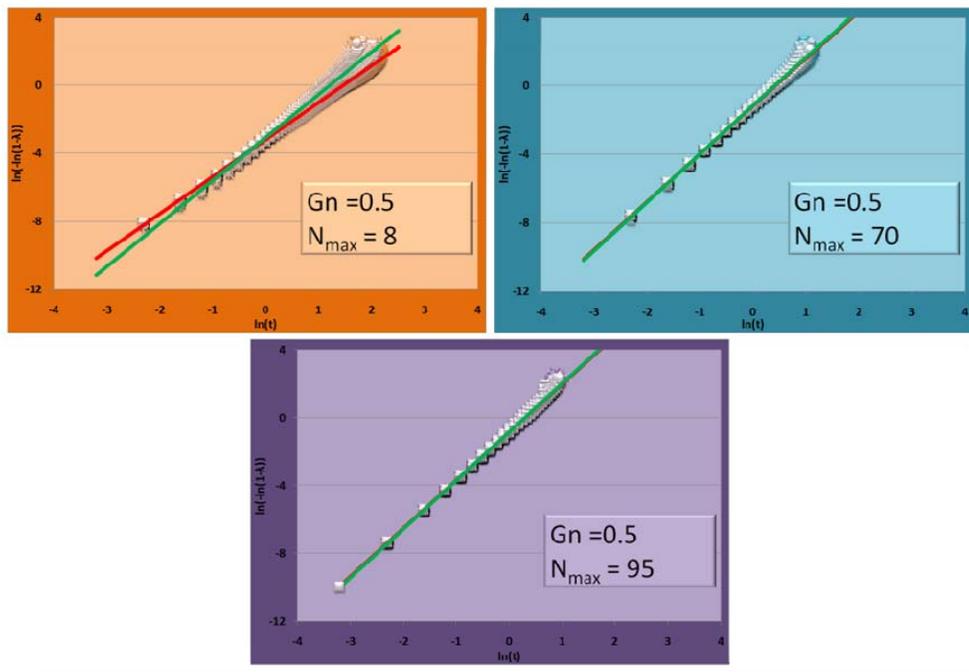


Figura 7.30. Gráficas de Avrami para las simulaciones cien repeticiones de nucleación térmica con $Gr = 0.1$, $G_n = 0.5$ y N_{Max} variable (se indican en insertos)

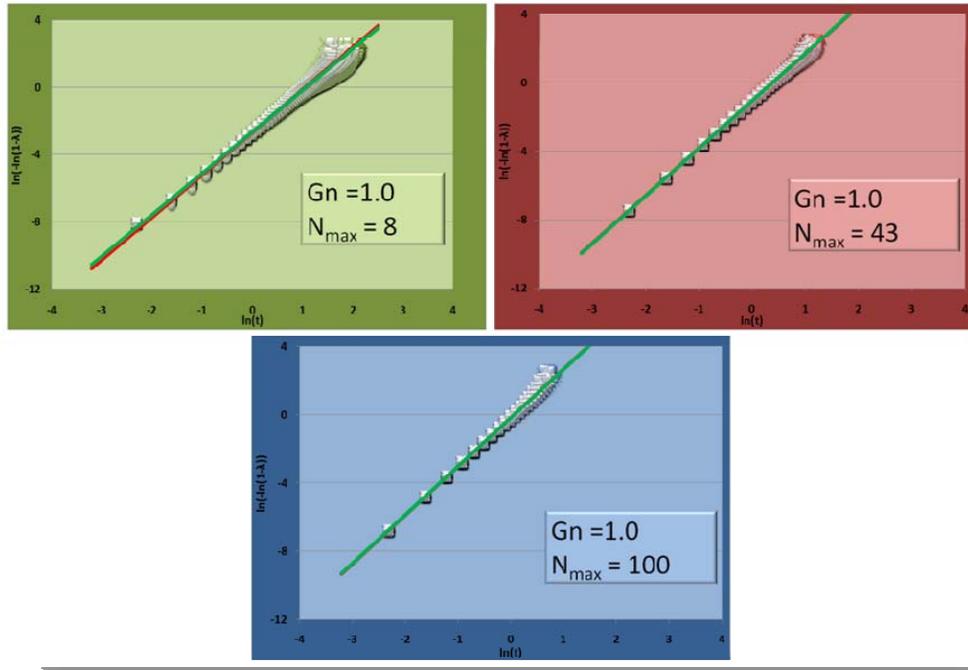


Figura 7.31. Gráficas de Avrami para las simulaciones cien repeticiones de nucleación térmica con $Gr = 0.1$, $G_n = 1.0$ y N_{Max} variable (se indican en insertos)

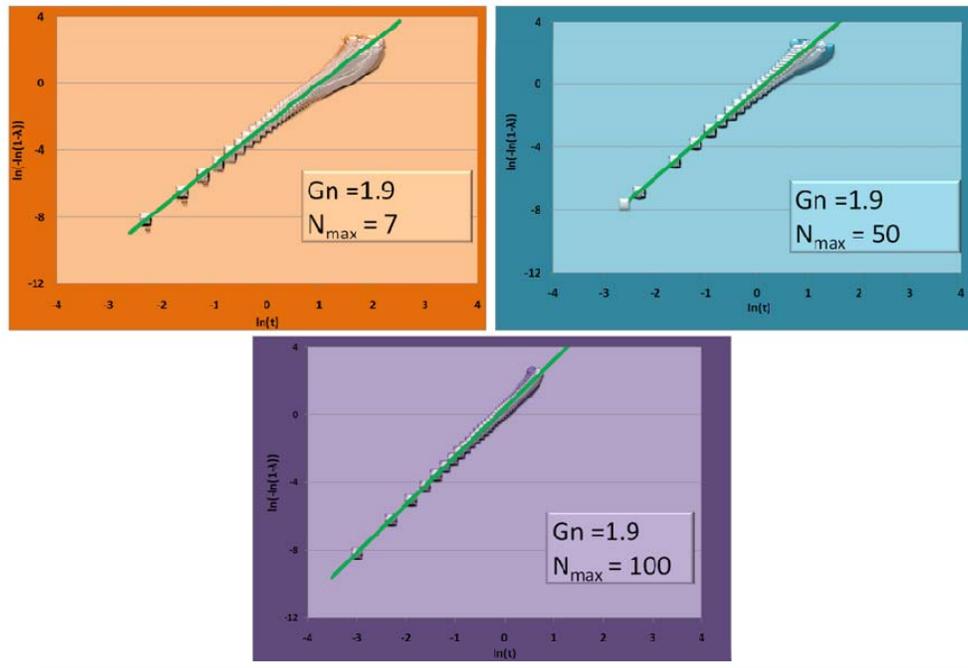


Figura 7.32. Gráficas de Avrami para las simulaciones cien repeticiones de nucleación térmica con $Gr = 0.1$, $G_n = 1.9$ y N_{Max} variable (se indican en insertos)

En la tabla 7.4 se presentan los resultados del número promedio de núcleos formados y el exponente de Avrami calculado con el *método propuesto*, incluyendo la desviación estándar. Aquí queda claro que el número de núcleos aumenta con G_n y con N_{Max} , reforzando la idea de la relación entre exactitud y precisión con el número promedio de núcleos.

Tabla 7.4 Resultados del número promedio de núcleos y exponente de Avrami y su desviación estándar para todas las simulaciones hechas

		Gn = 0.02			Gn = 0.2			Gn = 0.5				
N_{max}		N	n	σ	N_{max}	N	n	σ	N_{max}	N	n	σ
50		6.22	2.53	0.08	10	6.32	2.49	0.07	8	6.00	2.53	0.06
70		7.91	2.58	0.08	25	14.53	2.67	0.05	13	10.66	2.63	0.04
100		10.85	2.54	0.06	50	25.84	2.80	0.04	22	17.89	2.73	0.05
120		12.19	2.54	0.07	70	33.43	2.82	0.05	33	26.15	2.77	0.03
220		19.09	2.67	0.05	100	43.52	2.84	0.03	70	49.90	2.83	0.04
350		26.25	2.72	0.05	150	58.15	2.88	0.03	95	63.76	2.86	0.03
550		35.66	2.77	0.04	250	84.23	2.90	0.03				
1100		56.30	2.85	0.04								

		Gn = 1.0			Gn = 1.9			
N_{max}		N	n	σ	N_{max}	N	n	σ
8		6.23	2.53	0.04	7	6.00	2.48	0.05
18		16.00	2.64	0.04	15	13.81	2.62	0.04
28		25.62	2.72	0.03	30	27.85	2.73	0.03
43		38.94	2.79	0.03	50	47.96	2.78	0.04
70		60.59	2.83	0.03	70	68.70	2.83	0.02
100		83.93	2.86	0.03	80	76.50	2.84	0.03
					100	94.78	2.86	0.03

Entre los aspectos que se consideran de mayor importancia en este trabajo, destaca la posibilidad de determinar la función de nucleación a partir de las excentricidades de las hipérbolas que conforman los bordes de los núcleos al final de la “cristalización”. En las figura 7.33 y 7.34 se muestra este hecho, la Fig. 7.33 es una familia de curvas para una G_n constante e igual a 1.9, los puntos son los resultados calculados a partir de los vértices y las líneas son calculadas mediante la Ec. 6.2, que es la función de nucleación,

y los datos de entrada del simulador. Por su parte en la Fig. 7.34 se muestran estas mismas gráficas, escogidas al azar entre el lote de simulaciones llevadas a cabo, variando G_n así como N_{Max} . Considerando estos resultados y la gráfica de la Fig. 7.5, podemos concluir que indiscutiblemente considerando este modelo, se puede estimar la función de nucleación independientemente de sus características.

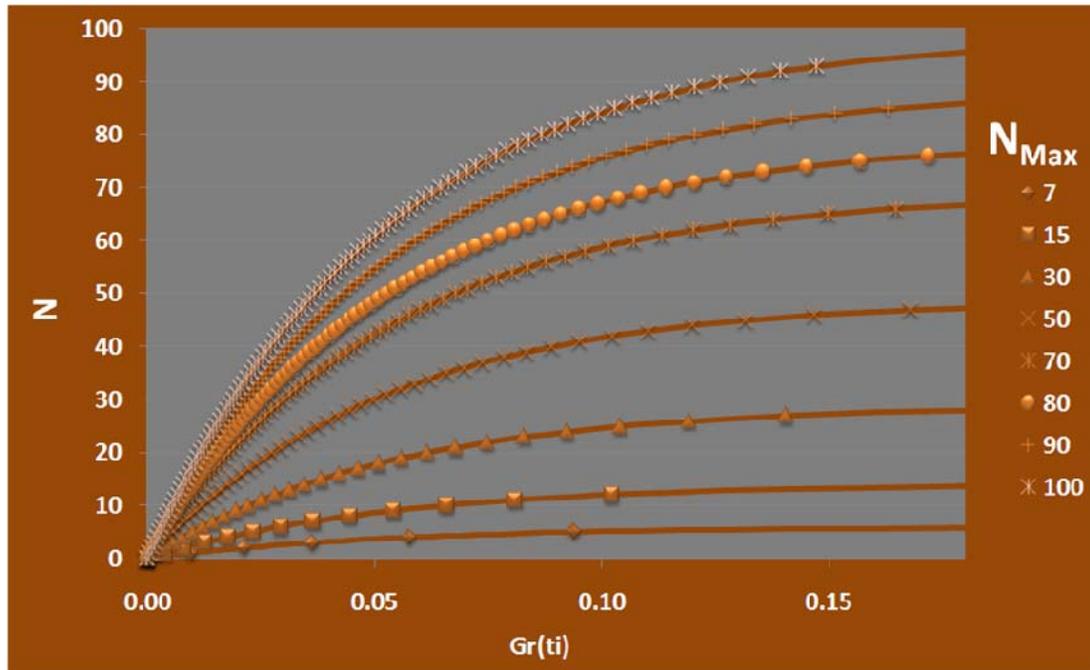


Figura 7.33. Número de núcleos vs. el producto $Gr(t_i)$ y su ajuste a la Ec. 6.2 para $G_n = 1.9$ y N_{Max} variable (se indica en el inserto).

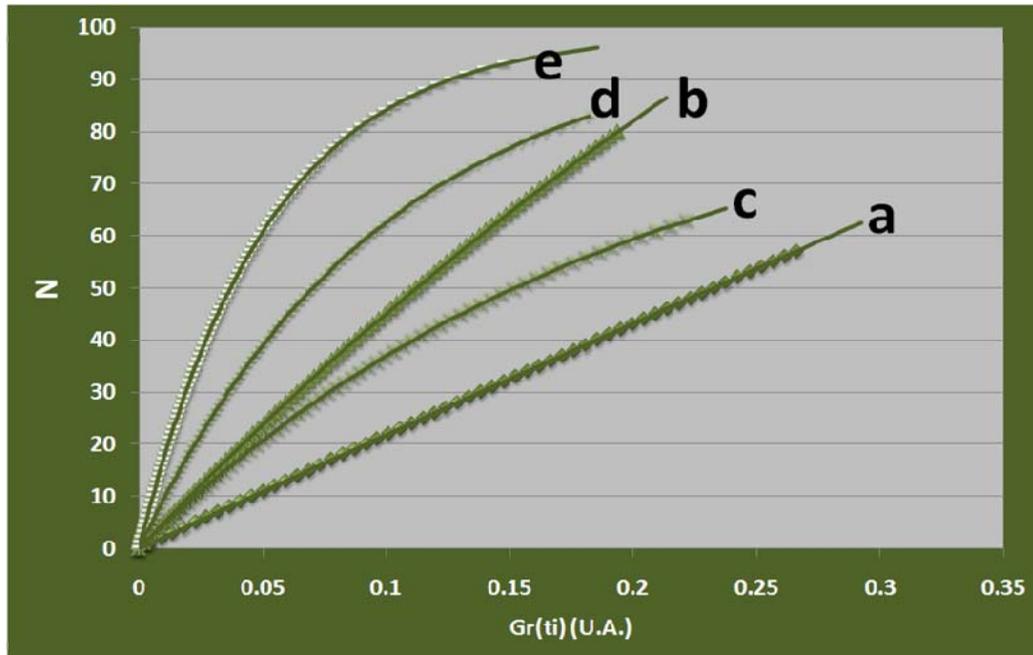


Figura 7.34, Ajustes de los datos de número de núcleos vs. $Gr(ti)$, calculado de los vértices de las fronteras, a la ecuación 6.2 con las siguientes variables de entrada: a). $Gn = 0.02$, $N_{Max} = 1100$, b) $Gn = 0.2$, $N_{Max} = 250$, c) $Gn = 0.5$, $N_{Max} = 95$, d) $Gn = 1.0$, $N_{Max} = 100$, e) $Gn = 1.9$, $N_{Max} = 100$

Por último la distribución de núcleos formados para cada conjunto de simulaciones, se muestra en forma normalizada en las gráficas de la Fig 7.35, apreciándose que además de aumentar el número de núcleos con Gn y N_{Max} , la distribución se hace cada vez mas estrecha. Es decir, entre menor sea la rapidez de nucleación, es necesario utilizar una muestra mayor para mantener una buen análisis de resultados.

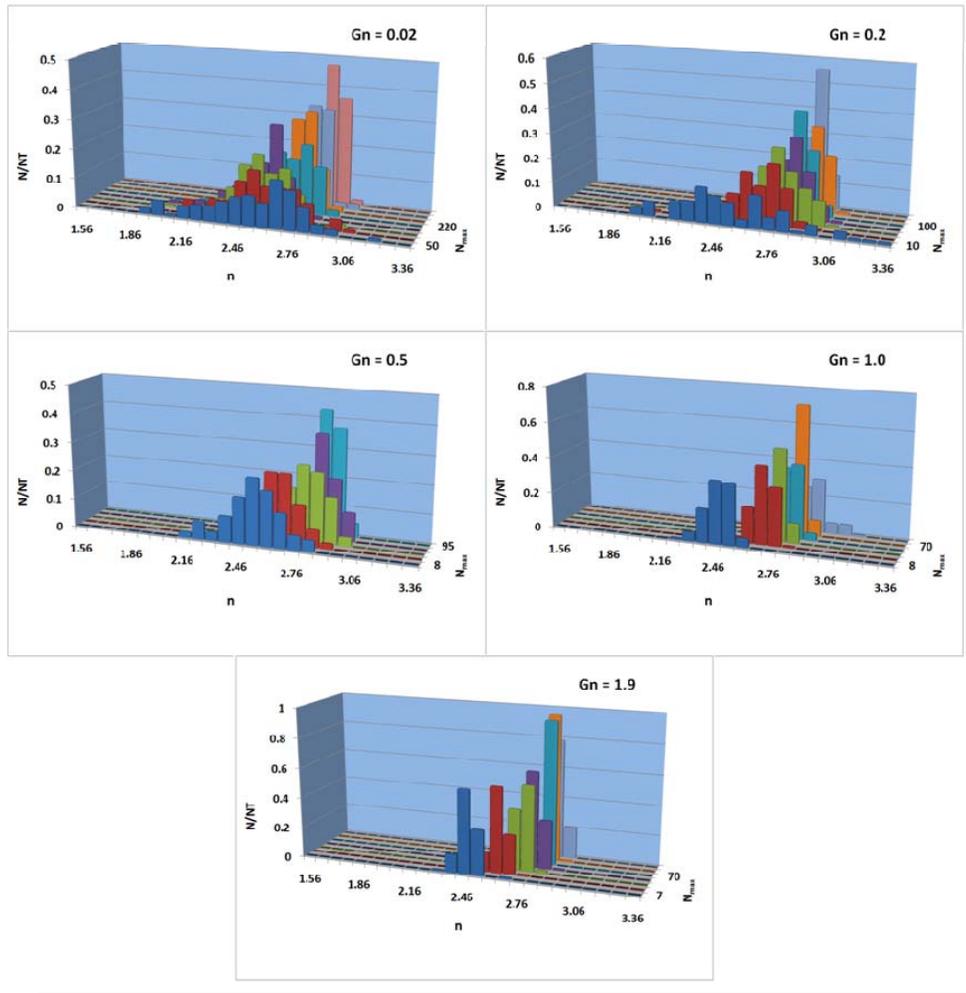


Figura 7.35. Distribuciones normalizadas del número de núcleos en función de G_n y N_{Max} .

7.3.- Función de nucleación constante respecto al área no transformada

El algoritmo de aproximaciones propuesto (Esquema 6.3), se probó con éxito, la gráfica de la Fig, 7.36 muestra cómo convergen para cada núcleo generado tanto la rapidez de nucleación y el tiempo de formación de los 72 núcleos que se forma en una simulación con G_n (inicial) de 176 núcleos por unidad de tiempo.

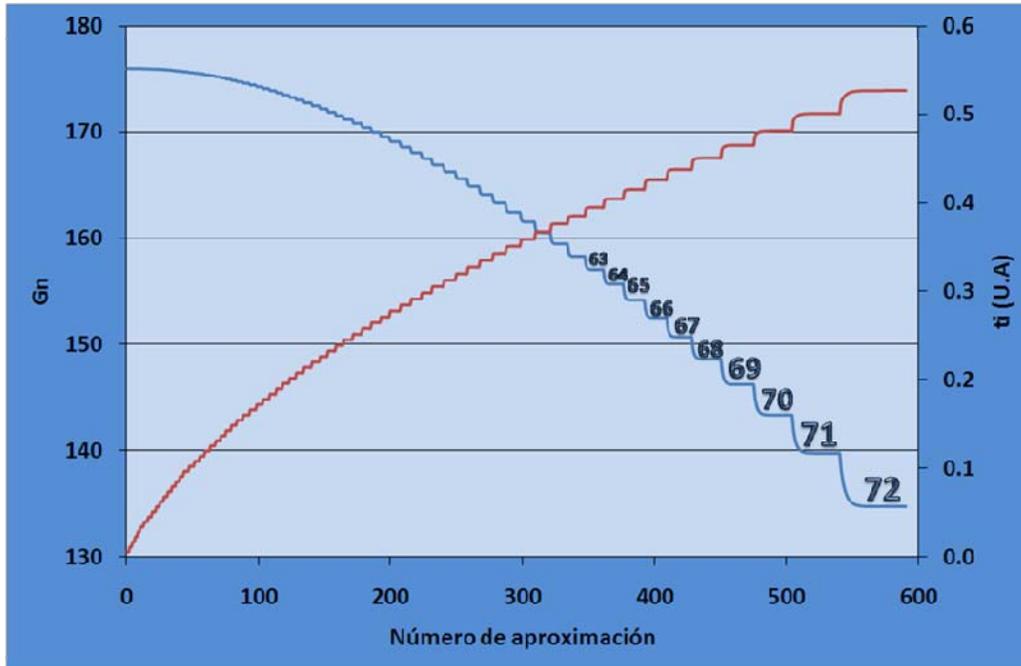


Figura 7.36. Convergencia del algoritmo de generación de núcleos (rapidez “ G_n ” y tiempo de nucleación “ t_i ”), cada escalón es un núcleo, (se indican los últimos 10). G_n inicial es de 176.

Por efectos de espacio en este trabajo, en la Fig. 7.37 solo se muestran cuatro imágenes correspondientes a los tiempos de 0.2, 0.9, 1.6 y 2.8 (U.A.). Aquí como en los dos casos anteriores se nota que los cristales están bien definidos. Pero además encontramos que en apariencia, existen muchas fronteras que parecen ser rectas, es decir resultado de choque entre núcleos que se formaron al mismo tiempo. Como veremos más adelante, esto no es así, todos los bordes son hiperbólicos, y como veremos de sus posiciones será necesario, igualmente que en los casos anteriores, determinar la función de nucleación, que en este caso es más complicada que cuando se consideró G_n una constante independiente del área y del tiempo (punto 7.1) o bien constante del área pero función asintótica del tiempo (punto 7.2).

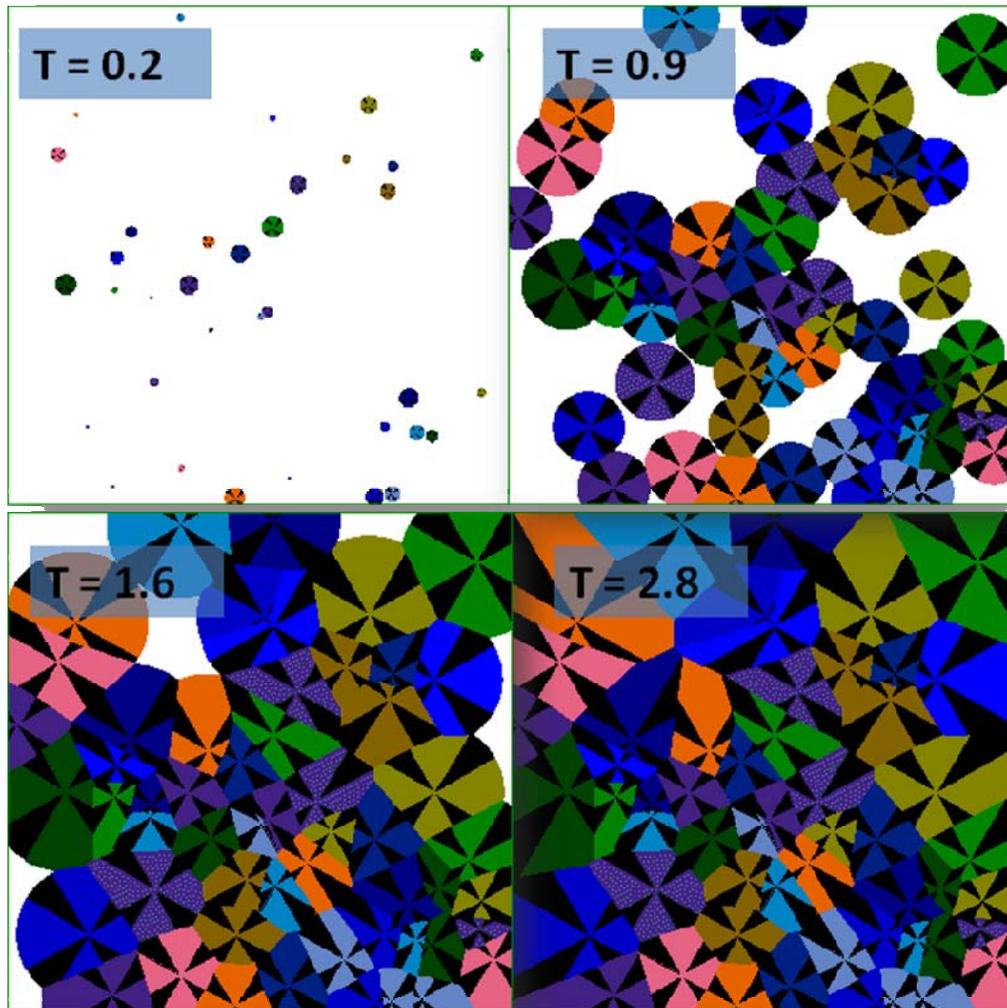


Figura 7.37. Cuatro imágenes correspondientes a diferentes tiempos de la simulación con $G_n = 176$ núcleos por unidad de área no cubierta

Las Figuras 7.38 y 7.39 reportan los conjuntos de isoterma de cristalización para diferentes G_n entre 8 y 176, aquí es notoria la disminución de la dispersión al aumentar G_n , es decir, no parece ser necesario considerar el uso de un mayor número de repeticiones para mejorar la reproducibilidad al disminuir G_n , pero desafortunadamente, la dispersión observada es equiparable a los casos de menor rapidez de nucleación cuando G_n es independiente del área residual (comparar con Figs 7.8 a 7.10 y 7.21 a 7.26). Lo anterior indica que el análisis experimental de la cristalinidad debe hacerse bajo bases estadísticas más sólidas que como se acostumbra (no más de tres repeticiones).

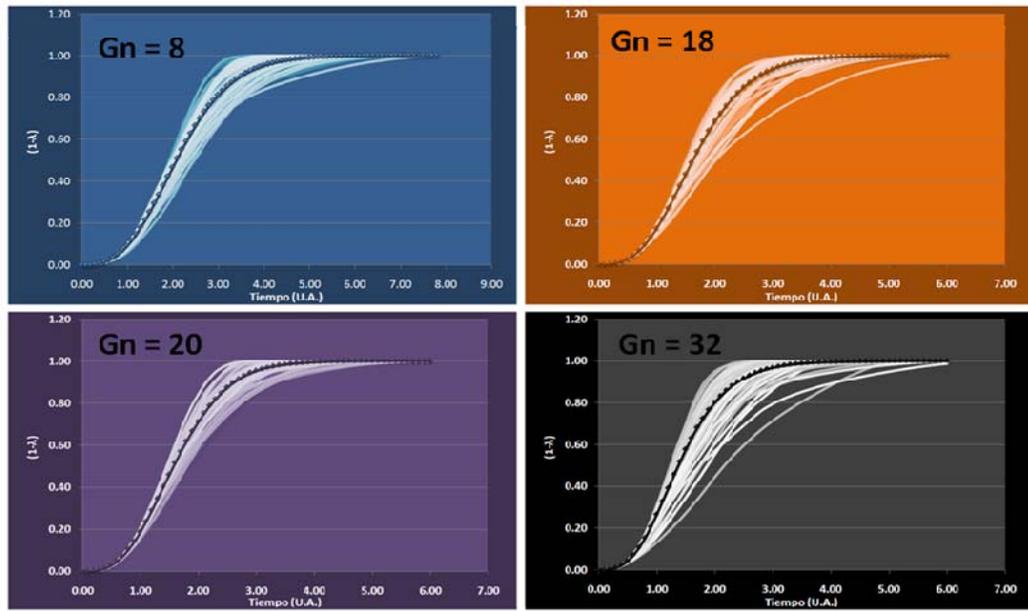


Figura 7.38. Isothermas de cristalización para G_n por unidad de área no cubierta constante (se indica en las figuras), cada una con cien repeticiones y la media indicada con línea punteada.

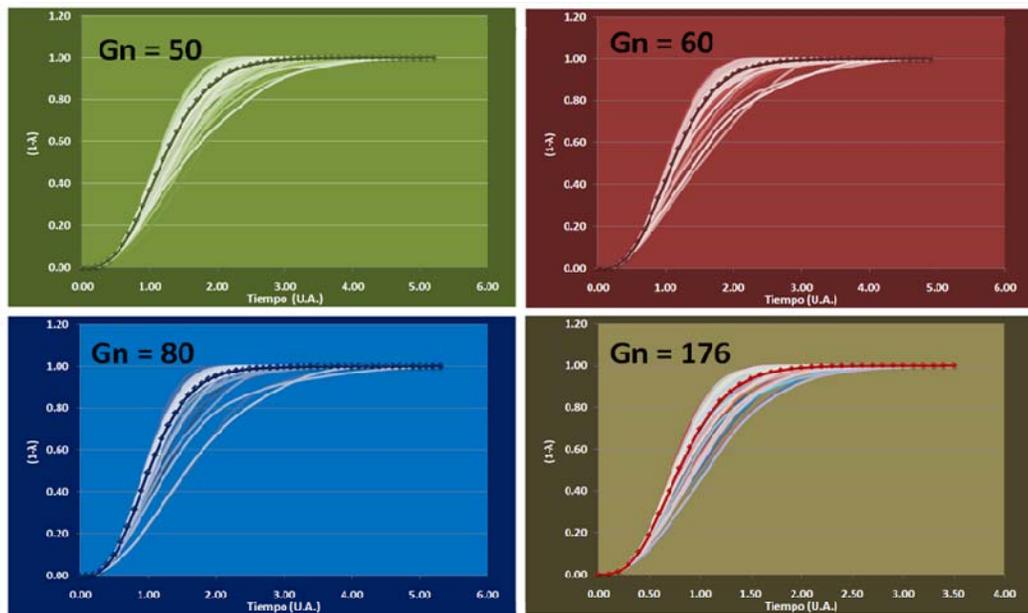


Figura 7.39. Isothermas de cristalización para las otras cuatro G_n por unidad de área no cubierta constante (se indica en las figuras), cada una con cien repeticiones y la media indicada con línea punteada.

La Fig. 7.40 muestra el comportamiento esperado de las isothermas promedio de cristalización en forma de “S” y disminuyendo el tiempo de terminación al aumentar Gn.

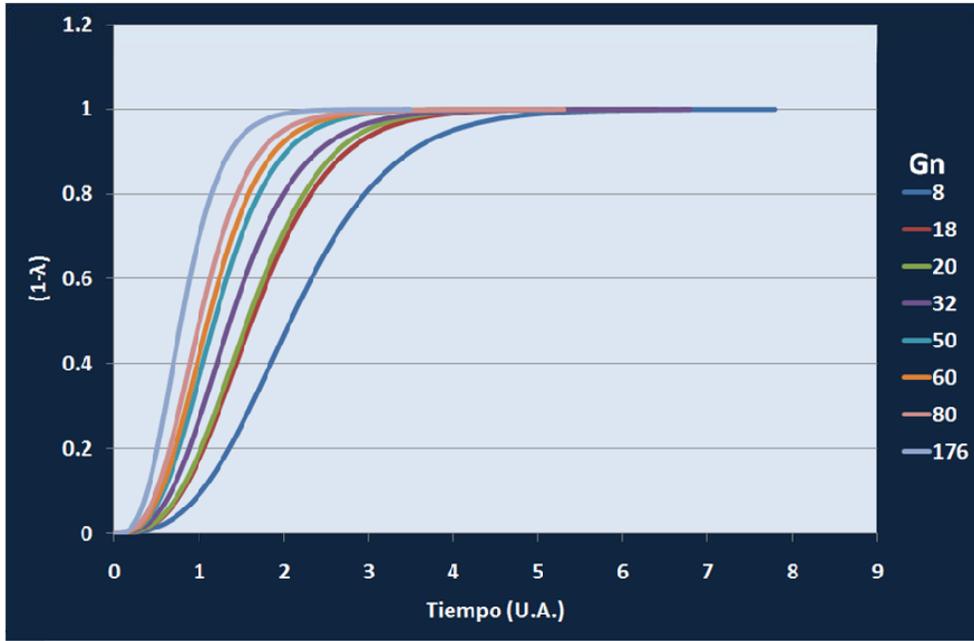


Figura 7.40. Isothermas promedio de cristalización en función de G_n

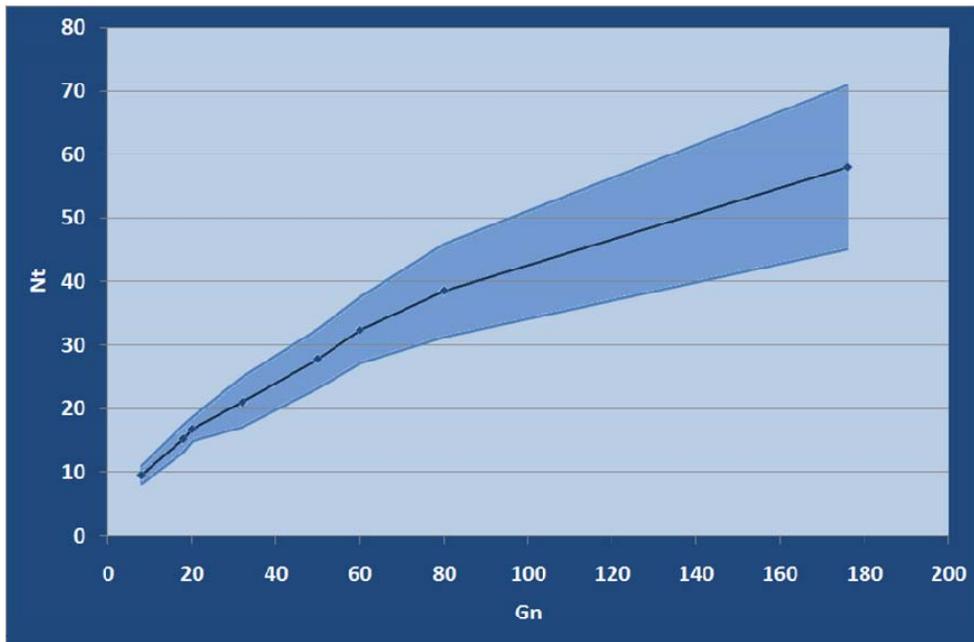


Figura 7.41 Número promedio de núcleos (N_t) y su desviación estándar en función de G_n .

Aunque en las figuras 7.42 y 7.43 las gráficas de n vs $\ln(t)$, parece haber una pendiente de cero, como lo estipula la teoría de Avrami, el acercamiento de las gráficas promedio, mostrado en la Fig. 7.44, muestra que sí hay una pendiente positiva, pero además se aprecia una subestimación del exponente respecto al valor teórico de nucleación térmica ($n = 3$).

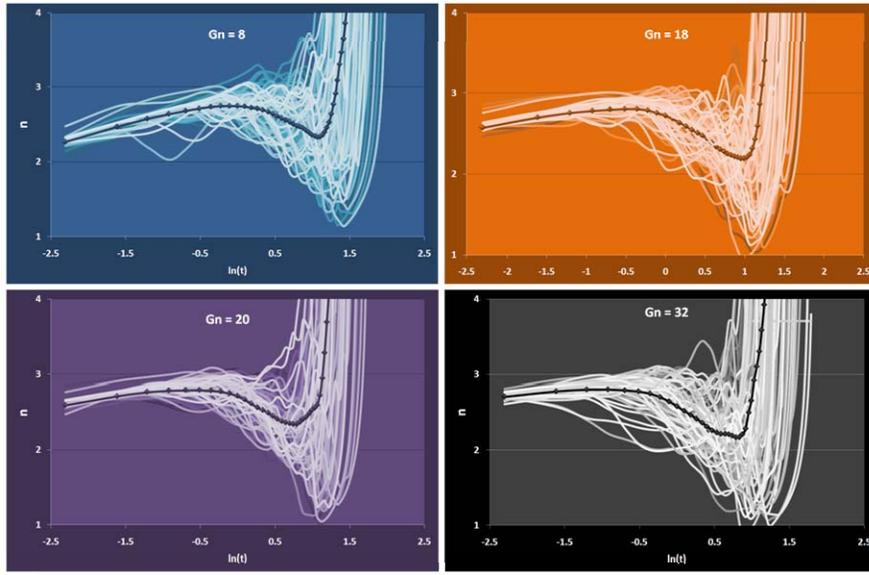


Figura 7.42 Exponente de Avrami en función del tiempo de nucleación para cada conjunto de 100 simulaciones ($8 \leq G_n \leq 32$).

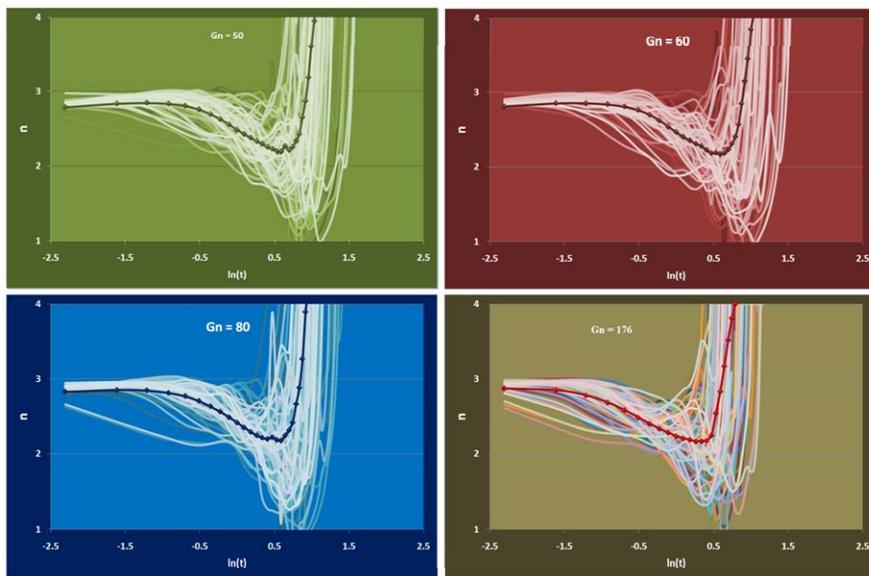


Figura 7.43 Exponente de Avrami en función del tiempo de nucleación para cada conjunto de 100 simulaciones ($50 \leq G_n \leq 174$).

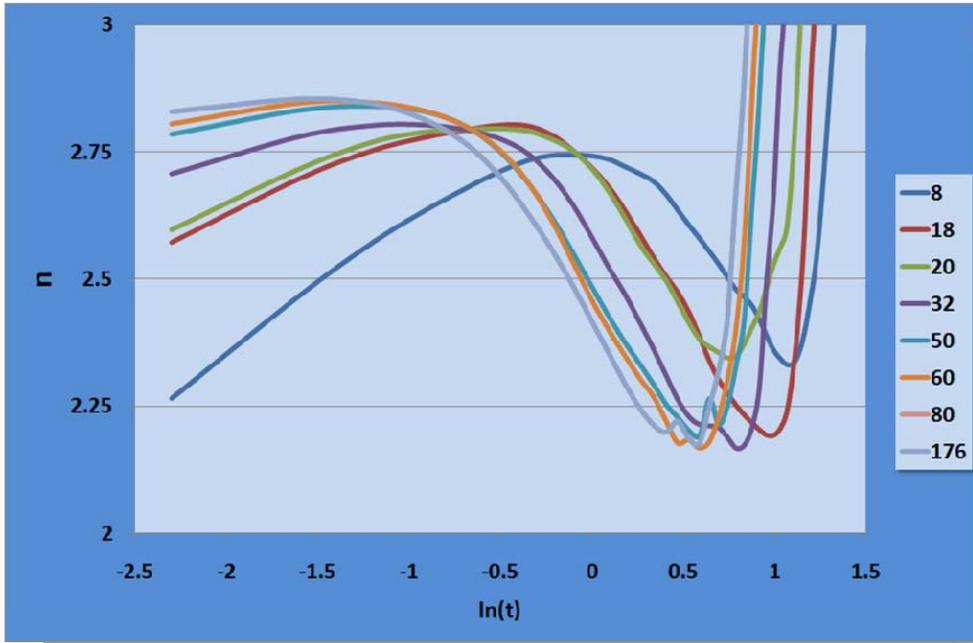


Figura 7.44 Amplificación de la región más horizontal de la función de n respecto al tiempo.

Por otra parte, los exponentes de Avrami calculados por el *método visual* y el *propuesto* parecen no diferir substancialmente (Figs. 7.45 y 7.46).

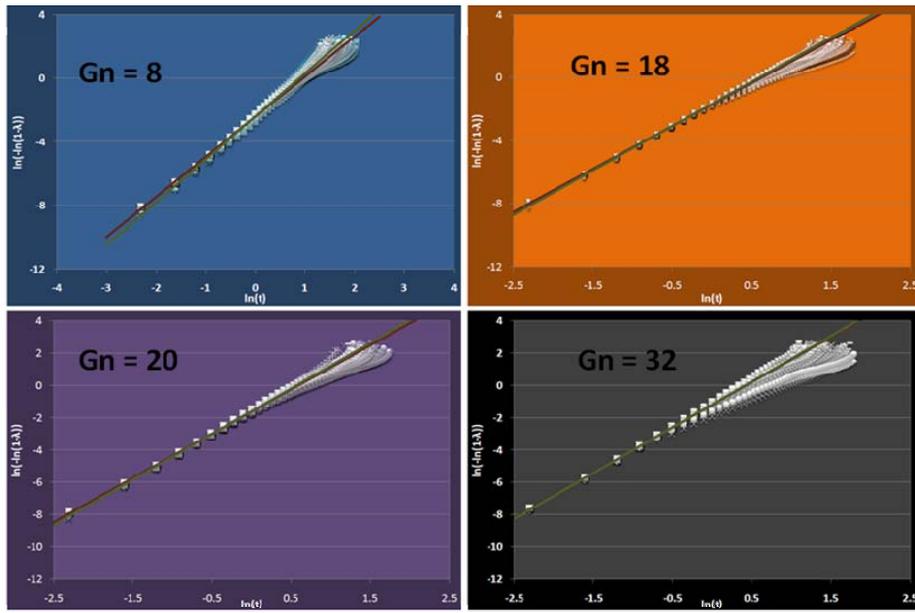


Figura 7.45. Gráficas de Avrami para los conjuntos de datos con G_n definida (se indica en los insertos)

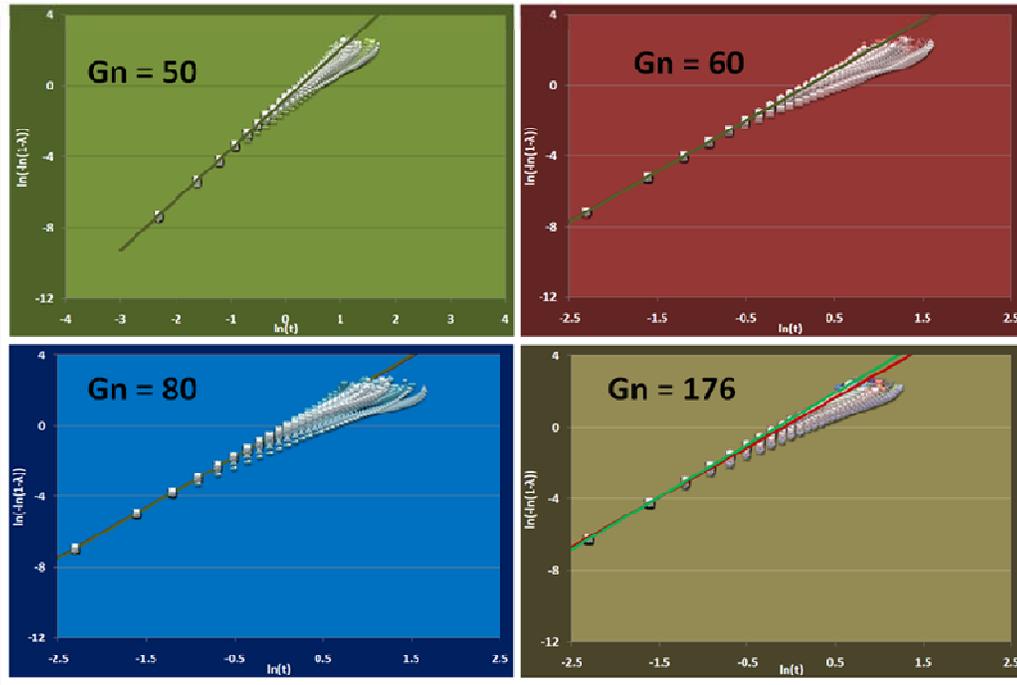


Figura 7.46. Gráficas de Avrami para los conjuntos de datos con G_n definida (se indica en los insertos)

Sin embargo, ya que “ n ” resulta de gráficas $\ln(t)$ vs. $\ln(-\ln(1-\lambda))$, los cambios de pendiente son a veces imperceptibles, también se aprecia un comportamiento menos errático cuando se hacen los cálculos mediante el *método propuesto*. (Figura 7.47). La distribución de exponentes de Avrami para cada conjunto de cien repeticiones, igual que en los casos anteriores, es más ancha cuando se calcula usando el método propuesto, aunque definitivamente el exponente de Avrami es menos subestimado (Fig. 7.48).

La mayor desventaja de usar el método visual destaca en los dos últimos puntos de la gráfica de la Fig. 7.47, el método puede ser muy errático, muy supeditado a la experiencia y conocimiento de quien lo calcula.

De nuevo parece ser que para llegar a tener resultados que alcancen el valor teórico de “ n ”, es necesario un número infinito de núcleos, lo cual no es posible hacer mediante simulación, ya que si partimos de G_n de alrededor de 170, la simulación dura ya más de un día, y el incremento del tiempo de cómputo es más que exponencial al aumentar G_n .

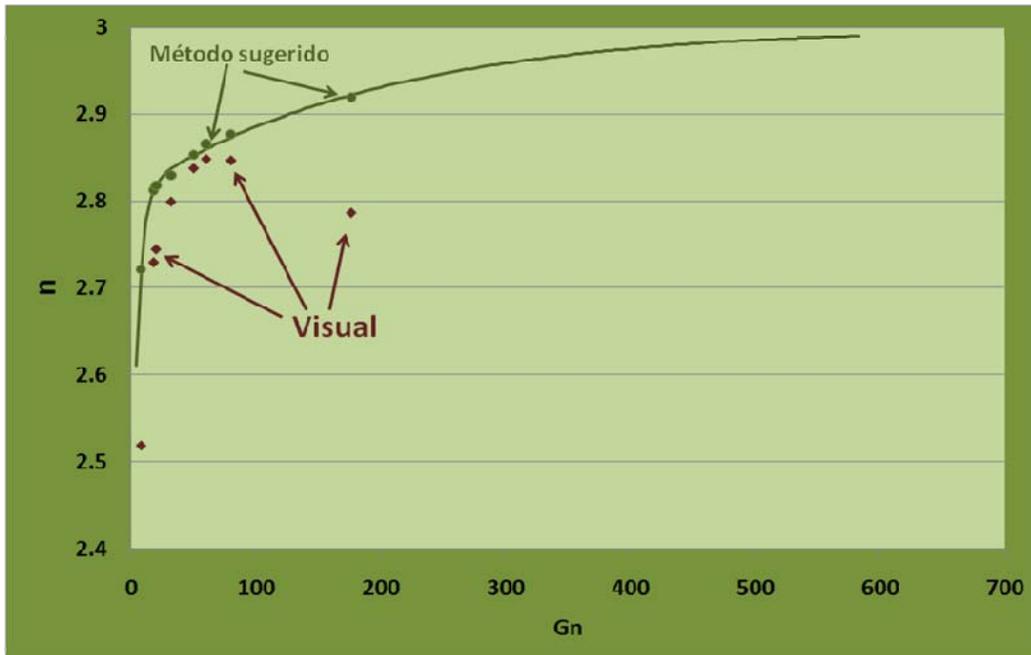


Figura 7.47. Exponente de Avrami calculado por ambos métodos, en función de Gn

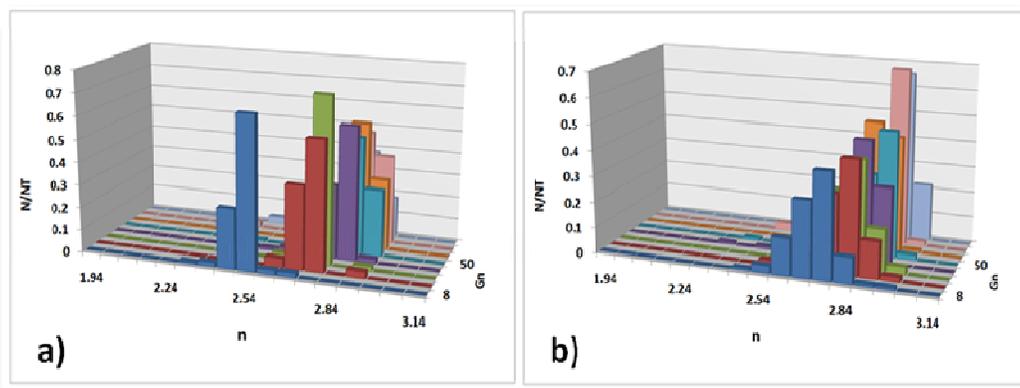


Figura 7.48 Distribuciones normalizadas del exponente de Avrami para las diferentes Gn. a) método visual, b) método propuesto.

Siguiendo el procedimiento de utilizar los vértices de las fronteras para calcular la función de nucleación, se construyeron las gráficas de la Fig. 7.49, es importante remarcar que en esta figura las líneas continuas en las gráficas son solo para guiar la vista, no son el resultado del ajuste a un determinado modelo. Pero sabiendo que en este

caso lo importante es calcular G_n , se procedió a obtener la derivada respecto al tiempo de estas gráficas. Estas derivadas se reportan en la Fig. 7.50.

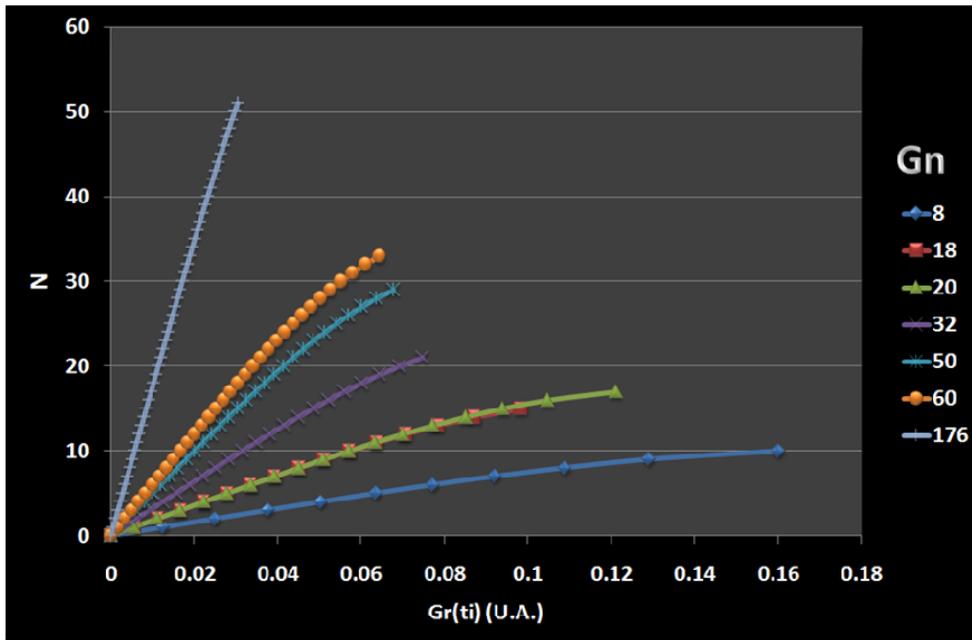


Figura 7.49 Funciones de nucleación para las diferentes G_n , las líneas continuas son solo para guiar la vista.

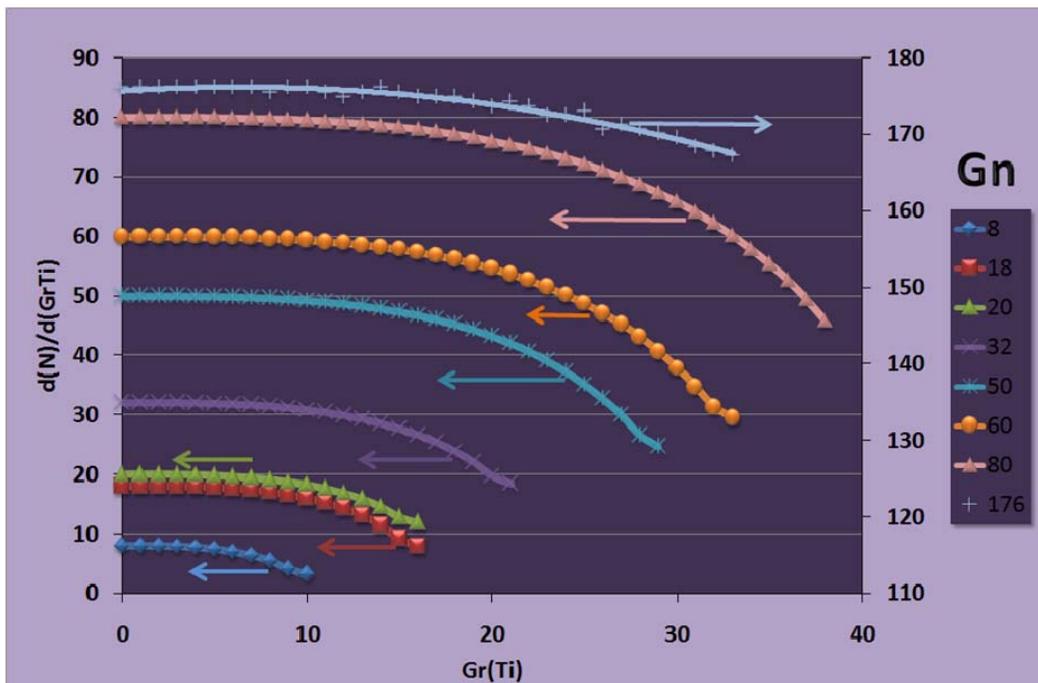


Figura 7.50. Derivadas de las funciones de nucleación respecto a $Gr(Ti)$. Las líneas continuas son solo para guiar la vista.

En esta última figura, encontramos que la derivada N respecto a Gr(ti) tiende a Gn cuando Gr(ti) tiende a cero (Ec. 7.4). En la tabla 7.5 se muestran los resultados

$$G_n = \lim_{t_i \rightarrow 0} \frac{d(N)}{d(Gr_{t_i})} \quad [7.4]$$

Tabla 7.5 Valores de Gn alimentados al programa de simulación y calculados mediante la Ec. 7.4

Alimentado	8.000	18.000	20.000	32.000	50.000	60.000	80.000	176.000
Calculado	8.000	17.999	20.000	32.010	50.000	60.024	80.000	176.056

Como se puede apreciar, al menos a partir de los resultados de la simulación, es posible determinar con gran precisión la rapidez de nucleación en términos del número de núcleos por unidad de área no cubierta.

Otro punto importante a discutir es la distribución de tamaños de núcleos en función de Gn, la figura 7.51 muestra dichas distribuciones.

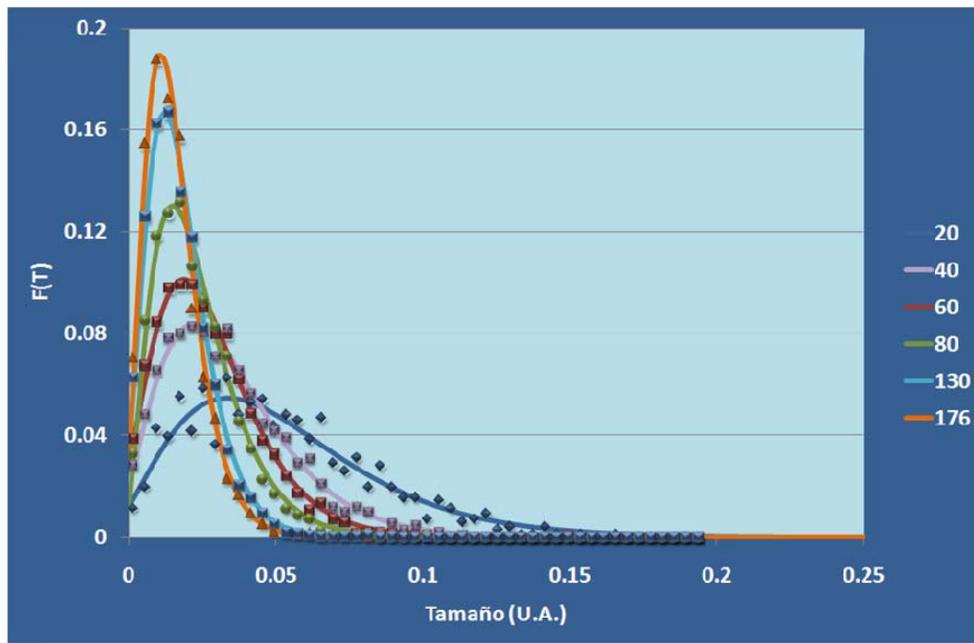


Figura 7.51. Distribuciones de tamaños de núcleos según la magnitud de Gn alimentada al programa de simulación.

Se buscó una función que ajustara a estas distribuciones y se encontró que la función pulso, (Ec. 7.5), es la que mejor describe este comportamiento, es necesario destacar que dicho ajuste es puramente empírico, aunque como se aprecia en la figura anterior el ajuste obtenido es aceptable.

$$F(N) = A \left\{ 1 - \exp \left[\frac{x_0 - x}{t_1} \right] \right\}^P \exp \left(\frac{x_0 - x}{t_2} \right) \quad [7.5]$$

Donde A, x_0 , t_1 , P y t_2 son constantes que toman los valores reportados en la tabla 7.6 para cada G_n

Tabla 7.6, Constantes de la función pulso en el ajuste de la distribución normalizada de tamaños de núcleos en función de G_n , se reporta también el coeficiente de determinación, r^2 .

Gn	20	40	60	80	130	176
x0	-0.01111	-0.00978	-0.0131	-0.00231	-0.00686	-0.0056
A	20	20	20	20	20	20
t1	0.19759	0.07859	0.03879	0.06945	0.01966	0.01655
P	2.3728	2.86722	4.31083	2.10234	4.47246	4.4427
t2	0.02115	0.0137	0.01143	0.00931	0.00718	0.00632
r²	0.93	0.995	0.995	0.995	0.995	0.995

De aquí se calcularon las razones entre el máximo de estas distribuciones y el tamaño correspondiente “ $F(N)/N$ ” y se graficaron contra G_n , encontrando la gráfica de la figura 7.52.

Esta gráfica parece una línea recta, solo que si así se considera, esta no interseca en el origen, por lo que se decidió ajustarla a una ecuación de potencia resultando con las constantes que se reportan en la ecuación 7.6.

$$G_n = 11.6 \left[\frac{F(N)}{N} \right]^{0.95} \quad [7.6]$$

La tabla 7.6 muestra los valores de G_n calculados usando este método, encontrando que es una buena aproximación a las G_n alimentadas al programa, aunque de menor exactitud que las G_n calculadas utilizando la Ec. 7.4.

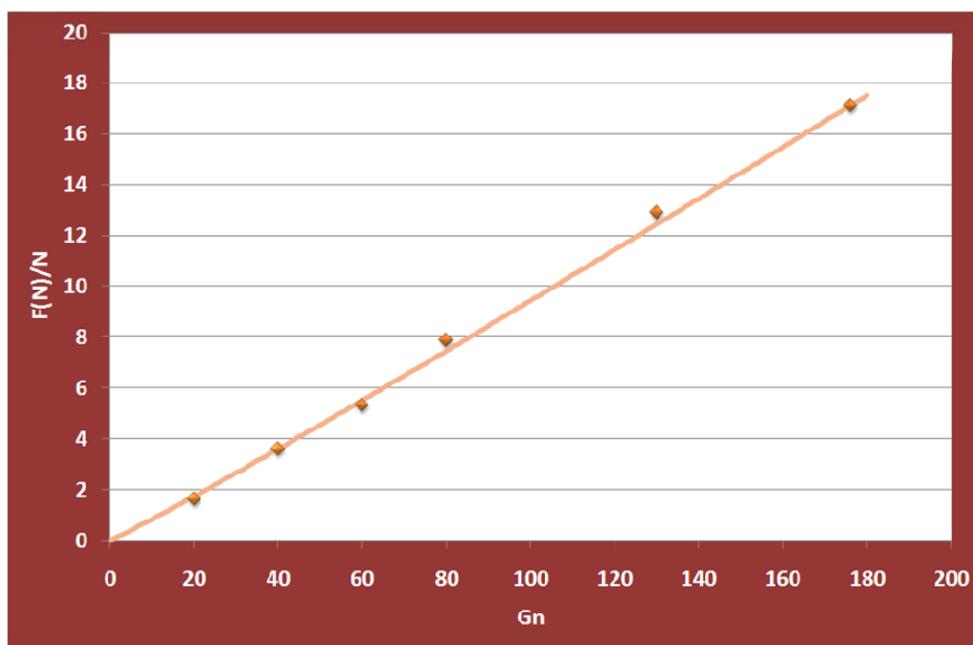


Figura 7.52. Gráfica de la razón entre la máxima fracción y el tamaño de las distribuciones de tamaño en función de Gn

Tabla 7.5 Valores de Gn alimentados al programa de simulación y calculados mediante la Ec. 7.6

Alimentado	20.0	40.0	60.0	80.0	130.0	176.0
Calculado	18.5	39.2	56.9	82.6	131.8	172.4

Experimentalmente, considerando que mediante el software comercial es relativamente sencillo obtener la distribución de tamaños de cristales, esta última técnica, aunque menos exacta, puede ser más atractiva.

7.4.- El error en mediciones experimentales

El análisis de los tamaños esferulíticos en polímeros, generalmente se desarrolla mediante microscopía óptica de transmisión, por lo que se utiliza una muestra en forma de película de menos de 1 μm de espesor, pero cuando se analizan los tamaños de grano en cerámicos o metálicos se hace mediante microscopía óptica de reflexión sobre una de las superficies de una pieza de cualquier dimensión. De acuerdo a los planteamientos aquí hechos, en tres dimensiones las fronteras intercristalinas son hiperboloides que al

simplificar el modelo a dos dimensiones resultan en hipérbolas cuya excentricidad nos proporciona información sobre la cinética de nucleación. Como vemos en una metalografía por ejemplo, o en la observación de esferulitas en películas de espesores relativamente grandes, se introducen dos errores al determinar las posiciones de los vértices.

El primero es debido al “corte” que puede estar en un plano perpendicular al plano paralelo al plano xz, pero fuera del origen (Fig. 7.53) , en este caso, entre mayor distancia al plano xz de la frontera observada, mayor será el error.

El error por la distancia “h” del plano de corte se puede calcular haciendo el siguiente tratamiento matemático:

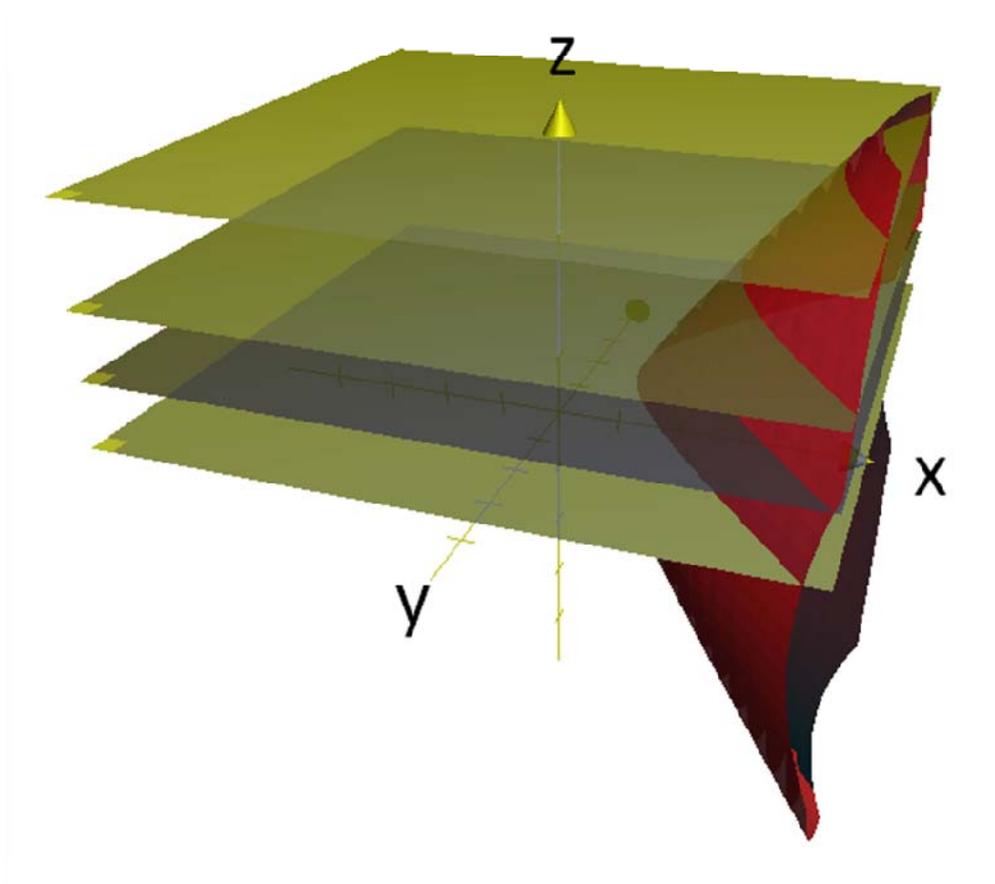


Figura 7.53. La altura del plano y la hipérbola medida.

Partimos de la ecuación del hiperboloide (Ec. 5.20)

$$\frac{x^2}{V^2} - \frac{y^2}{X^2 - V^2} - \frac{z^2}{X^2 - V^2} = 1 \quad [5.20]$$

Ya que el corte lo estamos haciendo a una altura “h”, podemos substituir “z” por “h” y se obtiene la expresión matemática:

$$\frac{x^2}{V^2} - \frac{y^2}{X^2 - V^2} = 1 + \frac{h^2}{X^2 - V^2} \quad [7.7]$$

Que reagrupando nos da:

$$\frac{(X^2 - V^2)x^2}{V^2(X^2 - V^2 - h^2)} - \frac{y^2}{X^2 - V^2 - h^2} = 1 \quad [7.8]$$

Definiendo V' como el vértice medido erróneamente y comparando [7.8] con [5.20] podemos definir:

$$V'^2 = \frac{V^2(X^2 - V^2 - h^2)}{(X^2 - V^2)} \quad [7.9]$$

Que simplificando queda:

$$V' = V \left(1 - \frac{h^2}{(X^2 - V^2)} \right)^{1/2} \quad [7.10]$$

Si definimos el error como:

$$\chi_e = 1 - \frac{V'}{V} \quad [7.11]$$

Entonces utilizando [7.10] podemos llegar fácilmente a la Ec. 7.12 como el error debido a la distancia del corte

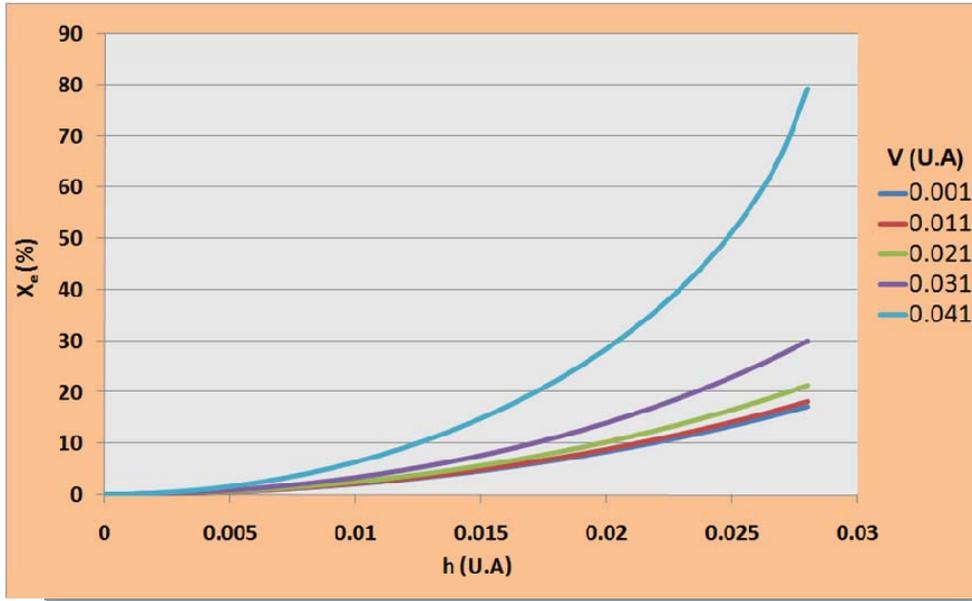


Figura 7.54. El error (en porcentaje) en la medición del vértice en función de la altura (h) y el vértice real, (se indica en el inserto).

$$\chi_e = 1 - \left(1 - \frac{h^2}{(X^2 - V^2)} \right)^{1/2} \quad \text{para } 0 \leq h^2 < X^2 - V^2 \quad [7.12]$$

La Figura 7.54 muestra como cambia el error con la altura.

Como se ve en la Fig. 7.54, el error puede ser muy grande con pequeños cambios de altura, por lo que en determinaciones experimentales, es necesario tener muestras en películas lo más delgadas posible.

Al error anterior, hay que agregarle la posible inclinación del plano respecto al plano xz, en la figura 7.55 se muestra una interpretación gráfica de esta situación. Aquí se aprecia como la geometría observada es la proyección de la hipérbola que se forma en el plano del corte, sobre el plano xz, lo que resulta ser una forma deformada de la hipérbola, donde las asíntotas no son simétricas. Esta observación nos permite saber, al menos cualitativamente cuando y en qué dirección está el plano de observación respecto a la hipérbola planteada en este modelo.

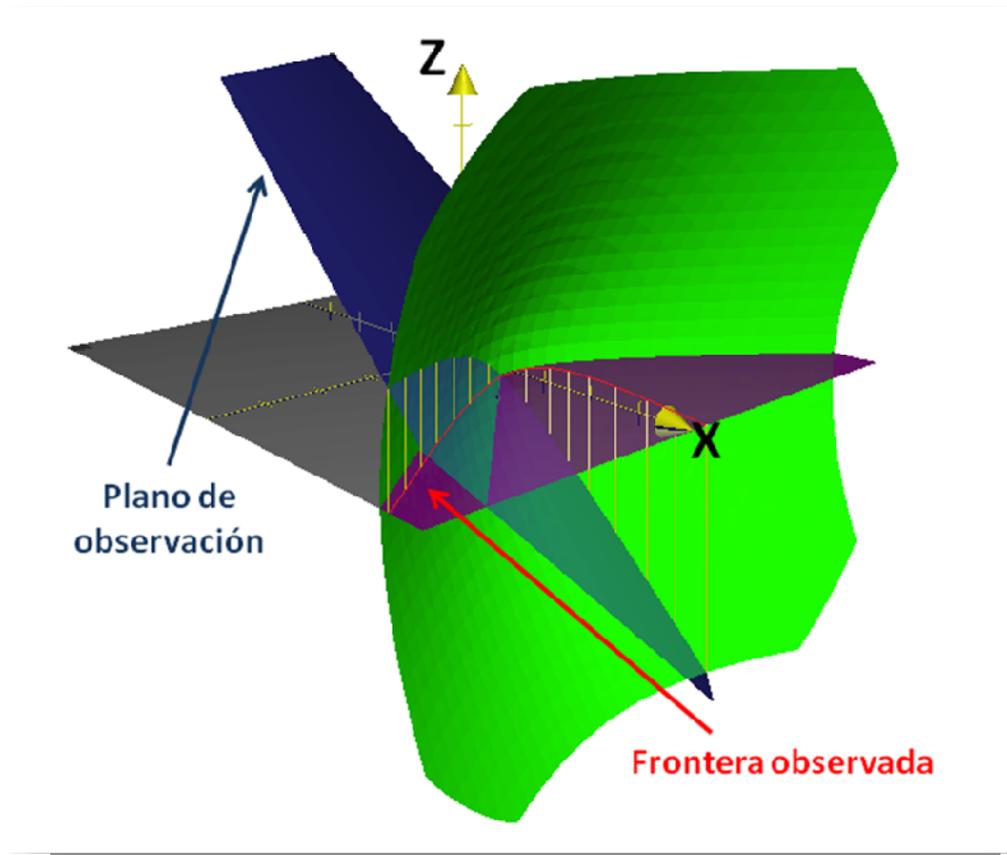


Figura 7.55. Deformación de la frontera intercristalina debido a su observación desde un plano inclinado

8.- Conclusiones

Se desarrolló un programa de computadora que simula las fronteras intercristalinas(esferulitas) del fenómeno de nucleación y crecimiento para fenómenos de nucleación térmica, las cuales son hiperbólicas, la solución del modelo se obtuvo analíticamente.

El algoritmo de generación de núcleos con una rapidez constante, definida como número de núcleos por unidad de área no cubierta, converge rápidamente.

El uso del simulador utilizando G_n expresada en términos de unidad de área no cubierta, mostró que el número de núcleos formados no es tan importante en la exactitud y precisión de los resultados.

El simulador es útil en la determinación de las funciones de nucleación en cualquier caso, ya sea cuando G_n es constante respecto al área total de la muestra, como función asintótica o como constante respecto del área residual o no cristalizada.

El simulador permite conocer los cambios en la cinética de cristalización como efecto de la rapidez de nucleación y el número de núcleos generados.

La exactitud y precisión de los resultados de la simulación aumentan con el número de núcleos generados, lo que puede implicar la necesidad de que experimentalmente sea conveniente analizar muestras con un número grande de núcleos.

Se demostró que durante la “cristalización”, el exponente de Avrami solo es constante al inicio de la “cristalización”, y bajo determinadas condiciones de las variables de G_n y N_{Max} en el caso utilizar G_n como función asintótica y en el caso de considerarla como constante respecto al área residual, siendo muy recomendable utilizar el *método propuesto* de selección del intervalo de análisis mediante la gráfica del exponente de Avrami vs. el tiempo.

El exponente de Avrami sólo es constante ó al inicio de la “cristalización” ó bajo condiciones muy específicas de G_n y N_{Max} .

Con la finalidad de tener reproducibilidad en los resultados del exponente de Avrami, es recomendable utilizar el *método propuesto* (graficando el exponente de Avrami vs. el tiempo).

El exponente de Avrami solo tiende al valor teórico de 3 cuando el número de núcleos tiende a infinito, siendo el caso de G_n constante respecto al área no cubierta, en el que más rápido se tiende a este valor.

Es posible determinar G_n constante con respecto al área residual, a partir de las excentricidades de las hipérbolas que representan las fronteras intercristalinas (mediante la ecuación 7.4), o bien mediante la estadística del tamaño de núcleos formados (Ec. 7.6)

Al hacer uso del modelo para el análisis de resultados experimentales, es muy importante tener el menor espesor posible de la muestra analizada, porque esto determina la inclusión de un error que puede alcanzar magnitudes inaceptables, tanto en cuanto al tamaño como a la forma de las fronteras.

9.- Trabajo futuro

Incluir en el simulador la nucleación atérmica y analizar los resultados cinéticos estadísticamente, comparando con la nucleación térmica y mezclas de ambas formas de nucleación

Ampliar el simulador para incluir las ecuaciones de la cinética de cristalización en función de la temperatura, para modelar la cristalización de materiales existentes.

Desarrollar el modelo de crecimiento y choque de los cristales para tres dimensiones, incluyendo la posibilidad de simular la técnica de la metalografía.

Desarrollar los modelos para condiciones dinámicas de cristalización, con técnicas de simulación y estudiar el comportamiento de las fronteras intercristalinas.

Aplicar el modelo para determinar G_n en materiales en los que no se conoce, como metálicos y cerámicos.

10.- Bibliografía

- 1.- Nittlera L.R, Alexander Conel M. O'D. "Pre-solar grains: outlook and opportunities for astrophysics", *Proceedings of the International Astronomical Union* (2006), 2, 357-360 Cambridge University Press.
- 2.- Kulmala M, "ATMOSPHERIC SCIENCE: How Particles Nucleate and Grow", *Science* 302(5647), 1000 – 1001, (2003).
- 3.- Medkour T, Ferrone F, Galactéros F, Hannaert P, "The double nucleation model for sickle cell haemoglobin polymerization: full integration and comparison with experimental data", *Acta biotheoretica* 56(1-2), 103-22, (2008).
- 4.- Sheng-Nian, Zheng L, Strachan A, Swift D. C, "Melting dynamics of superheated argon: Nucleation and growth", *The Journal of Chemical Physics*, 126, 034505 (2007).
- 5.- Woldt E, "New Kinetic Model for Primary Recrystallization of Pure Metals", *Met. And Mat. Trans A*, 32(a), 2465, (2001).
- 6.- Hu A.-M, Li M, Mao D.-L, "Crystallization Of Spodumene-Diopside In The Las Glass Ceramics With CaO And MgO Addition", *J. of Therm. Anal. and Cal*, 90(1), 185–189, (2007).
- 7.- Marangoni A. G, "On the Use and Misuse of the Avrami Equation in Characterization of the Kinetics of Fat Crystallization", *JAOCS*, 75(10), 1465, (1998).
- 8.- Toro-Vazquez J. F, Dibildox-Alvarado E, Charó-Alonso M, Herrera-Coronado V, Gómez-Aldapa C. A, "The Avrami Index and the Fractal Dimension in Vegetable Oil Crystallization", *JAOCS*, 79(9), 855, (2002).
- 9.- Saujanya C, Imai Y, Tateyama H, "Structure development and isothermal crystallization behaviour of compatibilized PET/expandable fluorine mica hybrid nanocomposite", *Polym. Bull.* 51, 85-92 (2003).
- 10.- Lümnen N, Kraska T, "Molecular dynamics investigation of homogeneous nucleation and cluster growth of platinum clusters from supersaturated vapour", *Nanotechnology*, 16 2870-2877, (2005).
- 11.- Lord Kelvin, "On the Division of Space with Minimum Partitional Area", *Philosophical Magazine*, 24(151), 503 (1887).
- 12.- Weaire D, Phelan R, "A counterexample to Kelvin's conjecture on minimal surfaces", *Phil. Mag. Lett.* 69, 107, (1994).
- 13.- Kolmogorov, A. N.; Aleksandrov, A. D., *La Matemática: Su Contenido, sus Métodos y su Significado. I* Ed. Alianza Universidad, Madrid, (1994).
- 14.- Descartes R, "Discours de la méthode", Francia, (1637)

- 15.- Lobachevski L. I, “Acerca de los principios de geometría”, *El mensajero de Kazán*, Rusia, (1829).
- 16.- Gauss C. F, “Disquisitiones generales circa superficies curva”, Göttingen, Alemania, (1828).
- 17.- Mandelbrot B, “Los objetos fractales: forma, azar y dimensión”, *Ed. Tusquets* Barcelona, (1975).
- 18.- Feder J, “Fractals”, *Ed. Premium Press*, New York, (1998).
- 19.- Grodzicki M, Seminario J. M, Politzer P, “Energy barriers of symmetry-forbidden reactions: Local density functional calculations”, *J. Chem. Phys.*, 94(2), 1668-1669, (1991).
- 20.- Russ J, “Fractal Surfaces”, *Ed. Plenum Press*, New York, (1994).
- 21.- Hinojosa M, Aldaco J. “Self-Affine Fracture Surface Parameters and Their Relationship with Microstructure in a Cast Aluminium Alloy”, *Journal of Materials Research Society*. Vol. 17 No 6, pp. 1276-1282, (2002)
- 22.- E. Reyes, C. Guerrero and M. Hinojosa, “Self-Affine Fractal Measurements on Fracture Surfaces of Polymers and Opal Glass”, *Proceedings of the 7th International Conference Fractal 2002*, World Scientific, pp. 385-391, (2002).
- 23.- Hinojosa M, Aldaco J, Ortiz U, González V, “Roughness Exponent of the Fracture Surface of Al-Si Alloy”, *Aluminum Transactions*, 3(1), 53-57, (2000).
- 24.- González V, Alanís M, Guerrero C, Ortiz U, “Surface Roughness in Dynamically Crystallized Isotactic Polypropylene Films”, *J. of Polym. Sci. Part. B, Polym. Phys.*, 42(4), 646-655, (2004).
- 25.- Smith C. S. , “Grain Shapes and other Metallurgical Applications of Topology”, *Metal Interfaces*. Ed. American Chemical Society, New York.(1952).
- 26.- Muthukumar M, “Shifting Paradigms in Polymer Crystallization”, *Lecture Notes in Physics*, 714, 1-18, (2007).
- 27.- Fatou J. G, “Crystallization Kinetics”, *Encyclopaedia of Polymer Science and Engineering, Supplement Volume*, 2nd Edition, J. Wiley and Sons, Inc., New York, (1989).
- 28.- Avila C. A, “Cristalización no-isotérmica y comportamiento en la fusión de copolímeros al azar de P(ET/CT)”, *Tesis Doctoral*, Centro de Investigación en Química Aplicada CIQA, (2001).
- 29.- Doye J. P. K, “Computer Simulations of the Mechanism of Thickness Selection in Polymer crystals”, *Polymer*, 41, 8857 – 8867, (2000).

- 30.- Phillips P. J, “Spherulitic Crystallization in Macromolecules”, *Handbook of Crystal Growth*. 2, 1168 – 1216, (1994).
- 31.- Johnson W. A, Mehl L. F, “Reaction kinetics in Processes of Nucleation and Growth”, *Trans. Amer. Inst. Mining and Mat. Eng*, 135-142, 426, (1939).
- 32.- Avrami M, “Kinetic of phase Change I”, *Chem. Phys.*, 7, 1103-1112, (1939).
- 33.- J.M. Avrami, “Kinetic of phase Change II”, *Chem. Phys.*, 8, 212-224, (1940).
- 34.- Saujanya C, Imai Y, Tateyama H, “Structure development and isothermal crystallization behaviour of compatibilized PET/expandable fluorine mica hybrid nanocomposite”, *Polymer Bulletin*, 51, 85-92, (2003).
- 35.- Gögebakan M, “The Effect of Si Addition on Crystallization Behavior of Amorphous Al-Y-Ni Alloy”, *Journal of Materials Engineering and Performance*, 13(4), 504, (2004).
- 36.- Hu A. -M, Li M, Mao D. -L, “Crystallization Of Spodumene-Diopside In The Las Glass Ceramicswith CaO And MgO Addition”, *Journal of Thermal Analysis and Calorimetry*, 90, 185–189, (2007).
- 37.- Toro-Vazquez J. F, Dibildox-Alvarado E, Charó-Alonso M, Herrera-Coronado V, Gómez-Aldapa C. A, “Avrami Index and the Fractal Dimension in Vegetable Oil Crystallization”, *Journal of the American Oil Chemistry Society*, 79(9), 855, (2002).
- 38.- Marangoni A. G, “On the Use and Misuse of the Avrami Equation in Characterization of the Kinetics of Fat Crystallization”, *Journal of the American Oil Chemistry Society*, 75(10), 1465, (1998).
- 39.- Sheng-Nian L, Zheng L, Strachan A, Swift D. C, “Melting dynamics of superheated argon: Nucleation and growth”, *The Journal Of Chemical Physics*, (2007), 126, 034505, (2007).
- 40.- Rikvold P. A, Ramos E. A, “How Good is \Avrami's Law"? Application to a kinetic Ising model in two dimensions”, *Reporte de proyecto a NSF, U. Florida*, <http://www.scri.fsu.edu/> (1997).
- 41.- Iwamatsua M., “Direct numerical simulation of homogeneous nucleation and growth in a phase-field model using cell dynamics method”, *The Journal Of Chemical Physics*, 128, 084504, (2008).
- 42.- Brandup J, Immergut E. H, Grulke E. A, *Polymer Handbook*, 4th Edition, John Wiley & Sons, New York, (1999).
- 43.- Lambrigger M, “Three Types of Avrami Master Curves for the Evaluation of Isothermal Polymer Crystallization Experiments”, *J. of Polym. Sci.; Part B: Polym. Phys.*, 35, 2559-2563, (1997).

- 44.- Di Lorenzo M. L, Silvestre C, “Non-isothermal crystallization of polymers”, *Prog. Polym. Sci.* 24, 917 – 950, **(1999)**.
- 45.- Ziabicki A, “Crystallization and melting behaviour of poly(butylene naphthalene-2,6-dicarboxylate)”, *Appl. Polym. Symp.*, 6, 1, **(1967)**.
- 46.- Ozawa,T. “Kinetics of non-isothermal crystallization”, *Polymer*, 12(3), 150-165, **(1971)**.
- 47.- Nakamura, N, Watanabe T, Katayama K, Amano T, “Some aspects of nonisothermal crystallization of polymers. I. Relationship between crystallization temperature, crystallinity, and cooling conditions”, *J. of Appl. Polym. Sci.*, 16(5), 1077-1091, **(1972)**.
- 48.- Dietz,W, “Spharolithwachstum in Polymeren”, *Colloid and Polym. Sci*, 259, 413-429, **(1981)**.
- 49.- Harnisch and Muschik, “Non-isothermal crystallization of UHMWPE filled during polymerization with inorganic fillers”, *Colloid and Polym. Sci*, 261, 908-912, **(1983)**.
- 50.- Anderson P. M, Srolovitz D. J, Grest G.S, Sahni P. S, “Computer Simulation of Grain Grow- I Kinetiks”, *Acta Metall.*, 32(5), 783-791, **(1984)**.
- 51.- Frost H, Thompson C. V, Walton D. T, “Grain Growth in Polycrystalline Materials”, ed. Abbruzzese G and Brosso P, *Brookfield, VT: Trans Tech Publications*, **(1992)**.
- 52.- Weiare D, Kermode J. P, “Computer Simulation of a 2-D Soap Froth”, *Phil. Mag. B*, 48, 245-259, **(1983)**.
- 53.- Chen L. -Q, Wang Y. -Z, “The Continuum Field Approach to Modeling Microstructural Evolution”, *JOM*, 48(11), 13–18, **(1996)**.
- 54.- Raabe D, “Grain Growth in Polycrystalline Materials III”, ed. Weiland H, Adams B. L, and Rollett A. D, *Warrendale, PA: TMS*, **(1998)**.
- 55.- Castro M, Domínguez-Adame F, Sánchez A, Rodríguez T, “Model for crystallization kinetics: Deviations from Kolmogorov-Johnson-Mehl-Avrami kinetics”, arXiv:cond-mat/9908463 v1 31 Aug **(1999)**.
- 56.- Hopkin G. H, Bhadeshia H. K. D. H, “Modeling Anisothermal Recrystallisation in Austenitic Stainless Steels”, *Proc. of the 21 Riso International Symposium on Materials Science*. 1-6, **(2000)**.
- 57.- Holm E. A., Battaile C.C., “The Computer Simulation of Microstructural Evolution. *JOM*. 53-59, **(2001)**.
- 58.- Burger M, Capasso V, Eder G, “Modelling of Polymer Crystallization in Temperature Fields”, *ZAMM Z. Angew. Math. Mech.* 82(1), 51-63, **(2002)**.

- 59.- Gránásy L, Börzsönyi T, Pusztai T, “Crystal nucleation and growth in binary phase-field theory”, *Journal of Crystal Growth*, 237-239, 1813-1817, **(2002)**.
- 60.- Jun S, Zhang H, Bechhoefer J., “Nucleation and growth in one dimension. I. The generalized Kolmogorov-Johnson-Mehl-Avrami model”, *Phys. Rev. E*, 71, 011908, **(2005)**.
- 61.- Yang J, McCoy B. J, Madras G, “Distribution kinetics of polymer crystallization and the Avrami equation”, *J. of Chem. Phys.*, 122, 064901/1-10, **(2005)**.
- 62.- Granasy L., Pusztai T., Tegze G., Warren J.A., Douglas J, “Growth and form of Spherulites”, *Physical Review, E* 72, 11605, pp. 1-14, **(2005)**.
- 63.- Vanherpe L, Moelans N, Blanpain B, Vandewalle S, “Bounding box algorithm for three-dimensional phase-field simulations of microstructural evolution in polycrystalline materials”, *Physical Review E* 76, 056702, **(2007)**.
- 64.- Iwamatsua M, “Direct numerical simulation of homogeneous nucleation and growth in a phase-field model using cell dynamics method”, *J. Of Chem. Phys.* 128, 084504, **(2008)**.
- 65.- Moelans N, Blanpain B, Wollants P, “An introduction to phase-field modeling of microstructure evolution”, *Computer Coupling of Phase Diagrams and Thermochemistry*, 32, 268-294, **(2008)**.
- 66.- Canada S. T, da Silva Assis W. L, Rangel R. P, “Simulation of Recrystallization in Iron Single Crystals”, *Materials Research*, 11(1), 109-115, **(2008)**.
- 67.- Dillon S. J, Rohrer G. S, “Mechanism for the development of anisotropic grain boundary character distributions during normal grain growth”, *Acta Materialia*, 57, 1-7, **(2009)**.
- 68.- Lehmann B, Karger-Kocsis J., “Isothermal and Non-Isothermal Crystallization Kinetics of pCBT and PBT Polymers as studied by DSC”, *Journal of Thermal Analysis and Calorimetry*, Vol. 95, 221-227, **(2009)**.
- 69.- Galeski A, Piórkowska E, “Method of Determining the Kinetics of Spherulitic Primary Nucleation from the Spherulite Shapes in Bulk Samples”, *Polymer Bulletin*, 2(1), 1-6, **(1980)**.
- 70.- Galeski A, “Computer simulation of two-dimensional spherulite growth”, *J. Polym. Sci. Polym. Phys.*, 19, 721-730, **(1981)**.
- 71.- González V. A, Guerrero S. C, Aguilar G. J, “Nucleación y crecimiento unidimensional Parte I. El modelo y su validación”, *Ingenierías*, 5(15), 38-43, **(2002)**.
- 72.- González V. A, Guerrero S. C, Aguilar G. J, “Nucleación y crecimiento unidimensional Parte II. Análisis de la ecuación de Avrami”, *Ingenierías*, 5(17), 46-50, **(2002)**.

73.- Garza F. J, Pinales M. A, González V. A, Hinojosa R. M, “Geometría de Fronteras Intercristalinas: Un Modelo Alternativo”, *Ingenierías*, 8(26), 6-11, (2005).

Índice de figuras

No.	Descripción	Página
1.1	a) Semillas de diente de león, b) esferulita de poli-cis-1,4-isopreno cristalizado durante 2 Hrs. a $-26\text{ }^{\circ}\text{C}$, observada mediante TEM (25,000X).	3
1.2	Diagrama esquemático de la arquitectura de las laminitas	4
1.3	Estructura Weaire-Phelan formada por tetracaidecaedros [12].	5
2.1	Rapidez de cristalización (Gr) en función de la temperatura, de acuerdo a la Ecuación de Hoffman (Ec. 2.1).	9
2.2	Representación esquemática del proceso de crecimiento de una laminita según la teoría de la superficie rugosa.	10
2.3	Conceptualización de la formación de un núcleo secundario en el crecimiento laminar de polímeros. L_0 es el ancho del cristal, ξ el espesor del cristal, a y b las dimensiones ortogonales del cristal, σ y σ_e las energías libres de superficie, g la rapidez de crecimiento lateral y Gr la rapidez de crecimiento laminar	11
2.4	Los regímenes de nucleación. g : rapidez de crecimiento lateral, i : rapidez de formación de núcleos en la dirección de crecimiento laminar	12
2.5	Simulación mediante el modelo de vértices del crecimiento policristalino isotrópico [56]	16
2.6	El exponente de Avrami (p) y el número de núcleos normalizados al máximo número obtenido ($n = N/N_{\max}$) según el modelo de campo de fases binario [57].	17
2.7	Variaciones en la forma del sólido en función del grado de conversión (X_c), según el modelo de campo de fases binario en condiciones isotrópicas, de izquierda a derecha $X_c=0.2, 0.5$ y 0.8 [57].	18
2.8	Resultado de la simulación tridimensional hecha por Vanherpe y Col.[60], mostrando la irregularidad de las fronteras.	19
2.9	Microestructuras obtenidas mediante simulación utilizando un autómata celular para la recristalización del hierro [63].	20
2.10	Exponente de Avrami calculado en función del número de núcleos formados atómicamente, expresado en fracción de núcleos iniciales (Φ_h). [69]	21
5.1	Coordenadas de los núcleos 1 y 2 en el espacio y tiempo.	26
5.2	Se muestra la superficie representada por la Ec. 5.20, hiperboloide con vértice en $(V,0,0)$.	30
5.3	El plano $x=0$ como frontera en nucleación atómica	31
5.4	Hipérbola como frontera de dos núcleos formados en diferentes momentos (Ec. 5.25).	32
5.5	El eje y como frontera de de dos núcleos formados al mismo tiempo (nucleación atómica).	33
5.6	Los dos casos para la determinación de límites de las fronteras, caso a) las fronteras quedan definidas por la intersección de dos círculos, caso b) una de las fronteras está definida como en el caso 1 y la otra por la intersección de tres círculos.	34
5.7	Delimitación del área A_1	35
5.8	Área del semicono truncado por la hipérbola (A_h)	37
5.9	Definición del Área A'_h	38

No.	Descripción	Página
6.1	Forma de la muestra virtual mostrando la posición de dos núcleos con centros en X_{c_i}, Y_{c_i} y X_{c_j}, Y_{c_j} , y una frontera hiperbólica con vértice en V_x, V_y	39
6.2	Núcleos e hipérbola sobre el eje x mostrándose algunas variables de Hpp	41
6.3	Determinación de taponamiento de las fronteras 4 y 5 por las fronteras 1 a 3 previamente formadas. Frontera 4 se desecha, frontera 5 se acepta.	47
7.1	Seis imágenes de instantáneas en la simulación con $Gr = 0.1$ y $Gn = 0.8$.	49
7.2	a) Gráfica de isotermas de cristalización en 100 simulaciones utilizando $Gr = 0.1$ y $Gn = 0.8$. La línea continua roja representa la isoterma promedio. b) Gráfica de Avrami, $n = 1.96$ utilizando el método visual y $n = 2.61$ el propuesto en este trabajo.	50
7.3	Histograma de valores de exponentes de Avrami, curva normalizada. ($Gr = 0.1, Gn = 0.8$).	51
7.4	Gráfica del exponente de Avrami en función del logaritmo del tiempo para la simulación con $Gr = 0.1$ y $Gn = 0.8$. Las líneas rectas representan el tiempo de formación de núcleos.	51
7.5	Gráfica de número de núcleos vs. el tiempo modificado por la rapidez de crecimiento.	53
7.6	Seis imágenes de instantáneas en la simulación con $Gr = 0.1$ y $Gn = 15.0$.	54
7.7	Seis imágenes de instantáneas en la simulación con $Gr = 0.1$ y $Gn = 52.0$.	54
7.8	Isotermas de cristalización para las 100 repeticiones y curva promedio (línea continua en rojo), de las simulaciones con $Gr = 0.1$ y Gn entre 1.8 y 8.5 (se indica en las gráficas).	55
7.9	Isotermas de cristalización para las 100 repeticiones y curva promedio (línea continua en rojo), de las simulaciones con $Gr = 0.1$ y Gn entre 9.6 y 30 (se indica en las gráficas).	56
7.10	Isotermas de cristalización para las 100 repeticiones y curva promedio (línea continua en rojo), de las simulaciones con $Gr = 0.1$ y Gn entre 36 y 52 (se indica en las gráficas).	56
7.11	Isotermas de cristalización para las 13 condiciones de rapidez de nucleación, $Gr = 0.1$, Gn se indica en la figura.	57
7.12	Derivadas de las gráficas de Avrami promedio para corridas con $Gr = 0.1$ y diversas Gn (indicadas en la figura)	58
7.13	Gráficas de Avrami para las 100 repeticiones y regresión de la curva promedio (línea continua en rojo), de las simulaciones con $Gr = 0.1$ y Gn entre 1.8 y 8.5 (se indica en las gráficas). La línea verde es la regresión a la Ec. de Avrami de acuerdo al método aquí propuesto.	59
7.14	Gráficas de Avrami para las 100 repeticiones y regresión de la curva promedio (línea continua en rojo), de las simulaciones con $Gr = 0.1$ y Gn entre 9.5 y 30 (se indica en las gráficas). La línea verde es la regresión a la Ec. de Avrami de acuerdo al método aquí propuesto.	60
7.15	Gráficas de Avrami para las 100 repeticiones y regresión de la curva promedio (línea continua en rojo), de las simulaciones con $Gr = 0.1$ y Gn entre 36 y 52 (se indica en las gráficas). La línea verde es la regresión a la Ec. de Avrami de acuerdo al método aquí propuesto.	60

No.	Descripción	Página
7.16	Exponentes de Avrami en función del número promedio de núcleos generados, tanto para el cálculo de apreciación visual como de la derivada.	62
7.17	Histogramas normalizados de las magnitudes de n , cuando el rango de ajuste se determina de la forma tradicional. Las 13 curvas con diferentes G_n están en profundidad.	63
7.18	Histogramas normalizados de las magnitudes de n , cuando el rango de ajuste se determina mediante la derivada de la gráfica de Avrami. Las 13 curvas con diferentes G_n están en profundidad.	63
7.19	Imágenes de simulación para tres tiempos (se indican en figura), con $G_n = 0.02$, asíntotica y dos número máximo de núcleos.	65
7.20	Imágenes finales de las simulaciones de diferente G_n y N_{Max} , valores que se indican en la figura.	65
7.21	Isotermas de cristalización para las simulaciones con función asíntotica con $G_n = 0.02$ y número variable de N_{Max} (Se indican en el inserto). La línea roja es la isoterma promedio de 100 repeticiones.	66
7.22	Isotermas de cristalización para las simulaciones con función asíntotica con $G_n = 0.2$ y número variable de N_{Max} (Se indican en el inserto). La línea roja es la isoterma promedio de 100 repeticiones.	66
7.23	Isotermas de cristalización para las simulaciones con función asíntotica con $G_n = 0.5$ y número variable de N_{Max} (Se indican en el inserto). La línea roja es la isoterma promedio de 100 repeticiones.	67
7.24	Isotermas de cristalización para las simulaciones con función asíntotica con $G_n = 1.0$ y número variable de N_{Max} (Se indican en el inserto). La línea roja es la isoterma promedio de 100 repeticiones.	67
7.25	Isotermas de cristalización para las simulaciones con función asíntotica con $G_n = 1.9$ y número variable de N_{Max} (Se indican en el inserto). La línea roja es la isoterma promedio de 100 repeticiones.	68
7.26	Efecto del número máximo de núcleos a formarse sobre las isotermas de nucleación promedio para las diversas rapidezces de nucleación (se indica en insertos).	68
7.27	Función del exponente de Avrami respecto al tiempo para las curvas promedio de todas las simulaciones. (se indica G_n y N_{Max} en los insertos), En negro se marca los datos utilizados para el <i>método propuesto</i> .	69
7.28	Gráficas de Avrami para las simulaciones cien repeticiones de nucleación térmica con $Gr = 0.1$, $G_n = 0.02$ y N_{Max} variable (se indican en insertos).	70
7.29	Gráficas de Avrami para las simulaciones cien repeticiones de nucleación térmica con $Gr = 0.1$, $G_n = 0.2$ y N_{Max} variable (se indican en insertos).	71
7.30	Gráficas de Avrami para las simulaciones cien repeticiones de nucleación térmica con $Gr = 0.1$, $G_n = 0.5$ y N_{Max} variable (se indican en insertos)	71

No.	Descripción	Página
7.31	Gráficas de Avrami para las simulaciones cien repeticiones de nucleación térmica con $Gr = 0.1$, $Gn = 1.0$ y N_{Max} variable (se indican en insertos).	72
7.32	Gráficas de Avrami para las simulaciones cien repeticiones de nucleación térmica con $Gr = 0.1$, $Gn = 1.9$ y N_{Max} variable (se indican en insertos)	72
7.33	Número de núcleos vs. el producto $Gr(t_i)$ y su ajuste a la Ec. 6.2 para $Gn = 1.9$ y N_{Max} variable (se indica en el inserto).	74
7.34	Ajustes de los datos de número de núcleos vs. $Gr(t_i)$, calculado de los vértices de las fronteras, a la ecuación 6.2 con las siguientes variables de entrada: a). $Gn = 0.02$, $N_{Max} = 1100$, b) $Gn = 0.2$, $N_{Max} = 250$, c) $Gn = 0.5$, $N_{Max} = 95$, d) $Gn = 1.0$, $N_{Max} = 100$, e) $Gn = 1.9$, $N_{Max} = 100$	75
7.35	Distribuciones normalizadas del número de núcleos en función de Gn y N_{Max} .	76
7.36	Convergencia del algoritmo de generación de núcleos (rapidez “ Gn ” y tiempo de nucleación “ t_i ”), cada escalón es un núcleo, (se indican los últimos 10). Gn inicial es de 176.	77
7.37	Cuatro instantáneas de la simulación con $Gn = 176$ núcleos por unidad de área no cubierta.	78
7.38	Isotermas de cristalización para Gn por unidad de área no cubierta constante (se indica en las figuras), cada una con cien repeticiones y la media indicada con línea punteada.	79
7.39	Isotermas de cristalización para las otras cuatro Gn por unidad de área no cubierta constante (se indica en las figuras), cada una con cien repeticiones y la media indicada con línea punteada.	79
7.40	Isotermas promedio de cristalización en función de Gn .	80
7.41	Número promedio de núcleos (Nt) y su desviación estándar en función de Gn .	80
7.42	Exponente de Avrami en función del tiempo de nucleación para cada conjunto de 100 simulaciones ($8 \leq Gn \leq 32$).	81
7.43	Exponente de Avrami en función del tiempo de nucleación para cada conjunto de 100 simulaciones ($50 \leq Gn \leq 174$).	81
7.44	Amplificación de la región más horizontal de la función de n respecto al tiempo.	82
7.45	Gráficas de Avrami para los conjuntos de datos con Gn definida (se indica en los insertos).	82
7.46	Gráficas de Avrami para los conjuntos de datos con Gn definida (se indica en los insertos).	83
7.47	Exponente de Avrami calculado por ambos métodos, en función de Gn	84
7.48	Distribuciones normalizadas del exponente de Avrami para las diferentes Gn . a) método visual, b) método propuesto.	84
7.49	Funciones de nucleación para las diferentes Gn , las líneas continuas son solo para guiar la vista.	85
7.50	Derivadas de las funciones de nucleación respecto a $Gr(T_i)$. Las líneas continuas son solo para guiar la vista.	85
7.51	Distribuciones de tamaños de núcleos según la magnitud de Gn alimentada al programa de simulación.	86

No.	Descripción	Página
7.52	Gráfica de la razón entre la máxima fracción y el tamaño de las distribuciones de tamaño en función de Gn.	88
7.53	La altura del plano y la hipérbola medida.	89
7.54	El error (en porciento) en la medición del vértice en función de la altura (h) y el vértice real, (se indica en el inserto).	91
7.55	Deformación de la frontera intercrystalina debido a su observación desde un plano inclinado.	92

Índice de esquemas

No.	Descripción	Página
2.1	Los modelos de cristalización en polímeros	8
6.1	Diagrama de bloques general	42
6.2	Diagrama de bloques de la generación aleatoria de núcleos con rapidez constante y asintóticos a N_{Max} .	43
6.3	Diagrama de bloques de la generación aleatoria de núcleos con rapidez constante respecto al área no cubierta por los núcleos previos.	44
6.4	Diagrama de bloques de proceso de generación de fronteras.	45
6.5	Diagrama de bloques de proceso para la determinación de intersecciones de las fronteras con la muestra.	46

Índice de tablas

No.	Descripción	Página
2.1	Magnitudes de las constantes de Avrami (K y n) en función de dimensionalidad de crecimiento, tipo de nucleación y rapidez de crecimiento.	13
7.1	Cálculo de la función de nucleación a partir de los vértices de las fronteras formadas mediante una simulación con $Gr = 0.1$ y $Gn = 8$.	52
7.2	Número de núcleos promedio, su desviación estándar y exponentes de Avrami calculados mediante los métodos <i>visual</i> y <i>propuesto</i> , para cada conjunto de experimentos con diferente Gn .	61
7.3	Variables Gn y N_{Max} de las simulaciones realizadas a Gn constante	64
7.4	Resultados del número promedio de núcleos y exponente de Avrami y su desviación estándar para todas las simulaciones hechas	73
7.5	Valores de Gn alimentados al programa de simulación y calculados mediante la Ec. 7.4	86
7.6	Constantes de la función pulso en el ajuste de la distribución normalizada de tamaños de núcleos en función de Gn , se reporta también el coeficiente de correlación.	87

Apéndice A

Determinación de la intersección entre dos fronteras inter-esferulíticas

Demostración Matemática

Se parte de la ecuación [1] al [8] que representan dos círculos (Ecs. [1] y [2]) con centros sobre el eje de las equis y en posiciones equidistantes al origen, un tercer círculo fuera de los ejes coordenados (Ec. [3]); las funciones de crecimiento lineal de los tres círculos (Ecs. [4], [5] y [6]); La ecuación de la hipérbola resultado del encuentro entre los dos primeros círculos (Ec. [7]) y la ecuación del vértice (Ec. [6]).

Después se parametriza con el tiempo la ecuación de la hipérbola para resultar las ecuaciones ([17] y [20], las cuales se substituyen en la ecuación del tercer círculo (Ec. [3]); al hacer operaciones algebraicas se llega a un polinomio de segundo orden (Ec. [41]) la cual, aunque sería conveniente intentar simplificar, puede resolverse fácilmente.

Utilizando los datos de la tabla 1, se encuentra como resultado el tiempo de intersección de $t_i = 7.9529$

Tabla 1 Datos de la simulación de ejemplo

X_1	1
X_3	1.5
Y_3	1.2
Gr	0.2
t_1	0
t_2	4
t_3	2

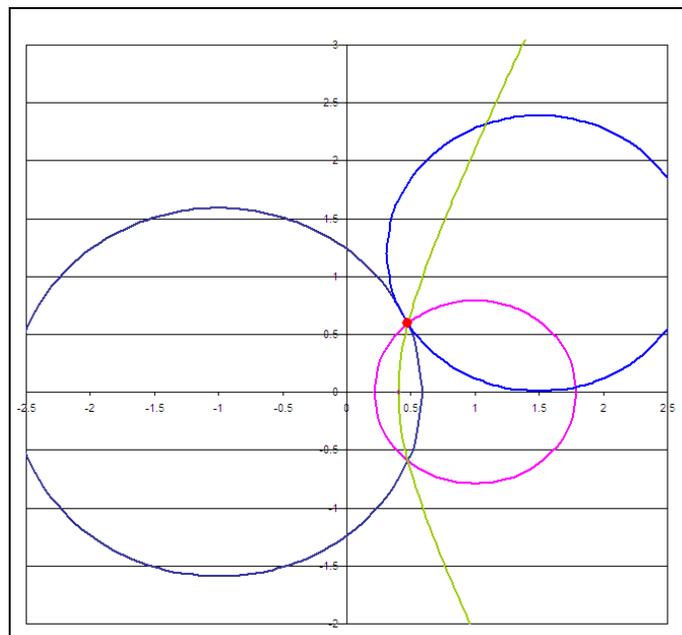


Fig. 1 Resultado de aplicar las Ecuaciones aquí deducidas al ejemplo de la tabla 1

$$(x + \bar{X})^2 + y^2 = r_1^2$$

$$(x - \bar{X})^2 + y^2 = r_2^2$$

$$(x - X_3)^2 + (y - Y_3)^2 = r_3^2$$

$$r_1 = G_r(t - t_1)$$

$$r_2 = G_r(t - t_2)$$

$$r_3 = G_r(t - t_3)$$

$$\frac{x^2}{V^2} - \frac{y^2}{\bar{X}^2 - V^2} = 1$$

$$V = \frac{G_r(t_2 - t_1)}{2}$$

Definimos:

$$D_1 = \bar{X}^2 - V^2$$

De 1 y 2

$$x^2 + 2\bar{X}x + \bar{X}^2 + y^2 = r_1^2$$

$$x^2 - 2\bar{X}x + \bar{X}^2 + y^2 = r_2^2$$

Restando [10] a [9]

$$4\bar{X}x = r_1^2 - r_2^2$$

[1]

[2]

[3]

[4]

[5]

[6]

[7]

[8]

[8a]

[9]

[10]

[11]

Substituyendo [4] y [5] en [11]

$$4\bar{X}x = G_r^2(t - t_1)^2 - G_r^2(t - t_2)^2 \quad [12]$$

$$x = \left(\frac{G_r^2}{4\bar{X}} \right) \left[(t^2 - 2t_1t + t_1^2) - (t^2 - 2t_2t + t_2^2) \right] \quad [13]$$

$$x = \left(\frac{G_r^2}{4\bar{X}} \right) \left[(2t_2 - 2t_1)t - (t_2^2 - t_1^2) \right] \quad [14]$$

$$x = \left(\frac{G_r^2(t_2 - t_1)}{2\bar{X}} t - \frac{G_r^2(t_2 - t_1)(t_2 + t_1)}{4\bar{X}} \right) \quad [15]$$

Usando [8] en [15]

$$x = \frac{G_r V}{\bar{X}} t - \frac{G_r V(t_2 + t_1)}{2\bar{X}} \quad [16]$$

Reagrupando

$$x = \left(\frac{G_r V}{\bar{X}} \right) (t - t') \quad \text{Donde} \quad t' = \frac{t_1 + t_2}{2} \quad [17]$$

Despejando y de [7] y [8a]

$$y^2 = \left(\frac{\bar{X}^2}{V^2} - 1 \right) x^2 - D_1 \quad [18]$$

Substituyendo [17] en [18]

$$y^2 = \left(\frac{\bar{X}^2}{V^2} - 1 \right) \left(\frac{G_r V}{\bar{X}} \right)^2 (t-t')^2 - D_1 \quad [19]$$

$$y^2 = D_1 \left[\frac{G_r^2}{X_1^2} (t-t')^2 - 1 \right] \quad [20]$$

Substituyendo [17] y [20] en [3]

$$\left(\left(\frac{G_r V}{\bar{X}} \right) (t-t') - X_3 \right)^2 + \left(D_1^{\frac{1}{2}} \left[\frac{G_r^2}{\bar{X}^2} (t-t')^2 - 1 \right]^{1/2} - Y_3 \right)^2 = r_3^2 \quad [21]$$

$$\left(\frac{G_r V}{\bar{X}} \right)^2 (t-t')^2 - 2X_3 \left(\frac{G_r V}{\bar{X}} \right) (t-t') + X_3^2 + D_1 \left[\frac{G_r^2}{\bar{X}^2} (t-t')^2 - 1 \right] - 2Y_3 D_1^{\frac{1}{2}} \left[\frac{G_r^2}{\bar{X}^2} (t-t')^2 - 1 \right]^{1/2} + Y_3^2 - r_3^2 = 0 \quad [22]$$

$$\left(\frac{G_r V}{\bar{X}} \right)^2 (t^2 - 2t't + t'^2) - 2X_3 \left(\frac{G_r V}{\bar{X}} \right) t + 2X_3 \left(\frac{G_r V}{\bar{X}} \right) t' + X_3^2 + D_1 \left[\frac{G_r^2}{\bar{X}^2} (t^2 - 2t't + t'^2) - 1 \right] - 2Y_3 D_1^{\frac{1}{2}} \left[\frac{G_r^2}{\bar{X}^2} (t-t')^2 - 1 \right]^{1/2} + Y_3^2 - r_3^2 = 0 \quad [23]$$

$$\left(\frac{G_r V}{\bar{X}} \right)^2 (t^2 - 2t't + t'^2) - 2X_3 \left(\frac{G_r V}{\bar{X}} \right) t + 2X_3 \left(\frac{G_r V}{\bar{X}} \right) t' + X_3^2 + D_1 \left(\frac{G_r^2}{\bar{X}^2} \right) (t^2 - 2t't + t'^2) - D_1 - 2Y_3 D_1^{\frac{1}{2}} \left[\frac{G_r^2}{\bar{X}^2} (t-t')^2 - 1 \right]^{1/2} + Y_3^2 - G_r^2 (t-t_3)^2 = 0$$

[24]

$$\left[\left(\frac{G_r V}{\bar{X}} \right)^2 t^2 - 2t' \left(\frac{G_r V}{\bar{X}} \right) t + \left(\frac{G_r V t'}{\bar{X}} \right)^2 - 2X_3 \left(\frac{G_r V}{\bar{X}} \right) t + 2X_3 \left(\frac{G_r V t'}{\bar{X}} \right) + X_3^2 + D_1 \left(\frac{G_r^2}{\bar{X}^2} \right) t^2 - 2t' D_1 \left(\frac{G_r^2}{\bar{X}^2} \right) t + D_1 \left(\frac{G_r^2}{\bar{X}^2} \right) t'^2 - D_1 - 2Y_3 D_1^{\frac{1}{2}} \left[\frac{G_r^2}{\bar{X}^2} (t-t')^2 - 1 \right]^{1/2} + Y_3^2 - G_r^2 t^2 + 2G_r^2 t_3 t - G_r^2 t_3^2 = 0 \right] \quad [25]$$

$$\left[\left[\left(\frac{G_r^2 V^2}{\bar{X}^2} \right) + (\bar{X}^2 - V^2) \left(\frac{G_r^2}{\bar{X}^2} \right) - G_r^2 \right] t^2 - \left[2t' \left(\frac{G_r V}{\bar{X}} \right) + 2X_3 \left(\frac{G_r V}{\bar{X}} \right) + 2t' (X_1^2 - V^2) \left(\frac{G_r^2}{\bar{X}^2} \right) - 2G_r^2 t_3 \right] t + \left(\frac{G_r V t'}{\bar{X}} \right)^2 + 2X_3 \left(\frac{G_r V t'}{\bar{X}} \right) + X_3^2 + (\bar{X}^2 - V^2) \left(\frac{G_r^2}{\bar{X}^2} \right) t'^2 - (\bar{X}^2 - V^2) - 2Y_3 D_1^{\frac{1}{2}} \left[\frac{G_r^2}{\bar{X}^2} (t-t')^2 - 1 \right]^{1/2} + Y_3^2 - G_r^2 t_3^2 = 0 \right] \quad [26]$$

$$\left[\left(\frac{G_r^2 V^2}{\bar{X}^2} + G_r^2 - \frac{G_r^2 V^2}{\bar{X}^2} - G_r^2 \right) t^2 - \left[2X_3 \left(\frac{G_r V}{\bar{X}^2} \right) + 2t' \left(\frac{G_r V}{\bar{X}} \right)^2 + 2t' G_r^2 - 2t' \left(\frac{G_r V}{\bar{X}} \right)^2 - 2G_r^2 t_3 \right] t + \left(\frac{G_r V t'}{\bar{X}} \right)^2 + 2X_3 \left(\frac{G_r V t'}{\bar{X}} \right) + X_3^2 + (\bar{X}^2 - V^2) \left(\frac{G_r^2}{\bar{X}^2} \right) t'^2 - (\bar{X}^2 - V^2) - 2Y_3 D_1^{\frac{1}{2}} \left[\frac{G_r^2}{\bar{X}^2} (t-t')^2 - 1 \right]^{1/2} + Y_3^2 - G_r^2 t_3^2 = 0 \right] \quad [27]$$

$$\left[\left[2(t_3 - t') G_r^2 - \frac{2X_3 G_r V}{\bar{X}} \right] t + \left(\frac{G_r V t'}{\bar{X}} \right)^2 + 2X_3 \left(\frac{G_r V t'}{\bar{X}} \right) + X_3^2 + G_r^2 t'^2 - \frac{G_r^2 V^2}{\bar{X}^2} t'^2 - (\bar{X}^2 - V^2) - 2Y_3 D_1^{\frac{1}{2}} \left[\frac{G_r^2}{\bar{X}^2} (t-t')^2 - 1 \right]^{1/2} + Y_3^2 - G_r^2 t_3^2 = 0 \right] \quad [28]$$

$$\left[\left[2(t_3 - t') G_r^2 - \frac{2X_3 G_r V}{\bar{X}} \right] t + 2X_3 \left(\frac{G_r V t'}{\bar{X}} \right) + X_3^2 + G_r^2 t'^2 - D_1 - 2Y_3 D_1^{\frac{1}{2}} \left[\frac{G_r^2}{\bar{X}^2} (t-t')^2 - 1 \right]^{1/2} + Y_3^2 - G_r^2 t_3^2 = 0 \right] \quad [29]$$

$$\left[2(t_3 - t')G_r^2 - \frac{2X_3G_rV}{X} \right] t + 2X_3 \left(\frac{G_rVt'}{X} \right) - 2Y_3D_1^{\frac{1}{2}} \left[\frac{G_r^2}{X^2} (t - t')^2 - 1 \right]^{1/2} + G_r^2(t'^2 - t_3^2) - D_1 + X_3^2 + Y_3^2 = 0 \quad [30]$$

$$-2Y_3D_1^{\frac{1}{2}} \left[\frac{G_r^2}{X^2} (t - t')^2 - 1 \right]^{1/2} = -2X_3 \left(\frac{G_rVt'}{X} \right) - \left[2(t_3 - t')G_r^2 - \frac{2X_3G_rV}{X} \right] t - G_r^2(t'^2 - t_3^2) + D_1 - X_3^2 - Y_3^2 \quad [31]$$

$$\left[\frac{G_r^2}{X^2} (t - t')^2 - 1 \right]^{1/2} = \frac{\left[2(t_3 - t')G_r^2 - \frac{2X_3G_rV}{X} \right] t + 2X_3 \left(\frac{G_rVt'}{X} \right) + G_r^2(t'^2 - t_3^2) - D_1 + X_3^2 + Y_3^2}{2Y_3D_1^{\frac{1}{2}}} \quad [32]$$

Definimos:

$$D_2 = X_3^2 + Y_3^2 \quad [32a]$$

$$\frac{G_r^2}{X^2} (t - t')^2 - 1 = \frac{\left\{ \left[2(t_3 - t')G_r^2 - \frac{2X_3G_rV}{X} \right] t + 2X_3 \left(\frac{G_rVt'}{X} \right) + G_r^2(t'^2 - t_3^2) - D_1 + D_2 \right\}^2}{4Y_3^2D_1} \quad [33]$$

$$(t - t')^2 = \frac{\bar{X}^2 \left\{ \left[2(t_3 - t')G_r^2 - \frac{2X_3G_rV}{X} \right] t + 2X_3 \left(\frac{G_rVt'}{X} \right) + G_r^2(t'^2 - t_3^2) - D_1 + D_2 \right\}^2}{4Y_3^2D_1G_r^2} + \frac{\bar{X}^2}{G_r^2} \quad [34]$$

$$(t-t')^2 = \frac{\left\{ \left[2\bar{X}(t_3-t')G_r^2 - 2X_3G_rV \right] t + 2X_3G_rVt' + G_r^2\bar{X}(t'^2-t_3^2) - \bar{X}(D_1-D_2) \right\}^2}{\left[2Y_3D_1^{\frac{1}{2}}G_r \right]^2} + \frac{\bar{X}^2}{G_r^2} \quad [35]$$

$$(t-t')^2 = \left\{ \frac{2\bar{X}(t_3-t')G_r^2 - 2X_3G_rV}{2Y_3D_1^{\frac{1}{2}}G_r} t + \frac{2X_3G_rVt' + G_r^2\bar{X}(t'^2-t_3^2) - \bar{X}(D_1-D_2)}{2Y_3D_1^{\frac{1}{2}}G_r} \right\}^2 + \frac{\bar{X}^2}{G_r^2} \quad [36]$$

$$(t-t')^2 = \left[\frac{\bar{X}(t_3-t')G_r - X_3V}{Y_3D_1^{\frac{1}{2}}} t + \frac{2X_3Vt' + \bar{X}G_r(t'^2-t_3^2) - \frac{\bar{X}}{G_r}(D_1-D_2)}{2Y_3D_1^{\frac{1}{2}}} \right]^2 + \frac{X_1^2}{G_r^2} \quad [37]$$

$$(t-t')^2 = \frac{\left[\bar{X}(t_3-t')G_r - X_3V \right]^2}{Y_3^2D_1} t^2 + \frac{\left[\bar{X}(t_3-t')G_r - X_3V \right] \left[2X_3Vt' + \bar{X}G_r(t'^2-t_3^2) - \frac{\bar{X}}{G_r}(D_1-D_2) \right]}{Y_3^2D_1} t + \frac{\left[2X_3Vt' + \bar{X}G_r(t'^2-t_3^2) - \frac{\bar{X}}{G_r}(D_1-D_2) \right]^2}{4Y_3^2D_1} + \frac{\bar{X}^2}{G_r^2} \quad [38]$$

$$\boxed{\left[\frac{[\bar{X}(t_3 - t')G_r - X_3V]^2}{Y_3^2 D_1} t^2 + \frac{[\bar{X}(t_3 - t')G_r - X_3V] \left[2X_3Vt' + \bar{X}G_r(t'^2 - t_3^2) - \frac{\bar{X}}{G_r}(D_1 - D_2) \right]}{Y_3^2 D_1} \right] t} \quad [40]$$

$$+ \left[\frac{[2X_3Vt' + \bar{X}G_r(t'^2 - t_3^2) - \frac{\bar{X}}{G_r}(D_1 - D_2)]^2}{4Y_3^2 D_1} + \frac{\bar{X}^2}{G_r^2} - t^2 + 2t't - t'^2 \right] = 0$$

$$\left\{ \left[\frac{[\bar{X}(t_3 - t')G_r - X_3V]^2}{Y_3^2 D_1} - 1 \right] t^2 + \left[\frac{[\bar{X}(t_3 - t')G_r - X_3V] \left[2X_3Vt' + \bar{X}G_r(t'^2 - t_3^2) - \frac{\bar{X}}{G_r}(D_1 - D_2) \right]}{Y_3^2 D_1} + 2t' \right] t \right\} \quad [41]$$

$$+ \left\{ \frac{[2X_3Vt' + \bar{X}G_r(t'^2 - t_3^2) - \frac{\bar{X}}{G_r}(D_1 - D_2)]^2}{4Y_3^2 D_1} + \frac{X_1^2}{G_r^2} - t'^2 \right\} = 0$$

La Ec. [41] es un polinomio de segundo orden que se resuelve mediante la conocida ecuación:

$$\boxed{t = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}} \quad [42]$$

Donde:

$$a = \left\{ \frac{\left[\bar{X}(t_3 - t')G_r - X_3V \right]^2}{Y_3^2 D_1} - 1 \right\} \quad [43]$$

$$b = \left\{ \frac{\left[\bar{X}(t_3 - t')G_r - X_3V \left[2X_3Vt' + \bar{X}G_r(t'^2 - t_3^2) - \frac{\bar{X}}{G_r}(D_1 - D_2) \right] \right]}{Y_3^2 D_1} + 2t' \right\} \quad [44]$$

$$c = \left\{ \frac{\left[2X_3Vt' + \bar{X}G_r(t'^2 - t_3^2) - \frac{\bar{X}}{G_r}(D_1 - D_2) \right]^2}{4Y_3^2 D_1} + \frac{\bar{X}^2}{G_r^2} - t'^2 \right\} \quad [45]$$

Sería recomendable intentar simplificar las expresiones [43], [44] y [45]

$$4ac = 4 \left\{ \frac{\left[\bar{X}(t_3 - t')G_r - X_3V \right]^2}{Y_3^2 D_1} - 1 \right\} \left\{ \frac{\left[2X_3Vt' + \bar{X}G_r(t'^2 - t_3^2) - \frac{\bar{X}}{G_r}(D_1 - D_2) \right]^2}{4Y_3^2 D_1} + \frac{\bar{X}^2}{G_r^2} - t'^2 \right\} \quad [46]$$

Definimos:

$$H = 2X_3Vt' + \bar{X}G_r(t'^2 - t_3^2) - \frac{\bar{X}}{G_r}(D_1 - D_2) \quad [47]$$

$$I = \bar{X}(t_3 - t')G_r - X_3V \quad [48]$$

$$a = \left\{ \frac{I^2}{Y_3^2 D_1} - 1 \right\} \quad [49]$$

$$b = \left\{ \frac{IH}{Y_3^2 D_1} + 2t' \right\} \quad [50]$$

$$4ac = 4 \left\{ \frac{I^2}{Y_3^2 D_1} - 1 \right\} \left\{ \frac{H^2}{4Y_3^2 D_1} + \frac{\bar{X}^2}{G_r^2} - t'^2 \right\} \quad [51]$$

$$b^2 - 4ac = \left\{ \frac{IH}{Y_3^2 D_1} + 2t' \right\}^2 - 4 \left\{ \frac{I^2}{Y_3^2 D_1} - 1 \right\} \left\{ \frac{H^2}{4Y_3^2 D_1} + \frac{\bar{X}^2}{G_r^2} - t'^2 \right\} \quad [52]$$

$$b^2 - 4ac = \left\{ \frac{IH}{Y_3^2 D_1} + 2t' \right\}^2 - \left\{ \frac{I^2 H^2}{(Y_3^2 D_1)^2} - \frac{H^2}{Y_3^2 D_1} + \frac{4\bar{X}^2 I^2}{G_r^2 Y_3^2 D_1} - \frac{4\bar{X}^2}{G_r^2} - \left(\frac{4I^2}{Y_3^2 D_1} \right) t'^2 + 4t'^2 \right\} \quad [53]$$

$$b^2 - 4ac = \frac{I^2 H^2}{(Y_3^2 D_1)^2} + \frac{4IHt'}{Y_3^2 D_1} + 4t'^2 - \left\{ \frac{I^2 H^2}{(Y_3^2 D_1)^2} - \frac{H^2}{Y_3^2 D_1} + \frac{4\bar{X}^2 I^2}{G_r^2 Y_3^2 D_1} - \frac{4\bar{X}^2}{G_r^2} - \left(\frac{4I^2}{Y_3^2 D_1} \right) t'^2 + 4t'^2 \right\} \quad [54]$$

$$b^2 - 4ac = \frac{I^2 H^2}{(Y_3^2 D_1)^2} + \frac{4IHt'}{Y_3^2 D_1} + 4t'^2 - \frac{I^2 H^2}{(Y_3^2 D_1)^2} + \frac{H^2}{Y_3^2 D_1} - \frac{\bar{X}^2 I^2}{G_r^2 Y_3^2 D_1} + \frac{4\bar{X}^2}{G_r^2} + \left(\frac{4I^2}{Y_3^2 D_1} \right) t'^2 - 4t'^2 \quad [55]$$

$$b^2 - 4ac = \frac{4IHt'}{Y_3^2 D_1} + \frac{H^2}{Y_3^2 D_1} - \frac{4\bar{X}^2 I^2}{G_r^2 Y_3^2 D_1} + \frac{4\bar{X}^2}{G_r^2} + \left(\frac{4I^2}{Y_3^2 D_1} \right) t'^2 \quad [56]$$

$$b^2 - 4ac = \frac{4}{Y_3^2 D_1} \left(-\frac{\bar{X}^2 I^2}{G_r^2} + \frac{\bar{X}^2 Y_3^2 D_1}{G_r^2} + I^2 t'^2 + IHt' + \left(\frac{H}{2} \right)^2 \right) \quad [57]$$

$$b^2 - 4ac = \frac{4}{Y_3^2 D_1} \left(-\frac{\bar{X}^2 I^2}{G_r^2} + \frac{\bar{X}^2 Y_3^2 D_1}{G_r^2} + \left(It' + \frac{H}{2} \right)^2 \right) \quad [58]$$

$$\sqrt{b^2 - 4ac} = \frac{2}{Y_3 D_1^{\frac{1}{2}}} \left(\frac{\bar{X}^2 Y_3^2 D_1}{G_r^2} - \frac{\bar{X}^2 I^2}{G_r^2} + \left(It' + \frac{H}{2} \right)^2 \right)^{\frac{1}{2}} \quad [59]$$

$$-b \pm \sqrt{b^2 - 4ac} = -\frac{IH}{Y_3^2 D_1} - 2t' \pm \frac{2}{Y_3 D_1^{\frac{1}{2}}} \left(\frac{\bar{X}^2 Y_3^2 D_1}{G_r^2} - \frac{\bar{X}^2 I^2}{G_r^2} + \left(It' + \frac{H}{2} \right)^2 \right)^{\frac{1}{2}} \quad [60]$$

$$\frac{-b \pm \sqrt{b^2 - 4ac}}{2a} = \frac{-\frac{IH}{Y_3^2 D_1} - 2t' \pm \frac{2}{Y_3 D_1^{\frac{1}{2}} \left(\frac{\bar{X}^2 Y_3^2 D_1}{G_r^2} - \frac{\bar{X}^2 I^2}{G_r^2} + \left(It' + \frac{H}{2} \right)^2 \right)^{\frac{1}{2}}}}{\frac{I^2}{Y_3^2 D_1} - 1} \quad [61]$$

$$\frac{-b \pm \sqrt{b^2 - 4ac}}{2a} = \frac{-IH - 2Y_3^2 D_1 t' \pm 2Y_3^2 D_1^{\frac{1}{2}} \left(\frac{\bar{X}^2 Y_3^2 D_1}{G_r^2} - \frac{\bar{X}^2 I^2}{G_r^2} + \left(It' + \frac{H}{2} \right)^2 \right)^{\frac{1}{2}}}{I^2 - Y_3^2 D_1} \quad [62]$$

$$J = Y_3^2 D_1 \quad [63]$$

$$\frac{-b \pm \sqrt{b^2 - 4ac}}{2a} = \frac{-IH - 2Jt' \pm \frac{2J}{D_1^{\frac{1}{2}} \left(\frac{\bar{X}^2 J}{G_r^2} - \frac{\bar{X}^2 I^2}{G_r^2} + \left(It' + \frac{H}{2} \right)^2 \right)^{\frac{1}{2}}}}{I^2 - J} \quad [64]$$

$$\frac{-b \pm \sqrt{b^2 - 4ac}}{2a} = \frac{-IH - 2Jt' \pm \frac{2\bar{X}J}{G_r D_1^{\frac{1}{2}} \left(J - I^2 + \frac{Gr^2}{\bar{X}^2} \left(It' + \frac{H}{2} \right)^2 \right)^{\frac{1}{2}}}}{I^2 - J} \quad [65]$$

Apendice B

Codificación del programa

```
{SI+}  
{R+} {Range checking on}  
{B+} {Boolean complete evaluation on}  
{S+} {Stack checking on}  
{N+} {Si numeric coprocessor}  
{M 65500,16384,655360} {Turbo 3 default stack and heap}
```

Uses

```
FDelay,Dos,Crt,Graph,Dibujo,AuxUT;
```

```
Function AnguloP(P1,P2 : Pnt) : Extended;
```

```
Var
```

```
Dist,Dx,Dy,Ang : Extended;
```

```
begin
```

```
Dx:=P2[1]-P1[1];
```

```
Dy:=P2[2]-P1[2];
```

```
Dist:=Sqrt(Sqr(Dx)+Sqr(Dy));
```

```
if Dx <> 0 then Ang:=Arctan(Dy/Dx) else
```

```
begin
```

```
if Dy > 0 then Ang:=Pi/2 else Ang:=3*Pi/2;
```

```
end;
```

```
if Dx < 0 then Ang:=Pi+Ang else
```

```
if (Dx <> 0)and(Dy < 0)then Ang:=2*Pi+Ang;
```

```
AnguloP:=Ang;
```

```
{Ang:=Ang*180/Pi;}
```

```
end;
```

```
Procedure CentroH(i,j : Word; Var Po : Pnt);
```

```

begin
Po[1]:=((Nu[i]^Pc[1]+Nu[j]^Pc[1])/2);
Po[2]:=((Nu[i]^Pc[2]+Nu[j]^Pc[2])/2);
end;

Procedure AlOrigen(i : Word);
begin
Nu[i]^Pc[1]:=Nu[i]^Pc[1]-P[1];
Nu[i]^Pc[2]:=Nu[i]^Pc[2]-P[2];
end;

Procedure DelOrigen(i : Word);
begin
Nu[i]^Pc[1]:=Nu[i]^Pc[1]+P[1];
Nu[i]^Pc[2]:=Nu[i]^Pc[2]+P[2];
end;

Function Vertice(i,j : Word) : Extended;
begin
Vertice:=GrowR*(Nu[j]^t0-Nu[i]^t0)/2;
end;

Function Xmedia(i,j : Word) : Extended;
begin
Xmedia:=Distancia(Nu[i]^Pc,Nu[j]^Pc)/2;
end;

Procedure InterH_R(P1,P2 : Pnt; K1,K2 : Extended; Var P3,P4 : Pnt);
Var
RAux,m,b,R : Extended;

```

```

a1,b1,c1 : Extended;
P5       : Pnt;
begin
if P1[1] <> P2[1] then
begin {Sera necesario este if?, Habra P[1,1] <> P[2,1]?}
m:=(P1[2]-P2[2])/(P1[1]-P2[1]);
b:=P1[2]-m*P1[1];
RAux:=sqr(m)-K2/K1;
a1:=1;
b1:=2*m*b/RAux;
c1:=(Sqr(b)+K2)/RAux;
RAux:=Sqr(b1)-4*c1;
if RAux < 0 then
begin
end else
begin
P3[1]:=(-b1+Sqrt(RAux))/2;
P4[1]:=(-b1-Sqrt(RAux))/2;
P3[2]:=m*P3[1]+b;
P4[2]:=m*P4[1]+b;
end;
end else
begin
RAux:=K2/K1*P1[1]-K2;
if RAux < 0 then
begin
end else
begin
P3[2]:=Sqrt(RAux);
P4[1]:=P3[2]{Sqrt(RAux)};

```

```
P3[1]:=P3[2]-b)/m;  
P4[1]:=P3[1]{(P4[2]-b)/m};  
end;  
end;  
end;
```

```
Function CruzaPun(Igual : Boolean; P1,P2,P3 : Real; Rdo : Word) : Boolean;  
begin  
CruzaPun:=False;  
P1:=Redondea(P1,Rdo[Rdo]);  
P2:=Redondea(P2,Rdo[Rdo]);  
P3:=Redondea(P3,Rdo[Rdo]);  
if Igual then  
begin  
if (P1 >= P2)and(P1 <= P3) then  
begin  
CruzaPun:=True;  
end;  
if (P1 <= P2)and(P1 >= P3)then  
begin  
CruzaPun:=True;  
end;  
end else  
begin  
if (P1 > P2)and(P1 < P3) then  
begin  
CruzaPun:=True;  
end;  
if (P1 < P2)and(P1 > P3)then  
begin
```

```

    CruzaPun:=True;
end;
end;
end;

Function Cruza(i,Nt : Word; Hp01,Hp03 : Hiperbola) : Boolean;
{ ~Esta tapada la Hp01 por alguna hiperbola (Hp02), del mismo nucleo i ? }
Var
kh,kk          : Word;
Cruza4         : Boolean;
CrX            : Array[1..3] of Boolean;
PcI,LV1i,LV2i  : PntI;
Px1i,Px2i,Px3i,Px4i  : PntI;
Px5i,Px6i      : PntI;
L1i,L2i        : PntI2;
Lx             : Array[1..4] of PntI;
L1,L2,L3       : Pnt2;
Px1,Px2,Px3,Px4,Vr1,Vr2,Ce : Pnt;
Px5,Px6        : Pnt;
Px11,Px21,Px31,Px41,Vr21  : Pnt;
Px51,Px61      : Pnt;
Hp02           : Hiperbola;
Paso2,Paso3    : Boolean;
Dh1,Dh2        : Array[1..3] of Extended;

begin
Cruza4:=False;
For Kh:=1 to 3 do Crx[Kh]:= False;
For Kh:=1 to 3 do
begin

```

```

Dh1[kh]:=0;
Dh2[kh]:=0;
end;
{Nucleo de Hiperola a evaluar }
Ce[2]:=0;
if (Hp01.Nuc[1] = i) then Ce[1]:=-Hp01.Xm else Ce[1]:=Hp01.Xm;
EscalaXY(Nu[i]^ .Pc,PcI);
if Nt > 0 then
begin
Kh:=1;
While Not Cruza4 and (Kh <= Nt) do
begin
Hp02:=Hpp[Nu[i]^ .id[kh]]^;
if Hp02.idh = Hp03.idh then
begin
Hp02:=Hp03;
end;
if Hp02.idh <> Hp01.idh then
begin
if Hp01.HipRec or Hp02.HipRec then
begin
{ Ambos segmentos y Vertice de Hip con quien cruzara }
Vr2:=Hp02.Vr;
if (Hp01.Pk[1] <> Hp02.Pk[1])or(Hp01.Pk[2] <> Hp02.Pk[2])then
begin
Vr2[2]:=(Hp02.Lim[1,2]+Hp02.Lim[2,2])/2;
Vr2[1]:=CalculaX(Vr2[2],Hp02.Pk,Hp02.HipRec);
Posiciona(Vr2,Hp02.Po,Hp02.A,True,Vr2); {Hp02 Lim a su posicion}
Posiciona(Hp02.Lim[1],Hp02.Po,Hp02.A,True,L2[1]); {Hp02 Lim a su posicion}
Posiciona(Hp02.Lim[2],Hp02.Po,Hp02.A,True,L2[2]);

```

```

EscalaXY(L2[1],L2i[1]);
Paso2:=Paso1;
EscalaXY(L2[2],L2i[2]);
Paso3:=Paso1;
Reposicio(L2[1],Hp01.Po,Hp01.A,L3[1]);  {Hp02 Lim a Pos respecto a Hp01}
Reposicio(L2[2],Hp01.Po,Hp01.A,L3[2]);
Reposicio(Vr2,Hp01.Po,Hp01.A,Vr21);
if Hp01.HipRec then
begin
InterH_R(Ce,L3[1],Hp01.Pk[1],Hp01.Pk[2],Px1,Px2);
InterH_R(Ce,Vr21, Hp01.Pk[1],Hp01.Pk[2],Px3,Px4);
InterH_R(Ce,L3[2],Hp01.Pk[1],Hp01.Pk[2],Px5,Px6);
end else
begin
Intersec(Ce,L3[1],Hp01.Lim[1],Hp01.Lim[2],L100,Px1);
Intersec(Ce,Vr21, Hp01.Lim[1],Hp01.Lim[2],L100,Px3);
Intersec(Ce,L3[2],Hp01.Lim[1],Hp01.Lim[2],L100,Px5);
Px2:=Px1;
Px4:=Px3;
Px6:=Px5;
Px2[1]:=-0.1;
Px4[1]:=-0.1;
Px6[1]:=-0.1;
end;
Posiciona(Px1,Hp01.Po,Hp01.A,True,Px11);
Posiciona(Px2,Hp01.Po,Hp01.A,True,Px21);
Posiciona(Px3,Hp01.Po,Hp01.A,True,Px31);
Posiciona(Px4,Hp01.Po,Hp01.A,True,Px41);
Posiciona(Px5,Hp01.Po,Hp01.A,True,Px51);
Posiciona(Px6,Hp01.Po,Hp01.A,True,Px61);

```

```

if Not CrX[1] then
begin
if CruzaPun(True,Px1[2],Hp01.Lim[1,2],Hp01.Lim[2,2],5) and (Px1[1] >= 0) then
begin
if (Redondea(Px1[2],Rdo[4]) = Redondea(Hp01.Lim[1,2],Rdo[4]))or
(Redondea(Px1[2],Rdo[4]) = Redondea(Hp01.Lim[2,2],Rdo[4])) then
begin
end else
CrX[1]:=Entre(Px1,Ce,L3[1]{Hp02.Lim[1]});
if CrX[1] then
begin
Dh1[1]:=Distancia(Nu[i]^Pc,Px11);
Dh2[1]:=Distancia(Nu[i]^Pc,L2[1]);
if Dh1[1] < Dh2[1] then
begin
CrX[1]:=False;
end;
end;
end;
end;
if Not CrX[1] then
begin
if CruzaPun(True,Px2[2],Hp01.Lim[1,2],Hp01.Lim[2,2],5) and (Px2[1] >= 0) then
begin
if (Redondea(Px2[2],Rdo[4]) = Redondea(Hp01.Lim[1,2],Rdo[4]))or
(Redondea(Px2[2],Rdo[4]) = Redondea(Hp01.Lim[2,2],Rdo[4])) then
begin
end else
CrX[1]:=Entre(Px2,Ce,L3[1]{Hp02.Lim[1]});
if CrX[1] then

```

```

begin
Dh1[1]:=Distancia(Nu[i]^Pc,Px21);
Dh2[1]:=Distancia(Nu[i]^Pc,L2[1]);
end;
end;
end;
if Not CrX[2] then
begin
if CruzaPun(True,Px3[2],Hp01.Lim[1,2],Hp01.Lim[2,2],5) and (Px3[1] >= 0) then
begin
if (Redondea(Px3[2],Rdo[4]) = Redondea(Hp01.Lim[1,2],Rdo[4]))or
(Redondea(Px3[2],Rdo[4]) = Redondea(Hp01.Lim[2,2],Rdo[4])) then
begin
end else
CrX[2]:=Entre(Px3,Ce,Vr21);
if CrX[2] then
begin
Dh1[2]:=Distancia(Nu[i]^Pc,Px31);
Dh2[2]:=Distancia(Nu[i]^Pc,Vr2);
if Dh1[2] < Dh2[2] then
begin
CrX[2]:=False;
end;
end;
end;
end;
if Not CrX[2] then
begin
if CruzaPun(True,Px4[2],Hp01.Lim[1,2],Hp01.Lim[2,2],5) and (Px4[1] >= 0) then
begin

```

```

if (Redondea(Px4[2],Rdo[4]) = Redondea(Hp01.Lim[1,2],Rdo[4]))or
  (Redondea(Px4[2],Rdo[4]) = Redondea(Hp01.Lim[2,2],Rdo[4])) then
begin
end else
CrX[2]:=Entre(Px4,Ce,Vr21);
if CrX[2] then
begin
Dh1[2]:=Distancia(Nu[i]^Pc,Px41);
Dh2[2]:=Distancia(Nu[i]^Pc,Vr2);
end;
end;
end;
if Not CrX[3] then
begin
if CruzaPun(True,Px5[2],Hp01.Lim[1,2],Hp01.Lim[2,2],5) and (Px5[1] >= 0) then
begin
if (Redondea(Px5[2],Rdo[4]) = Redondea(Hp01.Lim[1,2],Rdo[4]))or
  (Redondea(Px5[2],Rdo[4]) = Redondea(Hp01.Lim[2,2],Rdo[4])) then
begin
end else
CrX[3]:=Entre(Px5,Ce,L3[2]{Hp02.Lim[2]});
if CrX[3] then
begin
Dh1[3]:=Distancia(Nu[i]^Pc,Px51);
Dh2[3]:=Distancia(Nu[i]^Pc,L2[2]);
if Dh1[3] < Dh2[3] then
begin
CrX[3]:=False;
end;

```

```

end;
end;
end;
if Not CrX[3] then
begin
if CruzaPun(True,Px6[2],Hp01.Lim[1,2],Hp01.Lim[2,2],5) and (Px6[1] >= 0) then
begin
if (Redondea(Px6[2],Rdo[4]) = Redondea(Hp01.Lim[1,2],Rdo[4]))or
(Redondea(Px6[2],Rdo[4]) = Redondea(Hp01.Lim[2,2],Rdo[4])) then
begin
end else
CrX[3]:=Entre(Px6,Ce,L3[2]{Hp02.Lim[2]});
if CrX[3] then
begin
Dh1[3]:=Distancia(Nu[i]^Pc,Px61);
Dh2[3]:=Distancia(Nu[i]^Pc,L2[2]);
end;
end;
end;
end;
end;
end;
Inc(Kh);
if Crx[1] and Crx[2] and Crx[3] then Cruza4:=True;
if Not CrX[1] or Not CrX[2] or Not CrX[3] then
begin
if Cruza4 then Cruza4:=False;
end;
end;
end;
end;

```

```

if Cruza4 and (Dh2[1] > Dh1[1])and (Dh2[2] > Dh1[2])and (Dh2[3] > Dh1[3]) then
begin
  Quita:=True;
  ToQ:=Hp02.idh;
end;
Cruza:=Cruza4;
end;

```

```

Procedure MinimoHi(Kn : Word; Lo,Hi : Integer);
Var
is      : Integer;
Procedure Sort(l, r: Integer);
var
i,j,ix,iy : Integer;
x,y      : Extended;
begin
i := l; j := r; x := Ang1[(l+r) DIV 2];ix:=Nu[Kn]^id[(l+r) DIV 2];
repeat
  while Ang1[i] < x do i:=i+1;
  while x < Ang1[j] do j:=j-1;
  if i <= j then
  begin
    y:=Ang1[i];Ang1[i]:=Ang1[j];Ang1[j]:=y;
    iy:=Nu[Kn]^id[i];Nu[Kn]^id[i]:=Nu[Kn]^id[j];Nu[Kn]^id[j]:=iy;
    i :=i+1;j:=j-1;
  end;
until i > j;
if l < j then Sort(l, j);
if i < r then Sort(i, r);
end;

```

```

begin
Sort(Lo,Hi);
end;

Procedure OrdenaFro (i : Word);
Var
j,k,Hk : Word;
R      : Extended;
begin
With Nu[i]^ do
begin {Inicializa variables}
For k:=1 to N do
begin
Hk:=id[k];
Hpp[Hk]^ .An1[1]:=0;
Hpp[Hk]^ .An1[2]:=0;
For j:=1 to 2 do
begin
An[Hk,j,1]:=0;
An[Hk,j,2]:=0;
end;
end;
For k:=1 to N do
begin
Hk:=id[k];
Hpp[Hk]^ .An1[1]:=0;
Hpp[Hk]^ .An1[2]:=0;
An[Hk][1,2]:=(Hpp[Hk]^ .Lim[1,2]+Hpp[Hk]^ .Lim[2,2])/2;
An[Hk][1,1]:=CalculaX(An[Hk][1,2],Hpp[Hk]^ .Pk,Hpp[Hk]^ .HipRec);

```

```

Posiciona(An[Hk][1],Hpp[Hk]^Po,Hpp[Hk]^A,False,An[Hk][1]);
Ang1[k]:=(Angulo(Pc,An[Hk][1]));
For j:=1 to 2 do
begin
Posiciona(Hpp[Hk]^Lim[j],Hpp[Hk]^Po,Hpp[Hk]^A,False,An[Hk][j]);
end;
Hpp[Hk]^An1[1]:=(Angulo(Pc,An[Hk][1]));
Hpp[Hk]^An1[2]:=(Angulo(Pc,An[Hk][2]));
For j:=1 to 2 do
begin
An[Hk][j,1]:=(An[Hk][j,1]-Pc[1]);
An[Hk][j,2]:=(An[Hk][j,2]-Pc[2]);
end;
end;
if N > 1 then MinimoHi(i,1,N);
end;
end;

Function CruzaAlt(i,Nt : Word; H1,H3 : Hiperbola; Para : Boolean) : Boolean;
{ ~ Esta tapada la Hp01 por alguna hiperbola (H2), del mismo nucleo i ? }
Var
H4 : Hiperbola;
kh,kk,Sal : Word;
Cruza3,Repite : Boolean;
PcI,Pxi,L2i,L11i,L12i : PntI;
CrX,CrY : Array[1..3] of Boolean;
L1 : Array[1..3] of Pnt;
Px : Array[1..3,1..2] of Pnt;
Pxi : Array[1..3,1..2] of PntI;
L1i : Array[1..3] of PntI;

```

```

Ce,L5,L6,L7,L51,L61,L71,Ce1 : Pnt;
L8,L9,L10,L11,L12,L13      : Pnt;
L111,L121                  : Pnt;
H2                          : Hiperbola;
Delta,Delta1,D1,D2         : Extended;
Paso2                       : Boolean;
L3                          : Array[1..3] of Pnt;

AA1                          : Array[1..3] of Extended;
AAh                          : Array[1..3] of Extended;

```

```

Function Bool(P1,P2,P3 : Pnt) : Boolean;
Var B      : Boolean;
    ic     : Word;
    R,R1   : Real;
begin
B:=False;
if Not B then
begin
if (P1[1] >=0)and CruzaPun(True,P1[2],H2.Lim[1,2],H2.Lim[2,2],5) then
begin
B:=Entre(P1,Ce,P3);
if Not B then
begin
end;
end;
end;
if Not B then
begin
if (P2[1] >=0) and CruzaPun(True,P2[2],H2.Lim[1,2],H2.Lim[2,2],5) then

```

```

begin
B:=Entre(P2,Ce,P3);
if Not B then
begin
end;
end;
end;
Bool:=B;
end;

```

```

Procedure PintaAux;
Var
Pxp : Array[1..3,1..2] of Pnt;
k,j : Word;
begin
PintaAreaTrabajo(Pon);
For k:=1 to Nu[i]^N do
begin
EscalaHip(i,Hpp[Nu[i]^id[k]]^k,Bas);
end;
EscalaHip(i,H1,0,Bas);
EscalaHip(i,H2,14,Bas);
For k:=1 to 3 do
begin
For j:=1 to 1 do
begin
Posiciona(Px[k,j],H2.Po,H2.A,True,Pxp[k,j]);
EscalaXY(Pxp[k,j],Pxi[k,j]);
end;
SetColor(4);Circle(Pxi[k,1,1],Pxi[k,1,2],1);

```

```

SetColor(k);Line(PcI[1],PcI[2],L1i[k,1],L1i[k,2]);
end;
Temporal:=ReadKey;
end;

Procedure InterPos(Lin1,Lin2 : Pnt; Var Py,Py1 : Pnt);
begin
if H2.HipRec then
begin
InterH_R(Lin1,Lin2,H2.Pk[1],H2.Pk[2],Py,Py1);
end else
begin
Intersec(Lin1,Lin2,H2.Lim[1],H2.Lim[2],L100,Py);
Py1:=Py;
Py[1]:=-0.1;
end;
end;

Procedure PonEnLugar(L0 : Pnt; Var L : Pnt; Var Li : PntI);
begin
Posiciona(L0,H1.Po,H1.A,True,L);
EscalaXY(L,Li);
Paso2:=Paso1;
Reposicio(L,H2.Po,H2.A,L);
end;

Procedure CalculaEles1;
Var k : Word;
begin
L1[1,2]:=L5[2]; {Inferior}

```

```

L1[3,2]:=L7[2]; {Superior}
Delta:=(L1[3,2]-L1[1,2])/2; {Define Nuevo Delta como mitad Dist entre Sup e Inf}
L1[2,2]:=L1[1,2]+Delta; {Punto medio entre Sup e Inf}
L6[2]:=L1[2,2]; {Punto Medio}

```

```

For k:=1 to 3 do
begin {Calcula X de puntos anteriores}
L1[k,1]:=CalculaX(L1[k,2],H1.Pk,H1.HipRec);
if K = 2 then L6[1]:=L1[k,1];
PonEnLugar(L1[k],L1[k],Li[k]);
end;
if Abs(L1[1,1]) < 1e-12 then L1[1,1]:=0;
if Abs(L1[1,2]) < 1e-12 then L1[1,2]:=0;
if Abs(L1[2,1]) < 1e-12 then L1[2,1]:=0;
if Abs(L1[2,2]) < 1e-12 then L1[2,2]:=0;
if Abs(L1[3,1]) < 1e-12 then L1[3,1]:=0;
if Abs(L1[3,2]) < 1e-12 then L1[3,2]:=0;

```

```

if Sal = 1 then Inc(Sal);
end;

```

```

Var k1,Kr : Word;
Toler,R1,R2,R3,R4 : Extended;
Lim1 : Pnt2;
Cambia,FTF,Exepto : Boolean;
Cambio,DoIt : Boolean;
Lm : Pnt;
L2 : Array[1..3] of Pnt;
Mov5,Mov7 : Boolean;
Con01 : Word;

```

```

label 1,2,3,4,5,6,7;

begin {H1 : Hiperbola a evaluar}

Nu[NoNMax]^:=Nu[i]^;
OrdenaFro(NoNMax);
Cambia:=False;
Cambio:=False;
Lim1:=H1.Lim;
FTF:=False;
Toler:=1e-6;
1: Cruza3:=True;
For kh:=1 to 3 do CrX[kh]:=False;
Delta:=1;
EscalaXY(Nu[NoNMax]^,Pc,PcI); {Nucleo de Hiperola a evaluar }
if Nt > 0 then
begin
L5:=H1.Lim[1]; {Limite inferior}
L7:=H1.Lim[2]; {Limite superior}
L51:=L5;      {Respaldos}
L71:=L7;
Kh:=1;
Repite:=False;
Mov5:=False;
Mov7:=False;
While (Kh <= Nt)and((Not Crx[1])or(Not CrX[2])or(Not CrX[3])) do
begin
2: For Kk:=1 to 3 do Crx[Kk]:= True;

H2:=Hpp[Nu[NoNMax]^,id[kh]]^;      {Seleccion de H2}

```

```

{
  if Hpv[2]^ .idh = Hpp[Nu[NuNMax]^ .id[kh]]^ .idh then H2:=Hpv[2]^;
}
if H2.idh = H3.idh then H2:=H3;
Reposicio(Nu[NoNMax]^ .Pc,H2.Po,H2.A,Ce); {Pos centro Nuc respecto a H2}
if Abs(Ce[1]) < 1e-12 then Ce[1]:=0;
if Abs(Ce[2]) < 1e-12 then Ce[2]:=0;

if H2.idh <> H1.idh then
begin {Si no son la misma H1 y H2}
  Exepto:=False;
  DoIt:=False;
  if H2.HipRec then
  begin {Si H2 es hiperbola, no recta}
    if (H1.Pk[1] <> H2.Pk[1])or(H1.Pk[2] <> H2.Pk[2])then
    begin {Si H1 y H2 son diferentes}
      Lm[2]:=(H2.Lim[1,2]+H2.Lim[2,2])/2;
      Lm[1]:=CalculaX(Lm[2],H2.Pk,H2.HipRec);
      Sal:=0;
      While (CrX[1] or CrX[2] or CrX[3])and
        (Abs(Delta) >= Toler)and(Sal < 2) do
      begin
        if L7[2] > L5[2] then
        begin
          L9:=L7;
          L7:=L5;
          L5:=L9;
        end;
      end;

      CalculaEles1;
    end;
  end;
end;

```

```

(*)
  if i = 3 then
    begin
      Temporal:=ReadKey;
    end;
  *)

  For k1:=1 to 3 do
    begin
      InterPos(Ce,L1[k1],Px[k1,1],Px[k1,2]);
    end;

    CrX[1]:=Bool(Px[1,1],Px[1,2],L1[1]);
    if (L1[2,2] <> Ce[2])or(L1[2,1] <> Ce[1]) then
      begin
        CrX[2]:=Bool(Px[2,1],Px[2,2],L1[2]);
      end else
        begin
          CrX[2]:=False;
        end;
    CrX[3]:=Bool(Px[3,1],Px[3,2],L1[3]);

    if CrX[1] then
      begin
        if Redondea(Px[1,1,1],Rdo[4]) = Redondea(L1[1,1],Rdo[4]) then
          begin
            if Redondea(L5[2],Rdo[4]) <> Redondea(L6[2],Rdo[4]) then
              begin
                {  PintaAux;}

```

```

{   CrX[3]:=False;}
    CrX[1]:=False;
end else
begin
if Redondea(Px[3,1,1],Rdo[4]) = Redondea(L1[3,1],Rdo[4]) then
begin
if Nu[NoNMax]^N = 2 then
begin
{   PintaAux;}
    CrX[1]:=False;
    CrX[2]:=False;
    CrX[3]:=False;
    Exepto:=True;
{
    L7[2]-L5[2]:=0.1;
}
end;
end;
end;
end;
end;
if CrX[3] then
begin
if Redondea(Px[3,1,1],Rdo[4]) = Redondea(L1[3,1],Rdo[4]) then
begin
if Redondea(L5[2],Rdo[4]) <> Redondea(L6[2],Rdo[4]) then
begin
{   PintaAux;} {Si entra}
    CrX[3]:=False;
end;

```

```
end;  
end;
```

```
if Not CrX[1] and Not CrX[2] and CrX[3] then  
begin  
if Redondea(L7[2],Rdo[5]) = Redondea(H1.Lim[2,2],Rdo[5]) then  
begin  
if Redondea(Px[3,1,2],Rdo[5]) = Redondea(H2.Lim[1,2],Rdo[5]) then  
begin  
CrX[3]:=False;  
end;  
end;  
end;  
end;
```

```
if CrX[1] and CrX[2] and CrX[3] then  
begin  
Sal:=2;  
end;
```

```
(*  
if (i = 7)and Not H1.HipRec then  
begin  
PintaAux;  
end;
```

```
*)  
if Not CrX[1] and CrX[2] and Not CrX[3] then  
begin {FTF}  
if Not Cambia then  
begin  
if Not FTF then
```

```

begin
{   PintaAux;}
if (Abs(L7[2]-Lim1[1,2]) > 1e-6)and
   (Abs(L7[2]-Lim1[2,2]) > 1e-6) then
begin
{   PintaAux;}
FTF:=True;
Kr:=Kh;
L12:=L5;
L13:=L7;
{   Mov5:=True;}
GoTo 4;
end else
begin
{   PintaAux;}
FTF:=True;
Kr:=Kh;
L12:=L5;
L13:=L7;
Mov5:=True;
end;
end;
if Abs(Delta) <= Toler then
begin
L7:=L6;
L5:=L51;
Delta:=1;
end else
begin
L5:=L6;

```

```

    L51:=L5;
end;
end else
begin
if Cambio then
begin
{   PintaAux;}
GoTo 5;
end else
begin
4:  L7:=L6;
    L71:=L7;
    if ((Abs(Delta) <= Toler) or Cambio){ and Not CrX[2]} then
begin
L7:=L6;
L71:=L7;
L5:=L51;
Delta:=1;
if Cambio then Cambio:=False;
end else
begin
L7:=L6;
L71:=L7;
Delta:=1;
end;
end;
end;
end;

```

if CrX[1] and Not CrX[2] and CrX[3] then

```

begin {TFT}

if H1.HipRec then
begin
L5[2]:=L5[2]-Abs(L5[2]-L6[2])*1e-3;
L7[2]:=L7[2]+Abs(L7[2]-L6[2])*1e-3;

end else
begin
L5[2]:=L5[2]-1e-4;
L7[2]:=L7[2]+1e-4;
end;

L5[1]:=CalculaX(L5[2],H1.Pk,H1.HipRec);
L7[1]:=CalculaX(L7[2],H1.Pk,H1.HipRec);
L51:=L5;
L71:=L7;
end;
{
if (i = 3) and (H1.idh= 18) and (H1.Nuc[2] = 0) then
begin
PintaAux;
end;
}
if Not CrX[1] and Not CrX[2] and CrX[3] then
begin {FFT}
L10:=L6;
L11:=L7;
L8:=L11;
CrY[3]:=True;

```

```

Delta1:=1;
While (Delta1 > 1e-8){ or CrY[3]} do
begin
PonEnLugar(L8,L9,L2i);
InterPos(Ce,L9,Px[3,1],Px[3,2]);
CrY[3]:=Bool(Px[3,1],Px[3,2],L9);

Posiciona(Px[3,1],H2.Po,H2.A,True,Px[3,1]);
Reposicio(Px[3,1],H1.Po,H1.A,Px[2,1]);
(*)
if H1.idh = 17 then
begin
EscalaXY(Px[3,1],Pxi[3,1]);
SetColor(3);Circle(Pxi[3,1,1],Pxi[3,1,2],2);
Line(Pci[1],Pci[2],L2i[1],L2i[2]);
Temporal:=ReadKey;
end;
*)
if CrY[3] then
begin
L11:=L8;
end else
begin
L10:=L8;
end;
L8[2]:=(L11[2]+L10[2])/2;
L8[1]:=CalculaX(L8[2],H1.Pk,H1.HipRec);
Delta1:=Abs(L8[2]-L11[2]);
end;
if Not CrY[3] then

```

```

begin
L7:=L8;
if Delta1 < 1e-8 then
begin
{   PintaAux;}
  CrX[3]:=False;
end;
end else
begin
L6:=L8;
CrX[2]:=True;
end;
end;

if CrX[1] and Not CrX[2] and Not CrX[3] then
begin {TFF}
L10:=L6;
L11:=L5;
L8:=L5;
CrY[3]:=True;
Delta1:=1;

While (Delta1 > 1e-8){ or CrY[3]} do
begin
PonEnLugar(L8,L9,L2i);
InterPos(Ce,L9,Px[3,1],Px[3,2]);
CrY[3]:=Bool(Px[3,1],Px[3,2],L9);
Posiciona(Px[3,1],H2.Po,H2.A,True,Px[3,1]);

if CrY[3] then

```

```

begin
L11:=L8; {L5}
end else
begin
L10:=L8; {L6}
L11:=L5;
end;
{
if H1.idh = 5 then
begin

EscalaXY(Px[3,1],Pxi[3,1]);
SetColor(3);Circle(Pxi[3,1,1],Pxi[3,1,2],2);
SetColor(4);Line(Pci[1],Pci[2],L2i[1],L2i[2]);

Posiciona(L11,H1.Po,H1.A,True,L11i);
Posiciona(L10,H1.Po,H1.A,True,L12i);

EscalaXY(L11i,L11i);
EscalaXY(L12i,L12i);

SetColor(1);Circle(L11i[1],L11i[2],4);
SetColor(2);Circle(L12i[1],L12i[2],4);

Temporal:=ReadKey;
SetColor(1);Line(Pci[1],Pci[2],L2i[1],L2i[2]);
SetColor(0);Circle(L11i[1],L11i[2],4);
SetColor(0);Circle(L12i[1],L12i[2],4);
Temporal:=ReadKey;
end;

```

```

}

L8[2]:= (L11[2]+L10[2])/2;
L8[1]:= CalculaX(L8[2],H1.Pk,H1.HipRec);
Delta1:= Abs(L11[2]-L10[2]);
end;
if Not CrY[3] then
begin
L5:=L8;
if Delta1 < 1e-8 then
begin
CrX[1]:=False;
end;
end else
begin
{ PintaAux;}
L6:=L8;
CrX[2]:=True;
end;
end;

if CrX[1] and CrX[2] and Not CrX[3] then
begin

if Redondea(Px[3,1,2],Rdo[3]) = Redondea(H2.Lim[1,2],Rdo[3]) then
begin
CrX[3]:=True;
Cambio:=False;
GoTo 6;
end;

```

```

{5:}  L5:=L6;
      L51:=L5;          {Quite}
      if (Abs(Delta) <= Toler){ and Not CrX[2]} then
      begin
      L5:=L6;
      L51:=L5;
      L7:=L71;
      Delta:=1;
      end else
      begin
      L5:=L6;
      L51:=L5;

      Delta:=1;
      end;
      end;

      if Not CrX[1] and CrX[2] and CrX[3] then
      begin {FTT}
5:    L7:=L6;
      L71:=L7;
      if (Abs(Delta) <= Toler){ and Not CrX[2]} then
      begin
      L7:=L6;
      L71:=L7;
      L5:=L51;
      Delta:=1;
      end else
      begin

```

```
L7:=L6;  
L71:=L7;  
Delta:=1;  
end;  
end;  
end;
```

```
6: if CrX[1] and CrX[2] and CrX[3] and (Dobla or FTF) then  
begin  
if Not FTF then  
begin  
Kh:=1;  
{  
L5:=L12;  
L7:=L13;  
}  
Dobla:=False;  
Cambia:=True;  
Dobla:=False;  
Cambio:=True;  
FTF:=False;  
end else  
begin  
Kh:=Kr;  
{  
L5:=L12;  
L7:=L13;  
}  
L7:=L5;  
L5:=L12;
```

```
Cambia:=True;
Dobla:=False;
Cambio:=True;
FTF:=False;
GoTo 2;
end;
end;
```

```
if Not CrX[1] and Not CrX[2] and Not CrX[3] and (Dobla {or FTF}) then
```

```
begin
```

```
if Kh = Nt then
```

```
begin
```

```
if FTF then
```

```
begin
```

```
(*
```

```
3 : Kh:=Kr;
```

```
L5:=L12;
```

```
L7:=L13;
```

```
Cambio:=True;
```

```
Cambia:=True;
```

```
Dobla:=False;
```

```
FTF:=False;
```

```
GoTo 2;
```

```
*)
```

```
end else
```

```
begin
```

```
Cambio:=True;
```

```
Cambia:=True;
```

```
Dobla:=False;
```

```
FTF:=False;
```

```

    GoTo 1;
end;
end;
end;

end else
begin
Cruza3:=False;
For kk:=1 to 3 do CrX[kk]:=False;
end;
{Hasta Aqui}
end else
begin
Cruza3:=False;
For kk:=1 to 3 do CrX[kk]:=False;
end;
end else
begin
Cruza3:=False;
For kk:=1 to 3 do CrX[kk]:=False;
end;
if (Abs(L7[2]-L5[2]) < Toler)and(H1.idh <> H2.idh)and Not Exepto then
begin
if Not FTF then
begin
For kk:=1 to 3 do CrX[kk]:=True;
end else
begin
Kh:=Kr;
L5:=L12;

```

```
L7:=L13;  
Cambio:=True;  
Cambia:=True;  
Dobla:=False;  
FTF:=False;  
GoTo 2;  
end;  
end;
```

```
if Not Repite and (kh = Nu[NoNMax]^N) then  
begin  
if Not Cambio then  
begin  
Repite:=True;  
kh:=1;  
end else  
begin  
Cambio:=False;  
end;  
end;  
Inc(Kh);  
end;
```

```
if Not Crx[1] or Not Crx[2] or Not Crx[3] then Cruza3:=False;  
if Crx[1] and Crx[2] and Crx[3] then Cruza3:=True;  
(*  
if CrX[1] and CrX[2] and CrX[3] and Cambio then  
begin  
PintaAux;  
Temporal:=ReadKey;
```

```

L5:=L12;
L7:=L13;
Cruza3:=False;
end;
*)
end else Cruza3:=False;
CruzaAlt:=Cruza3;
end;

```

```

Procedure AcotaLimite(inN,Nt : Word; Var Cruzo : Boolean); {Define Ljmite por interseccion de hiperbola N1-N2}
Var
      {con nucleo N3}
k1,K2      : Word;
Ni         : Array[1..4] of Word;
Tp1,Tp2,D1,D2,Xprim  : Extended;
Dy,H,I,a1,b,c    : Extended;   {Auxiliares de calculo}
Tch,Lx       : Pnt;
Cp         : Array [1..4] of Pnt;   {Centro del nucleo N3 girado y corrido}
Cch        : Array [1..2] of Pnt2; {Puntos de choque}
CchA       : Array [1..2] of Pnt2; {Puntos de Ch En posicion}
iCch       : Array [1..2] of Pnt2; {Puntos de Ch paa Graf}
CruzaX     : Boolean;
AuxCh      : Pnt;
Hc         : Array[1..4] of Hiperbola;

```

```

Procedure CalcParalela(H1,H2 : Hiperbola);
Var
YAux0,YAux1,YAux2    : Extended;
begin

```

```

if Cp[3,1] < Cp[4,1] then XPrim:=Cp[3,1]+H2.Xm else XPrim:=Cp[4,1]+H2.Xm;

```

```

Tch[1]:= (YAux0+YAux1)/YAux2;
YAux0:= H1.Xm*H2.Xm*XPrim;
YAux1:= GrowR*(H2.Xm*H1.V*Tp1+H1.Xm*H2.V*Tp2);
YAux2:= GrowR*(H2.Xm*H1.V+H1.Xm*H2.V);
Tch[2]:= (YAux0+YAux1)/YAux2;
Cch[1][1,1]:= GrowR*H1.V/H1.Xm*(Tch[1]-Tp1);
Cch[1][1,2]:= ((Sqr(Cch[1][1,1]/H1.V)-1)*(Sqr(H1.Xm)-Sqr(H1.V)));
Cch[1][2,1]:= GrowR*H2.V/H2.Xm*(Tch[1]-Tp2);
Cch[1][2,2]:= ((Sqr(Cch[1][2,1]/H2.V)-1)*(Sqr(H2.Xm)-Sqr(H2.V)));
if Cch[1][1,2] <> Cch[1][2,2] then
begin
Cch[1][1,1]:= GrowR*H1.V/H1.Xm*(Tch[2]-Tp1);
Cch[1][1,2]:= ((Sqr(Cch[1][1,1]/H1.V)-1)*(Sqr(H1.Xm)-Sqr(H1.V)));
Cch[1][2,1]:= GrowR*H2.V/H2.Xm*(Tch[2]-Tp2);
Cch[1][2,2]:= ((Sqr(Cch[1][2,1]/H2.V)-1)*(Sqr(H2.Xm)-Sqr(H2.V)));
Tch[1]:= Tch[2];
end else if Tch[1] <> Tch[2] then
begin
Tch[2]:= Tch[1];
end;
Cch[1][2,1]:= Redondea(Cch[1][1,1],Rdo[7]);
Cch[1][1,2]:= Redondea(Sqrt(Cch[1][1,2]),Rdo[7]);
Cch[1][2,2]:= -Cch[1][1,2];
end;

```

```

Procedure CalculaGrl(H1 : Hiperbola);
Var R,R1,R2 : Extended;
    Kk,Kj,Ki : Word;
    YAux0 : Extended;
    YAux : Extended;

```

```

Px2      : Pnt;
begin
With H1 do
begin
D1:=Sqr(Xm)-Sqr(V);
D2:=Sqr(Cp[4,1])+Sqr(Cp[4,2]);
Dy:=Sqr(Cp[4,2])*D1;
H:=Xm*GrowR*(Nu[Ni[4]]^t0-Tp1)-Cp[4,1]*V;
I:=2*Cp[4,1]*V*Tp1+Xm*GrowR*(Sqr(Tp1)-Sqr(Nu[Ni[4]]^t0));
I:=I-Xm/GrowR*(D1-D2);
a1:=Sqr(H)/Dy-1;
b:=H*I/Dy+2*Tp1;
c:=Sqr(I)/4/Dy+Sqr(Xm/GrowR)-Sqr(Tp1);
Tch[1]:=(-b-Sqrt(Sqr(b)-4*a1*c))/2/a1; {Tiempos de choque}
Tch[2]:=(-b+Sqrt(Sqr(b)-4*a1*c))/2/a1;
if ((Tch[2] < Tch[1])and(Tch[2] > 0))or(Tch[1] <= 0) then
begin      { Coloca los tiempos de choque de menor a mayor,}
YAux:=Tch[1]; { evitando tiempos negativos en tch[1] }
Tch[1]:=Tch[2];
Tch[2]:=YAux;
end;
if (Tch[1] <= tiempo)and(Tch[1] > 0) then
begin      { si ya choc• en Tch[1], Calcula interseccion Cch[1][1] }
Cch[1][1,1]:=(GrowR*V/Xm*(Tch[1]-Tp1));
Cch[1][1,2]:=(D1*(Sqr(GrowR/Xm*(Tch[1]-Tp1))-1));
Cch[1][1,2]:=(Sqrt(Cch[1][1,2]));
YAux0:=(GrowR*(Tch[1]-Nu[Ni[4]]^t0));
YAux:=(Distancia(Cp[4],Cch[1][1]));
if Cp[4,2] < 0 then
begin      { Si el n—cleo extra est... debajo del eje x, hace y negativa }

```

```

Cch[1][1,2]:=-Cch[1][1,2];
YAux:=(Distancia(Cp[4],Cch[1][1]));
if (YAux <> YAux0)and(Cch[1][1,2] >0) then
begin
Cch[1][1,2]:=-Cch[1][1,2];
end;
end;
end else
begin { Como el Tiempo de choque es menor que el tiempo del analisis, }
Cch[1][1,1]:=0; { no ha chocado }
Cch[1][1,2]:=0;
end;
if (Tch[2] <= tiempo)and(Tch[2] > 0) then
begin { si ya choc• en Tch[2], Calcula interseccion Cch[1][2] }
Cch[1][2,1]:=(GrowR*V/Xm*(Tch[2]-Tp1));
Cch[1][2,2]:=(D1*(Sqr(GrowR/Xm*(Tch[2]-Tp1))-1));
Cch[1][2,2]:=(-Sqrt(Cch[1][2,2]));
Px2[1]:=(Distancia(Cp[4],Cch[1][2]));
Px2[2]:=(GrowR*(Tch[2]-Nu[Ni[4]]^t0));
if Redondea(Px2[1],Rdo[4]) <> Redondea(Px2[2],Rdo[4]) then
begin { Cambia de signo el y de choque (Cch[1][2]) }
Cch[1][2,2]:=-Cch[1][2,2];
end;
end else
begin { Como el Tiempo de choque es menor que el tiempo del analisis, }
Cch[1][2,1]:=0; { no ha chocado }
Cch[1][2,2]:=0;
end;
end;
For kk:=1 to 2 do For Kj:=1 to 2 do

```

```

begin
Cch[1][kk,kj]:=Redondea(Cch[1][kk,kj],Rdo[7]);
end;
if (Cch[1][1,2] < Cch[1][2,2])and
  (Cch[1][2,1] <> 0)and(Cch[1][2,2] <> 0)and
  (Cch[1][1,1] <> 0)and(Cch[1][1,2] <> 0)then
begin
IntercambioPnt(Cch[1][2],Cch[1][1]);
IntercambioRl(Tch[1],Tch[2]);
end;
if Tch[1] > 0 then
begin
R:=GrowR*(Tch[1]-Nu[Ni[4]]^t0);
R1:=Distancia(Cp[4],Cch[1][1]);
Cch[1][1,2]:=-Cch[1][1,2];
R2:=Distancia(Cp[4],Cch[1][1]);
if Abs(R-R1) < Abs(R-R2) then
begin
Cch[1][1,2]:=-Cch[1][1,2];
end;
end;
if Tch[2] > 0 then
begin
R:=GrowR*(Tch[2]-Nu[Ni[4]]^t0);
R1:=Distancia(Cp[4],Cch[1][2]);
Cch[1][2,2]:=-Cch[1][2,2];
R2:=Distancia(Cp[4],Cch[1][2]);
if Abs(R-R1) < Abs(R-R2) then
begin
Cch[1][2,2]:=-Cch[1][2,2];

```

```

end;
end;
if (Cch[1][1,1] < 0) then
begin
Cch[1][1,1]:=0;
Cch[1][1,2]:=0;
Tch[1]:=-1;
end;
For Ki:=1 to 2 do
begin
if Not ((Tch[Ki] > 0)and(Tch[Ki] <= Tiempo)) then
begin
Cch[1][Ki,1]:=0;
Cch[1][Ki,2]:=0;
Tch[Ki]:=-1;
end;
end;

end;

```

Procedure CalcHipRec(H1,H2 : Hiperbola);

Var

L1 : Pnt;

L2 : Pnt2;

iCh : Pnt12;

Ki : Word;

begin

Posiciona(H1.Lim[1],H1.Po,H1.A,True,L2[1]);

Posiciona(H1.Lim[2],H1.Po,H1.A,True,L2[2]);

```

Reposicio(L2[1],H2.Po,H2.A,L2[1]);
Reposicio(L2[2],H2.Po,H2.A,L2[2]);
InterH_R(L2[1],L2[2],H2.Pk[1],H2.Pk[2],Cch[2][1],Cch[2][2]);
L1[2]:=0;
if H2.Nuc[1] = inN then
begin
L1[1]:=-H2.Xm;
end else
begin
L1[1]:=H2.Xm;
end;
Tch[1]:=Distancia(L1,Cch[2][1])/GrowR+Nu[iNN]^t0;
Tch[2]:=Distancia(L1,Cch[2][2])/GrowR+Nu[iNN]^t0;
For Ki:=1 to 2 do
begin
if Not ((Tch[ki] > 0)and(Tch[ki] <= Tiempo))or(Cch[2][Ki,1] < 0) then
begin
Cch[2][ki,1]:=0;
Cch[2][ki,2]:=0;
Tch[Ki]:=-1;
end;
end;
if (Cch[2][1,1] = 0)and(Cch[2][1,2] = 0)and((Cch[2][2,1] <> 0)or(Cch[2][2,2] <> 0)) then
begin
IntercambioPnt(Cch[2,1],Cch[2][2]);
IntercambioRl(Tch[1],Tch[2]);
end;

end;

```

```

Procedure CalculaChoques(H1,H2 : Hiperbola);
Var
Ki,Kj : Word;
begin
Tch[1]:=0;
Tch[2]:=0;
NucPr:=Nu[inN]^;

if H1.HipRec then
begin
{Define nucleos involucrados}
Ni[1]:= H1.Nuc[1];
Ni[2]:= H1.Nuc[2];
Ni[3]:= H2.Nuc[2];
Ni[4]:= H2.Nuc[1];
if (Ni[1] <> 0) and (Ni[2] <> 0) then
begin
Tp1:=(Nu[Ni[1]]^.t0+Nu[Ni[2]]^.t0)/2; {Tiempos medios}
end else
begin
if (Ni[1] <> 0) then
begin
Tp2:=(Nu[Ni[1]]^.t0)/2;
end else
begin
Tp2:=(Nu[Ni[2]]^.t0)/2;
end;
end;
if (Ni[3] <> 0) and (Ni[3] <> 0) then
begin

```

```

Tp2:=(Nu[Ni[3]]^.t0+Nu[Ni[4]]^.t0)/2;
end else
begin
if (Ni[3] <> 0) then
begin
Tp2:=(Nu[Ni[3]]^.t0)/2;
end else
begin
Tp2:=(Nu[Ni[4]]^.t0)/2;
end;
end;
if Nu[H1.Nuc[1]]^.t0 > Nu[H1.Nuc[2]]^.t0 then
begin
Ni[1]:=H1.Nuc[2];
Ni[2]:=H1.Nuc[1];
end;
if (H2.Nuc[1] = Ni[1])or(H2.Nuc[1] = Ni[2]) then
begin
Ni[3]:=H2.Nuc[1];
Ni[4]:=H2.Nuc[2];
end;
With H1 do
begin
For Ki:=1 to 4 do {Pone los cuatro nucleos involucrados en posicion de la}
begin {hiperbola a analizar}
if Ni[Ki] <> 0 then
begin
Reposicio(Nu[Ni[Ki]]^.Pc,Po,A,Cp[Ki]);
end else
begin

```

```

    Cp[ki,1]:=0;
    Cp[ki,2]:=0;
end;
end;
if Cp[4,2] = 0 then {Si las hipérbolas son paralelas}
begin
    CalcParalela(H1,H2);
end else
begin
    CalculaGr1(H1);
end;
end;
end else
begin
    CalcHipRec(H1,H2);
end;

```

```

if HPV[1]^HipRec then Ki:=1 else Ki:=2;

```

```

For kj:=1 to 2 do
begin
if Cch[Ki][Kj,1] < 0 then
begin
    Cch[Ki][Kj,1]:=0;
    Cch[Ki][Kj,2]:=0;
    Tch[Kj]:=-1;
end;
end;
if (Tch[1] < 0)and(Tch[2] >=0) then
begin

```

```

IntercambioPnt(Cch[Ki][1],Cch[Ki][2]);
IntercambioRl(Tch[1],Tch[2]);
end;
if (Tch[2] < Tch[1])and((Cch[ki][2,1] <> 0)or(Cch[ki][2,2] <> 0)) then
begin
if Cch[ki][2,1] > 0 then
begin
IntercambioPnt(Cch[Ki,1],Cch[Ki][2]);
IntercambioRl(Tch[1],Tch[2]);
end else
begin
Cch[Ki][2,1]:=0;
Cch[Ki][2,2]:=0;
end;
end;

For kj:=1 to 2 do
begin

if (Tch[kj] > 0)and(Tch[kj] < Tiempo) then
begin
Posiciona( Cch [Ki][kj],Hpv[Ki]^Po ,Hpv[Ki]^A,True,CchA[Ki][kj]);
RePosicio( CchA[Ki][kj],Hpv[3-Ki]^Po,Hpv[3-Ki]^A, Cch[3-Ki][kj] );
if Hpv[1]^HipRec then
begin
EscalaXY(CchA[1][Kj],iCch[1][Kj]);
end else
begin
EscalaXY(CchA[2][Kj],iCch[2][Kj]);
end;

```

end;

end;

For Ki:=1 to 2 do For Kj:=1 to 2 do

begin

Cch[Ki][Kj,1]:=Redondea(Cch[Ki][Kj,1],Rdo[7]);

Cch[Ki][Kj,2]:=Redondea(Cch[Ki][Kj,2],Rdo[7]);

end;

end;

Function Cruza2(Hp10, Hp20 : Hiperbola; L4, L9 : Pnt; T : Real; CualBy : Word) : Boolean;

Var

Ki : Word;

D : Extended;

Pc10, Pc20 : Pnt;

L5, L6, L7, L10, L20 : Pnt2;

L5i, L6i, L1i, L2i : PntI2;

L3 : Array[1..3] of Pnt;

L12 : Array[1..5] of Pnt;

Px1 : Array[1..6] of Pnt;

Px : Array[1..10] of Pnt;

L3i : Array[1..3] of PntI;

Pxi : Array[1..6] of PntI;

Hp30 : Hiperbola;

Cruza5, Prueba : Boolean;

CrX1 : Array[1..5] of Boolean;

R, R1 : Extended;

Label 1;

```

Procedure PintAux;
Var Bas : Extended;
begin
  PintaAreaTrabajo(Pon);
  Circle(CxIn[1],CxIn[2],3);
  EscalaHip(Inn,Hp10,1,Bas);
  Temporal:=ReadKey;
  EscalaHip(Inn,Hp20,5,Bas);
  Temporal:=ReadKey;

  SetColor(9); Circle(L1i[1,1],L1i[1,2],3);
  SetColor(10); Circle(L1i[2,1],L1i[2,2],3);

  SetColor(1); Line(CxIn[1],CxIn[2],L1i[1,1],L1i[1,2]);
  SetColor(2); Line(CxIn[1],CxIn[2],L1i[2,1],L1i[2,2]);
  SetColor(3); Line(CxIn[1],CxIn[2],L3i[1,1],L3i[1,2]);

  Temporal:=ReadKey;
  SetColor(1); Line(CxIn[1],CxIn[2],L3i[2,1],L3i[2,2]);
  Temporal:=ReadKey;
  SetColor(2); Line(CxIn[1],CxIn[2],L3i[3,1],L3i[3,2]);

  Temporal:=ReadKey;
  Temporal:=ReadKey;

  SetColor(1); Circle(Pxi[1,1],Pxi[1,2],3);
  SetColor(2); Circle(Pxi[2,1],Pxi[2,2],3);

  SetColor(3); Circle(Pxi[3,1],Pxi[3,2],3);

```

```

SetColor(6); Circle(Pxi[4,1],Pxi[4,2],3);
SetColor(0); Circle(Pxi[5,1],Pxi[5,2],3);
{
SetColor(0); Circle(L2i[1,1],L2i[1,2],3);
SetColor(0); Circle(L2i[2,1],L2i[2,2],3);

SetColor(4); Circle(L5i[1,1],L5i[1,2],3);
SetColor(5); Circle(L5i[2,1],L5i[2,2],3);

SetColor(9); Circle(L2i[1,1],L2i[1,2],3);
SetColor(9); Circle(L2i[2,1],L2i[2,2],3);
}
Temporal:=ReadKey;
Temporal:=ReadKey;
end;

begin

EscalaXY(NucPr.Pc,CxIn);
L3[1]:=L4; {L4 es Respaldo de punto de EX }
Hp30:=Hp10; {H30 es Respaldo de hiperbola 1 (H1)}

1 :
D:=1e-4; {LA D}
Prueba:=False;
Cruza5:=False; {Inicializa booleana }
For Ki:=1 to 5 do CrX1[Ki]:= False; { Inicializa }

L3[2,2]:=(L3[1,2]+Abs(Hp10.Lim[1,2]-L3[1,2])*D);

```

```

L3[3,2]:=(L3[1,2]-Abs(Hp10.Lim[2,2]-L3[1,2])*D);

{
if (Hp10.idh = 34)and(Hp20.idh = 16) then
begin
  Temporal:=ReadKey;
end;
}

Posiciona(L3[1],Hp10.Po,Hp10.A,True,L3[1]);      { Intersecciones en Hiper 2 }
Reposicio(L3[1],Hp20.Po,Hp20.A,L12[3]);        { Interseccion a H2 }
EscalaXY(L3[1],L3i[1]);                        { Punto de Choque }

While Not CrX1[4] and Not CrX1[5] and Not Prueba{(D > 1e-5)} do
begin
For Ki:=2 to 3 do
begin
L3[Ki,1]:=CalculaX(L3[Ki,2],Hp10.Pk,Hp10.HipRec);
Posiciona(L3[Ki],Hp10.Po,Hp10.A,True,L3[Ki]);  { Intersecciones en Hiper 2 }
Reposicio(L3[Ki],Hp20.Po,Hp20.A,L12[Ki+2]);  { Interseccion a H2 }
EscalaXY(L3[Ki],L3i[Ki]);                    { Punto de Choque }
end;
For ki:=1 to 2 do
begin
Posiciona(Hp20.Lim[Ki],Hp20.Po,Hp20.A,True,L20[Ki]); { Ambos segmentos de H2 }
Reposicio(L20[Ki],Hp10.Po,Hp10.A,L5[Ki]);      { L2 de H2 al campo de H1 }
Posiciona(Hp10.Lim[Ki],Hp10.Po,Hp10.A,True,L10[Ki]); { Ambos segmentos de H1 }
Reposicio(L10[Ki],Hp20.Po,Hp20.A,L12[Ki]);    { L de H1 al campo de H2 }
EscalaXY(L20[Ki],L2i[Ki]);                    { Limite 1 de Hip2 }

```

```

EscalaXY(L10[Ki],L1i[Ki]); { Limite 1 de Hip1 }
end;
Reposicio(NucPr.Pc,Hp20.Po,Hp20.A,Pc10);      { Nucleo al Eje X de H2 }
Reposicio(NucPr.Pc,Hp10.Po,Hp10.A,Pc20);      { Nucleo al Eje X de H2 }
if Hp10.hipRec then
begin
InterH_R(Pc10,L5[1],Hp10.Pk[1],Hp10.Pk[2],L5[1],L6[1]); {Limite 1  }
InterH_R(Pc10,L5[2],Hp10.Pk[1],Hp10.Pk[2],L5[2],L6[2]); {Limite 1  }
end else
begin
Intersec(Pc10,L5[1],Hp10.Lim[1],Hp10.Lim[2],L100,L5[1]);
Intersec(Pc10,L5[2],Hp10.Lim[1],Hp10.Lim[2],L100,L5[2]);
end;
Posiciona(L5[1],Hp10.Po,Hp10.A,False,L5[1]);
Posiciona(L5[2],Hp10.Po,Hp10.A,False,L5[2]);
EscalaXY(L5[1],L5i[1]);
EscalaXY(L5[2],L5i[2]);
Reposicio(L5[1],Hp20.Po,Hp20.A,L6[1]);
Reposicio(L5[2],Hp20.Po,Hp20.A,L6[2]);

For Ki:=1 to 10 do {Intersecciones de la hiperbola 2, de las rectas entre}
begin          {el centro del nucleo y:          }
Px[Ki,1]:=0;
Px[Ki,2]:=0;
end;

For Ki:=1 to 5 do {Intersecciones de la hiperbola 2, de las rectas entre}
begin          {el centro del nucleo y:          }
if Hp20.hipRec then
begin

```

```

InterH_R(Pc10,L12[Ki],Hp20.Pk[1],Hp20.Pk[2],Px[Ki],Px[Ki+5]); {Limite 1 }
end else
begin
Intersec(Pc10,L12[Ki],Hp20.Lim[1],Hp20.Lim[2],L100,Px[Ki]); {Limite 1 }
Px[Ki+5]:=Px[Ki];
Px[Ki+5,1]:=-1;
end;
Posiciona(Px[Ki],Hp20.Po,Hp20.A,False,Px1[Ki]);
EscalaXY(Px1[Ki],Pxi[Ki]); {Interseccion H2, Recta con L1 de H1}

```

```

if Px[Ki,1] >= 0 then
begin
CrX1[Ki]:=Entre(Px[Ki],Pc10,L12[Ki]);
if CrX1[Ki] then CrX1[Ki]:=CruzaPun(True,Px[Ki,2],Hp20.Lim[1,2],Hp20.Lim[2,2],5);
end else CrX1[Ki]:=False;
if Not CrX1[Ki] and (Px[Ki+5,1] >= 0) then
begin
CrX1[Ki]:=Entre(Px[Ki+5],Pc10,L12[Ki]);
if CrX1[Ki] then CrX1[Ki]:=CruzaPun(True,Px[Ki+5,2],Hp20.Lim[1,2],Hp20.Lim[2,2],5);
if CrX1[Ki] then Px[Ki]:=Px[Ki+5];
end;
end;
if Not CrX1[2] and Not CrX1[3] and Not CrX1[4] and Not CrX1[5] and CrX1[1] then
begin
(* PintAux;*)
CrX1[4]:=True;
end;

```

```

if Not CrX1[1] then
begin
if (Redondea(Hp10.Lim[1,1],Rdo[3]) = Redondea(L4[1],Rdo[3]))and
(Redondea(Hp10.Lim[1,2],Rdo[3]) = Redondea(L4[2],Rdo[3])) then
begin
CrX1[1]:=True;
if (CrX1[4] = False)and(CrX1[4] = CrX1[5]) then
begin
CrX1[4]:=True;
end else
begin
if (CrX1[4] = False)and(CrX1[5] = True)and Not CrX1[2] then
begin
{ PintAux;}
CrX1[5]:=False;
end;
end;
end;
end;

```

```

if Not CrX1[2] then
begin
if (Redondea(Hp10.Lim[2,1],Rdo[3]) = Redondea(L4[1],Rdo[3]))and
(Redondea(Hp10.Lim[2,2],Rdo[3]) = Redondea(L4[2],Rdo[3])) then
begin
{ PintAux;}
CrX1[2]:=True;
if (CrX1[5] = False)and(CrX1[4] = CrX1[5]) then
begin
CrX1[5]:=True;

```

```

end;
end;
end;

if CrX1[1] and CrX1[2] and Not CrX1[3] and CrX1[4] and Not CrX1[5] then
begin
if (Redondea(Cch[1,1,2],Rdo[4]) = Redondea(Hp10.Lim[1,2],Rdo[4])) and
(Redondea(Cch[2,1,2],Rdo[4]) = Redondea(Hp20.Lim[1,2],Rdo[4])) then
begin
CrX1[1]:=False;
CrX1[2]:=True;
CrX1[4]:=False;
CrX1[5]:=True;
end;
end;

if CrX1[1] and CrX1[2] and CrX1[3] and (Not CrX1[4] or Not CrX1[5]) then
begin
if Redondea(L12[2,2],Rdo[4]) = Redondea(L9[2],Rdo[4]) then
begin
if Redondea(Hp10.Lim[2,2],Rdo[5]) <> Redondea(L4[2],Rdo[5]) then
begin
CrX1[3]:=False;
CrX1[4]:=True;
CrX1[5]:=False;
end;

end;

end;
if Redondea(L12[1,2],Rdo[4]) = Redondea(L9[2],Rdo[4]) then

```

```

begin
if Redondea(L4[2],Rdo[4]) <> Redondea(Hp10.Lim[1,2],Rdo[4]) then
begin
CrX1[3]:=False;
CrX1[4]:=False;
CrX1[5]:=True;
end else
begin
CrX1[3]:=False;
CrX1[4]:=True;
CrX1[5]:=False;
end;
end;
end;

{
if (inn = 7) and (Hp10.idh = 28)and(Hp20.idh = 2) then
begin
PintAux;
end;
}
if (CrX1[4] = CrX1[5]) then
begin
D:=D/10;
if D < 1e-5 then Prueba:=True;
end;

{
if (CrX1[4] <> CrX1[5]) then
begin

```

```

if (CrX1[1] <> CrX1[2]) then
begin
if CrX1[1] and Not CrX1[4] then
begin
if L12[3,2] = L12[4,2] then
begin
Temporal:=ReadKey;
end;
end;
end;
end;
}
if Not CrX1[1] then
begin
if Not CrX1[2] then
begin
if CrX1[3] and CrX1[4] <> CrX1[5] and (AEval = 1) then
begin
if CrX1[5] then
begin
if (L4[2] <> Hp10.Lim[2,2]){and(L9[2] <> Hp20.Lim[1,2])}then
begin

Dobla:=True;
{
L3[1,2]:=(L4[2]+Hp10.Lim[2,2])/2;
L3[2,1]:=CalculaX(L3[2,2],Hp10.Pk,Hp10.HipRec);
PintAux;
Temporal:=ReadKey;
}
}

```

```

end;
end;
end else
begin
if Not CrX1[5] and CrX1[3] then
begin
Dobla:=True;
end;
end;
end;
end;

if (CrX1[4] = CrX1[5]) and Prueba and (CrX1[1] = CrX1[3])and(CrX1[1] <> CrX1[2]) then
begin
if Redondea(Hp10.Lim[1,2],Rdo[5]) = Redondea(L4[2],Rdo[5]) then
begin
CrX1[1]:=True;
CrX1[2]:=False;
CrX1[3]:=False;
CrX1[4]:=True;
CrX1[5]:=False;
end;
end;
end;

if CrX1[1] or CrX1[2] or CrX1[3] then Cruza5:=True else Cruza5:=False;

ByAux[CualBy]:=0;
if Not CrX1[4] and CrX1[5] then
begin

```

```

if Redondea(Hp10.Lim[1,2],Rdo[5]) <> Redondea(L4[2],Rdo[5]) then
  begin
  ByAux[CualBy]:=2;
end else
  begin
  ByAux[CualBy]:=2;
end;
end else
begin
if CrX1[4] and Not CrX1[5] then
  begin
if Redondea(Hp10.Lim[2,2],Rdo[5]) <> Redondea(L4[2],Rdo[5]) then
  begin      {Aguas, aumente el if}
  ByAux[CualBy]:=1;
end else
  begin
  ByAux[CualBy]:=1 {2};
end;
end else
begin
if Not CrX1[4] and Not CrX1[5] and Not CrX1[2] then
  begin
if CrX1[1] and Not CrX1[3] then
  begin
if CruzaPun(True,Redondea(L9[2],Rdo[4]),Redondea(Hp20.Lim[1,2],Rdo[4]),Redondea(Hp20.Lim[2,2],Rdo[4]),5) then
  begin
  ByAux[CualBy]:=1;
end;
end else
begin

```

```

if Not CrX1[1] and CrX1[3] then
  begin
  if CruzaPun(True,Redondea(L9[2],Rdo[4]),Redondea(Hp20.Lim[1,2],Rdo[4]),Redondea(Hp20.Lim[2,2],Rdo[4]),5) then
    begin
    ByAux[CualBy]:=1;
    end;
  end;
end else
begin
begin
if Not CrX1[4] and Not CrX1[5] and Not CrX1[1] then
  begin
  if CrX1[2] and Not CrX1[3] then
    begin
    ByAux[CualBy]:=2; {Acabo de abrir}
    end else
    begin
    if Not CrX1[2] and CrX1[3] then
      begin
      ByAux[CualBy]:=2;
      end else
      begin
      if CrX1[2] and CrX1[3] then
        begin
        ByAux[CualBy]:=2;
        end;
      end;
    end;
  end else
  begin
  begin

```

```

if Not CrX1[1] and CrX1[2] and CrX1[3] and Crx1[4] THEN
  begin
  ByAux[CualBy]:=2
end else
  begin
  if CrX1[1] and Not CrX1[2] and CrX1[3] and Crx1[4] THEN
    begin
    ByAux[CualBy]:=1;
    end;
  end;

end;

end;
end;
end;
if ByAux[CualBy] <> 0 then Cruza5:=True;
Cruza2:=Cruza5;
end;

```

```

Procedure Asigna0(Ch,Ci: Word; Hip,Hip1,Lm : Word);
Var Cr1 : Boolean;
  R,R11 : Pnt; {A¥ADI Hip1 y R11 28/Ag 13:00 Hrs}
begin
if (Cch[1][Ci,1] >= 0)and(Cch[2][Ci,1] >= 0) then
  begin
  if Ch = 1 then
    begin
    R:=Cch[1][Ci];
    R11:=Cch[2][Ci];

```

```

end else
begin
R:=Cch[2][Ci];
R11:=Cch[1][Ci];
end;

if CruzaPun(True,Redondea(R[2],Rdo[4]),Redondea(Hpv[Hip]^Lim[1,2],Rdo[4]),Redondea(Hpv[Hip]^Lim[2,2],Rdo[4]),5)or
((Ch = 2)and(Asigno[1,1])) then

begin
{
if CruzaPun(True,Redondea(R11[2],Rdo[4]),Redondea(Hpv[Hip1]^Lim[1,2],Rdo[4]),Redondea(Hpv[Hip1]^Lim[2,2],Rdo[4])) then
begin
}
{
if (Hpv[1]^idh = 20)and(Hpv[2]^idh = 18) then
begin
Temporal:=ReadKey;
end;
}
}
Case Hip of
3: Hc[1]:=Hpv[Hip]^;
5: Hc[3]:=Hpv[Hip]^;
4: Hc[2]:=Hpv[Hip]^;
6: Hc[4]:=Hpv[Hip]^;
end;
Hpv[Hip]^Lim[Lm]:=R;
if Hpv[Hip]^Lim[1,2]*Hpv[Hip]^Lim[2,2] > 0 then Hpv[Hip]^Split:=True;
Case Hip of
3: begin Hc[1].Lim[3-Lm]:=R; if Hc[1].Lim[1,2]*Hc[1].Lim[2,2] > 0 then Hc[1].Split:=True;end;

```

```

5: begin Hc[3].Lim[3-Lm]:=R; if Hc[3].Lim[1,2]*Hc[3].Lim[2,2] > 0 then Hc[3].Split:=True;end;
4: begin Hc[2].Lim[3-Lm]:=R; if Hc[2].Lim[1,2]*Hc[2].Lim[2,2] > 0 then Hc[2].Split:=True;end;
6: begin Hc[4].Lim[3-Lm]:=R; if Hc[4].Lim[1,2]*Hc[4].Lim[2,2] > 0 then Hc[4].Split:=True;end;
end;
{
if (Hpv[1]^ .idh = 2)and(Hpv[2]^ .idh = 1) then
begin
EscalaHip(Inn,Hpv[Hip]^,11);
Temporal:=ReadKey;
EscalaHip(Inn,Hc[3],12);

Temporal:=ReadKey;
Temporal:=ReadKey;
end;
}
Asigno[Ch,Ci]:=True;
{
end;
}
end;
end;
end;

```

```

Procedure ChecaHpv(iK : Word);
Var Color : Word;
begin
Case iK of
1 : begin HPV[7]^:=HPV[3]^;Color:=04; end;
2 : begin HPV[7]^:=HPV[5]^;Color:=12; end;

```

```

3 : begin HPV[7]^:=HPV[4]^;Color:=05; end;
4 : begin HPV[7]^:=HPV[6]^;Color:=14; end;
end;
With HPV[7]^ do
begin
if Lim[1,2]*Lim[2,2] >= 0 then
begin
if (iK = 1)or
((iK = 2)and(Cambio[1,1]))or
(iK = 3)or
((iK = 2)and(Cambio[2,1])) then
begin
if Not Split then
Case ik of
1 : Cambio[1,1]:=True;
2 : Cambio[1,2]:=True;
3 : Cambio[2,1]:=True;
4 : Cambio[2,2]:=True;
end;
Split:=True;
end;
end;
end;
end;
{
if (Hpv[1]^ .idh = 2)and(Hpv[2]^ .idh = 1) then
begin
PintaAreaTrabajo(Pon);

EscalaHip(Inn,HpV[1]^,9);

```

```
EscalaHip(Inn,HpV[2]^,10);
```

```
Temporal:=ReadKey;
```

```
EscalaHip(Inn,HpV[7]^,0);
```

```
SetColor(0);
```

```
Circle(iCch[1][1,1],iCch[1][1,2],3);
```

```
Temporal:=ReadKey;
```

```
Temporal:=ReadKey;
```

```
end;
```

```
}
```

```
Case iK of
```

```
1 : HPv[3]^:=HPv[7]^;
```

```
2 : HPv[5]^:=HPv[7]^;
```

```
3 : HPv[4]^:=HPv[7]^;
```

```
4 : HPv[6]^:=HPv[7]^;
```

```
end;
```

```
end;
```

```
Procedure DefByAux(T,Ch,H1,H2,CualBy : Word);
```

```
begin
```

```
if (Cch[Ch][T][2] <> 0)and((Redondea(Cch[Ch][T][1],Rdo[4]) <= Redondea(Hpv[H1]^Lim[1,1],Rdo[4]))or  
  (Redondea(Cch[Ch][T][1],Rdo[4]) <= Redondea(Hpv[H1]^Lim[2,1],Rdo[4]))) then
```

```
begin
```

```
if ByAux[CualBy] <> 0 then Asigna0(Ch,T,H1,H2,ByAux[CualBy]);
```

```
if Not Asigno[Ch,T] then Hpv[H1]^:=Hpv[H1-2]^;
```

```
end else
```

```
begin
```

```
Hpv[H1]^:=Hpv[H1-2]^;
```

```
end;  
end;
```

```
Function CruzaDef(T,Ch,H1,H2 : Word): Boolean;  
begin  
  ByAux[1]:=0;  
  ByAux[2]:=0;  
  CruzaDef:=False;  
  if ((Tch[T] > 0) and (Tch[T] <= tiempo)) then  
    begin  
      CruzaDef:=Cruza2(Hpv[H1]^,Hpv[H2]^,Cch[1][T],Cch[2][T],Tch[T],1);  
      CruzaDef:=Cruza2(Hpv[H2]^,Hpv[H1]^,Cch[2][T],Cch[1][T],Tch[T],2);  
    end else CruzaDef:=False;  
  end;  
end;
```

```
Var iCh1      : Array[1..2] of PntI;  
  Ki         : Word;  
  Pc10,L6    : Pnt;  
  LCh        : Pnt;  
  Lx1        : Pnt;  
  Lx1i,Pc10I,L6I : PntI;  
  Bas       : Extended;  
  Label 1,2;
```

```
begin  
For k1:=1 to 2 do  
  begin  
For K2:=1 to 2 do  
  begin  
    Asigno[K1,K2]:=False;
```

```

Cambio[K1,K2]:=False;
Modif[K1,K2]:=False;
For ki:=1 to 2 do Cch[k1,k2,ki]:=0;
end;
Suma[K1]:=False;
end;

(*)
if (iNN = 1)and Not (HPv[1]^HipRec) {and(Hpv[2]^idh = 39)} then
begin
Temporal:=ReadKey;
end;
*)
CalculaChoques(Hpv[1]^Hpv[2]^);

(*)
if (iNN = 6)and(HPv[1]^idh = 40)and(Hpv[2]^idh = 39) then
begin
Temporal:=ReadKey;
end;
*)

For k1:= 3 to 6 do Hpv[k1]^:=Hpv[k1-2]^; {5 = 3 = 1, 6 = 4 = 2}

EscalaXY(Nu[inn]^Pc,Pc10I);

Cruzo:=False;

AEval:=0;

```

```

Anula:=False;
Igual6:=True;
Igual5:=True;

(*
if (iNN = 16)and(HPv[1]^.idh = 64)and(Hpv[2]^ .idh = 58) then
begin
PintaAreaTrabajo(Pon);

SetColor(Inn);Circle(Pc10I[1],Pc10I[2],2);

EscalaHip(inn,Hpv[1]^,1,Bas);
Temporal:=ReadKey;

EscalaHip(inn,Hpv[6]^,2,Bas);

SetColor(4);Circle(iCch[1,1,1],iCch[1,1,2],1);
SetColor(3);Circle(iCch[1,2,1],iCch[1,2,2],1);
SetColor(4);Circle(iCch[2,1,1],iCch[2,1,2],1);
SetColor(3);Circle(iCch[2,2,1],iCch[2,2,2],1);

Temporal:=ReadKey;
Temporal:=ReadKey;
end;
*)
if (Tch[1] > 0)and (Tch[1] <= Tiempo) then
begin
if ((Cch[1][1,1] >= 0)or(Cch[1][1,2] <> 0))or((Cch[2][1,1] >= 0)or(Cch[2][1,2] <> 0)) then
begin
if CruzaPun(True,Cch[1][1,2],Hpv[1]^ .Lim[1,2],Hpv[1]^ .Lim[2,2],3) then

```

```

begin
if CruzaPun(True,Cch[2][1,2],Hpv[2]^Lim[1,2],Hpv[2]^Lim[2,2],3) then
{
if CruzaPun(True,Redondea(Cch[2][1,2],Rdo[4]),Redondea(Hpv[2]^Lim[1,2],Rdo[4]),Redondea(Hpv[2]^Lim[2,2],Rdo[4])) then
if CruzaPun(True,Redondea(Cch[2][1,2],Rdo[3]),Redondea(Hpv[2]^Lim[1,2],Rdo[3]),Redondea(Hpv[2]^Lim[2,2],Rdo[3])) then
}
begin
if (Cch[1][1,1] = 0)and(Cch[1][1,2] = 0)and
(Cch[2][1,1] = 0)and(Cch[2][1,2] = 0) then
begin
end else
begin
AEval:=1;
end;
end;

end;
end;
if(Tch[2] > 0)and (Tch[2] <= Tiempo) then
begin
if ( (Cch[1][2,1] >= 0)or(Cch[1][2,2] <> 0))or((Cch[2][2,1] >= 0)or(Cch[2][2,2] <> 0)) then
begin
if CruzaPun(True,Cch[1][2,2],Hpv[1]^Lim[1,2],Hpv[1]^Lim[2,2],5) then
begin
if CruzaPun(True,Cch[2][2,2],Hpv[2]^Lim[1,2],Hpv[2]^Lim[2,2],5) then
begin
if (Cch[1][2,1] = 0)and(Cch[1][2,2] = 0)and
(Cch[2][2,1] = 0)and(Cch[2][2,2] = 0) then else
if AEval = 0 then
begin

```

```

    AuxCh:=Cch[1][1];
    Cch[1][1]:=Cch[1][2];
    Cch[1][2]:=AuxCh;
    AuxCh:=Cch[2][1];
    Cch[2][1]:=Cch[2][2];
    Cch[2][2]:=AuxCh;
    AuxCh[1]:=Tch[1];
    Tch[1]:=Tch[2];
    Tch[2]:=AuxCh[1];
    AEval:=1;
end else
begin
    AEval:=2;
end;
end;
end;
end;
end;
end;

if AEval = 2 then
begin
if (Redondea(Hpv[1]^Lim[1,2],Rdo[4]) =Redondea(Cch[1,1,2],Rdo[4]))and
(Redondea(Hpv[1]^Lim[2,2],Rdo[4]) =Redondea(Cch[1,2,2],Rdo[4]))and
(Redondea(Hpv[2]^Lim[1,2],Rdo[4]) =Redondea(Cch[2,1,2],Rdo[4]))and
(Redondea(Hpv[2]^Lim[2,2],Rdo[4]) =Redondea(Cch[2,2,2],Rdo[4])) then
begin
GoTo 2;
end;
if (Redondea(Hpv[1]^Lim[1,2],Rdo[4]) =Redondea(Cch[1,2,2],Rdo[4]))and

```

```

(Rdondea(Hpv[1]^Lim[2,2],Rdo[4]) =Redondea(Cch[1,1,2],Rdo[4]))and
(Rdondea(Hpv[2]^Lim[1,2],Rdo[4]) =Redondea(Cch[2,2,2],Rdo[4]))and
(Rdondea(Hpv[2]^Lim[2,2],Rdo[4]) =Redondea(Cch[2,1,2],Rdo[4])) then
begin
GoTo 2;
end;
end;

if AEval > 0 then
begin
if AEval >= 1 then
begin
{5 con 6, t1, ch1}
Dobla:=False;
(*)
if (iNN = 4) and (HPv[1]^idh = 24)and (Hpv[2]^idh = 12) then
begin
PintaAreaTrabajo(Pon);

SetColor(Inn);Circle(Pc10I[1],Pc10I[2],2);

EscalaHip(inn,Hpv[5]^4,Bas);
Temporal:=ReadKey;

EscalaHip(inn,Hpv[6]^2,Bas);
Temporal:=ReadKey;
{
EscalaHip(inn,Hc[3],3);
Temporal:=ReadKey;

```

```

EscalaHip(inn,Hc[4],4);
Temporal:=ReadKey;
}
SetColor(3);Circle(iCch[1,1,1],iCch[1,1,2],2);
SetColor(3);Circle(iCch[2,1,1],iCch[2,1,2],3);

Temporal:=ReadKey;
Temporal:=ReadKey;
end;
*)
CruzaX:=CruzaDef(1,1,5,6); {T,Ch,H1,H2}
if CruzaX then
begin
DefByAux(1,1,5,6,1); {(T,Ch,H1,H2}
DefByAux(1,2,6,5,2); {(T,Ch,H1,H2}

if Asigno[1,1] then
begin
ChecaHpv(2);
ChecaHpv(4);
end;
(*)
if (iNN = 1) and (HPv[1]^idh = 9)and (HPv[2]^idh = 4) then
begin
PintaAreaTrabajo(Pon);

SetColor(Inn);Circle(Pc10I[1],Pc10I[2],2);

EscalaHip(inn,Hpv[5]^4,Bas);
Temporal:=ReadKey;

```

```
EscalaHip(inn,Hpv[6]^2,Bas);  
Temporal:=ReadKey;
```

```
EscalaHip(inn,Hc[3],3,Bas);  
Temporal:=ReadKey;
```

```
EscalaHip(inn,Hc[4],4,Bas);  
Temporal:=ReadKey;
```

```
SetColor(3);Circle(iCch[1,1,1],iCch[1,1,2],2);  
SetColor(3);Circle(iCch[2,1,1],iCch[2,1,2],3);
```

```
Temporal:=ReadKey;  
Temporal:=ReadKey;
```

```
end;
```

```
*)
```

```
{
```

```
if Not Asigno[1,1] and Not Asigno[2,1] then  
begin
```

```
Temporal:=ReadKey;  
end;
```

```
if Asigno[1,1] <> Asigno[2,1] then  
begin
```

```
Temporal:=ReadKey;  
end;
```

```
}
```

```
end;
```

```
end;
```

```

if (AEval = 1)and Dobla then
begin

(*)
if (iINN = 4)and(HPv[1]^.idh = 18)and(Hpv[2]^ .idh = 15) then
begin
PintaAreaTrabajo(Pon);

SetColor(Inn);Circle(Pc10I[1],Pc10I[2],2);

EscalaHip(inn,Hpv[5]^,1);
Temporal:=ReadKey;

EscalaHip(inn,Hpv[6]^,4);
Temporal:=ReadKey;

EscalaHip(inn,Hc[3],3);
Temporal:=ReadKey;
{
EscalaHip(inn,Hc[4],4);
Temporal:=ReadKey;
}
SetColor(3);Circle(iCch[1,2,1],iCch[1,2,2],2);

Temporal:=ReadKey;
Temporal:=ReadKey;
end;
*)
if Asigno[1,1] then

```

```

begin
{
Dobla:=False;
}
Alt1:=False;

end;

{ else
begin
if Asigno[2,1] then
begin
Hpv[3]^:=Hc[3];
Hpv[4]^:=Hpv[6]^;
Asigno[1,2]:=True;
Asigno[2,2]:=True;
Iguales:=False;
Anula:=True;
Suma[1]:=True;
end;
end;
}

end;
if AEval = 2 then
begin

Lx[2]:=(Cch[1][1,2]+Cch[1][2,2])/2;
Lx[1]:=CalculaX(Lx[2],Hpv[1]^Pk,Hpv[1]^HipRec);

```

```

Posiciona(Lx,Hpv[1]^Po,Hpv[1]^A,True,Lx1);
EscalaXY(Lx1,Lx1I);
Reposicio(Nu[inn]^Pc,Hpv[2]^Po,Hpv[2]^A,Pc10);      { Nucleo al Eje X de H2 }
Reposicio(Lx1,Hpv[2]^Po,Hpv[2]^A,Lx1);
if Hpv[2]^HipRec then
begin
InterH_R(Pc10,Lx1,Hpv[2]^Pk[1],Hpv[2]^Pk[2],LCh,L6); {Limite 1  }
end else
begin
Intersec(Pc10,Lx1,Hpv[2]^Lim[1],Hpv[2]^Lim[2],L100,LCh);
end;
Posiciona(LCh,Hpv[2]^Po,Hpv[2]^A,True,L6);
(*
if (HPV[1]^idh = 34)and(Hpv[2]^idh = 10) then
begin
PintaAreaTrabajo(Pon);
SetColor(Inn);Circle(Pc10I[1],Pc10I[2],2);
EscalaHip(inn,Hpv[5]^1,Bas);
Temporal:=ReadKey;
EscalaHip(inn,Hpv[6]^2,Bas);
Temporal:=ReadKey;

EscalaHip(inn,Hc[3],3,Bas);
Temporal:=ReadKey;
EscalaHip(inn,Hc[4],4,Bas);
Temporal:=ReadKey;

SetColor(2);Circle(Lx1I[1],Lx1I[2],2);
SetColor(0);Line(Pc10I[1],Pc10I[2],Lx1I[1],Lx1I[2]);
SetColor(3);Circle(iCch[1,2,1],iCch[1,2,2],2);

```

```

Temporal:=ReadKey;
Temporal:=ReadKey;
end;
*)
if Entre(LCh,Pc10,Lx1) then
begin
Hpv[3]^:=Hc[3];
Anula:=True;
Igual5:=False;
end else
begin
Hpv[3]^:=Hpv[5]^;
Anula:=False;
Igual5:=True;
end;

if CruzaPun(True,Lx1[2],Hpv[6]^Lim[1,2],Hpv[6]^Lim[2,2],5)and(Abs(Lx1[1]) > Hpv[6]^V) then
begin
Hpv[4]^:=Hpv[6]^;
Igual6:=True;
end else
begin
if CruzaPun(True,Lx1[2],Hc[4].Lim[1,2],Hc[4].Lim[2,2],5) then
begin
Hpv[4]^:=Hc[4];
Igual6:=True;
end else
begin
Hpv[4]^:=Hpv[6]^;
end;

```

```

end;

(*)
if (iNN = 7) and (HPv[1]^idh = 21) and (HPv[2]^idh = 18) then
begin
PintaAreaTrabajo(Pon);
SetColor(Inn);Circle(Pc10I[1],Pc10I[2],2);
Temporal:=ReadKey;
EscalaHip(inn,HPv[3]^,1,Bas);
Temporal:=ReadKey;
EscalaHip(inn,HPv[4]^,2,Bas);
{
Temporal:=ReadKey;

EscalaHip(inn,HPv[5]^,3,Bas);
Temporal:=ReadKey;
EscalaHip(inn,HPv[6]^,4,Bas);
Temporal:=ReadKey;
}
SetColor(0);Line(Pc10I[1],Pc10I[2],Lx1I[1],Lx1I[2]);
SetColor(3);Circle(iCch[2,2,1],iCch[2,2,2],2);
Temporal:=ReadKey;
Temporal:=ReadKey;
end;
*)
{3 con 4, t2, ch1}
{Cambie el 2}
CruzaX:=CruzaDef(2,1,3,4); {T,Ch,H1,H2}
if CruzaX then
begin

```

```

DefByAux(2,1,3,4,1); {(T,Ch,H1,H2}
DefByAux(2,2,4,3,2); {(T,Ch,H1,H2}
if Asigno[1,2] then
  begin
    ChecaHpv(2);
    ChecaHpv(4);
  end;
end;
if CruzaPun(True,Lx1[2],Hpv[6]^Lim[1,2],Hpv[6]^Lim[2,2],5)and(Abs(Lx1[1]) > Hpv[6]^V) then
  begin
    if Asigno[2,2] then
      begin
        Hpv[6]^:=Hpv[4]^
      end else
        begin
          Hpv[4]^:=Hpv[6]^;
        end;
      Igual6:=True;

    end else
      begin
        if CruzaPun(True,Lx1[2],Hc[4].Lim[1,2],Hc[4].Lim[2,2],5) then
          begin
            Hpv[4]^:=Hc[4];
            Igual6:=True;
          end else
            begin
              Hpv[4]^:=Hpv[6]^;
            end;
          end;
        end;
      end;

```

```

(*)
if (HPv[1]^.idh = 5)and(Hpv[2]^.idh = 1) then
begin
PintaAreaTrabajo(Pon);
SetColor(Inn);Circle(Pc10I[1],Pc10I[2],2);
Temporal:=ReadKey;
EscalaHip(inn,HPv[5]^,1,Bas);
Temporal:=ReadKey;
EscalaHip(inn,HPv[6]^,2,Bas);
Temporal:=ReadKey;
EscalaHip(inn,HPv[3]^,3,Bas);
Temporal:=ReadKey;
EscalaHip(inn,HPv[4]^,4,Bas);
Temporal:=ReadKey;
Temporal:=ReadKey;
end;
*)
if Igual5 {and Not Hpv[5]^HipRec} then
begin
Hpv[5]^:=Hpv[3]^;
end;

if Igual6 {and Not Hpv[5]^HipRec} then
begin
Hpv[6]^:=Hpv[4]^;
end;

if Igual5 and Igual6 then Suma[1]:=False;
{
if (HPv[1]^idh = 34)and(Hpv[2]^idh = 16) then

```

```

begin
PintaAreaTrabajo(Pon);

EscalaHip(inn,Hpv[3]^,1);
Temporal:=ReadKey;
EscalaHip(inn,Hpv[4]^,2);
Temporal:=ReadKey;

EscalaHip(inn,Hpv[5]^,3);
Temporal:=ReadKey;
EscalaHip(inn,Hpv[6]^,4);

Temporal:=ReadKey;
Temporal:=ReadKey;
end;
}
end;
end;
{
if (HPv[1]^ .idh = 10)and(Hpv[2]^ .idh = 2) then
begin
PintaAreaTrabajo(Pon);

EscalaHip(inn,Hpv[3]^,1);
Temporal:=ReadKey;
EscalaHip(inn,Hpv[4]^,2);
Temporal:=ReadKey;

EscalaHip(inn,Hpv[5]^,3);
Temporal:=ReadKey;

```

```

EscalaHip(inn,Hpv[6]^,4);

Temporal:=ReadKey;
Temporal:=ReadKey;
end;
}
2 :
if Not Asigno[1,1] and Not Asigno[1,2] and Not Asigno[2,1] and Not Asigno[2,2] then
begin
Cruzo:=True;
end;

if Asigno[1,1] and Asigno[2,1] then
begin
Cruzo:=True;
if Asigno[1,2] and Asigno[2,2] then
begin
Suma[1]:=True;
if Igual5 {and Not Hpv[5]^HipRec} then
begin
Suma[1]:=False;
end;
end;
end;

For ki:=3 to 6 do
begin
if (Hpv[ki]^Lim[1,2]*Hpv[ki]^Lim[2,2] > 0)and Not Hpv[ki]^Split then
begin
Hpv[ki]^Split:=True;

```

```
end;  
end;  
1 : end;
```

```
Procedure Quitar(Nuc1,Indice : Word);
```

```
Var
```

```
Ki,Kj : Word;
```

```
N1,N2 : Word;
```

```
begin
```

```
{  
  EscalaHip(i,Hpp[ToQ]^,5);  
  Temporal:=ReadKey;  
  Temporal:=ReadKey;  
}
```

```
Ki:=1;
```

```
N1:=Hpp[ToQ]^ Nuc[1];
```

```
N2:=Hpp[ToQ]^ Nuc[2];
```

```
For Ki:= 1 to NoNuc do
```

```
begin
```

```
  Kj:=1;
```

```
  While (Nu[ki]^id[Kj] < ToQ)and(Kj < Nu[Ki]^N)do Inc(kj);
```

```
  if Nu[ki]^id[Kj] = 0 then Dec(kj);
```

```
  While (Kj <= Nu[Ki]^N)and(Kj <> 0) do
```

```
begin
```

```
  if Kj > 0 then
```

```
begin
```

```
  if (Ki = N1)or(Ki = N2) then
```

```
begin
```

```

    Nu[ki]^id[Kj]:=Nu[ki]^id[Kj+1];
end;
if Nu[ki]^id[Kj] > ToQ then Nu[ki]^id[Kj]:=Nu[ki]^id[Kj]-1;
Inc(Kj);
end;
end;
if (Ki = N1)or(Ki = N2) then
begin
if Nu[Ki]^N > 0 then
begin
Dec(Nu[Ki]^N);
end;
end;
end;
Ki:=ToQ;
While Ki < NoFro do
begin
Hpp[Ki]^:=Hpp[Ki+1]^;
Hpp[Ki]^idh:=Hpp[Ki]^idh-1;
Inc(Ki);
end;
Dec(NoFro);

if Hpv[1]^idh = ToQ then
begin
Hpv[1]^:=Hpp[Nu[Nuc1]^id[Indice]]^;
end else
begin
if Hpv[2]^idh = ToQ then
begin

```

```
if (Indice <= Nu[Nuc1]^N)and(Indice > 0) then
begin
  Hpv[2]^:=Hpp[Nu[Nuc1]^id[Indice]]^;
end;
end;
end;
```

```
if Hpv[1]^idh >= ToQ then
begin
  Dec(Hpv[1]^idh);
end else
begin
  GoToXY(1,32);Write('Nada: ',ToQ:2);
end;
```

```
{
GoToXY(56,25);Write('Quito ',ToQ:3);
}
end;
```

```
Procedure SoloAsigna(i,j,Nt : Word);
Var
k,Kl,Hi1,KiH : Word;
Alt : Boolean;
Existe : Boolean;
```

```
Procedure Ya;
Var
L1,L2 :Pnt;
```

```

begin
Posiciona(Hpv[1]^Lim[1],Hpv[1]^Po,Hpv[1]^A,True,L1);
Posiciona(Hpv[1]^Lim[2],Hpv[1]^Po,Hpv[1]^A,True,L2);
{
if Entre(L1,Ori,Uno)or Entre(L2,Ori,Uno)then
begin
}
if Nt > 0 then
begin
Hpp[Nu[i]^id[k]]^:=Hpv[2]^,
{
if Hpv[1]^idh > Hpp[Nu[i]^id[Nu[i]^N]]^idh then
}
if Hpv[1]^idh > NoFro then
begin
Inc(NoFro);
Nu[i]^:=NucPr;
Inc(Nu[i]^,N); {Incrementa el número de fronteras del nucleo i}
Nu[i]^id[Nu[i]^N]:=NoFro; {Se asigna identificadores de la hiperbola a los nucleos}
Hpp[NoFro]^:=Hpv[1]^;
if j > 0 then
begin
Inc(Nu[j]^,N); {Incrementa el numero de fronteras del nucleo j}
Nu[j]^id[Nu[j]^N]:=NoFro; {Se asigna identificadores de la hiperbola a los nucleos}
end;
NucPr:=Nu[i]^;
end else
begin
Hpp[Hpv[1]^idh]^:=Hpv[1]^;
end

```

```

end else
begin
  Inc(NoFro);
  Nu[i]^:=NucPr;
  Inc(Nu[i]^N);           {Incrementa el numero de fronteras del nucleo i}
  Nu[i]^id[Nu[i]^N]:=NoFro; {Se asigna identificadores de la hiperbola a los nucleos}
  Hpp[NoFro]^:=Hpv[1]^;
  if j > 0 then
  begin
    Inc(Nu[j]^N);           {Incrementa el numero de fronteras del nucleo j}
    Nu[j]^id[Nu[j]^N]:=NoFro; {Se asigna identificadores de la hiperbola1 a los nucleos}
  end;
  NucPr:=Nu[i]^,
end;
{
end;
}
end;

```

```

Var Cxi : PntI;
    Bas : Extended;
    Va,Kii : Word;
    Ve,Quito : Boolean;
    Label 1,2,3,4,5;

```

```

Procedure DibujaAux(C1,C2 : Word);
Var ki,Kj : Word;
begin
PintaAreaTrabajo(Pon);
Kj:=2;
For ki:=1 to NoNuc do
begin
EscalaXY(Nu[ki]^Pc,Cxi);
if Kj < 13 then Inc(Kj) else Kj:=1;
SetColor(kj);Circle(Cxi[1],Cxi[2],2);
end;
Kj:=2;
For ki:=1 to Nu[i]^N do
begin
EscalaHip(i,Hpp[Nu[i]^id[ki]]^kj,Bas);
if Kj < 13 then Inc(Kj) else Kj:=1;
end;
SetColor(i);EscalaXY(Nu[i]^Pc,CxIn);
Circle(CxIn[1],CxIn[2],3);

EscalaHip(i,Hpv[1]^C1,Bas);
EscalaHip(i,Hpv[2]^C2,Bas);
Temporal:=ReadKey;
Temporal:=ReadKey;
end;

{
For kii:=1 to Nu[3]^N do
begin
EscalaHip(3,Hpp[Nu[3]^id[kii]]^kii,Bas);

```

```
end;  
  Temporal:=ReadKey;  
  Temporal:=ReadKey;  
}
```

```
begin  
  1 : Cruza1:=True;  
  Suma[1]:=False;  
  Suma[2]:=False;  
  Hpv[10]^:=Hpv[1]^;  
  k:=1;  
  Alt:=False;  
  Alt1:=False;
```

```
While k <= Nt do  
  begin
```

```
    HPV[2]^:=Hpp[Nu[i]^].id[k]^;
```

```
    if Not Hpv[1]^HipRec and Not Hpv[2]^HipRec then
```

```
      begin
```

```
        if Hpv[1]^idh > Nu[i]^id[Nu[i]^].N] then
```

```
          begin
```

```
            Alt1:=CruzaAlt(i, Nu[i]^N,Hpv[1]^, Hpv[2]^,False)
```

```
            if Not Alt1 then
```

```
              begin
```

```
                Ya;
```

```
              end;
```

```

end;
end;

if (Hpv[1]^Pk[1] <> Hpv[2]^Pk[1])or(Hpv[1]^Pk[2] <> Hpv[2]^Pk[2]) then
begin

AcotaLimite(i,Nu[i]^N,Cruza1);

if (Asigno[1,1]{ and Not Hpv[1]^HipRec}) or (k = Nu[i]^N) then
begin
if Not Anula then
begin
Hpv[1]^:=Hpv[5]^;
(*)
if (i = 13) and (Hpv[01]^idh= 57) and Not Hpv[01]^HipRec{ and (Hpv[2]^idh = 11)} then
begin
DibujaAux(1,2);
Temporal:=ReadKey;
end;
*)
Alt:=CruzaAlt(i,Nu[i]^N,Hpv[1]^,Hpv[2]^,False);
if Alt then
begin
GoTo 5;
Alt:=False;
end else
begin
if AEval = 1 then
begin

```

```

Alt1:=CruzaAlt(i,Nu[i]^N,Hpv[2]^,Hpv[1]^,False);
if Not Alt1 then
begin
Hpv[2]^:=Hpv[6]^;
Ya;
end else
begin
Existe:=False;
Hi1:=1;
While Not Existe and (Hi1 <= Nu[i]^N) do
begin
if Nu[i]^id[Hi1] = Hpv[2]^idh then Existe:=True else Inc(Hi1);
end;
if Existe then
begin

ToQ:=Hpv[2]^idh;
Quitar(i,Hi1-1);
Nt:=Nu[i]^N;
NucPr:=Nu[i]^;
Alt1:=False;
if k > 1 then Dec(k);
end;
end;
end;
end;
end;
end;
end;
Inc(k);

```

```

end;

Ve:=False;
2: k:=1;
While k <= Nt{NucPr.N} do
begin
HPv[2]^:=Hpp[Nu[i]^id[k]]^;
Va:=k;
if (Hp[1]^Pk[1] <> Hp[2]^Pk[1])or(Hp[1]^Pk[2] <> Hp[2]^Pk[2]) then
begin

Quita:=False;
AcotaLimite(i,Nu[i]^N,Cruza1);
if Igual5 and Igual6 then Suma[1]:=False;
Quito:=False;
3: Hp[1]^:=Hp[5]^;

if (Hp[6]^Lim[1,2] <> Hp[6]^Lim[2,2]) then
begin
Hp[2]^:=Hp[6]^;
end;
(*)
if (i = 3) and (Hp[1]^idh = 18) and Not Hp[01]^HipRec{ and (Hp[2]^idh = 11)} then
begin
DibujaAux(0,14);
Temporal:=ReadKey;
end;
*)
if Not Alt then Alt:=CruzaAlt(i,Nu[i]^N,Hp[1]^,Hp[2]^,False);

```

```
if Alt and Not Dobla then
begin
if Hpv[1]^idh > NoFro then
begin
GoTo 4;
end;
end;
```

```
if Not Alt and Cruza1 then
begin
Ya;
```

```
if Nu[i]^N > Nt then Nt:=Nu[i]^N;
if Not Alt1 then Alt1:=CruzaAlt(i,Nu[i]^N,Hpv[2]^,Hpv[1]^,False);
if Alt1 then
begin
ToQ:=Hpv[2]^idh;
Quitar(i,k);
Nt:=Nu[i]^N;
NucPr:=Nu[i]^;
Alt1:=False;
if k > 1 then Dec(k);
end;
end else
begin
Hpv[1]^:=Hpv[10]^;
if Alt then
begin
Existe:=False;
```

```

Hi1:=1;
While Not Existe and (Hi1 <= Nu[i]^N) do
  begin
  if Nu[i]^id[Hi1] = Hpv[1]^idh then Existe:=True else Inc(Hi1);
  end;
if Existe then
  begin
  ToQ:=Hpv[1]^idh;

  Quitar(i,Hi1-1);
  Nt:=Nu[i]^N;
  NucPr:=Nu[i]^;
  Quito:=True;
  k:=1;
  Alt:=False;
  end;
end;
end;

if Suma[2] then
  begin
  Hpv[8]^:=Hpv[1]^;
  Hpv[1]^:=Hpv[3]^;
  Alt:=CruzaAlt(i,Nu[i]^N,Hpv[1]^,Hpv[2]^,False);
  if Not Alt and Cruza1 then Ya;
  Hpv[1]^:=Hpv[8]^;
  {
  GoToXY(60,5);Write('SUMA2');
  }
  end;

```

```

if Suma[1] then
begin

Hpv[8]^:=Hpv[1]^;
Hpv[3]^ .idh:=NoFro+1;
Hpv[1]^:=Hpv[3]^;
(*)
if (i = 6) and (Hpv[1]^ .idh = 40){ and (Hpv[2]^ .idh = 18)} then
begin
DibujaAux(0,14);
end;
*)
Alt:=CruzaAlt(i,Nu[i]^ .N,Hpv[1]^,Hpv[2]^,False);

if Not Alt and Cruza1 then
begin
Ya;
if Not Quito then Hpv[1]^:=Hpv[8]^;
Inc(Nt);
Ve:=True;
Quito:=False;
end else
begin
Hpv[1]^:=Hpv[8]^;
Alt1:=False;
Alt:=False;
end;
end;
end;

```

```

k:=k+1;
end;

if Ve then
begin
Hpv[1]^:=Hpp[Nu[i]^].id[Nu[i]^].N]]^;
GoTo 1;
{
Temporal:=ReadKey;
}
4 : end;
if NucPr.N = 0 then Ya;
(*
*)
5: end;

Label 12;

Procedure HipProv(i,j : Word); {Identificadores de Hiperbola y n£cleos}
Var
Principal : Word;
Ang      : Extended;

Procedure IntEjes;
Var XL : array [1..2] of Pnt;
    l : Word;
begin
For l:=1 to 4 do
begin
Reposicio(XYe[l],Hpv[1]^Po,Hpv[1]^A,XL[l]);

```

```

if l < 4 then Reposicio(XYe[l+1],Hpv[1]^Po,Hpv[1]^A,XL[2]) else
Reposicio(XYe[l],Hpv[1]^Po,Hpv[1]^A,XL[2]);
InterH_R(XL[1],XL[2],Hpv[1]^Pk[1],Hpv[1]^Pk[2],XL[1],XL[2]);
if (XL[1,1] >=0)then
begin
if (XL[1,2] >= 0)and(XL[1,2] < Hpv[1]^Lim[1,2]) then
begin
Hpv[1]^Lim[1]:=XL[1];
end else if (XL[1,2] <= 0)and(XL[1,2] > Hpv[1]^Lim[2,2]) then
begin
Hpv[1]^Lim[2]:=XL[1];
end;
end;
if (XL[2,1] >= 0)then
begin
if (XL[2,2] >= 0)and(XL[2,2] < Hpv[1]^Lim[1,2]) then
begin
Hpv[1]^Lim[1]:=XL[2];
end else
begin
if (XL[2,2] <= 0)and(XL[2,2] > Hpv[1]^Lim[2,2]) then Hpv[1]^Lim[2]:=XL[2];
end;
end;
end;
end;

begin

if Nu[i]^t0 > Nu[j]^t0 then Principal:=i else Principal:=j;
NucPr:=Nu[i]^;

```

```

With Hpv[1]^ do
begin
idh:=NoFro+1;
Nuc[1]:=i;      {Se asigna identificadores de nucleos a hip,rbola}
Nuc[2]:=j;      {Se asigna identificadores de nucleos a hip,rbola}
if Nu[j]^ .t0 < NucPr.t0 then{Se orden de viejo a joven}
begin
Nuc[1]:=j;
Nuc[2]:=i;
end;
Xm:=Xmedia(i,j);
V :=Posit(Vertice(i,j));
Tv:=(Xm+V)/GrowR+Nuc[Principal]^ .t0;
CentroH(Nuc[1],Nuc[2],Po);      {Se calcula el centro de la Hip.}
A:=Angulo(Nu[Nuc[1]]^.Pc,Nu[Nuc[2]]^.Pc); {Se calcula el ngulo a girar desde el eje X}
Pk[1]:=Sqr(V);                  {Las constantes a,b}
Pk[2]:=Sqr(Xm)-Pk[1];
Vr[1]:=V;                       {Se calcula el vřrtice ya girado}
Vr[2]:=0;
Posiciona(Vr,Po,A,True,Vr);
Lim[1,1]:=Sqr(GrowR)*(Sqr(Tiempo-Nu[Hpv[1]^ .Nuc[1]]^.t0)-Sqr(Tiempo-Nu[Hpv[1]^ .Nuc[2]]^.t0))/(4*Hpv[1]^ .Xm);
Lim[2,1]:=Hpv[1]^ .Lim[1,1];
Lim[1,2]:=Sqrt((Sqr(Hpv[1]^ .Lim[1,1])/(Hpv[1]^ .Pk[1])-1)*Hpv[1]^ .Pk[2]);
Lim[2,2]:=-Hpv[1]^ .Lim[1,2];
HipRec:=True;
Split:=False;
end;
IntEjes;
SoloAsigna(i,j,NucPr.N);
end;

```

```

Const
  NoNucl =NoNMax;

Var
  OrK : Array [1..NoNucl] of Word;

Procedure Minimo(i : Word);

Var
  Dist : Array [1..NoNucl] of Extended; {Cambiar el 4 Por NoNuc}
  j : Word;
  Min : Extended;

Procedure Sort(l, r: Integer);
var
  i1,j1,Ok : integer;
  x,y : Extended;
begin
  i1:=l;
  j1:=r;
  x:=Dist[(l+r) DIV 2];
  repeat
  while Dist[i1] < x do i1:=i1+1;
  while x < Dist[j1] do j1:=j1-1;
  if i1 <= j1 then
  begin
  y:=Dist[i1];
  Ok:=Ork[i1];
  Dist[i1]:=Dist[j1];

```

```

OrK[i1]:=OrK[j1];
Dist[j1]:= y;
OrK[j1]:=OK;
i1:=i1+1;
j1:=j1-1;
end;
until i1 > j1;
if l < j1 then Sort(l,j1);
if i1 < r then Sort(i1,r);
end;

begin
OrK[i]:=i;
Dist[i]:=0;
For j:= 1 to NucRed do
begin
if j <> i then
begin
Dist[j]:=Distancia(Nu[i]^Pc,Nu[j]^Pc);
OrK[j]:=j;
end;
end;
Sort(1,NucRed);
end;

Procedure LimEjes(i : Word);
Var Rad    : Extended;
    P1,P2  : Pnt;
    k      : Word;
    Modifica : Boolean;

```

```

Paso    : Boolean;

Procedure AsignaFroR (j : Word);
  Label 1;
begin
Paso:=False;
1: With Hpv[1]^ do
begin
if (Lim[1,1] <> Lim[2,1]) or
  (Lim[1,2] <> Lim[2,2]) then
begin
if Not CruzaAlt(i,Nu[i]^ .N,Hpv[1]^,Hpv[2]^,False) then
begin
if Nu[i]^ .N = 0 then
begin
With NucPr do
begin
Inc(NoFro);
Inc(N);      {Incrementa el número de fronteras del nucleo i}
id[N]:=NoFro; {Se asigna identificadores de la hipórbola a los nucleos}
Nu[i]^:=NucPr;
Hpp[NoFro]^:=Hpv[1]^;
end;
end else
if Hpv[1]^ .idh > Hpp[Nu[i]^ .id[Nu[i]^ .N]]^ .idh then
begin
With NucPr do
begin
Inc(NoFro);
Inc(N);      {Incrementa el número de fronteras del nucleo i}

```

```

id[N]:=NoFro; {Se asigna identificadores de la hiperbola a los nucleos}
Nu[i]^:=NucPr;
Hpp[NoFro]^:=Hpv[1]^;
end;
end else
begin
Hpp[Hpv[1]^ .idh]^:=Hpv[1]^;
end;

if Modifica and (Nu[i]^ .N > 1) then
begin
Hpp[Hpv[2]^ .idh]^:=Hpv[2]^;
end;

if Suma[1] and Not Paso then
begin
Hpv[1]^:=Hpv[3]^;
Inc(Hpv[1]^ .idh);
Modifica:=False;
Paso:=True;
GoTo 1;
end;
end;
end;
end;
end;

Procedure Auxiliar;
begin
SoloAsigna(i,0,Nu[i]^ .N);

```

```

end;

begin
NucPr:=Nu[i]^;
Rad:=Radio(i);{GrowR*(Tiempo-NucPr.t0)};
if Rad > 0 then
begin
With Hpv[1]^ do With NucPr do
begin
Nuc[1]:=i;
Nuc[2]:=0;
Split:=False;
HipRec:=False;
Lim[1,1]:=0;
Lim[2,1]:=0;
Pk[1]:=0;
Pk[2]:=0;
if (Nu[i]^ .Pc[2]+Rad > 0) then
begin
Tv:=(Pc[2])/GrowR+t0;
if Tv < Tiempo then
begin { Con el eje y = 0 }
idh:=NoFro+1;
V:=Pc[2];
Vr[1]:=Pc[1];
Vr[2]:=0;
Po:=Vr;
A:=Pi/2*3;
P1[2]:=Sqrt(Sqr(Rad)-Sqr(V));
P2[2]:=-P1[2];

```

```

if Pc[1]+P1[2] < 1 then Lim[1,2]:=P1[2] else Lim[1,2]:=1-Pc[1];
if Pc[1]+P2[2] > 0 then Lim[2,2]:=P2[2] else Lim[2,2]:=-Pc[1];
SoloAsigna(i,0,Nu[i]^N);
end;
end;
if (Nu[i]^Pc[2]+Rad > 1) then
begin {Con el eje y = 1}
Tv:=(1-Pc[2])/GrowR+t0;
if Tv < Tiempo then
begin
idh:=NoFro+1;
V:=1-Pc[2];
Vr[1]:=Pc[1];
Vr[2]:=1;
Po:=Vr;
A:=Pi/2;
P1[2]:=Sqrt(Sqr(Rad)-Sqr(V));
P2[2]:=-P1[2];
if Pc[1]-P2[2] < 1 then Lim[2,2]:=P2[2] else Lim[2,2]:=Pc[1]-1;
if Pc[1]-P1[2] > 0 then Lim[1,2]:=P1[2] else Lim[1,2]:=Pc[1];
SoloAsigna(i,0,Nu[i]^N);
end;
end;
if (Pc[1]-Rad < 0) then
begin {Con el eje x = 0}
Tv:=Pc[1]/GrowR+t0;
if Tv < Tiempo then
begin
idh:=NoFro+1;
V:=Pc[1];

```

```

Vr[1]:=0;
Vr[2]:=Pc[2];
Po:=Vr;
A:=Pi;
P1[2]:=Sqrt(Sqr(Rad)-Sqr(V));
P2[2]:=-P1[2];
if Pc[2]-P1[2] > 0 then Lim[1,2]:=P1[2] else Lim[1,2]:=Pc[2];
if Pc[2]-P2[2] < 1 then Lim[2,2]:=P2[2] else Lim[2,2]:=Pc[2]-1;
SoloAsigna(i,0,Nu[i]^N);
end;
end;
if (Nu[i]^Pc[1]+Rad > 0) then
begin { Con el eje x = 1 }
Tv:=(1-Pc[1])/GrowR+t0;
if Tv < Tiempo then
begin
idh:=NoFro+1;
V:=(1-Pc[1]);
Vr[1]:=1;
Vr[2]:=Pc[2];
Po:=Vr;
A:=0;
P1[2]:=Sqrt(Sqr(Rad)-Sqr(V));
P2[2]:=-P1[2];
if P1[2]+Pc[2] < 1 then Lim[1,2]:=P1[2] else Lim[1,2]:=1-Pc[2];
if P2[2]+Pc[2] > 0 then Lim[2,2]:=P2[2] else Lim[2,2]:=-Pc[2];
SoloAsigna(i,0,Nu[i]^N);
end;
end;
end;

```

```
end;  
end;
```

```
Procedure InicaHiperbolas;  
Var i,j : Word;  
begin  
Assign(ArFron,'Fronter.Dat');  
ReWrite(ArFron);  
Seek(ArFron,0);  
NoFro:=0;  
For i:=1 to NoTFro do  
begin  
Hpp[i]^idh:=0;  
Hpp[i]^Nuc[1]:=0; Hpp[i]^Nuc[2]:=0;  
Hpp[i]^Po[1]:=0; Hpp[i]^Po[2]:=0;  
Hpp[i]^Lim[1,1]:=0;  
Hpp[i]^Lim[1,2]:=0;  
Hpp[i]^Lim[2,1]:=0;  
Hpp[i]^Lim[2,2]:=0;  
Hpp[i]^HipRec:=True;  
Hpp[i]^Split:=False;  
Hpp[i]^A:=0;  
Write(ArFron,Hpp[i]^);  
end;  
For j:=1 to NoNMax do  
begin  
ANuc[j]:=0;  
For i:=1 to NoFrXNu do Nu[j]^id[i]:=0;  
Nu[j]^N:=0;
```

```
end;  
Close(ArFron);  
end;
```

```
Procedure InicializaNuc;  
Var  
i,j : Word;  
begin  
For i:=1 to NoNMax do  
begin  
Are[i]:=0;  
Termino[i]:=False;  
With Nu[i]^ do  
begin  
Ni:=0;  
N:=0;  
Pc[1]:=0;  
Pc[2]:=0;  
t0:=0;  
For j:=1 to NoFrXNu do id[j]:=0;  
end;  
end;  
end;  
end;
```

```
Procedure IniciaNucleos;  
Var i,j : Word;  
Rep : LongInt;  
PAux : Array [1..2] of Word;  
Acepta : Boolean;  
R,D : Extended;
```

```

    MaximoN : Word;
begin
InicializaNuc;
NucRed:=0;
Seek(ArNucl,0);
i:=1;
Nu[i]^Ni:=i;

Nu[i]^Pc[1]:=Redondea(Random(65535)/65534,Rdo[7]);
Nu[i]^Pc[2]:=Redondea(Random(65535)/65534,Rdo[7]);
{
Nu[i]^Pc[1]:=0.5;
Nu[i]^Pc[2]:=0.5;
}
Nu[i]^t0:=0;
Write(ArNucl,Nu[i]^);
Inc(i);
Rep:=0;
MaxRep:=0;
if GnCte then MaximoN:=NoNMax else MaximoN:=Round(MaxNToDo);
While (i <= MaximoN)and(MaxRep < 100000)and Not KeyPressed do
begin
{
Nu[i]^t0:=(i-1)/GrowN;
Nu[i]^t0:=-ln((50.01-i)/(50.01-1))/(0.70);
Nu[i]^t0:=-ln((30.01-i)/(30.01-1))/(0.50);
Nu[i]^t0:=-ln((70.01-i)/(70.01-1))/(1.9);
Nu[i]^t0:=-ln((80.01-i)/(80.01-1))/(1.9);
Nu[i]^t0:=-ln((90.01-i)/(90.01-1))/(1.9);
}
}

```

```

Nu[i]^t0:=(i-1)/GrowN;
Nu[i]^t0:=-ln((MaxNToDo-i)/(MaxNToDo-1))/(KRate);

Nu[i]^Pc[1]:=Redondea(Random(65535)/65534,Rdo[7]);
Nu[i]^Pc[2]:=Redondea(Random(65535)/65534,Rdo[7]);
j:=1;
Acepta:=True;
While Acepta and (j < i) do
begin
R:=GrowR*(Nu[i]^t0-Nu[j]^t0);
D:=Distancia(Nu[j]^Pc,Nu[i]^Pc);
if D <= R+0.01 then
begin
Acepta:=False;
end else
begin
Inc(j);
end;
end;
if Acepta then
begin
Nu[i]^Ni:=i;
Write(ArNucl,Nu[i]^);
{
GoToXY(1,26);Write(i:3);
}
Inc(i);
Rep:=0;

```

```

    Inc(NucRed);
end else
begin
    Inc(Rep);
    if Rep > MaxRep then
        begin
            MaxRep:=Rep;
        {
            GoToXY(1,27);Write(MaxRep:7);
        }
        end;
    end;
end;
Close(ArNucl);
end;

```

```

Procedure IniPrueba;
Var i : Word;
begin
    NoNuc:=NucRed;    {Numero de nucleos existentes}
    InicaHiperbolas; {Inicializar fronteras    }
    ReSet(ArNucl);   {Leer Nucleos de archivo  }
    Seek(ArNucl,0);
    NoNuc:=FileSize(ArNucl)-1;
    NucRed:=NoNuc;
    For i:=1 to NoNuc do Read(ArNucl,Nu[i]^);
    Close(ArNucl);
end;

```

```

Var i,j,k,K1 : Word;
  R          : Extended;
  it         : Word;
  Cxi        : Array[1..NoNMax] of PntI;
  Basura,Entro,Cinetica : Boolean;
  Color,Col1 : Word;
  NucR       : Word;
  S,S1,S2    : String;
  Chc01,Chc02,Chc03 : Char;
  Cha01,Cha02,Cha03 : Char;
  NoRepet,Rpt,Fallo : Word;
  DeltaT     : Extended;
  Label 1,2;

Procedure Inicializacion01;
Begin
  {Separar memoria de nucleos, fronteras y auxiliares }
  {Asignación de nombres de archivos auxiliares y apertura }
  For i:=1 to NoNMax do GetMem(Nu[i],SizeOf(Nu[1]^));
  For i:=1 to NoTFro do GetMem(Hpp[i],SizeOf(Hpp[1]^));
  For i:=1 to 10 do GetMem(Hpv[i],SizeOf(Hpv[1]^));
  S:= 'Image??.BMP';
  S1:='Areas???.DAT';
  S2:='Kinti???.DAT';
  GrowR:=0.1;
  Assign(ArNucl,'Nucleos.Dat');
  Assign(NucTot,'NucTot.Dat');
  ReWrite(NucTot);
  DeltaT:=0.1;
  Rpt:=1;

```

```
Rpt:=0;  
Fallo:=0;  
LimNuc:=2;  
RetrH:=0;  
RetrC:=0;  
Corre:=0;  
Pon:=False;  
Chc01:='0';  
Chc02:='0';  
Chc03:='0';  
Cha01:='0';  
Cha02:='0';  
Cha03:='0';  
end;
```

```
begin  
Iniciar;  
Inicializacion01;  
{Variables generales de control}  
Cinetica:=True;  
CaptGrafica:=False;  
Raya:=False;  
Grafica:=True;  
Genera:=True;  
NoRepet:=2;  
GnCte:=False;  
if GnCte then GrowN:=40.0 end else  
begin  
MaxNToDo:=120.01;
```

```

KRate:=0.02;
end;
While (Rpt < NoRepet)and Not KeyPressed do
begin
AreaFin:=0;
S1[6]:=Cha01;
S1[7]:=Cha02;
S1[8]:=Cha03;
S2[6]:=Cha01;
S2[7]:=Cha02;
Assign(ArResA,S1);
GoToXY(43,10);Write('      ARCHIVO: ',S1);
if Cinetica then Assign(ArKin,S2);
Assign(AreNuc,'NucAr.Dat');
ReWrite(AreNuc);
2:
ReWrite(ArResA);
if Cinetica then ReWrite(ArKin);
if Genera then ReWrite(ArNucl);
if Genera then
begin
IniciaNucleos;
end else
begin
InicializaNuc;
end;
IniPrueba;
ReSet(ArNucl);
SetAreaTrabajo(1,1,330,1,15);
PintaAreaTrabajo(False);

```

```

it:=0;
Conos:=NucRed;
Tiempo:=0.0;
While Not KeyPressed and ((Conos > 0)or(NucR < NucRed))and
  (Abs(1-AreaFin) > 1e-6) do
begin
SetBkColor(0);
if Grafica then PintaAreaTrabajo(Pon);
AreaFin:=0;
AConF:=0;
AHipFin:=0;
CalcArea:=False;
NucR:=0;
Conos:=0;
InicaHiperbolas;
For i:=1 to NoNuc do
begin
Minimo(i);    { Ordenar Nucleos Por distancia al i }
Entro:=False;
if Radio(i) > 0 then
begin
Inc(NucR);    {Si i ya inicio el crecimiento}
For j:=1 to NoNuc do
begin
if (i < OrK[j]) then
begin
if (Nu[i]^Pc[1] = Nu[OrK[j]]^Pc[1])or(Nu[i]^Pc[2] =
  Nu[OrK[j]]^Pc[2]) then
begin
Close(ArResA);

```

```

if Cinetica then Close(ArKin);
GoTo 1;
end;
if Radio(OrK[j]) > 0 then
begin {Si Ork[j] ya inicio el crecimiento}
if XMedia(i,OrK[j]) <= (Radio(i)+Radio(OrK[j]))/2 then
begin {Si ya chocaron i y OrK[j] entonces calcula hiperbola}
HipProv(i,OrK[j]);
Entro:=True;
end;
end;
end;
end;
end;
if Not Entro and (Nu[i]^N > 1) then
begin
j:=2;
While j <= Nu[i]^N do
begin
Hpv[1]^:=Hpp[Nu[i]^id[j]]^;
SoloAsigna(i,j,Nu[i]^N);
Inc(j);
end;
end;
if Nu[i]^N > 1 then
begin
Kl:=1;
While Kl <= Nu[i]^N do
begin
Basura:=CruzaAlt(i,Nu[i]^N,Hpp[Nu[i]^id[Kl]]^,Hpv[2]^,True);

```

```

if Basura then
  begin
    ToQ:=Nu[i]^id[K1];
    Quitar(i,K1);
  end else Inc(K1);
end;
end;
if Radio(i) > 0 then
  begin
    LimEjes(i);
  end;
end;
if NoFro = 0 then Conos:=NucRed;
For i:=1 to NoNuc do
  begin
    if Radio(i) > 0 then
      begin
        OrdenaFro(i);
      end;
    end;
  end;
if Grafica then PintaAreaTrabajo(Pon);
Color:=0;
Color:=1; {Analizando }
CalcArea:=True;
For i:=1 to NucRed do
  begin
    ANuc[i]:=0;
    if Radio(i) > 0 then
      begin
        PintaNucleo(i,Color);
      end;
    end;
  end;
end;

```

```

R:=Pi*sqr(Radio(i));
if Not Termino[i] then
begin
ANuc[i]:=ACon[i]+AHp[i];
end else
begin
ANuc[i]:=Are[i];
end;
Are[i]:=ANuc[i];
AConF:=AConF+ACon[i];
AHipFin:=AHipFin+AHp[i];
AreaFin:=AConF+AHipFin;
if Color = 14 then Color:=1 else Inc(Color);
end;
Write(AreNuc,Anuc[i]:15,',');
end;

WriteLn(AreNuc,"");
GoToXY(43,2);Write('Nucleos Totales : ',NucRed:3,' Formados: ',NucR:3);
GoToXY(43,3);Write('Fronteras Libres: ',Conos:3,' No libre: ',NoFro:3);

GoToXY(43,4);Write('Tiempo transcurrido: ',tiempo:2:4);
GoToXY(43,5);Write('Area de Fronteras libres: ',AConF:1:6);
GoToXY(43,6);Write('Area de Fronteras No libres:',AHipFin:1:6);
GoToXY(43,7);Write('          Area total: ',AreaFin:1:6);
GoToXY(43,8);Write('No de repeticiones: ',Rpt:2);

WriteLn(ArResA,Tiempo:3,',',AreaFin:1:7);

S[6]:=Chc01;

```

```

S[7]:=Chc02;
if CaptGrafica then
begin
CaptGra(0,0,639,449,S);
if Chc02 < '9' then
begin
Inc(Chc02);
end else
begin
Chc02:='0';
Inc(Chc01);
end;
end;
if AreaFin > 1+1e-6 then
begin
GoTo 1;
end;
Tiempo:=Tiempo+DeltaT;
inc(it);
end;
R:=0;
k:=1;
if Cinetica then
begin
For i:=1 to NoFro do
begin
if Hpp[i]^Nuc[2] <> 0 then
begin
WriteLn(ArKin,i:2,',',Hpp[i]^Nuc[1]:2,',',Hpp[i]^Nuc[2]:2,',',Hpp[i]^V:1:6);

```

```

end;
end;
end;
Close(ArResA);
if Cinetica then Close(ArKin);
if Abs(1-AreaFin) < 1e-6 then
begin
Write(NucTot,NucRed:2,',');
Rpt:=Rpt+1;
if Cha03 < '9' then
begin
Inc(Cha03);
end else
begin
if Cha02 < '9' then
begin
Inc(Cha02);
Cha03:='0';
end else
begin
Cha03:='0';
Cha02:='0';
Inc(Cha01);
end;
end;
end else
begin
1:Inc(Fallo);
if Not KeyPressed and (AreaFin > 1) then
begin

```

```
Close(ArResA);
if Cinetica then Close(ArKin);
GoToXY(43,9);Write('Fallidos : ',Fallo:2{' ',''},AreaFin:2:7});
GoTo 2;
end;
end;
Close(AreNuc);
end;
Close(NucTot);
Temporal:=ReadKey;
Temporal:=ReadKey;
SalidaFinal;
end.
```