

UNIVERSIDAD NACIONAL DE GENERAL SAN MARTÍN
COMISIÓN NACIONAL DE ENERGÍA ATÓMICA
INSTITUTO DE TECNOLOGÍA
“Prof. Jorge A. Sabato”

Estudio y modelización de la transformación
martensítica en un acero ASTM A335 P91*

Por Florencia Antonella Correa

Director de trabajo
Ing. Jorge Ignacio Besoky

Codirector
Dr. Claudio Ariel Danón

*Trabajo de Seminario para optar al título de Ingeniero en
Materiales

República Argentina
2021

Resumen

En este trabajo se aborda el análisis de información experimental -previamente obtenida por medio de la técnica de dilatometría- relativa a dos aspectos de la transformación martensítica, a saber, la temperatura de inicio de la transformación y el curso de la transformación como una función de la temperatura, durante un proceso de enfriamiento continuo después de un mantenimiento en austenita.

El material en estudio es un acero ASTM A335 grado P91 (9Cr1MoVNbN) caracterizado por una alta resistencia a la corrosión y a la termofluencia, propiedades físicas que le permiten un muy buen desempeño durante prestaciones a temperaturas cerca de, o más altas que, 600 °C. La combinación de estas propiedades con una alta conductividad térmica, un bajo coeficiente de expansión térmica y una excelente resistencia al hinchado bajo irradiación hacen de este acero un candidato de referencia para aplicaciones estructurales en reactores nucleares de Generación IV.

Durante el recorrido previo del grupo de investigación que propuso este trabajo se estudiaron aspectos de la metalurgia física de este tipo de aleaciones en ciclos térmicos diseñados bajo diferentes condiciones de calentamiento, mantenimiento a alta temperatura y enfriamiento.

Se realizó entonces una búsqueda bibliográfica acotada sobre los dos aspectos seleccionados y se desarrolló una serie de rutinas computacionales que permitieron implementar el análisis de los datos experimentales y su ajuste con modelos metalúrgicos ya informados en la literatura existente, usando el lenguaje de programación Python.

Los resultados obtenidos constituyen un primer paso de aporte a la comparación sistemática de métodos para determinar la temperatura de inicio de la transformación martensítica y corroboran la superioridad de los modelos de tipo sigmoidal por sobre los exponenciales para describir el curso de la transformación en aceros 9% cromo.

Abstract

This work addresses the analysis of experimental information previously obtained by means of the dilatometry technique and relative to two aspects of the martensitic transformation, namely, the transformation start temperature and the course of the transformation as a function of temperature during a continuous cooling process after high-temperature austenite holding.

The material under study is an ASTM A335 grade P91 (9Cr1MoNbVN) steel, characterized by a high resistance to corrosion and creep, these properties allowing a good performance during service at temperatures close to, or even higher than, 600 °C. The combination of these properties with a high thermal conductivity, a low thermal expansion coefficient and an excellent resistance to void swelling under irradiation makes this steel a reference candidate for structural applications in Generation IV nuclear reactors.

In previous studies carried out by the research group that proposed this work, aspects of the physical metallurgy of this type of steel have been investigated in thermal cycles designed under different conditions of heating, holding at high temperature and cooling.

A delimited bibliographic search over the two selected aspects was carried out and a series of computational scripts was developed that allowed to implement the analysis of experimental data as well as data adjustment with metallurgical models already reported in the previous literature, using the Python programming language.

The obtained results constitute a first step of contribution to the systematic comparison of methods to determine the martensite start temperature, and corroborate the superior performance of the sigmoidal models over the exponential ones in describing the course of the martensitic transformation in 9 % Cr steels.

Índice de contenido

1	Revisión de conceptos básicos sobre aceros	11
1.1	Introducción teórica	11
1.1.1	Definiciones	11
1.1.2	Variedades alotrópicas del hierro [1]	11
1.1.3	Diagrama Fe-C	11
1.1.4	Diagrama Fe-Cr-C	12
1.1.5	Transformación martensítica [1][4]	13
1.1.6	Tratamientos térmicos en aceros	15
2	Aceros Ferrítico/Martensíticos para uso nuclear	19
2.1	Generación IV	19
2.2	Selección de materiales	19
2.3	Aceros de alto contenido en Cromo	20
2.4	Aceros ferrítico-martensíticos (F/M)	21
2.4.1	Segundas fases	21
2.4.2	Diagrama CCT	22
3	Dilatometría. Fundamentos de la técnica experimental y aplicaciones en transformaciones martensíticas	25
3.1	Dilatometría	25
3.1.1	Introducción	25
3.1.2	Generalidades sobre deformación térmica, defectos y transformaciones de fases [20]	25
3.1.3	Metodología para el análisis dilatométrico de las transformaciones de fases[20]	26
3.1.4	Equipo experimental [21]	26
3.2	Análisis de las curvas dilatométricas	28
3.3	Métodos para la determinación de la temperatura de inicio de transformación martensítica	31
3.3.1	Modificación de la ecuación empírica para la determinación de M_S en aceros con alto contenido de Cr	31
3.3.2	Métodos para la determinación de la M_S a partir de curvas dilatométricas	32
4	Análisis de la evolución de la fracción transformada a martensita	35
4.1	Aspectos experimentales	35
4.2	Modelado	35
4.2.1	Modelo de Koistinen y Marburger (K-M) [30]	35
4.2.2	Modelo de van Bohemen y Sietsma [32]	36
4.2.3	Modelo de van Bohemen [33]	37
4.2.4	Modelo de Lee y van Tyne[34]	37
4.2.5	Modelo de Huyan y colaboradores [36]	38
4.3	Método CALPHAD y software Thermocalc	39

5	Procesamiento y ajuste de los datos experimentales	41
5.1	Tratamientos térmicos aplicados	41
5.2	Aplicación de métodos de determinación de M_S	41
5.2.1	Método de offset: Etapa de Definición	42
5.2.2	Método de offset: Etapa de Evaluación	42
5.3	Análisis de la dispersión en la aplicación de ajustes lineales	43
5.3.1	Localización del ajuste	44
5.3.2	Segmento excluido	44
5.4	Aplicación de modelos de ajuste para la fracción transformada a martensita	45
5.4.1	Parámetros libres ajustados	46
6	Resultados	49
6.1	Ciclos térmicos	49
6.2	Curvas dilatométricas	50
6.3	Determinación de M_S	53
6.3.1	Método offset	53
6.3.2	Método ASTM	55
6.3.3	Comparación entre métodos experimentales de determinación de M_S	56
6.4	Modelos aplicados a la determinación de fracción martensítica transformada	58
6.4.1	Ajustes de parámetros en los modelos de fracción transformada uti- lizando la M_S obtenida a partir de ensayos dilatométricos.	58
6.4.2	Ajustes de parámetros en los modelos de fracción transformada uti- lizando la composición química del acero	63
6.5	Comparación entre determinaciones experimentales y ecuaciones empíricas	66
7	Conclusiones	67
8	Anexo	69
8.1	Macros de Thermocalc	69
8.2	Determinación de M_S por método ASTM	70
8.3	Determinación de M_S por método offset	78
8.4	Determinación de M_S por todos los métodos	85
8.5	Aplicación de ajustes de modelos de Koistinen-Marburger, Lee-Van Tyne y Huyan	93
8.6	Curvas de dilatometría y tratamientos térmicos	115
8.7	Funciones	121
9	Referencias	139

Índice de tablas

5.1	Tratamientos térmicos analizados	41
6.1	Resultados para la M_S ($^{\circ}\text{C}$) según localización de intervalos utilizando el método de offset.	53
6.2	Resultados de la M_S ($^{\circ}\text{C}$) promedio para las muestras #541, 584, 586, 590 según los distintos segmentos considerados para el ajuste con el método offset.	55
6.3	Resultados para M_S ($^{\circ}\text{C}$) según los distintos intervalos de localización y utilizando el método ASTM.	55
6.4	Resultados de la M_S ($^{\circ}\text{C}$) promedio para las muestras #541, 584, 586, 590 según los distintos segmentos considerados para el ajuste con el método ASTM.	56
6.5	Resultados de M_S ($^{\circ}\text{C}$) determinada para todas las muestras por los distintos métodos de análisis.	57
6.6	Promedio y dispersión de M_S determinada por distintos métodos de análisis, para todas las muestras sometidas al tratamiento térmico 1.	57
6.7	Resumen de los parámetros determinados para la media de fracción obtenida para las muestras con tratamiento térmico 1.	60
6.8	Resumen de los parámetros determinados para el ajuste de la curva de fracción transformada obtenida para la muestra #587.	61
6.9	Resumen de parámetros calculados	63
6.10	Resumen de los parámetros calculados durante el ajuste de la fracción transformada a martensita para los modelos de van Bohemen y Sietsma, van Bohemen y Lee-van Tyne.	64
6.11	Resultados de la M_S ($^{\circ}\text{C}$) obtenida de forma experimental y mediante la aplicación de ecuaciones empíricas.	66

Índice de figuras

1.1	Diagrama Fe-C metaestable [2]	12
1.2	Diagrama de fase: detalle del diagrama de fases Fe-Cr	13
1.3	Diagrama de fase: detalle del diagrama de fases del sistema Fe-Cr-0,1 % C; la notación (CrFe) ₄ C refiere al carburo M ₂₃ C ₆ [3].	14
1.4	Tratamiento térmico de Q&P	17
2.1	Sumario de materiales considerados en reactores de Generación IV.[11]	20
2.2	Evolución temporal de los aceros ferrítico-martensíticos.[12]	21
2.3	Composición del acero ASTM A335 P91	22
2.4	Diagrama CCT del acero ASTM A335 grado P91 [18]. “A” es austenita, “C” carburos, “F” ferrita y “M” martensita.	23
3.1	Dilatómetro Bähr modelo DIL 805 A, que proveyó los resultados analizados en este trabajo.	27
3.2	Esquema de componentes principales de un dilatómetro.	27
3.3	Esquema de variación de longitud en una probeta isotrópica	28
3.4	Dependencia del CTE con la temperatura para las fases ferrita y austenita en aceros [23]	29
3.5	Deformación durante una transformación isotérmica.	30
3.6	Deformación durante una transformación en enfriamiento continuo	31
3.7	Determinación de M _S según método offset	33
3.8	Determinación de M _S según método de las tangentes	33
5.1	Configuraciones de fases del sistema y análisis por el método de offset	43
5.2	Curva dilatométrica con puntos representativos para el análisis	44
5.3	Dependencia de la fuerza impulsora de la transformación austenita a martensita con la temperatura para un sistema simplificado del acero ASTM A335 grado P91.	47
6.1	Ciclo térmico 1, muestras 541, 584 586 y 590.	49
6.2	Ciclo térmico 2, muestra 587.	49
6.3	Curva dilatométrica de la muestra #541	51
6.5	Curva dilatométrica de la muestra #586	51
6.4	Curva dilatométrica de la muestra #584	52
6.6	Curva dilatométrica de la muestra #587	52
6.7	Curva dilatométrica de la muestra #590	53
6.8	Determinación de la M _S para todas las muestras por el método de offset	54
6.9	Promedio y dispersión de M _S determinada por el método offset, para todas las muestras sometidos al tratamiento térmico 1 según la localización del ajuste	54
6.10	M _S determinada para todas las muestras por el método ASTM.	55
6.11	Promedio y dispersión de M _S determinada por el método ASTM, para todas las muestras sometidas al tratamiento térmico 1 según la localización del ajuste.	56
6.12	M _S (°C) determinada por los distintos métodos de análisis utilizados para todas las muestras.	57

6.13 Promedio y dispersión de M_S determinada por distintos métodos de análisis, para todas las muestras sometidas al tratamiento térmico 1.	58
6.14 Modelos de ajuste a la curva de fracción transformada para M_S definida por el método de las rectas tangentes.	59
6.15 Modelos de ajuste a la curva de fracción transformada para M_S definida por el método offset.	60
6.16 Modelos de ajuste a la curva de fracción transformada para M_S definida por el método ASTM	61
6.17 Modelos de ajuste de la curva de fracción transformada, para M_S definida por el método de las rectas tangentes y para el tratamiento térmico 2.	62
6.18 Modelos de ajuste de la curva de fracción transformada, para M_S definida por el método de offset y para el tratamiento térmico 2.	62
6.19 Modelos de ajuste de la curva de fracción transformada, para M_S definida por el método ASTM y para el tratamiento térmico 2.	63
6.20 Modelos de ajuste aplicados a la curva de fracción transformada para el tratamiento térmico 1 utilizando el valor de M_S calculado por Finkler y Schirra	64
6.21 Modelos de van Bohemen y van Bohemen - Sietsma aplicados a la curva experimental de fracción transformada obtenida por la regla de la palanca para el tratamiento térmico 1.	65
6.22 Modelo de Lee y van Tyne aplicados a la curva experimental de fracción transformada obtenida por la regla de la palanca para el tratamiento térmico 1.	66

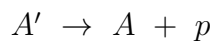
1 Revisión de conceptos básicos sobre aceros

1.1. Introducción teórica

En este capítulo se presentan algunos conceptos de uso frecuente a lo largo de este trabajo.

1.1.1. Definiciones

- Fase: es una porción físicamente distinguible de un sistema, con propiedades físicas y composición homogéneas.
- Solución sólida: cuando la adición de átomos de un dado elemento a una fase no altera su estructura cristalina se dice que el elemento en cuestión se encuentra en solución sólida. Esto puede lograrse de dos formas: el átomo puede reemplazar un átomo de la red cristalina (solución sólida sustitucional) o ubicarse en intersticios de la red (solución sólida intersticial).
- Precipitación: transformación a través de la cual un sistema en fase A' sobresaturada reduce su energía libre a partir de la formación de una segunda fase p .



1.1.2. Variedades alotrópicas del hierro [1]

El hierro puro presenta, a presiones menores que 100 kbar, tres variedades alotrópicas. La ferrita o α -Fe (con estructura cristalina BCC) es la fase estable dentro de un amplio rango de temperaturas que se extiende desde temperaturas bajo cero hasta los 910°C. A mayores temperaturas, la fase estable pasa a ser la austenita o γ -Fe (estructura FCC). La estructura FCC posee mayor compacidad que la BCC, por lo que la transformación $\alpha \rightarrow \gamma$ es acompañada por una contracción volumétrica que se encuentra en el orden del 1 %. Por encima de los 1390°C y hasta alcanzarse la temperatura de fusión $T_f = 1536^\circ\text{C}$ la fase estable, δ -Fe, vuelve a tener la estructura BCC.

1.1.3. Diagrama Fe-C

Debido a constituir una base de análisis para un muy extendido conjunto de aplicaciones industriales, uno de los diagramas de equilibrio más ampliamente utilizados en el campo de la metalurgia física ha sido el Hierro – Carbono (Fe-C).

En el diagrama Fe-C estable aparece la fase grafito. Sin embargo, la cinética de nucleación y crecimiento de esta fase suele ser muy lenta en ausencia de determinados aleantes y consecuentemente, en lugar de formarse dicha fase, se observa una fase metaestable conocida como cementita (Fe_3C). Por este motivo en la práctica se suele recurrir al diagrama Fe-C metaestable, o diagrama Fe- Fe_3C , [Figura 1.1](#).

El diagrama Fe-Fe₃C presenta cuatro campos monofásicos: uno ferrítico, que admite una baja concentración de C en solución sólida con una solubilidad máxima de 0,022 %C a 723°C, uno austenítico-con una solubilidad de C 100 veces mayor (concentración máxima de C: 2,14 %C a 1147°C)-, uno ferrítico de alta temperatura (ferrita δ) y finalmente uno correspondiente a la fase cementita (Fe₃C).

Es importante señalar en la [Figura 1.1](#) algunas temperaturas características, a saber, A1, A2, A3 y A4. A1 es la temperatura a la que ocurre una transformación conocida como eutectoide. A3, por su parte, marca el límite superior del campo bifásico: por encima de esta temperatura la fase austenítica pasa a ser la única fase estable. A4 indica la transformación de austenita a ferrita de alta temperatura. Por último, A2 es la temperatura a la cual la ferrita cambia sus propiedades magnéticas, o temperatura de Curie (769°C en hierro puro).

El estudio de estas temperaturas se logra a partir del análisis de las variaciones que experimenta alguna propiedad del material sensible a las transformaciones de fase -por ejemplo, resistividad, volumen específico-; entre otros factores, su valor dependerá de si se está enfriando o calentando el material durante la transformación. Por este motivo se recurre a la notación AX_c , AX_r y AX_e para indicar si la temperatura fue medida en calentamiento, enfriamiento o corresponde al equilibrio, respectivamente.

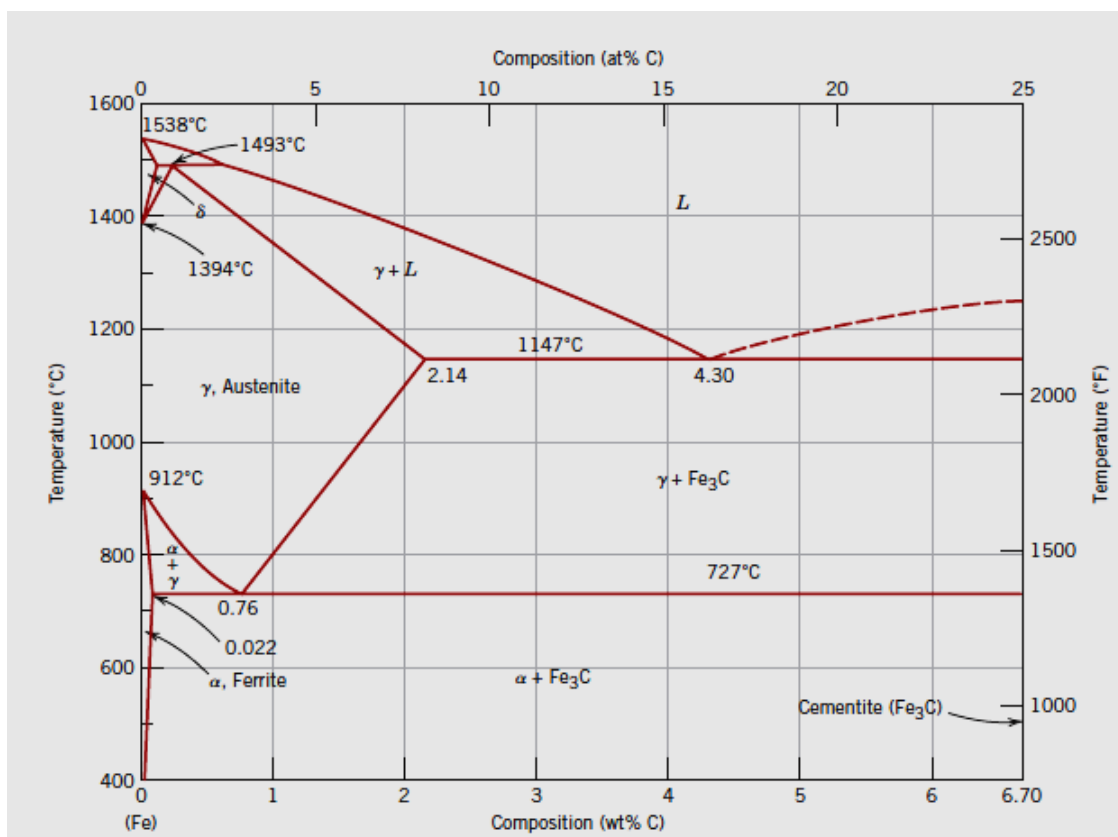


Figura 1.1. Diagrama Fe-C metaestable [2]

1.1.4. Diagrama Fe-Cr-C

La adición de aleantes al sistema puede alterar radicalmente la forma del diagrama Fe-C, pudiendo modificar tanto las temperaturas de transformación como las fases presentes. Va de suyo que, además de la configuración básica del sistema aleado, cuando se estudia un acero se debe tener presente al resto de los elementos minoritarios y considerar el comportamiento de las fases principales en cuanto a estabilidad frente al agregado de

cada uno de ellos. Así, los elementos aleantes pueden clasificarse en:

1. estabilizadores de la austenita (gamágenos): C, N, Ni, Mn, Cu, Co, los cuales extienden el campo γ .
2. estabilizadores de la ferrita (alfágenos): Cr, Mo, Nb, V, W, Si, Ti, Al los cuales extienden el campo α .

El Cromo (Cr), un fuerte alfégeno, forma una solución sólida con el Fe en las fases BCC (α -Fe y δ -Fe) restringiendo el campo austenítico. La adición de Cr al sistema Fe puro (Figura 1.2)-produce una disminución de la temperatura A4 y un aumento de A3. Para concentraciones de Cr mayores a 12,7% desaparece el campo austenítico y se obtiene un acero con una fase ferrítica estable en un amplio rango de temperaturas.

La adición de pequeñas cantidades de C al sistema Fe-Cr modifica sensiblemente el diagrama de fases, Figura 1.3. El C, un aleante gamágeno, aumenta el rango de temperaturas en el que la austenita es estable -esto es, aumenta A4 y disminuye A3- y extiende el rango bifásico ($\alpha - \gamma$) a mayores concentraciones de Cr.

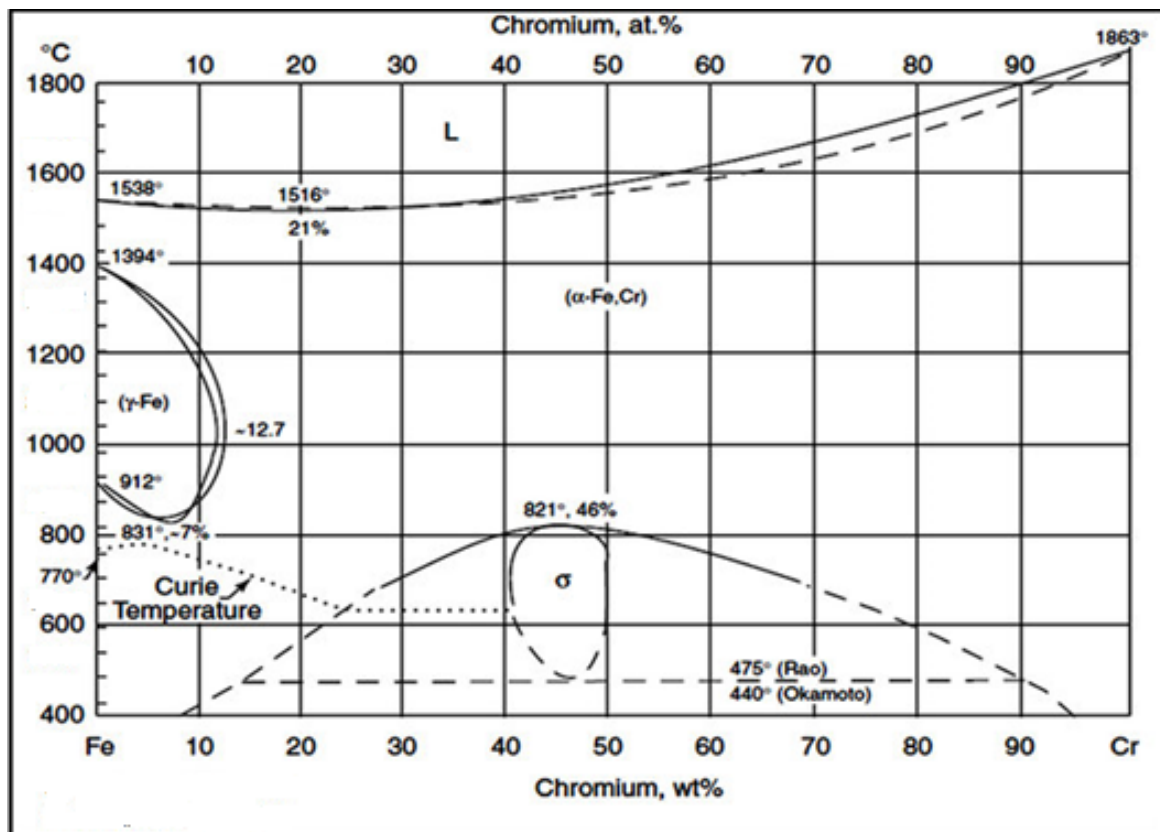


Figura 1.2. Diagrama de fase: detalle del diagrama de fases Fe-Cr

1.1.5. Transformación martensítica [1][4]

La transformación martensítica es una transformación de fase fuera del equilibrio que se produce en ausencia de difusión y que tiene lugar cuando el material es austenizado y luego enfriado en forma continua con una velocidad de enfriamiento lo suficientemente rápida como para que la austenita no pueda experimentar una transformación asistida por difusión. La fase producto (martensita) es metaestable, posee estructura cristalina BCT y puede presentar una microestructura en paquetes de listones o de discos, que

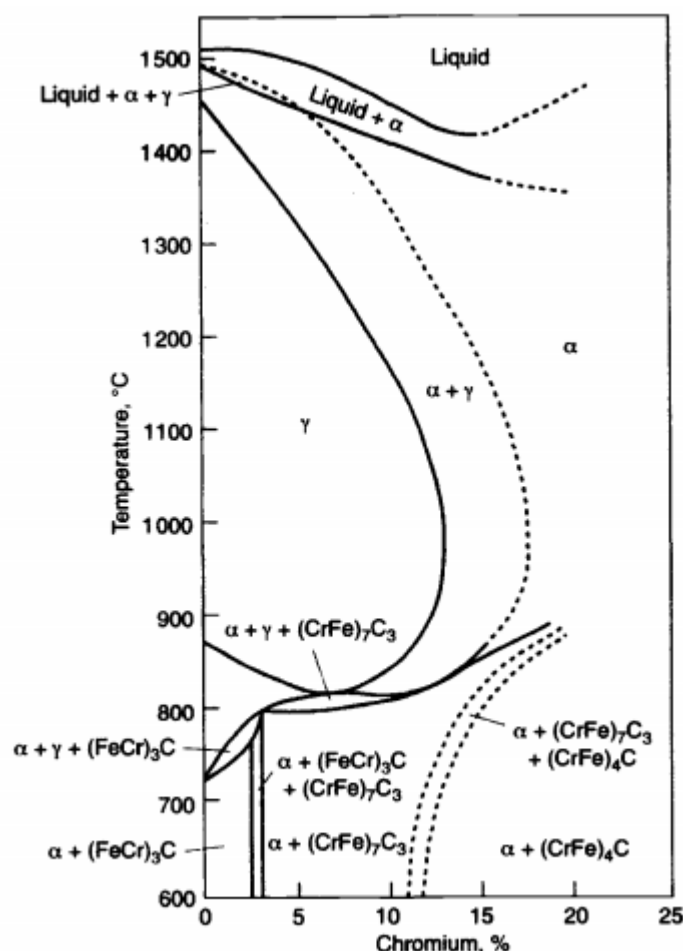


Figura 1.3. Diagrama de fase: detalle del diagrama de fases del sistema Fe-Cr-0,1 % C; la notación $(\text{CrFe})_4\text{C}$ refiere al carburo M_{23}C_6 [3].

mantienen -en ambos casos- cierta relación de orientación con la fase madre (austenita) y que presentan una elevada cantidad de deformación.

Durante una transformación de esta naturaleza los átomos de la red se desplazan, con un movimiento cooperativo, distancias menores al espaciado interatómico, manteniendo así la composición de la fase madre (en particular, la fracción de C en solución sólida) y la disposición de los átomos vecinos. Asimismo, al no haber difusión, los listones de martensita no atraviesan los ex-bordes de grano austeníticos, de modo que la longitud máxima de los listones estará de alguna manera relacionada con el tamaño de grano de la fase madre.

La martensita es una fase dura y frágil, y de gran importancia a nivel industrial. Así, la transformación martensítica ha sido estudiada en una gran variedad de familias de aceros, entre las que pueden mencionarse ciertos aceros inoxidables, aceros templados y revenidos, aceros para rodamientos y, más recientemente, aceros maraging (martensita endurecida por precipitación), aceros TRIP (transformación inducida por deformación plástica), aceros tratados termomecánicamente (austenita deformada plásticamente antes del temple) y aceros dual phase (mezclas de ferrita + martensita obtenidas templando desde el campo $\alpha + \gamma$).

Temperatura y curso de la transformación

La transformación martensítica es de primer orden, es decir, involucra la nucleación y el crecimiento de cúmulos de la nueva fase. La fase madre comienza a transformarse a martensita cuando alcanza una temperatura determinada M_S durante un proceso de enfriamiento continuo, y finaliza al alcanzarse la temperatura de finalización M_F . En la mayoría de los casos -y en el caso particular del acero estudiado en este trabajo-, la transformación no es completa y se observa una determinada fracción de austenita retenida entre los listones martensíticos. A lo largo de este trabajo se denominará M_{F*} a la temperatura para la cual cesa la transformación, independientemente del porcentaje transformado.

Se considera que la transformación martensítica es *atérmica* porque la fracción transformada depende del subenfriamiento bajo la temperatura de inicio de transformación (M_S). Es decir, cuanto mayor es la diferencia entre M_S y T (temperatura menor que M_S o temperatura de subenfriamiento) mayor es la cantidad de martensita transformada. El carácter atérmico se define de manera experimental, esto es, se observa que la transformación ocurre en una escala de tiempo mucho menor que la asequible a través de las técnicas de seguimiento de mayor resolución. Por otra parte, se conoce como efecto *autocatalítico* a aquel en el que la formación de una placa inicial de martensita estimula la formación de otras y la tensión generada por la primera da la orientación de estas últimas.

El rango de temperaturas en el que ocurre la transformación es de particular importancia puesto que posee una fuerte influencia tanto en el tipo de martensita obtenido como en la cantidad de austenita retenida. En general, a valores altos de M_S se obtiene una martensita en paquetes de listones internamente dislocados, mientras que para valores menores de M_S la martensita adopta una forma más caótica en discos internamente maclados. Por otro lado, cuanto menor es la temperatura a la que se produce la transformación, mayor es la fracción de austenita retenida.

Influencia de la composición sobre la temperatura M_S

El valor de M_S mantiene una fuerte dependencia de la composición de la fase madre. Se observa en general que un aumento en el contenido de aleantes reduce el valor de M_S . Esto se ejemplifica en la ecuación 1.1, donde se muestra una relación empírica -ecuación de Andrews- que permite estimar la temperatura M_S en función de la composición de la fase madre para aceros de bajo contenido de C y baja aleación [5].

$$M_S = 539 - 423.w_C - 30,4.w_{Mn} - 12,1.w_{Ni} - 17,7.w_{Ni} - 7,5.w_{Si} + 10.w_{Co} \quad (1.1)$$

donde w_i es la fracción en peso de cada aleante y la M_S es determinada en °C.

1.1.6. Tratamientos térmicos en aceros

Un tratamiento térmico es un programa temperatura vs. tiempo al que es sometido un dado material con el fin de modificar sus propiedades, ya sea con el objeto de prepararlo para otras etapas del ciclo de manufactura (otros tratamientos térmicos, etapas de mecanizado, etapas de conformado, etc.) o para su entrada final en servicio.

A continuación se describen en forma general algunos de los tratamientos térmicos más empleados en la manufactura de piezas de acero.

- Austenizado: Un austenizado consiste en calentar el acero por encima de la temperatura A_3 y mantenerlo durante un tiempo lo suficientemente largo como para

lograr la transformación total de la ferrita a austenita, la disolución de segundas fases (total o parcial) y la homogeneización de la matriz. Las características de la austenita formada en este paso (uniformidad de la composición química, morfología y tamaño de grano; tamaño, composición y distribución espacial de segundas fases si las hubiera, etc.) se pueden diseñar a través del control de la microestructura inicial, la velocidad de calentamiento, la temperatura y tiempo de mantenimiento y las tecnologías de procesamiento termomecánico. El austenizado prepara el material para tratamientos térmicos subsiguientes.

- Normalizado: Es un tratamiento recurrente en la industria, que consiste en austenizar el acero y luego enfriarlo al aire. Si bien el normalizado puede ser realizado con distintos fines, en general se busca lograr una homogeneización química y estructural del material.
- Temple: Un temple consiste en un austenizado del material seguido por un enfriamiento realizado con una velocidad lo suficientemente rápida como para dar lugar a una transformación martensítica de la matriz.
- Revenido: Dado que la martensita es una fase muy frágil, los aceros sometidos a temple deben ser tratados térmicamente antes de entrar en servicio. Un revenido es un tratamiento térmico cuyo fin es aumentar la ductilidad y la tenacidad de un acero templado sin reducir significativamente su dureza y su resistencia mecánica. Consiste en calentar el material a una temperatura subcrítica ($T < A_1$) y mantenerlo a esa temperatura durante aproximadamente 1h (el tiempo exacto de mantenimiento dependerá de la aplicación, y puede extenderse significativamente).

Tratamiento de temple y partición (Q&P: quenching and partitioning) [6]

Con el objetivo de incrementar la ductilidad y resistencia en los aceros, en los últimos años se han diseñado variaciones de tratamientos térmicos convencionales con ese fin; un ejemplo de ello es el denominado "Quenching and partitioning". El proceso original de Q&P puede describirse en 4 etapas, la primera de las cuales es equivalente a la descripta como austenizado en el parágrafo anterior; estas etapas han sufrido modificaciones a lo largo de años de desarrollo a fin de responder a los requerimientos necesarios:

2. La etapa de temple, durante la cual el acero es templado desde el estado total o parcialmente austenítico a una temperatura entre M_S y M_F (temperatura final de transformación martensítica). El objetivo de esta parte del tratamiento es conseguir una fina y espacialmente homogénea mezcla de martensita-austenita o mezcla de martensita-austenita-ferrita.
3. La etapa de partición, durante la cual el acero se mantiene a la temperatura de temple (para Q&P de un paso) o a una temperatura más alta que M_S (para Q&P de dos pasos) para facilitar la partición del carbono de la martensita hacia la austenita o dar a lugar a otras transformaciones de fase.
4. La etapa de enfriamiento final, durante el cual se controla la formación de martensita fresca.

En la [Figura 1.4](#) se representa el tratamiento térmico; los símbolos representan las siguientes fases, α' : martensita, α'' : martensita revenida, α_B : ferrita de bainita, γ : austenita, γ' : austenita enriquecida en carbono, η o ϵ : carburos de transición y θ : cementita.

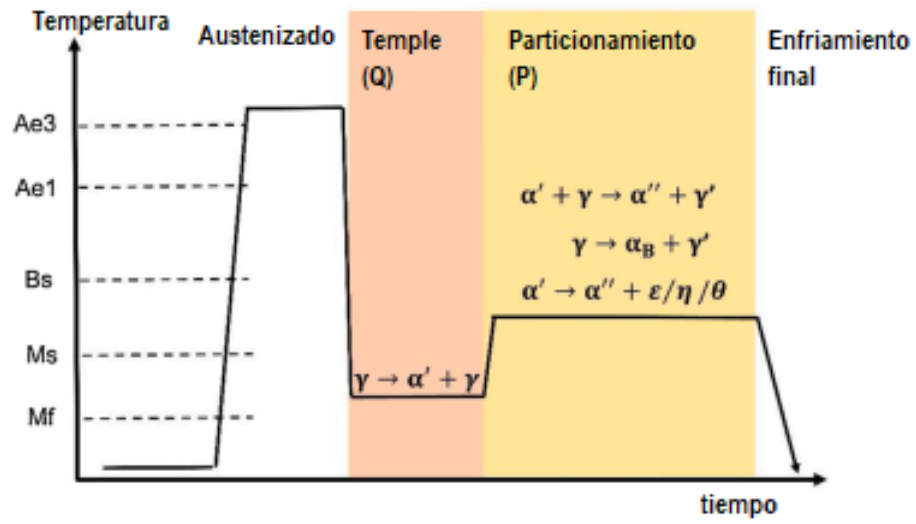


Figura 1.4. Tratamiento térmico de Q&P

2 Aceros Ferrítico/Martensíticos para uso nuclear

2.1. Generación IV

En el año 2000 la población mundial rondaba los 6.000 millones de habitantes, al día de la fecha excede los 7.600 millones y se espera que en el término de 35 años la cifra ascienda a 10 mil millones. Semejante incremento en la población mundial traerá consigo un aumento comparable en el consumo de energía y es menester de la ciencia y la ingeniería el proveer a la humanidad de los medios necesarios para satisfacer dicha demanda con energía limpia, abundante y económica a fin de poder mantener y mejorar los estándares de calidad de vida.

Por otro lado, esta explosión demográfica también hace patente el carácter limitado del espacio y los recursos de los que dispone la humanidad y por lo tanto es necesario considerar la problemática de la eficiencia y el impacto ambiental. Los nuevos medios de producción de energía deben asegurar una mayor optimización de recursos y reducir al mismo tiempo los efectos adversos sobre el medio ambiente.

En este marco y con el fin de asegurar un rol para la energía nuclear en el siglo XXI se creó en el año 2000 el Foro Internacional de Generación IV o GIF -por sus siglas en inglés-. El GIF, al que adhirieron inicialmente 9 países (Argentina, Canadá, Francia, Japón, Corea del Sur, Sudáfrica, Reino Unido y Estados Unidos) y al que luego se unieron Suiza (2002), el consorcio Euratom (2003), China (2006) y Rusia (2006), definió la ruta a seguir para el diseño e implementación de reactores nucleares que puedan satisfacer las necesidades de la vida moderna. El GIF planteó los objetivos que debían cumplir los reactores de Generación IV [7] sobre los pilares de la sustentabilidad, competitividad económica, seguridad y control de proliferación; en base a estos criterios se seleccionaron 6 diseños de reactores que satisfacen de distinta forma y en distinto grado dichos objetivos [8].

2.2. Selección de materiales

Las altas temperaturas y el elevado flujo neutrónico traen como consecuencia que materiales extensamente empleados en reactores convencionales -LWR y PWR- y sobre los que hay abundante experiencia en servicio -tales como las aleaciones de base Zr [9] – deban ser reemplazados. Es por este motivo que la cuestión de la selección de materiales para este tipo de reactores ha sido tratada recurrentemente en la literatura [7][10]. Se enumeran a continuación características generales que deben cumplir los materiales a utilizar en la construcción de reactores de Generación IV:

- Estabilidad dimensional: resistencia a la termofluencia (creep), al creep asistido por radiación, al hinchado por radiación neutrónica.
- Resistencia a mecanismos de degradación: corrosión seca, corrosión electroquímica.
- Propiedades mecánicas adecuadas, particularmente a alta temperatura.

- Resistencia al daño por radiación.
- Otras características: mantenimiento, costo de fabricación.

En la [Figura 2.1](#) se muestra un resumen de algunos de los materiales considerados para distintas aplicaciones en reactores de Generación IV.

Diseño	SFR	GFR	LFR	VHTR	SCWR	MSR
Refrigerante	Na fundido	He	Pb o Pb/Bi	He	Agua	Sales fundidas
P(MPa)	0,1	7	0,1	7	24	0,1
T(°C)	352-552	482-852	552-802	602-1002	282-552	502-722
Materiales	Aceros F/M	SiC ₂ /SiC	Aceros F/M	Grafito C/C	Aceros F/M	Grafito
estructurales (Núcleo)	F/M ODS	Compuestos	F/M ODS	SiC ₂ /SiC	Aceros base Ni	Hastelloy
T(°C)	392-702	602-1202	352-482	602-1602	352-622	702-802

Figura 2.1. Sumario de materiales considerados en reactores de Generación IV.[11]

2.3. Aceros de alto contenido en Cromo

La aplicabilidad de los materiales metálicos a problemas de utilización bajo condiciones de alta temperatura y sollicitación mecánica tuvo un extraordinario impulso con la aparición de los aceros con contenidos de Cr entre 7 y 18 % en peso. Estas aleaciones fueron desarrolladas durante la primera mitad del siglo pasado y con su irrupción se logró mejorar drásticamente propiedades tales como la resistencia a la oxidación y la resistencia al creep. Por ello, han sido empleadas extendidamente en calderas y turbinas para la industria de generación de energía.

Dentro de este grupo de aleaciones, los aceros martensítico-ferríticos 9%Cr-1%Mo han demostrado un excelente desempeño en prestaciones convencionales de alta temperatura. Entre sus propiedades se destacan, además de las ya mencionadas -elevadas tensiones de ruptura por creep y resistencia a la oxidación- la tenacidad y una adecuada estabilidad microestructural. Durante la segunda mitad del siglo pasado, estos aceros comenzaron a considerarse también para su eventual uso en plantas de energía nuclear de fisión. En el caso de posibles aplicaciones estructurales en los reactores nucleares de fisión de Generación IV, la utilización estará condicionada no solo por los exigentes requerimientos usuales -esto es, altas presiones y temperaturas de trabajo (20 a 25 MPa y 400 a 550 °C)- sino también por la presencia de campos de radiación neutrónica directa. Del mismo modo, los aceros 7-12%Cr fueron identificados como materiales aptos para la fabricación de componentes estructurales en sistemas de reactores de fusión. En el caso de esta última aplicación se desarrolló el concepto de *activación reducida*: los aceros diseñados a este fin contienen W, V, Mn, Ta y Ti como elementos aleantes, y se reduce en ellos el tenor de elementos como Mo, Nb y Ni, no deseados en términos del decaimiento radiactivo durante la disposición final. Así, las aplicaciones previstas en reactores nucleares tanto de fisión como de fusión requieren resistencia al daño por radiación (inducido por el bombardeo de neutrones altamente energéticos), pero además el acero debe retener sus propiedades mecánicas a altas temperaturas durante todo el período de servicio.

En términos de su microestructura, los aceros con alto contenido en Cr son completamente austeníticos o tienen una estructura dúplex (austenita y ferrita) a temperaturas en el rango entre 850 y 1200 °C. A temperatura ambiente, en cambio, son martensíticos, es decir, la austenita se transforma en martensita durante el enfriamiento al aire o el

enfriamiento rápido; luego, mediante la aplicación de tratamientos térmicos específicos, puede obtenerse una buena combinación de resistencia mecánica, ductilidad y tenacidad.

2.4. Aceros ferrítico-martensíticos (F/M)

Bajo el rótulo aceros ferrítico-martensíticos nos referimos a una serie de aceros de alta temperatura que fueron diseñados para ser utilizados en plantas térmicas. En la [Figura 2.2](#) se muestran distintos aceros de este tipo y su evolución a lo largo de la segunda mitad del siglo XX.

Generación	Años	Detalles	Tensión de ruptura por creep, 10 ⁵ h a 600°C (MPa)	Aceros	T max (°C)
0	1940-1960		40	T22, T9	520- 538
1	1960-1970	Adición de Mo, Nb, V a aceros Cr-Mo.	60	EM12, HCM9M, HT9, HT91	565
2	1970-1985	Optimización de C, Nb, V, N.	100	HCM12, T/P91, HCM2S	593
3	1985-1995	Sustitución del W por Mo, adición de Cu y B.	140	NF616, E911, HCM12A	620
4	Futuro	Incremento de W y adición de Co.	180	NF12, SAVE12	650

Figura 2.2. Evolución temporal de los aceros ferrítico-martensíticos.[12]

La necesidad de alcanzar temperaturas operativas mayores -con el objeto de aumentar la eficiencia térmica- llevó a la constante modificación de la composición de estos aceros a fin de mejorar las propiedades frente a los mecanismos de degradación limitantes: la fluencia lenta y la corrosión.

Esto se logró mediante la adición de aleantes que formasen segundas fases finamente distribuidas en la matriz. Su función es anclar, dependiendo de su tamaño, dislocaciones y/o interfases -bordes de listones, bordes de bloque o paquete, ex-bordes de grano-, aumentando así la resistencia a la fluencia lenta del material.

Sumado a esto, la estructura cristalina de la matriz de estos aceros (BCC) en combinación con una microestructura templada y revenida, ofrece una excelente resistencia a distintos mecanismos de daño por radiación y en particular al hinchado por radiación -o void swelling.

Estos aceros -y en particular los aceros 9Cr1Mo modificados [10][13]- se convirtieron en los principales candidatos para distintas aplicaciones en reactores de Generación IV (por ejemplo, SFR, LFR y SCWR) e incluso en reactores de fusión y también para otras aplicaciones estructurales en zonas de elevado flujo neutrónico, pero de temperaturas no superiores a 650°C.

El acero ASTM A335 grado P91 es un acero 9Cr1Mo modificado, esto es, aleado con Nb, V y N (entre otros). A diferencia de la mayoría de los aceros F/M fue diseñado para su implementación en el reactor rápido reproductor de Oak Ridge (Oak Ridge Fast Breeder Reactor) [14]. La composición del acero se detalla en la [Figura 2.3](#), los valores indicados corresponden a por ciento en peso.

2.4.1. Segundas fases

Según la composición del acero y los tratamientos térmicos a los que son sometidos, los aceros F/M pueden presentar distintos tipos de segundas fases [15]. En particular,

Elemento	Porcentaje	Elemento	Porcentaje
C	0.10	Nb	0.075
Cu	0.16	Ca	0.0008
Al	0.009	P	0.014
N	0.0605	Cr	8.37
Sb	0.001	V	0.211
Mn	0.36	O	0.0017
Sn	0.008	S	0.001
Ti	0.004	Mo	0.88
B	0.0001	Co	0.01
Pb	0.001	As	0.005
Si	0.24	Fe	88.3379
Ni	0.15	-	-

Figura 2.3. Composición del acero ASTM A335 P91

los aceros 9Cr modificados presentan dos grupos principales de segundas fases. El primer grupo corresponde a los carburos de Cr: estos son principalmente los M_7C_3 y los $M_{23}C_6$, donde M puede ser Cr y/o Fe. El segundo grupo corresponde a los precipitados de tipo MX, donde M puede ser Niobio (Nb), Vanadio (V), Titanio (Ti) y X es C y/o Nitrógeno (N). Los MX -con tamaños típicos del orden de los 20 a los 150 nm- son particularmente importantes puesto que le confieren al material resistencia a los distintos mecanismos de creep. Dependiendo de la composición, las partículas MX pueden crecer adoptando distintas morfologías [16]:

- Tipo I: precipitados esferoidales ricos en Nb estables a alta temperatura.
- Tipo II: precipitados facetados ricos en V.
- Tipo III: Precipitados complejos.

En estudios realizados sobre estos aceros sometidos a tensiones y temperaturas elevadas (650°C) durante tiempos prolongados se detectaron otras fases tales como precipitados $Fe_2(MoW)$, también llamados fases de Laves [14][16][17].

2.4.2. Diagrama CCT

En la [Figura 2.4](#) se muestra el diagrama CCT del acero ASTM A335 grado P91 informado en la literatura [18]. Como puede verse, el acero P91 austenizado a 1050°C durante 30 minutos exhibirá una microestructura 100 % martensítica al ser enfriado dentro de un amplio rango de velocidades. Se determinó en trabajos previos que la velocidad de enfriamiento crítica para la obtención de esta microestructura (V_m) se encuentra acotada en el intervalo [190-200] °C/h [19].

Se encuentra en bibliografía que la temperatura de inicio de la transformación martensítica para velocidades de enfriamiento mayores a V_m adopta el valor promedio $M_S = 373^\circ\text{C}$.

La ausencia de un rango bainítico en el diagrama CCT para una composición basada en 9%Cr está asociada a la elevada cantidad de aleantes y al hecho de que nunca se haya informado bainita en bibliografía bajo ninguna condición de enfriamiento.

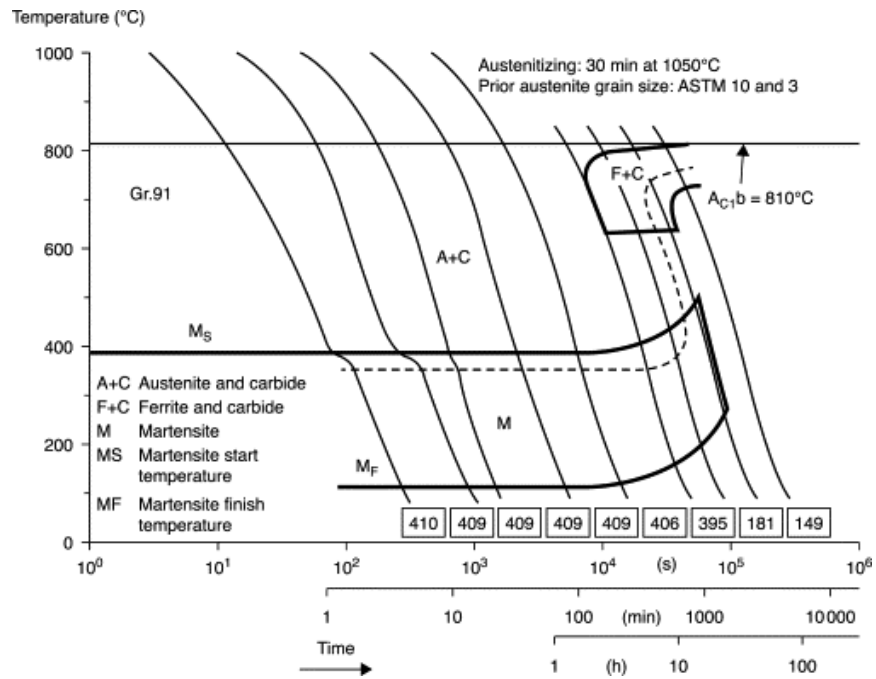


Figura 2.4. Diagrama CCT del acero ASTM A335 grado P91 [18]. “A” es austenita, “C” carburos, “F” ferrita y “M” martensita.

3 Dilatometría. Fundamentos de la técnica experimental y aplicaciones en transformaciones martensíticas

3.1. Dilatometría

3.1.1. Introducción

En general, los diferentes métodos para medir la expansión o contracción de un material cuando se lo somete a cambios de temperatura se pueden englobar dentro de las técnicas dilatométricas. A lo largo de este trabajo cuando se mencione el término dilatometría nos referiremos a la técnica de medición que utiliza pequeñas probetas para la determinación de coeficientes de expansión térmica y transformaciones de fases en el estado sólido. Así, dependiendo del tipo y modelo de dilatómetro, es posible aplicar múltiples ciclos temperatura- tiempo combinando segmentos de calentamiento, mantenimiento isotérmico y enfriamiento apropiados. Durante estos ciclos se monitorea con precisión el cambio en una dimensión característica de la probeta; este cambio, con simplificaciones adecuadas, da cuenta a su vez de la variación del volumen específico de la probeta y puede correlacionarse directamente con coeficientes de expansión térmica, transiciones de fases y defectos en las redes cristalinas. [20]

3.1.2. Generalidades sobre deformación térmica, defectos y transformaciones de fases [20]

Por encima de la temperatura de cero absoluto los átomos y moléculas dentro de un material sólido cristalino o amorfo se encuentran vibrando continuamente. Con el aumento de temperatura la amplitud de dichas vibraciones se acrecienta y comienza a forzarse e incrementarse la distancia efectiva entre los átomos; este fenómeno microscópico provoca, a su vez, que los sólidos se expandan de manera tal que esa expansión puede ser macroscópicamente medida con un dilatómetro. Para un metal, el coeficiente de expansión lineal puede definirse como la fracción en longitud que se expande por efecto de la temperatura, al comparar la longitud final con la longitud inicial de una muestra:

$$\alpha = \frac{dl}{dT} \quad (3.1)$$

En cristales de baja simetría cristalina y en materiales compuestos el coeficiente de expansión térmica (CTE) puede tener un valor distinto dependiendo de la dirección del cristal que se considere, dando lugar a un efecto de anisotropía. Estas características del CTE se refieren siempre a sistemas de una sola fase y por lo tanto muestran siempre un cambio continuo con la temperatura.

Los defectos cristalinos que se incorporan como un volumen de exceso también pueden ser detectados por dilatometría de alta resolución. Por ejemplo, las vacancias son defectos

termodinámicamente necesarios por encima de 0 K y se encuentran presentes en todos los cristales; la dependencia de la densidad de vacancias con la temperatura está dada por una ley de tipo Arrhenius. Por este motivo los defectos medibles en un dilatómetro son los presentes en el régimen de altas temperaturas.

Las transformaciones de fases en estado sólido se relacionan con un cambio en la estructura cristalina, por ejemplo, hierro cúbico centrado en el cuerpo a hierro cúbico centrado en las caras para la transformación ferrita a austenita en el caso del hierro puro y en los aceros. Este cambio en la red cristalina conlleva un efecto adicional al cambio en la dimensión característica que se está midiendo debido a la expansión térmica, y puede ser positivo o negativo dependiendo de las respectivas densidades de empaquetamiento que tengan la fase madre y la fase hija. También es posible detectar transiciones de segundo orden, por ejemplo, transformaciones de orden-desorden como la transición desde el estado ferromagnético al paramagnético en el hierro.

3.1.3. Metodología para el análisis dilatométrico de las transformaciones de fases[20]

Además de variadas aplicaciones específicas, la dilatometría se utiliza para medir los cambios en longitud producidos por transformaciones de fases durante calentamientos o enfriamientos continuos o durante etapas de mantenimiento isotérmico. Como resultado de estas mediciones es posible obtener, por ejemplo, diagramas de transformación en enfriamiento continuo (diagramas CCT) y diagramas de transformación isotérmica (diagramas TTT). Sin embargo existen otros tipos de diagramas que pueden ser generados con la ayuda de un dilatómetro: tiempo - temperatura - austenización (TTA), tiempo - temperatura - recocido, tiempo - temperatura - precipitación, tiempo- temperatura- recristalización, entre otros.

En el caso particular de los aceros, los análisis dilatométricos más comunes incluyen el estudio de transformaciones durante un ciclo de enfriamiento o calentamiento continuo y de transformaciones durante un mantenimiento isotérmico. En este último caso la probeta se lleva a una meseta de temperatura homogénea a través de un enfriamiento o calentamiento y se mantiene a temperatura constante; durante este mantenimiento se evalúa la deformación en el tiempo debida a la transformación de fase. La alternativa de estudiar los aceros durante el calentamiento o enfriamiento continuo consiste en realizar experiencias modificando la transferencia de calor de forma específica pero suficientemente flexible como para compensar cualquier calor liberado o absorbido por la muestra durante la transformación.

3.1.4. Equipo experimental [21]

Existen diferentes configuraciones que puede adoptar un dilatómetro, dependiendo del método que emplea para cuantificar la expansión experimentada por el material y el sistema de calentamiento/enfriamiento que permite llevar adelante el ciclado térmico de la probeta.

En la [Figura 3.1](#) se muestra el dilatómetro empleado para la obtención de los resultados que se analizaron en este trabajo, a saber, un dilatómetro de alta resolución y alta velocidad Bähr modelo DIL 805 A perteneciente a la Gerencia Materiales de CNEA. El mismo consta de una cámara sellada conectada a un sistema de vacío que cuenta con dos bombas: una bomba mecánica que lleva adelante la evacuación inicial de la cámara y otra turbomolecular, que, en una situación óptima, genera presiones del orden de $1 \cdot 10^{-6} \text{ mbar}$.

En la Figura 3.2 se muestra un esquema de la cámara sellada donde se observa el dispositivo de medición constituido esencialmente por una bobina, dos varillas de vidrio de sílice y un LVDT (transformador diferencial de variación lineal). La probeta -un cilindro macizo de 10 mm de longitud y 4 mm de diámetro- se coloca en el centro de la bobina, y se mantiene posicionada entre las varillas de vidrio de sílice únicamente por presión. Haciendo circular corriente alterna a través de la bobina se genera un campo magnético variable que a su vez -dado el carácter conductor del material en estudio- induce corrientes parásitas en la probeta. Consecuentemente, la probeta se calienta por efecto Joule. Para monitorear la temperatura se sueldan sobre la misma los dos terminales de una termocupla tipo S. El enfriamiento de la probeta puede realizarse por distintos medios: si las velocidades de enfriamiento requeridas no son excesivas, el enfriamiento se alcanza en vacío por radiación. Por otro lado, si son necesarias velocidades de enfriamiento mayores, la probeta puede ser enfriada inyectando Ar o He en la cámara mediante un chorro dirigido directamente sobre ella. Una de las dos varillas de vidrio de sílice se encarga de transmitir el efecto de la expansión térmica de la probeta al LVDT.



Figura 3.1. Dilatómetro Bähr modelo DIL 805 A, que proveyó los resultados analizados en este trabajo.

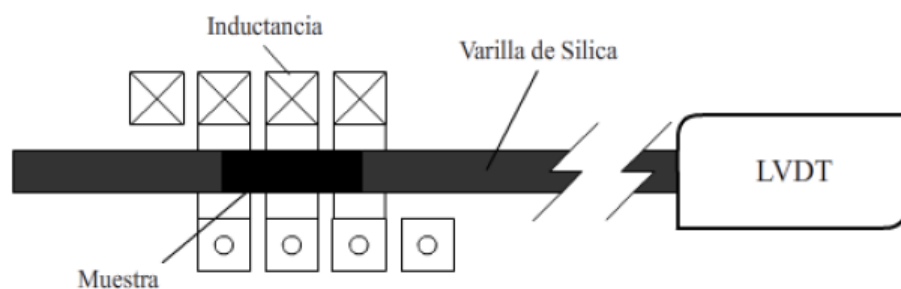


Figura 3.2. Esquema de componentes principales de un dilatómetro.

Un LVDT es un dispositivo electromecánico sencillo que permite medir desplazamientos lineales de hasta 100 nm. Consiste en un arreglo de tres bobinas dispuestas en dos circuitos (primario y secundario) y un vástago con un núcleo ferromagnético. El circuito primario (bobina central) está conectado a un generador de corriente alterna de alta frecuencia (1 a 10 kHz), mientras que las bobinas del secundario se encuentran conectadas en serie e invertidas de modo que la tensión medida en el secundario es la diferencia entre las corrientes inducidas en ambas bobinas.

El vástago se mueve libremente modificando la tensión inducida en cada una de las bobinas del secundario. Cuando el núcleo ferromagnético se ubica en el centro del LVDT, las tensiones inducidas en cada bobina son iguales y de sentido opuesto, con lo que la tensión de salida es nula. Por el contrario, si el vástago se desplaza en una u otra dirección con respecto al centro la señal de salida adoptará un comportamiento oscilatorio con una determinada amplitud y fase. Esta señal puede ser convenientemente rectificada para producir una tensión de salida continua, sensible a la amplitud y fase de la señal alterna y directamente proporcional al desplazamiento dentro de cierto rango de desplazamientos llamado rango lineal del LVDT.

3.2. Análisis de las curvas dilatométricas

Las variables medidas en una experiencia de dilatometría convencional son tiempo, temperatura y cambio de longitud. Típicamente la unidad de medición del cambio de longitud dl es el μm . A su vez es posible definir la deformación como:

$$\varepsilon = \frac{dl}{l_0} \quad (3.2)$$

donde dl es la variación de la longitud y l_0 la longitud inicial. La definición ingenieril y con la que se trabajará está dada por la siguiente relación:

$$\varepsilon = \frac{l - l_0}{l_0} \quad (3.3)$$

En la [Figura 3.3](#) se esquematizan los cambios de longitud en una probeta cilíndrica en las direcciones radial (d) y longitudinal (l). En el caso de un material isotrópico es posible considerar que $\varepsilon = \varepsilon_l = \varepsilon_d$.

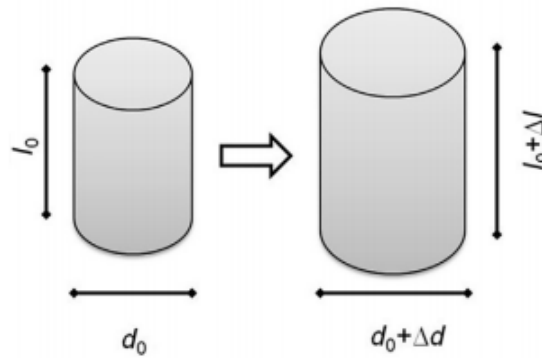


Figura 3.3. Esquema de variación de longitud en una probeta isotrópica

En una región de temperatura donde el material no sufra una transformación de fase, es posible describir la deformación en función de la temperatura a partir de la siguiente ecuación:

$$\varepsilon(T) = \varepsilon_0 + \int_{T_0}^T \alpha(T') \cdot dT' \quad (3.4)$$

donde ε_0 es una deformación inicial, α es el CTE definido en la ecuación 3.1 y T_0 es una temperatura de referencia. Para altas temperaturas el CTE es casi constante y la integración de la ecuación anterior resulta sencilla. Sin embargo, se ha encontrado [22] que el comportamiento para varios metales puede predecirse a través de la ecuación 3.5, que tiene en cuenta que el CTE debe anularse cuando la temperatura se acerca al cero absoluto:

$$\alpha(T) = \alpha_D \left[1 - \exp\left(\frac{-T}{T_D}\right) \right] \quad (3.5)$$

En la Figura 3.4 se muestra la dependencia del CTE con la temperatura para las fases austenita y ferrita presentes en la mayoría de los aceros.

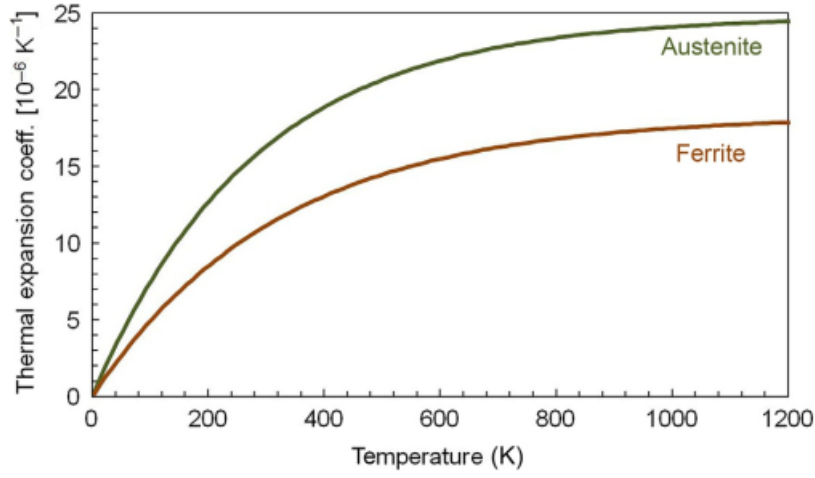


Figura 3.4. Dependencia del CTE con la temperatura para las fases ferrita y austenita en aceros [23]

Dado que la expansión térmica depende del material, cuando se tienen distintas fases simultáneamente presentes es posible determinar la deformación a través de la siguiente ecuación:

$$\varepsilon(T) = \sum_i^N p_i \cdot \left(\varepsilon_{i0} + \int_{T_0}^T \alpha_i(T') \cdot dT' \right) \quad (3.6)$$

donde p_i es la fracción en volumen de la fase i , α_i es el CTE correspondiente a dicha fase y ε_{i0} es el coeficiente de deformación independiente de cada fase; naturalmente $\sum_{fases} p_i = 1$.

La ecuación 3.6 puede aplicarse también para medir la deformación durante las transformaciones de fases. En esta situación, como el contenido de cada fase $p_i(t)$ depende del tiempo, la deformación también dependerá del tiempo $\varepsilon(t, T)$. En el caso de dos fases p_m y p_h , donde m corresponde a la fase madre y h corresponde a la fase hija respectivamente, se deduce que $p_m = 1 - p_h$. Para una transformación isotérmica, la fracción transformada estará dada por

$$p_h(t) = \frac{\varepsilon(t) - \varepsilon_m(T)}{\varepsilon_h(T) - \varepsilon_m(T)} \quad (3.7)$$

La ecuación 3.7 se conoce como regla de la palanca y puede ser también utilizada para transformaciones en calentamiento o enfriamiento continuo. Es importante tener en cuenta las siguientes consideraciones para su correcta aplicación:

- El coeficiente de dilatación térmica es constante en el rango de temperatura estudiado, por lo tanto, la contracción térmica de las fases que intervienen en la transformación durante un enfriamiento está descrita, en cada caso, por una recta.
- La deformación es linealmente dependiente de la fracción transformada.
- Sólo se consideran dos fases intervinientes durante la transformación.
- La deformación $\varepsilon(t)$ si bien no lo explicita, tiene una dependencia indirecta con la temperatura a través de t .

En las siguientes figuras se muestra esquemáticamente la dependencia de la deformación con el tiempo y la temperatura para los casos isotérmico y en enfriamiento continuo respectivamente.

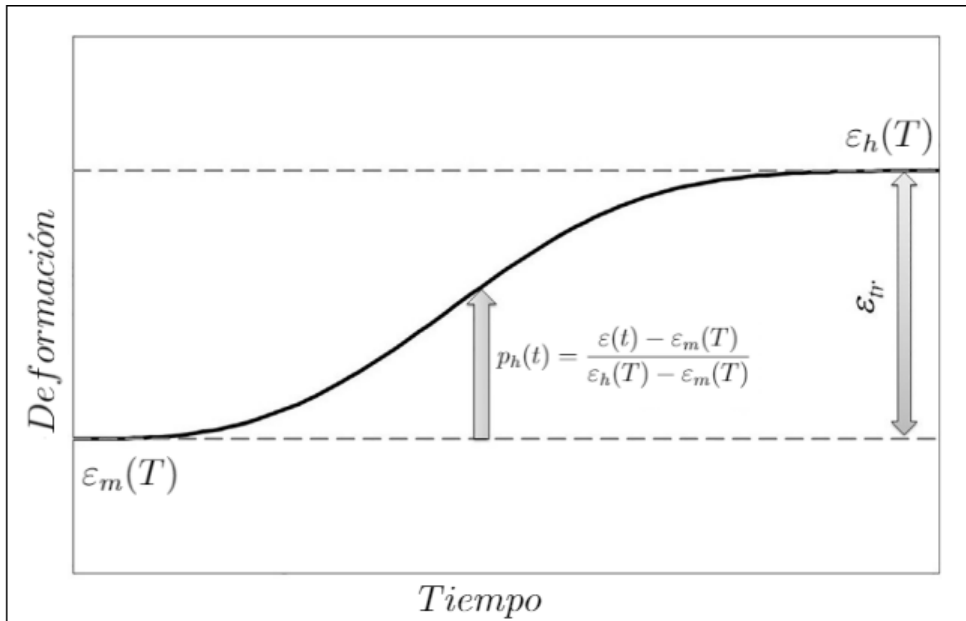


Figura 3.5. Deformación durante una transformación isotérmica.

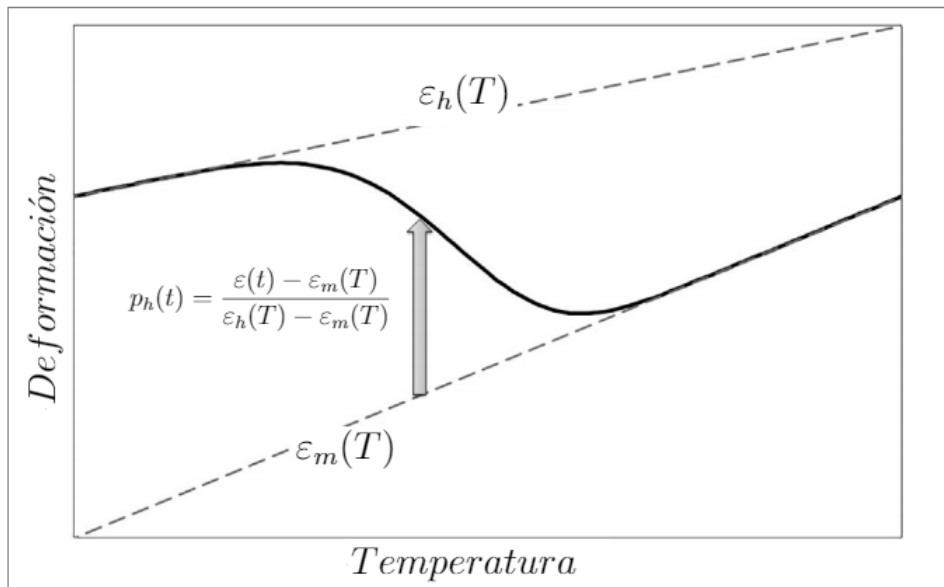


Figura 3.6. Deformación durante una transformación en enfriamiento continuo

3.3. Métodos para la determinación de la temperatura de inicio de transformación martensítica

Para poder diseñar tratamientos térmicos específicos, formular modelos predictivos de la fracción transformada o asistir el desarrollo o elección del material consumible para un proceso de soldadura -considerando el factor que juega la transformación martensítica en la relajación de tensiones [24]- es de suma importancia conocer el valor de la temperatura de inicio de la transformación martensítica (M_S) durante el enfriamiento.

En las siguientes subsecciones se explicarán algunos métodos utilizados para la determinación de M_S , ya sea basándose en la composición química de la fase madre o utilizando los resultados obtenidos con un dilatómetro.

3.3.1. Modificación de la ecuación empírica para la determinación de M_S en aceros con alto contenido de Cr

El desarrollo de Finkler y Schirra [25] retoma el tipo de ecuación empírica de Andrews (presentada en el Capítulo 1) basándose, a su vez, en un análisis precedente de Steven y Haynes [26]. Este análisis predice con buena exactitud los valores de M_s para aceros de baja aleación teniendo en cuenta la composición química de la fase madre. Finkler y Schirra modificaron la expresión existente y agregaron algunos factores teniendo en cuenta la relación entre V y W con el Mo:

$$M_S = 635 - 474 [pw_C + 0,86 (pw_N - 0,15(pw_{Nb} + pw_{Zr})) - 0,066(pw_{Ta} + pw_{Hf})] - [17pw_{Cr} + 33pw_{Mn} + 21pw_{Mo} + 17pw_{Ni} + 39pw_V + 11pw_W] \quad (3.8)$$

donde pw_i es el porcentaje en peso del compuesto i . Esta ecuación es válida para aceros con contenidos de 8-14 wt % Cr y < 1 at %C, Mn, Ni, Mo, W, V.

3.3.2. Métodos para la determinación de la M_S a partir de curvas dilatométricas

Método 1: ASTM A1033 [27]

Para implementar este método se utiliza la regla de la palanca, que se introdujo en la sección 3.2, ecuación 3.7, y se define arbitrariamente a la M_S como el valor de temperatura para el que se registra un 1 % en volumen de martensita transformada.

El procedimiento consiste en ajustar dos regresiones lineales: una sobre los puntos que describen el enfriamiento de la austenita, hasta poco antes de la desviación dada para la transformación y la otra luego de completada la transformación. Dichas regresiones se extrapolan para temperaturas menores y mayores, respectivamente, y a partir de estas dos rectas se aplica la ecuación 3.7. En definitiva, según este método, se define M_S como la temperatura para la cual la regla de la palanca predice un 1 % en volumen de martensita transformada.

Método 2: Método de offset [28]

Este método se inspira en el procedimiento utilizado para la determinación de la tensión de fluencia en un ensayo mecánico de tracción $R_{0,2}$, para lo cual define un corrimiento (offset) que se relaciona directamente con los parámetros de red de ambas fases involucradas, austenita y martensita, junto con su composición.

El primer paso consiste en realizar una regresión lineal en la zona de contracción de la austenita hasta antes del inicio de la transformación. A esta recta se le suma un corrimiento (offset) ϵ , dado por la ecuación 3.11 y determinado a partir de a_γ (parámetro de red de la austenita) y $a_{\alpha'}$ (parámetro de red de la martensita). Estos parámetros dependen, a su vez, de la composición del acero, y también se tiene presente la fracción volumétrica de martensita transformada (V). El punto dado por la intersección de la recta offset y la curva dilatométrica definirá M_S ; en las siguientes expresiones se resume la variación de los parámetros de red con la composición química del acero y en la Figura 3.7 se muestra un esquema del método. En la ecuación 3.9, x_i representa la fracción molar de la especie i , mientras que en 3.10 w_i es la fracción másica.

$$a_\alpha = 0,28664 + \frac{(a_{Fe} - 0,0279x_C^\alpha)^2 \cdot (a_{Fe} + 0,2496x_C^\alpha) - a_{Fe}^3}{3a_{Fe}^2} - 0,003x_{Si}^\alpha + 0,006x_{Mn}^\alpha + 0,007x_{Ni}^\alpha + 0,031x_{Mo}^\alpha + 0,005x_{Cr}^\alpha + 0,0096x_V^\alpha \quad (3.9)$$

$$a_\gamma = 0,3573 + 3,3 \cdot 10^{-1}w_C^\gamma + 9,5 \cdot 10^{-3}w_{Mn}^\gamma - 2,1 \cdot 10^{-3}w_{Ni}^\gamma + 6,1 \cdot 10^{-3}w_{Cr}^\gamma + 3,1 \cdot 10^{-2}w_{Mo}^\gamma + 1,8 \cdot 10^{-2}w_V^\gamma \quad (3.10)$$

$$(1 + \epsilon)^3 = a_\gamma^{-3} [2Va_\alpha^3 + (1 - V)a_\gamma^3] \quad (3.11)$$

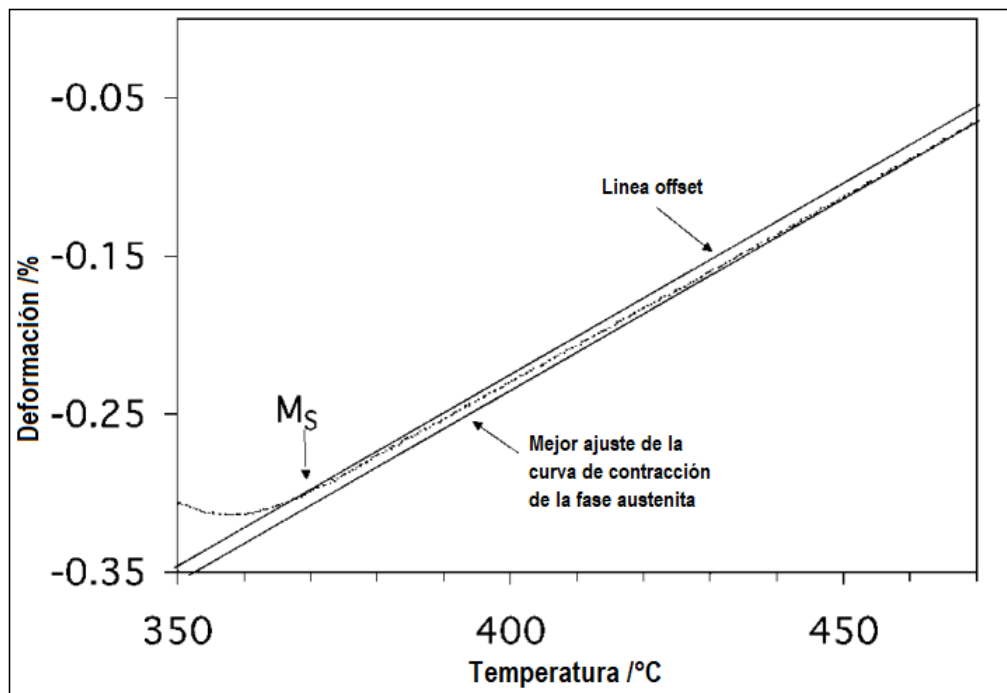


Figura 3.7. Determinación de M_S según método offset

Método 3: Rectas tangentes [29]

A diferencia de los planteados en lo anterior, este método es significativamente dependiente del criterio del observador. Se eligen cuatro puntos de forma arbitraria, ubicados antes y después de la región en donde se estima que se encuentra la temperatura M_S (reconocida gráficamente). Estos puntos se esquematizan en la Figura 3.8 como p_1, p_2, p_3 y p_4 y con ellos se definen las pendientes de dos rectas tangentes, $Recta_1$ y $Recta_2$. La intersección de ambas rectas determinará la temperatura de inicio de la transformación martensítica M_S .

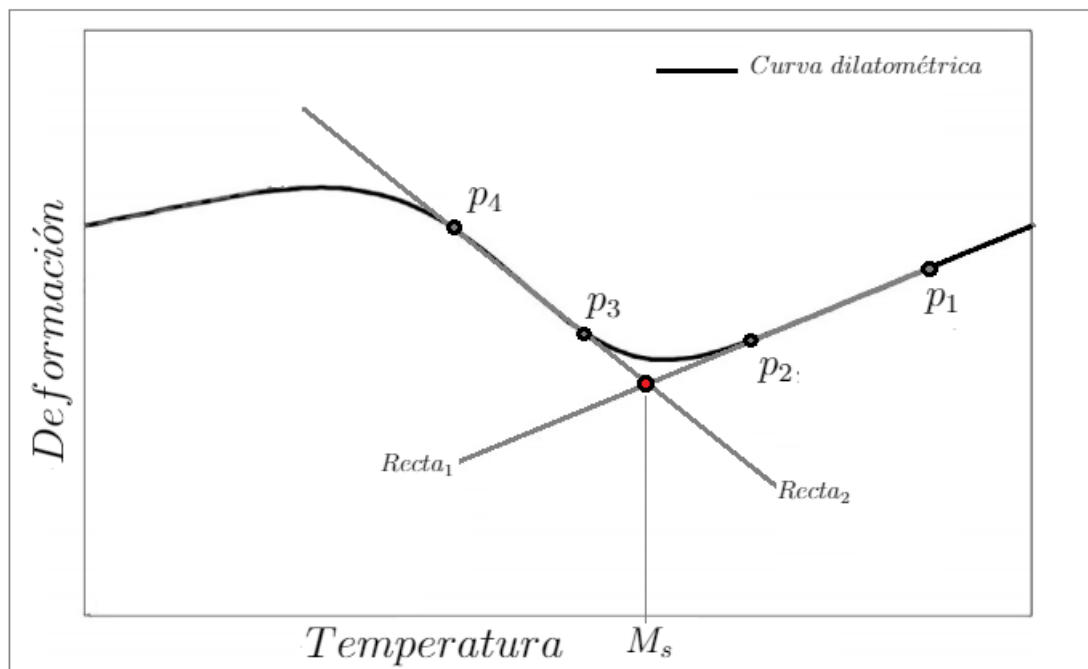


Figura 3.8. Determinación de M_S según método de las tangentes

4 Análisis de la evolución de la fracción transformada a martensita

4.1. Aspectos experimentales

Desde el punto de vista experimental, las dos metodologías más utilizadas para estudiar la evolución de la fracción transformada a martensita con la temperatura son las basadas en técnicas dilatómetricas y en métodos metalográficos. Cuando se considera la dilatometría, es posible utilizar la regla de la palanca para determinar la fracción transformada aun en presencia de efectos como redistribución de carbono y dependencias no lineales del CTE con la temperatura, dado que no introducen errores significativos en el estudio de la transformación austenita-martensita. El beneficio de esta metodología está dado por la posibilidad de seguir el curso completo de la transformación -esto es, la curva de fracción transformada final- con una sola probeta.

La otra técnica consiste en analizar metalográficamente una probeta templada hasta una temperatura objetivo durante la transformación y luego mantenida a esa temperatura o a una superior para finalmente enfriarla hasta temperatura ambiente. Este tratamiento permite distinguir en el microscopio óptico -luego de un ataque con el reactivo correspondiente- la martensita revenida (obtenida durante el primer temple) de la martensita final. Teniendo en cuenta la relación de una fase con la otra y repitiendo este procedimiento para varias temperaturas es posible reconstruir la curva de fracción transformada en función de la temperatura.

4.2. Modelado

Con el objetivo de mejorar las propiedades de los aceros se ha prestado especial atención al estudio y optimización de modelos empíricos y semi-empíricos que puedan predecir la fracción de martensita como función del subenfriamiento. A continuación, se describirán los modelos más utilizados en la bibliografía y algunos desarrollos recientes que se utilizaron para analizar la evolución de la fracción martensítica en el presente trabajo.

4.2.1. Modelo de Koistinen y Marburger (K-M) [30]

Uno de los primeros y más frecuentemente utilizados modelos empíricos para medir la fracción de martensita transformada es el propuesto por Koistinen y Marburger en 1959, ecuación 4.1, donde $f_{\alpha'}$ es la fracción en volumen transformada a martensita, α un parámetro de ajuste y $T < M_S$, la temperatura:

$$f_{\alpha'} = 1 - \exp[-\alpha(M_S - T)] \quad (4.1)$$

Esta expresión cobró mayor relevancia cuando Magee, en 1970 [31], dio un significado físico a la constante utilizada a partir de las siguientes consideraciones:

1. El aumento en el número de placas martensíticas por unidad de volumen de muestra, dN , es linealmente dependiente del incremento de la fuerza impulsora de la transformación martensítica $\Delta G_V^{\gamma \rightarrow \alpha'}$, como se describe en la ecuación 4.2 donde φ es una constante de proporcionalidad.

$$dN = -\varphi d(\Delta G_V^{\gamma \rightarrow \alpha'}) \quad (4.2)$$

2. El volumen (Ω) de cada placa de martensita se mantiene constante, y la nueva fracción en volumen de martensita es

$$df = \Omega(1 - f)dN \quad (4.3)$$

3. Combinando las dos ecuaciones anteriores, considerando que la fuerza impulsora y el volumen son constantes e integrando se tiene

$$f = 1 - \exp \left[\varphi \Omega \frac{d(\Delta G_V^{\gamma \rightarrow \alpha'})}{dT} (M_S - T) \right] \quad (4.4)$$

donde $\varphi \Omega \frac{d(\Delta G_V^{\gamma \rightarrow \alpha'})}{dT}$ equivale a α en la ecuación original de K-M. Una desventaja de esta ecuación es que la función que representa tiene una forma del tipo C, en lugar de una forma tipo S que es la observada experimentalmente para este tipo de transformaciones, por lo que es necesario extrapolar un valor M_S en lugar de utilizar el valor de M_S medido experimentalmente. Además, esta expresión suele presentar desviaciones considerables en el inicio de la transformación, por debajo del 20 % con respecto a los valores obtenidos en forma experimental a través de la regla de la palanca. Una explicación posible es que la teoría desarrollada no tiene en cuenta el fenómeno de nucleación autocatalítica -que modificaría el número de sitios de nucleación-, la composición química -dado que se ha demostrado que α es dependiente de la misma- y el hecho de que el volumen de las placas martensíticas no es constante.

4.2.2. Modelo de van Bohemen y Sietsma [32]

Este modelo se basa en el de K-M y le asigna a la constante α de la ecuación 4.1 una dependencia con la composición química a partir del estudio experimental de 19 aceros al carbono y de baja aleación. También define una M_S teórica a partir del parámetro T_{KM} , que reemplaza la M_S experimental, la cual también es función de la composición química. En las siguientes ecuaciones se pueden ver estas dependencias:

$$T_{KM} = 462 - 273.pw_C - 26.pw_{Mn} - 16.pw_{Ni} - 13.pw_{Cr} - 30.pw_{Mo} \quad (4.5)$$

$$\alpha_m = 0,0224 - 0,0107pw_C - 0,0007pw_{Mn} - 0,00005pw_{Ni} - 0,00012pw_{Cr} - 0,0001pw_{Mo} \quad (4.6)$$

$$f_{\alpha'} = 1 - \exp[-\alpha_m(T_{KM} - T)] \quad (4.7)$$

La ecuación 4.7 es válida para tenores de 0,3-1,1 wt % C, y se torna más precisa con la disminución de la concentración de elementos sustituciones tales como Cr y Ni. La influencia del C es mayor que la de los otros aleantes porque este elemento interfiere directamente en la cinética de la transformación a través de un fuerte efecto endurecedor en solución.

4.2.3. Modelo de van Bohemen [33]

Este estudio de van Bohemen fue una extensión del tratado en el párrafo anterior, donde se analizaron 115 aceros y se encontró que existe una dependencia exponencial de la temperatura M_S con el contenido de carbono. Esta constatación permitió modificar las ecuaciones 4.6 y 4.5 y remplazar el valor teórico de T_{KM} por el calculado a partir de la ecuación 4.8; a su vez, el valor de α también fue modificado, ecuación 4.9. La inclusión de estos dos nuevos parámetros en la ecuación 4.7 permitió mejorar el ajuste de los datos experimentales.

$$M_S = 565 - (31.pw_{Mn} + 13.pw_{Si} + 10.pw_{Cr} + 18.pw_{Ni} + 12.pw_{Mo}) - 600. [1 - \exp(-0,96.pw_C)] \quad (4.8)$$

$$\alpha(10^{-3}K^{-1}) = 27,2 - 0,14pw_{Mn} - 0,21pw_{Si} - 0,11pw_{Cr} - 0,08pw_{Ni} - 0,05pw_{Mo} - 19,8 [1 - \exp(-1,56pw_C)] \quad (4.9)$$

Donde pw_i es el porcentaje en peso del compuesto i y T_{KM} se llama temperatura de inicio de transformación martensítica teórica.

Este modelo es aplicable para aceros que posean el rango en composición 0,1-1,9wt %C, 0-3wt %Mn, 0-2wt %Si, 0-4wt %Cr, 0-5wt %Ni y 0-0,8wt %Mo y no contienen adiciones significativas de Cu, W, Nb, V, Co, Al, B. Es importante mencionar que todos los elementos de aleación deberán estar completamente disueltos en la austenita. Según el autor, las discrepancias entre la M_S calculada por 4.8 y la obtenida experimentalmente se correlacionan con el tamaño de grano austenítico previo, ya que T_{KM} se aproxima al valor experimental de M_S en el caso de microestructuras con un tamaño de grano austenítico muy fino.

4.2.4. Modelo de Lee y van Tyne[34]

En este modelo se añadió un parámetro adicional, β , a la ecuación 4.1, con el propósito de corregir las discrepancias observadas entre el modelo de K-M y los datos experimentales al inicio de la transformación (ecuación 4.10). A diferencia de los modelos anteriores la función asociada a este modelo tiene una forma de tipo S y la temperatura M_S utilizada es efectivamente la temperatura de inicio de la transformación martensítica a través de la ecuación 4.11 propuesta por Capdevila et al. [35]. Las ecuaciones y parámetros asociados se encuentran en las siguientes expresiones:

$$f_{\alpha'} = 1 - \exp[-\alpha_m(M_S - T)^\beta] \quad (4.10)$$

$$M_S = 764,2 - 302,6.pw_C - 30,6.pw_{Mn} - 14,5.pw_{Si} - 16,6.pw_{Ni} \\ - 8,9.pw_{Cr} + 2,4.pw_{Mo} - 11,3.pw_{Cu} + 8,58.pw_{Co} + 7,4.pw_W \quad (4.11)$$

$$\alpha(K^{-1}) = 0,0231 - 0,0105.pw_C - 0,0017.pw_{Ni} \\ + 0,0074.pw_{Cr} - 0,0193.pw_{Mo} \quad (4.12)$$

$$\beta = 1,4304 - 1,1836.pw_C + 0,7527.pw_C^2 - 0,0258.pw_{Ni} \\ - 0,0739.pw_{Cr} + 0,3108.pw_{Mo} \quad (4.13)$$

Donde pw_i es el porcentaje en peso del compuesto i . El rango válido de composición para las ecuaciones 4.11, 4.12 y 4.13 está dado por 0,2-1,2wt %C, 0,0-2,1wt %Ni, 0,0-1,2wt %Cr, 0,0-0,3wt %Mo, 0,75wt %Mn y 0,25wt %Si, la composición de estos dos últimos elementos son los valores medios encontrados en aceros al carbono y de baja aleación.

4.2.5. Modelo de Huyan y colaboradores [36]

Este modelo ofrece una característica adicional interesante, a saber, además de responder a la forma tipo S, relaciona los parámetros de ajuste con funciones termodinámicas, lo que le confiere una interpretación física más robusta. Se asume que la nueva fracción de láminas o placas de martensita, df , es proporcional al cambio en la fuerza impulsora, $d(\Delta G_m^{\gamma \rightarrow \alpha'})$ y se propone la siguiente ecuación,

$$df_{\alpha'} = B \cdot f \cdot (1 - f) \cdot \frac{1}{\Delta G_m^{\gamma \rightarrow \alpha'}} \cdot d(\Delta G_m^{\gamma \rightarrow \alpha'}) \quad (4.14)$$

donde B es una constante del material, f se añade para tener en cuenta el efecto autocatalítico de la martensita formada previamente y $\frac{1}{d(\Delta G_m^{\gamma \rightarrow \alpha'})}$ se relaciona con la energía de deformación en la fracción de austenita que no transformó, $(1 - f)$. Esta austenita no transformada se estabiliza por deformación y, si se asume que la energía de deformación almacenada es proporcional a la fuerza impulsora, el incremento de la fracción de martensita, df , es inversamente proporcional a la fuerza impulsora $d(\Delta G_m^{\gamma \rightarrow \alpha'})$. Al integrar la ecuación 4.14 desde M_S hasta cierta temperatura T por debajo de M_S se obtiene,

$$f_{\alpha'} = \frac{1}{1 + A^{-1} \cdot (\Delta G)^{-B}} \quad (4.15)$$

donde A es una constante y ΔG la fuerza impulsora en exceso, esto es, la diferencia entre la fuerza impulsora a T y a M_S

$$\Delta G = \Delta G_m^{\gamma \rightarrow \alpha'}(T) - \Delta G_m^{\gamma \rightarrow \alpha'}(M_S) \quad (4.16)$$

Las fuerzas impulsoras se calculan a partir de la ecuación

$$\Delta G_m^{\gamma \rightarrow \alpha'}(T) = \Delta G_m^{\gamma}(T) - \Delta G_m^{\alpha'}(T) - \Delta G_z(T) \quad (4.17)$$

en la cual G_m^{γ} y $G_m^{\alpha'}$ son las energías molares de Gibbs de la austenita y ferrita respectivamente y ΔG_z es la energía de ordenamiento de Zener, relacionada con la redistribución del carbono luego de la formación de la martensita.

4.3. Método CALPHAD y software Thermocalc

La formulación de modelos termodinámicos parametrizados y la optimización de los valores de dichos parámetros para la construcción de diagramas de fase sobre la base de datos experimentales de constitución y/o termoquímicos se denomina generalmente cálculo de diagramas de fase (o aproximación CALPHAD, CALculation of PHase Diagrams por sus siglas en inglés).

Los diagramas de fase proveen información básica e importante, especialmente para el diseño de nuevos materiales. Utilizando solo técnicas experimentales se requiere mucho trabajo y tiempo para la construcción de tan solo una región de un diagrama de fase, ya que en la práctica, los materiales están compuestos de aleaciones de múltiples componentes más que de sistemas binarios o ternarios. Con el objetivo de hacer más eficaz el trabajo experimental relacionado con la obtención de diagramas de fase en sistemas complejos multicomponentes, se desarrolló el método CALPHAD [37]. En este método una variedad de valores experimentales relacionados con los límites de fase y las propiedades termodinámicas se analizan de acuerdo a un modelo termodinámico apropiado y se evalúan las energías de interacción entre los átomos. Usando esta técnica, partes de diagramas de fase que están fuera del rango experimental accesible pueden calcularse basándose en datos termodinámicos. La dificultad en la extensión a sistemas de orden mayor de los resultados calculados para sistemas de bajo orden es mucho menor que la que aparece en el caso del trabajo experimental, dado que la esencia del cálculo no cambia mucho entre un sistema binario o ternario y uno de orden superior. Este método provee un entendimiento muy preciso de las propiedades macroscópicas del material estudiado [38].

Mediante la utilización del software THERMOCALC, se realizaron los cálculos correspondientes para estimar algunas de las composiciones que se alcanzaron durante los tratamientos realizados. El programa presenta la limitación de trabajar solamente en condiciones de equilibrio, es decir, no incluye la variable tiempo, por lo que no es posible calcular estados intermedios. Aun así, es posible estimar resultados a partir de los cálculos realizados con el programa.

5 Procesamiento y ajuste de los datos experimentales

El procesamiento de los datos experimentales requirió la implementación de códigos computacionales que permitieran ajustar distintos modelos físicos a las curvas experimentales obtenidas con anterioridad a la realización de este trabajo. Dichos códigos fueron elaborados utilizando el lenguaje Python y se exponen en el Apéndice destinado a tal fin. Antes de introducir la descripción de los métodos de procesamiento de datos utilizados, se listan los tratamientos térmicos que fueron objeto de análisis.

5.1. Tratamientos térmicos aplicados

Solo se analizaron las muestras que presentaron la martensítica como única transformación, y éstas, según lo que se observó en las curvas dilatométricas, solo se obtuvieron para los tratamientos térmicos descritos en la [Tabla 5.1](#). Por otro lado cabe señalar que para el tratamiento térmico 1, se puede analizar la repetitividad de los datos medidos o determinados.

Tabla 5.1. Tratamientos térmicos analizados

Tratamiento térmico	Descripción	Muestras
1	Calentamiento a 1°C/s hasta 1050°C por 30min y enfriamiento a 1°C/s	#541, 584, 586, 590
2	Calentamiento a 1°C/s hasta 1050°C por 30min y enfriamiento a 10°C/s	#587

5.2. Aplicación de métodos de determinación de M_s

La aplicación de los métodos descritos en la sección [3.3](#) -a través del desarrollo de algoritmos que permitan automatizarlos- se puede describir en las siguientes etapas:

1. Definición de parámetros: se identifican los parámetros necesarios según el método aplicado. Para ello se cuenta con un ejecutable principal de Python donde se tiene la información necesaria y a través del cual se ejecutarán algoritmos específicos para resolver acciones concretas llamados **funciones**; de esta forma se logra simplificar y reducir el código principal.
2. Evaluación: esta etapa está relacionada con un ejecutable secundario de Python que tiene definidas todas las funciones necesarias y también una secuencia de instrucciones para, luego del cálculo de un conjunto de datos discretos, poder obtener los valores de las magnitudes de interés a través del planteo de cotas para las mismas. Estas cotas pueden fijarse en función de la fracción transformada (método ASTM) o de la deformación (métodos offset y de las rectas tangentes).

3. Obtención de la fracción transformada: se calcula la fracción transformada por medio de la regla de la palanca.

Para ejemplificar esta secuencia de etapas se muestra a continuación la implementada para el método de offset.

5.2.1. Método de offset: Etapa de Definición

Para la etapa de definición se tomaron en consideración dos factores: el primero está asociado con la composición de las fases γ y α' (desarrollado en el ítem 1 abajo) y el segundo está asociado a la fracción de martensita con la cual se determinó el inicio de la transformación martensítica (desarrollado en el ítem 2).

1. Los parámetros de red se determinan, de acuerdo con lo explicado en la subsección 3.3.2, con las ecuaciones 3.9 y 3.10; luego estos participan en la definición del offset o corrimiento de la recta de interés. Así, deben tenerse en cuenta los tratamientos térmicos aplicados y lo descrito en la subsección 2.4.1 con respecto a las segundas fases, dado que de ello depende la composición de las fases de interés (austenita y martensita). Por esta razón se evaluaron dos configuraciones diferentes del sistema, a alta y baja temperatura. Dichas configuraciones fueron:
 - 100 % austenita (γ) a alta temperatura y 100 % martensita (α') formada durante el enfriamiento (Figura 5.1 con leyendas de color negro); en este caso para la ejecución de los cálculos se tomó la composición nominal del acero, es decir, se planteó que a altas temperaturas todos los elementos de aleación están disueltos en una única fase austenítica.
 - Un porcentaje p de fase γ , considerando la existencia de los carburos y nitruros no disueltos a la temperatura de trabajo, y la misma cantidad p % de martensita formada durante el enfriamiento (Figura 5.1 con leyendas de color rojo); en este caso se tomó la composición dada por el software Thermocalc.

Para la operación de Thermocalc se usó la base de datos termodinámicos tcf5; en el Anexo se presenta la secuencia de comandos empleados. Se definió la composición y la temperatura de austenizado (1050°C) y se obtuvieron las fases termodinámicamente estables en esas condiciones, a saber, 3 fases con estructura FCC, de las cuales se consideró la primera, esto es, un carbonitrúro de la forma MX rico en Nb. El caso señalado en verde en la Figura 5.1 corresponde al equilibrio a altas y bajas temperaturas, y no se tiene en cuenta.

2. Por otro lado se tuvo en cuenta la ecuación 3.11, expresión para la que también se planteó la variación del parámetro fracción en volumen transformada (V), seleccionándose los siguientes valores: 0.01, 0.015, 0.02 y 0.03.

Como no se obtuvieron variaciones significativas en la M_S determinada, se tomó $V=0.02$ dado que $f_{\alpha'}(M_S) \approx 0.02$, es decir, la fracción de martensita (determinada según la regla de la palanca) a la temperatura M_S es equivalente a un 2 % de martensita transformada.

5.2.2. Método de offset: Etapa de Evaluación

Cuando se tienen dos curvas definidas a través de series continuas de valores, la intersección de estas puede encontrarse localizando un punto de coordenadas (x, y) perteneciente

a ambas curvas, de manera tal que la diferencia entre los valores de las expresiones analíticas correspondientes sea nula, siendo además $y = \text{fun}(x)$ para cada una de ellas.

En nuestro caso, y teniendo presente que la serie de datos que se tiene constituye un conjunto discreto, la evaluación de la intersección entre la recta que mejor ajusta en el segmento lineal de la curva dilatométrica y la recta desplazada se realiza estimando una cota. Como para el método de offset se tiene que las coordenadas de los puntos corresponden a la temperatura (T_i) y deformación (ε_i), resulta $(x_i, y_i) = (T_i, \varepsilon_i)$; por lo tanto la cota tomada para evaluar la intersección está definida de la siguiente forma:

- $T_i^{\text{Recta1}} = T_i^{\text{Recta2}}$
- $|\varepsilon_i^{\text{Recta1}} - \varepsilon_i^{\text{Recta2}}| < \varepsilon_{\text{cota}} = 1$

donde *Recta 1* y *Recta 2* corresponden a la recta ajustada a la curva dilatométrica y a la recta corrida o desplazada.

La construcción para el caso del método de rectas tangentes es equivalente, y lo mismo ocurre para el método ASTM, con la distinción de que en este último caso se evalúa fracción transformada, es decir, $y_i = f_{\alpha',i}$.

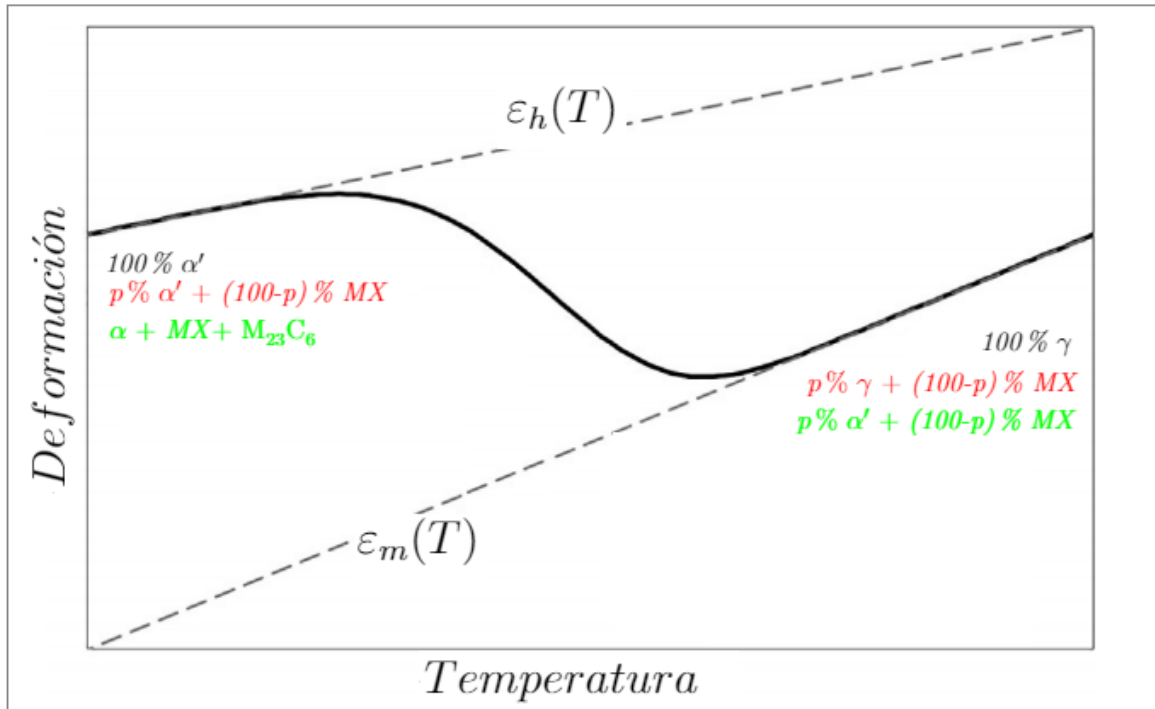


Figura 5.1. Configuraciones de fases del sistema y análisis por el método de offset

5.3. Análisis de la dispersión en la aplicación de ajustes lineales

Tanto el método ASTM como el método de offset mostraron una dependencia significativa de los resultados con respecto al intervalo seleccionado para ajustar cada región aproximadamente lineal de la curva dilatométrica, en especial en lo que hace a la región de contracción de la fase austenita. Este hecho sugirió la conveniencia de realizar un análisis de la desviación estándar asociada a M_S como resultado de la variación de dichos intervalos. Así, se definieron 4 intervalos de aplicación para el ajuste lineal de la contracción de la austenita, modificando el extremo derecho como se muestra en la [Figura 5.2](#) (es decir,

considerando los puntos 0, 1, 2 y 3) y tomando como extremo izquierdo fijo el punto a 400°C.

5.3.1. Localización del ajuste

Llamaremos localización del ajuste a la selección del conjunto de puntos medidos más pequeño y cercano al punto en donde se estima que se encuentra M_S para realizar el ajuste lineal que va a definir la fracción transformada. Es decir, se localiza el ajuste cuando se varía desde el punto 0 al 3 en el extremo derecho del intervalo de ajuste, como se muestra en la Figura 5.2.

Al localizar el ajuste se obtiene un mejor coeficiente de correlación para el mismo, dado que, si bien se considera en primera instancia que el coeficiente de dilatación térmica es constante para las fases participantes, se sabe que en realidad sufre pequeñas variaciones con la temperatura, como se mostró en la subsección 3.2.

Cabe aclarar que esta localización en los ajustes, se realizó solo para los métodos ASTM y offset.

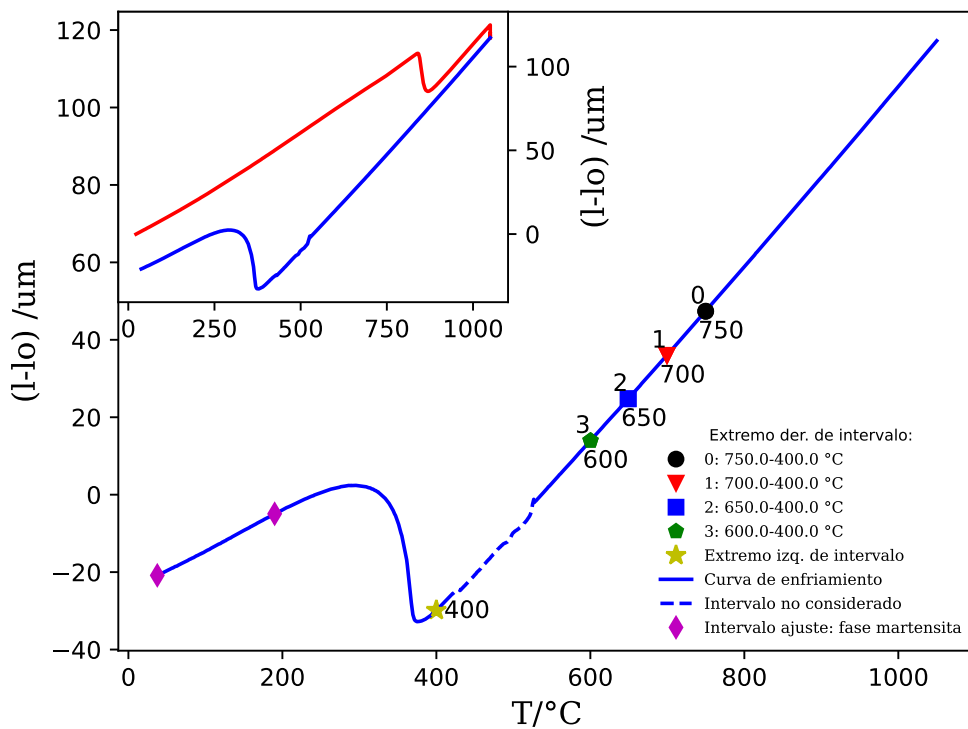


Figura 5.2. Curva dilatométrica con puntos representativos para el análisis

5.3.2. Segmento excluido

En las muestras enfriadas a $1^{\circ}\text{C}/\text{s}$ (códigos #541, 584, 586, 590) se observa un rango de temperatura entre aproximadamente 530 y 580°C en el cual existe una perturbación en la tendencia de la curva dilatométrica, lo cual dificulta la definición de M_S . Esta perturbación está asociada con parámetros experimentales del equipo necesarios para controlar la velocidad de enfriamiento.

Se aplicaron filtros de promedio móvil centrados para disminuir el ruido de las mediciones, tanto en los puntos de deformación medida como en la fracción de martensita

determinada por la regla de la palanca, sin conseguir un resultado apreciablemente mejor; por este motivo, se procedió a suprimir el segmento donde se registraban estas perturbaciones. Se ensayaron distintas formas de trabajar con el segmento excluido.

- Forma 1:

Tanto en la evaluación de M_S como en la definición de la fracción transformada se suprimen los puntos que integran el segmento de línea a trazos en la [Figura 5.2](#) y con los puntos restantes se definen los parámetros de ajuste lineal.

- Forma 2:

En la evaluación de M_S se suprimen los puntos medidos que definen el segmento de línea a trazos en la [Figura 5.2](#) y con los puntos restantes se definen los parámetros del ajuste lineal. En este caso, para la definición de la fracción transformada el intervalo suprimido se redefinió a partir de los parámetros de ajuste dados anteriormente, y se rehicieron los ajustes lineales para aplicar la regla de la palanca.

- Forma 3:

En la evaluación de M_S se suprimen los puntos medidos que definen el segmento de línea a trazos en la [Figura 5.2](#) y con los puntos restantes se definen los parámetros de ajuste lineal. Para la definición de la fracción transformada, el intervalo suprimido se redefinió a partir de los parámetros obtenidos por los puntos extremos del intervalo suprimido, y se rehicieron los ajustes lineales para aplicar la regla de la palanca.

Se seleccionó la primera forma de trabajo por simplicidad, dado que no se observaron modificaciones en los resultados obtenidos con respecto a los restantes procedimientos.

5.4. Aplicación de modelos de ajuste para la fracción transformada a martensita

A partir de lo descrito en la sección 4.2, el desarrollo de los algoritmos para obtener las curvas que definirán las fracciones transformadas en función de la temperatura de subenfriamiento se puede categorizar en dos tipos:

1. Parámetros característicos ajustados: en este caso, los parámetros de los modelos presentados se consideran libres y se obtienen a través de ajustes numéricos utilizando un paquete predefinido de Python: *scipy.optimize*. Este método se aplicó usando las M_S determinadas experimentalmente (por los métodos ASTM, offset y de las rectas tangentes descritos anteriormente; los resultados se presentarán en el Capítulo 6, subsección 6.4.1) y la M_S obtenida según la formulación de Finkler y Schirra (ídem subsección 6.4.2).
2. Parámetros característicos y M_S calculados: este caso se refiere a que tanto los parámetros de los modelos como la M_S ya tienen una formulación determinada empíricamente en función de la composición del acero (cabe también remarcar que estas formulaciones, según sea el caso, están restringidas a ciertos tipos de aceros). Para ello se define una función que realice las operaciones matemáticas y devuelva los valores según el modelo.

En otras palabras, solo se aplican las siguientes ecuaciones, descritas para cada modelo:

- van Bohemen y Sietma: ecuaciones 4.5, 4.6 y 4.7
- van Bohemen: ecuaciones 4.7, 4.8 y 4.9.
- Lee y van Tyne: ecuaciones 4.10, 4.11, 4.12 y 4.13.

Para el modelo de Lee y van Tyne, los algoritmos se desarrollaron según los dos procedimientos mencionados.

5.4.1. Parámetros libres ajustados

Consideraciones sobre los modelos de Koistinen-Marburger y Lee-van Tyne

Ambos modelos se ajustaron directamente a las curvas experimentales obtenidas a través de la regla de la palanca (ecuación 3.7) utilizando los parámetros correspondientes. En el caso de Koistinen-Marburger se ajustó el parámetro α , mientras que para Lee-van Tyne se ajustaron α y β .

Consideraciones sobre modelo de Huan y col.

En este caso se aplicó un ajuste indirecto en la curva experimental obtenida a través de la regla de la palanca. Según lo descrito en la subsección 4.2.5, despejando ΔG de la ecuación 4.15 se deduce la relación 5.1 y, a partir de ella, puede obtenerse un conjunto de pares de valores $(\Delta G_i, f_{\alpha'i})$. Por otra parte, para la dependencia de ΔG con la temperatura se asume, una relación lineal (ecuación 5.2), y se realiza de este modo un ajuste para la diferencia $(T - M_s)$. Los parámetros por ajustar son, entonces, a y b en dicha relación lineal.

$$\Delta G = 100 \cdot \left[\frac{f_{\alpha'}}{A \cdot (1 - f_{\alpha'})} \right]^{1/B} \quad (5.1)$$

$$\Delta G = a \cdot (T - M_s) + b \quad (5.2)$$

Finalmente, los parámetros A y B se tomaron de los valores dados en [36], tal como se ha descrito en la subsección 4.2.5.

La hipótesis que supone que la fuerza impulsora ΔG depende linealmente de la temperatura se fundamenta en el cálculo termodinámico de las energías libres de la fase austenita y ferrita (la cual simula a la martensita de bajo carbono encontrada en este acero) para un sistema simplificado del ASTM A335 grado P91: 0,1 wt % C y 8,37 wt % Cr. A través del software Thermocalc se calculó la energía libre de Gibbs en función de la temperatura asumiendo la presencia única de cada una de las fases involucradas. Así, se realizaron dos cálculos, uno para la austenita y otro para la ferrita, y luego se calculó la fuerza impulsora de la transformación austenita a martensita como se expresó en la ecuación 4.17 sin tener en cuenta la energía de ordenamiento de Zener en esta primera aproximación.

En la Figura 5.3 se observan los resultados obtenidos con respecto a la dependencia de la fuerza impulsora para la transformación de austenita a martensita en función de la temperatura. Claramente en el rango de interés de esta transformación – 200 a 400 °C – es posible estimar un comportamiento lineal de la fuerza impulsora con la temperatura.

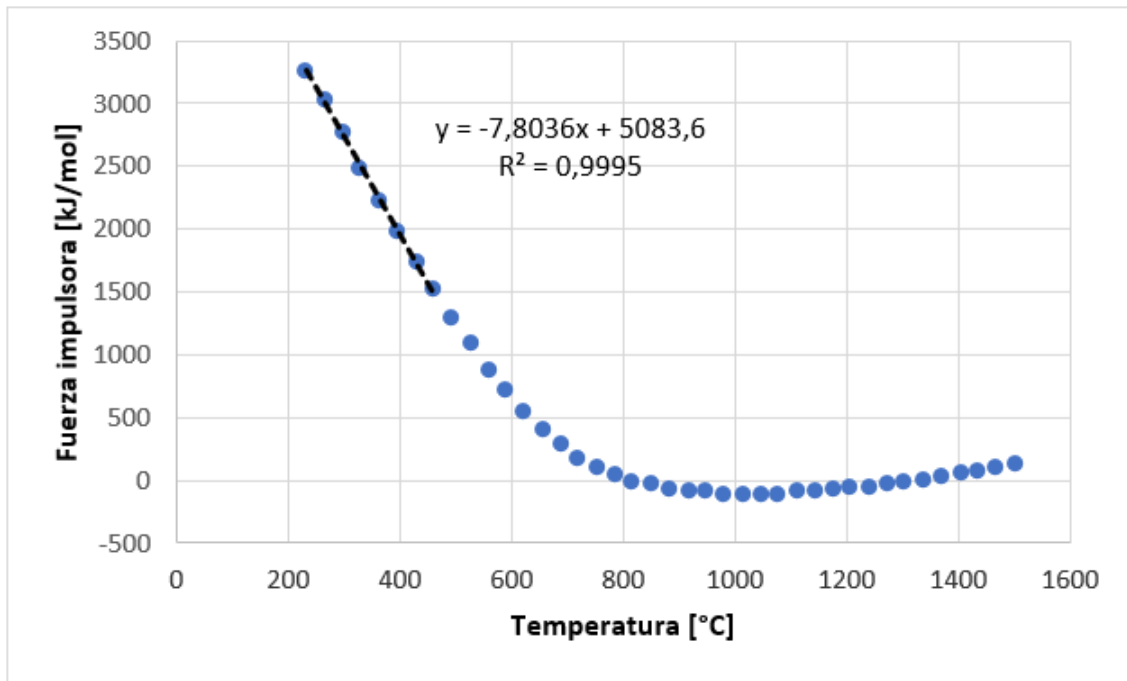


Figura 5.3. Dependencia de la fuerza impulsora de la transformación austenita a martensita con la temperatura para un sistema simplificado del acero ASTM A335 grado P91.

6 Resultados

6.1. Ciclos térmicos

En la [Figura 6.1](#) y la [Figura 6.2](#) se muestran los ciclos térmicos a los que fueron sometidas las muestras analizadas. Ambos ciclos consistieron en un austenizado a 1050°C durante 30 minutos; la velocidad de calentamiento fue de 1°C/s y sólo se modificó la velocidad de enfriamiento: 1°C/s para el ciclo térmico denominado 1 y 10°C/s para el ciclo térmico 2. Se realizaron 4 repeticiones del ciclo térmico 1 (probetas códigos 541, 584, 586 y 590), dado que esta velocidad resultaba de sumo interés para actividades futuras.

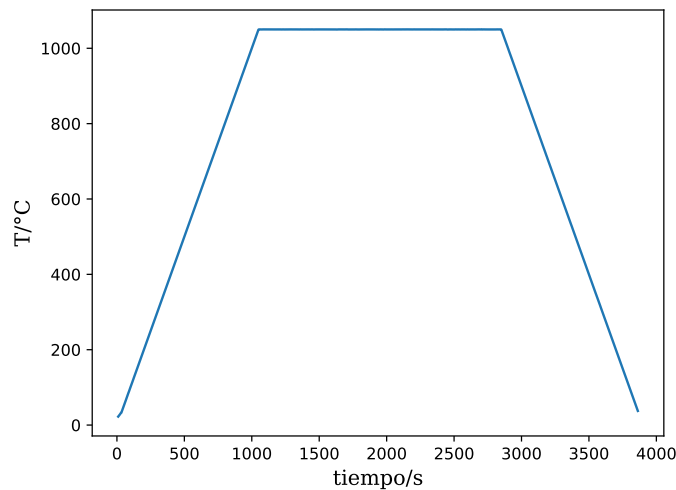


Figura 6.1. Ciclo térmico 1, muestras 541, 584 586 y 590.

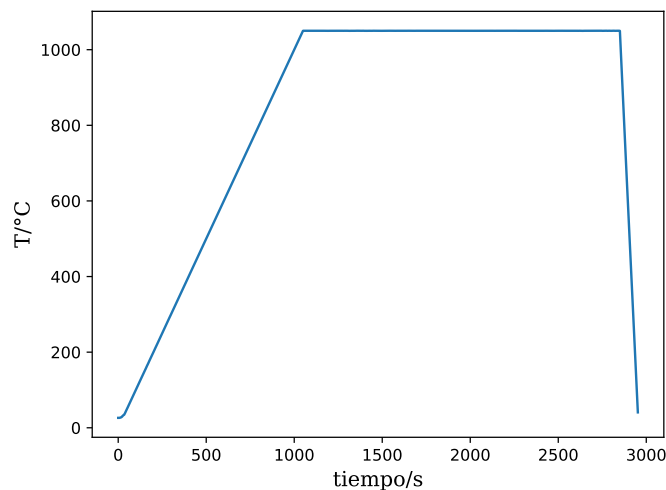


Figura 6.2. Ciclo térmico 2, muestra 587.

6.2. Curvas dilatométricas

En las [Figura 6.3](#), [Figura 6.4](#), [Figura 6.5](#), [Figura 6.6](#) y [Figura 6.7](#) se muestran los resultados dilatométricos en función de la temperatura para todos los ciclos presentados en la sección anterior. Los insertos que se encuentran en cada esquina superior izquierda hacen referencia a lo ocurrido durante todo el tratamiento térmico, la parte en trazo rojo durante el calentamiento y la parte en trazo azul durante el enfriamiento. Dado que el mayor interés de este trabajo reside en el estudio durante el enfriamiento continuo, la parte azul se encuentra graficada con mayor detalle y con observaciones que serán descriptas más adelante.

La evolución del cambio de longitud -y en consecuencia de la deformación- puede describirse en tres etapas que se correlacionan con el tratamiento térmico aplicado:

1. Durante el calentamiento, en el inserto izquierdo y demarcado en rojo, se observa primeramente un comportamiento lineal asociado con una dilatación constante de la fase ferrítica-martensítica (α) debida al incremento en la temperatura, luego de los 760°C aproximadamente aparece un quiebre y una contracción en la curva que se asocia a la transformación de fase ferrita (α) a austenita(γ). La primera fase tiene una estructura cristalina menos compacta que la segunda, cúbica centrada en el cuerpo (BCC) y cúbica centrada en las caras (FCC) respectivamente. Por último, se observa nuevamente un segmento con comportamiento lineal pero esta vez asociado con la dilatación sufrida por la fase austenítica.
2. Durante el mantenimiento a 1050°C es posible observar una pequeña contracción, que podría estar relacionada con la disolución y precipitación de carburos y/o carbonitruros; el análisis detallado de esta hipótesis excede los objetivos de este trabajo. En algunos casos se observa que el inicio de la curva de enfriamiento no tiene la misma pendiente que la registrada para la curva de calentamiento, es decir, el mismo CTE. Luego del proceso de austenizado, esto pueden deberse a dos efectos: inhomogeneidades en el material (diferencias térmicas entre la superficie y el centro de la probeta medida) o ligeras variaciones en el mecanismo de enfriamiento del dilatómetro. Este fenómeno se percibe más claramente en las curvas correspondiente a las muestras #584, 587 y 590.
3. Como se mencionó anteriormente, al inicio del enfriamiento se puede observar la contracción de la austenita por el cambio continuo en la temperatura. En la curva ampliada se muestran los puntos de trabajo para los diferentes métodos de determinación de M_S y el intervalo de puntos excluidos. Cabe señalar que para la muestra #587 no fue necesario excluir dicho intervalo, dado que la perturbación se vio desplazada hacia temperaturas más altas y no interfirió en la determinación de M_S .

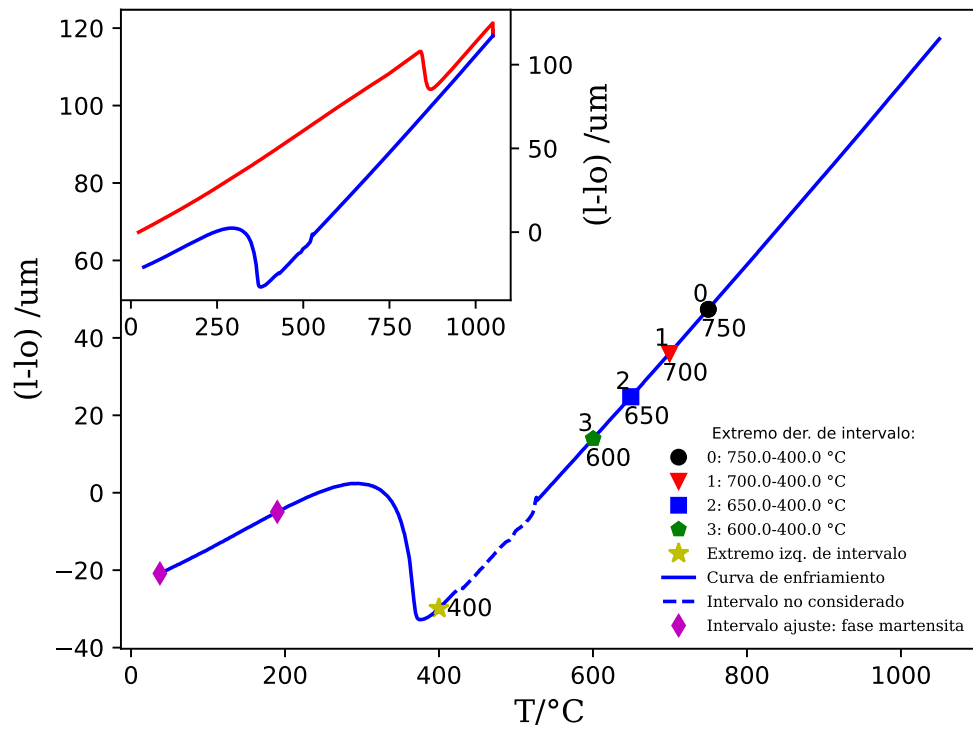


Figura 6.3. Curva dilatométrica de la muestra #541

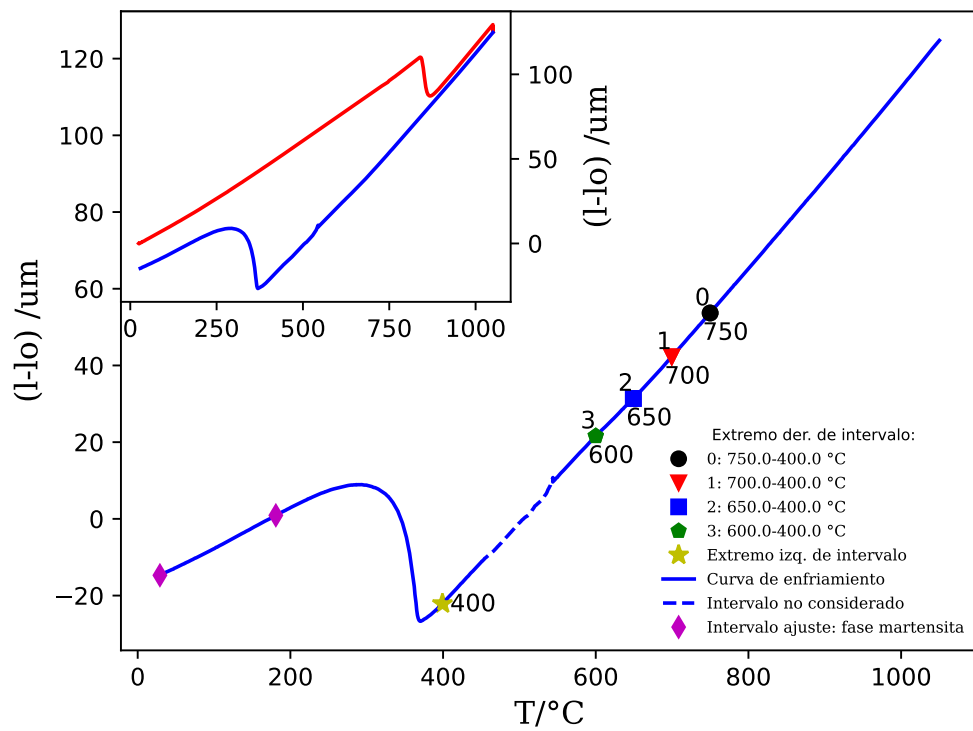


Figura 6.5. Curva dilatométrica de la muestra #586

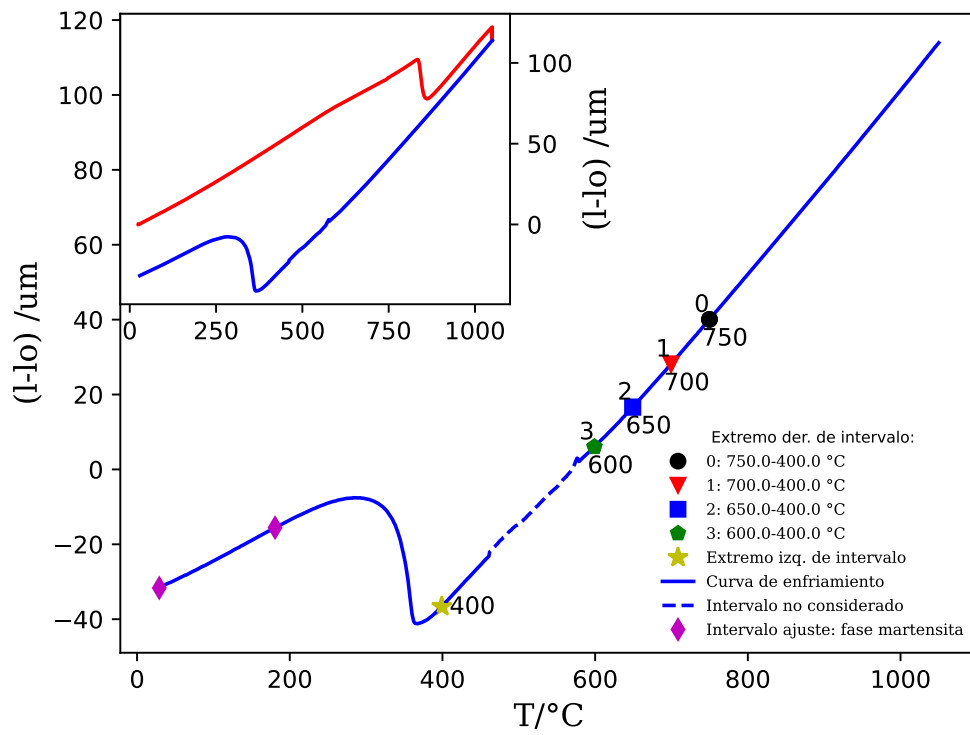


Figura 6.4. Curva dilatométrica de la muestra #584

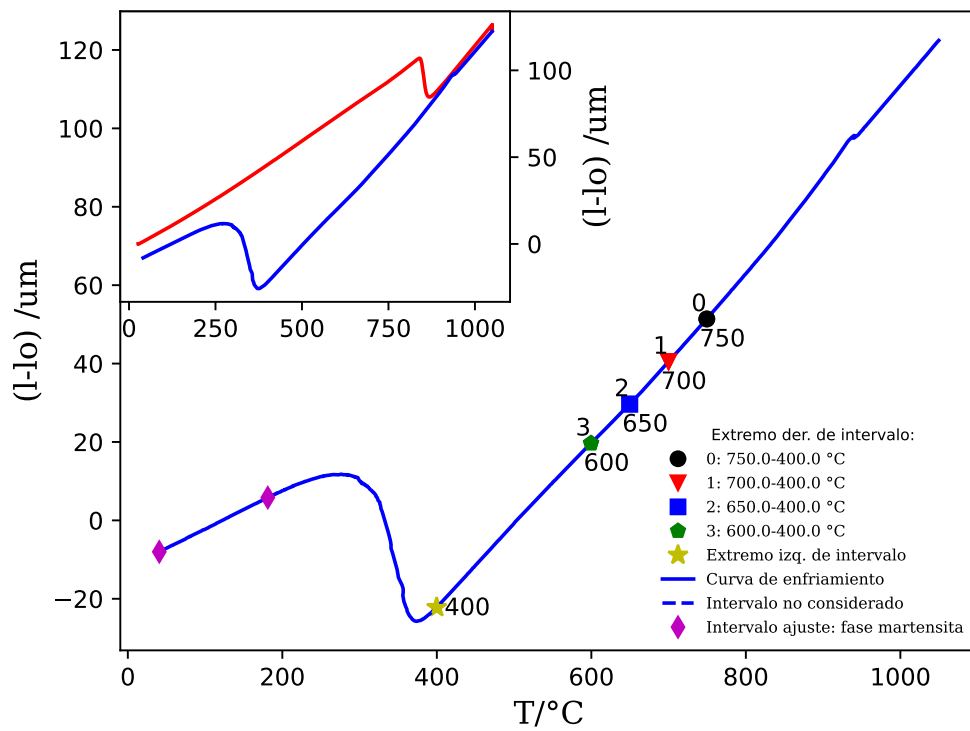


Figura 6.6. Curva dilatométrica de la muestra #587

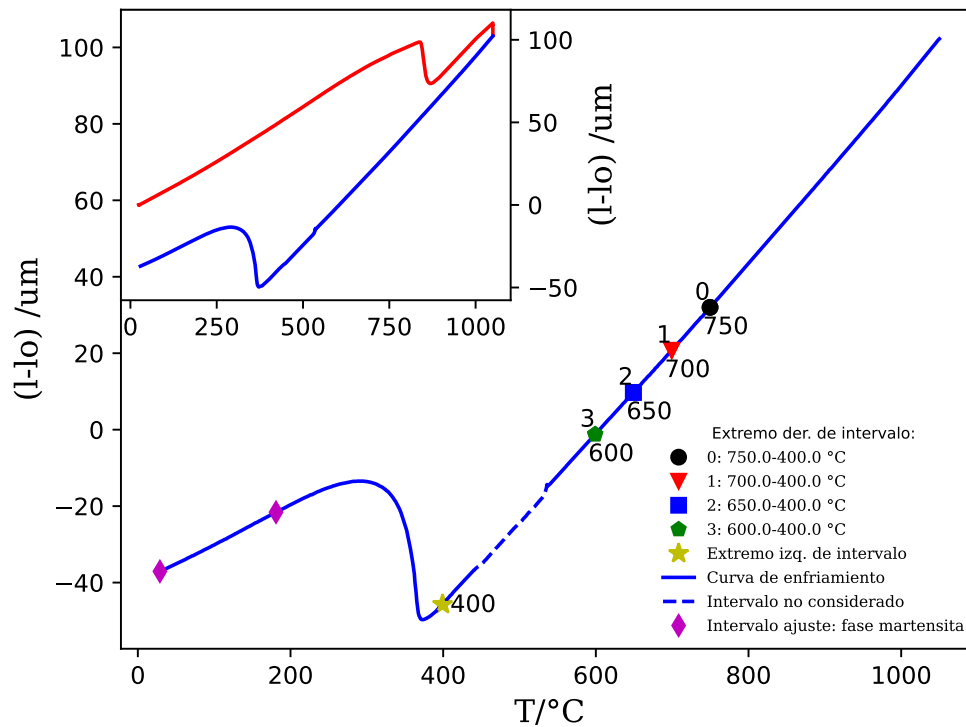


Figura 6.7. Curva dilatométrica de la muestra #590

6.3. Determinación de M_S

6.3.1. Método offset

En la [Tabla 6.1](#) y en la [Figura 6.8](#) se resumen los resultados de M_S obtenidos para todas las muestras analizadas según el método de offset y variando la localización del ajuste como se explicó en la subsección [5.3.1](#).

Como se puede observar, para este método la variación de la localización del ajuste del segmento lineal de la austenita para cada muestra no involucra una gran influencia en la determinación de la M_s , a saber, 3°C en el peor de los casos. Además, no se encontró una relación evidente entre la localización del ajuste y el resultado de la M_S calculada.

Tabla 6.1. Resultados para la M_S (°C) según localización de intervalos utilizando el método de offset.

Intervalo	541	584	586	587	590
0	387.95	377.58	373.65	377.69	377.58
1	387.06	375.72	373.65	375.86	377.58
2	385.91	374.54	376.62	377.69	378.54
3	385.91	374.54	378.70	378.61	378.54

En la [Tabla 6.2](#) y [Figura 6.9](#) se presentan los resultados del análisis de los promedios y desviación estándar de M_S teniendo en cuenta solamente el ciclo térmico 1, enfriamiento a 1°C/s, que involucró las muestras 541, 584, 586, 590. Se observa que los valores para los intervalos donde se consideraron los puntos más cercanos al inicio de la transformación presentaron una menor dispersión.

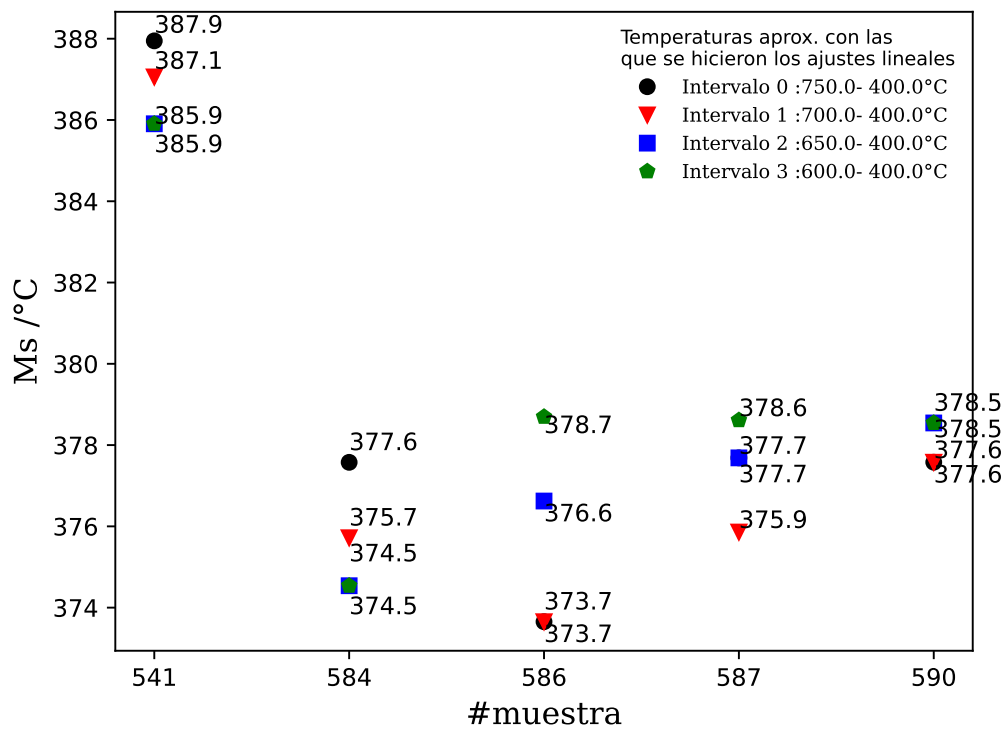


Figura 6.8. Determinación de la M_S para todas las muestras por el método de offset

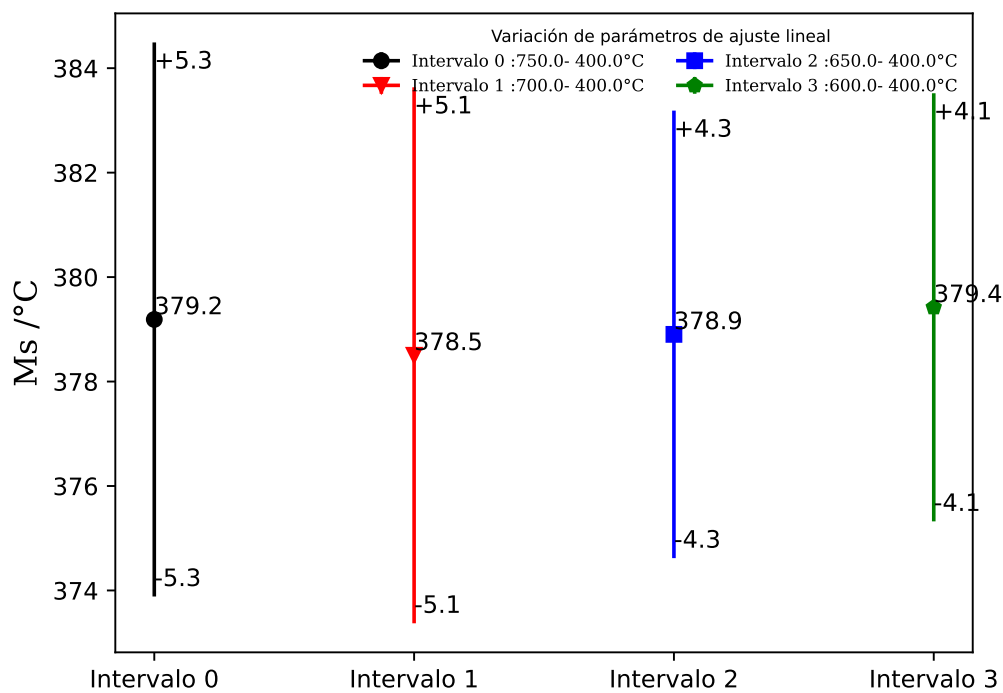


Figura 6.9. Promedio y dispersión de M_S determinada por el método offset, para todas las muestras sometidos al tratamiento térmico 1 según la localización del ajuste

Tabla 6.2. Resultados de la M_S ($^{\circ}\text{C}$) promedio para las muestras #541, 584, 586, 590 según los distintos segmentos considerados para el ajuste con el método offset.

Intervalo	0	1	2	3
M_S (media)	379.2	378.5	378.9	379.4
Desviación estándar	5.3	5.1	4.3	4.1

6.3.2. Método ASTM

En la [Tabla 6.3](#) y en la [Figura 6.10](#) se resumen los resultados de M_S obtenidos para todas las muestras analizadas según el método ASTM y variando la localización del ajuste como se explicó en la subsección [5.3.1](#).

En este método la mayor diferencia encontrada en dichos valores debido a la localización del ajuste fue de 6°C , muestra 586; no se evidenció una clara tendencia del inicio de la transformación martensítica con respecto a la localización del ajuste.

Tabla 6.3. Resultados para M_S ($^{\circ}\text{C}$) según los distintos intervalos de localización y utilizando el método ASTM.

Intervalo	541	584	586	587	590
0	398.2	385.8	381.0	381.5	382.6
1	394.2	380.7	379.7	379.5	382.6
2	392.9	379.6	383.9	381.5	383.7
3	392.9	380.7	386.0	383.4	384.9

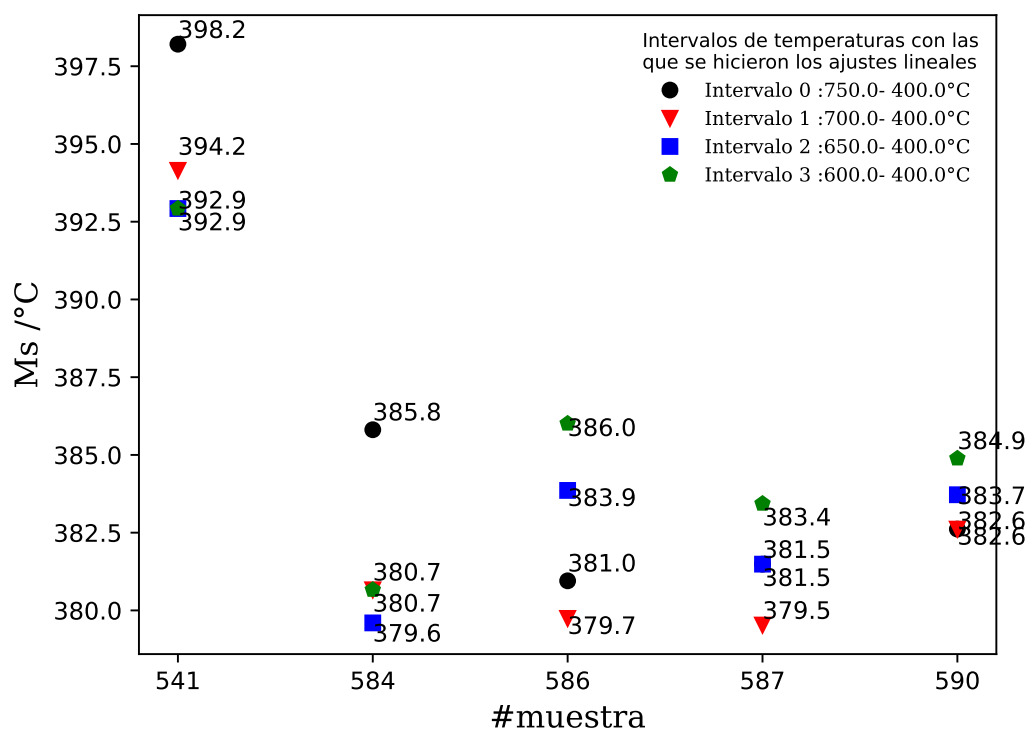


Figura 6.10. M_S determinada para todas las muestras por el método ASTM.

Al igual que lo expuesto para el método de offset, en la [Tabla 6.4](#) y [Figura 6.11](#) se

presentan los resultados del análisis de los promedios y desviación estándar de M_S teniendo en cuenta solamente el ciclo térmico 1. Del mismo modo, se observa que los valores presentaron una menor dispersión para el intervalo donde se consideraron los puntos más cercanos al inicio de la transformación.

Tabla 6.4. Resultados de la M_S ($^{\circ}\text{C}$) promedio para las muestras #541, 584, 586, 590 según los distintos segmentos considerados para el ajuste con el método ASTM.

Intervalo	0	1	2	3
$M_S(\text{media})$	386.9	384.3	385.0	386.1
Desviación estándar	6.8	5.8	4.9	4.4

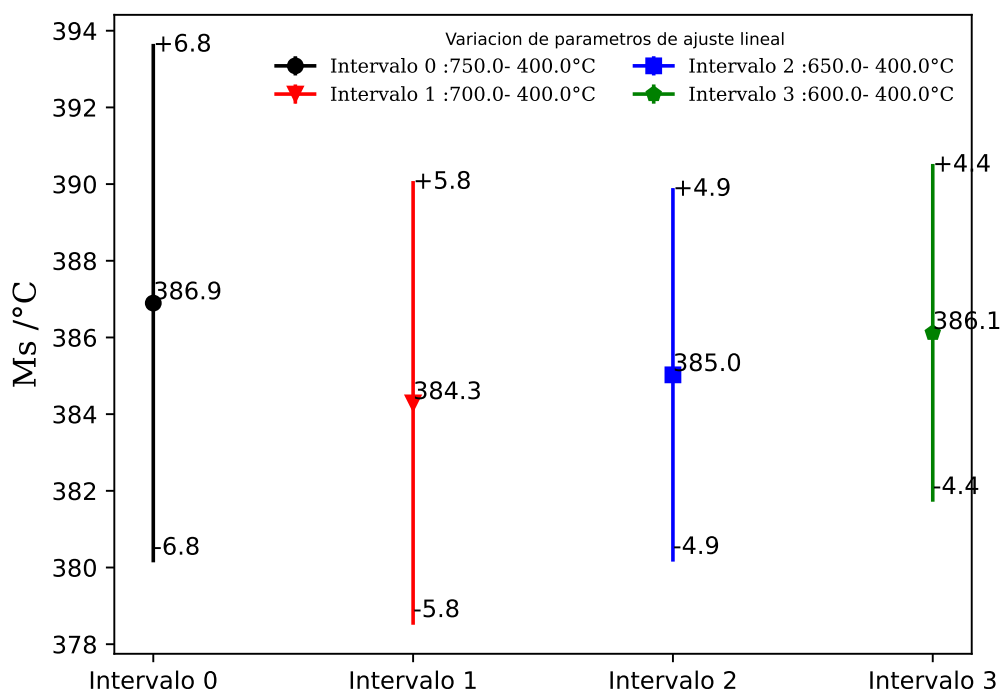


Figura 6.11. Promedio y dispersión de M_S determinada por el método ASTM, para todas las muestras sometidas al tratamiento térmico 1 según la localización del ajuste.

6.3.3. Comparación entre métodos experimentales de determinación de M_S

En la [Figura 6.12](#) y [Tabla 6.5](#) se resumen los valores determinados por todos los métodos utilizados para la determinación de M_S . En el caso de los métodos ASTM y offset se seleccionaron los resultados asociados con el intervalo de menor desviación estándar, es decir, se consideró la localización del ajuste lineal de la austenita más cercana al inicio de la transformación martensítica. Claramente se observa que el método de las rectas tangentes siempre determina una M_S menor, seguido por el offset y por último el ASTM.

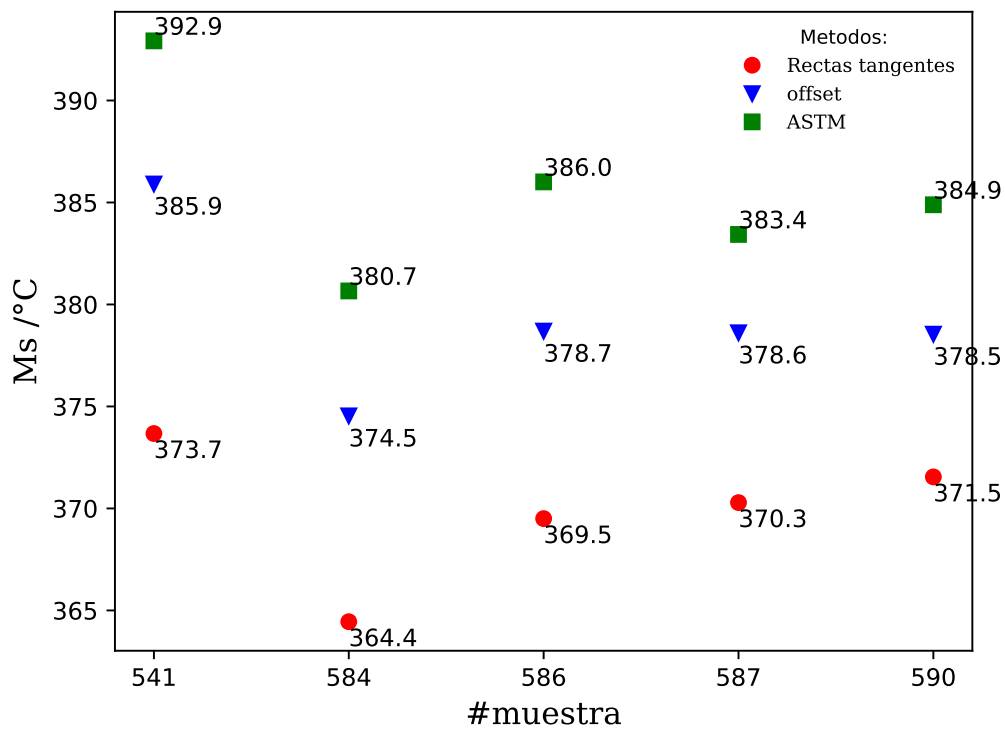


Figura 6.12. M_S (°C) determinada por los distintos métodos de análisis utilizados para todas las muestras.

Tabla 6.5. Resultados de M_S (°C) determinada para todas las muestras por los distintos métodos de análisis.

Método	541	584	586	587	590
ASTM	392.9	380.7	386.0	383.4	384.9
offset	385.9	374.5	378.7	378.6	378.5
Rectas tangentes	373.7	364.4	369.5	370.3	371.5

Los resultados de la [Tabla 6.6](#) y la [Figura 6.13](#) surgen de reunir los obtenidos para las probetas 541, 584, 586 y 590 -que fueron sometidas al mismo ciclo térmico- y calcular tanto el promedio de las M_S como la desviación estándar. De esta forma se evidencia con más claridad lo mencionado anteriormente: el método de rectas tangentes estima una M_S menor, seguido del método offset y por último el método ASTM.

Tabla 6.6. Promedio y dispersión de M_S determinada por distintos métodos de análisis, para todas las muestras sometidas al tratamiento térmico 1.

Intervalo	Rectas tangentes	offset	ASTM
M_S (media)	369.8	379.4	386.1
Desviación estándar	3.4	4.1	4.4

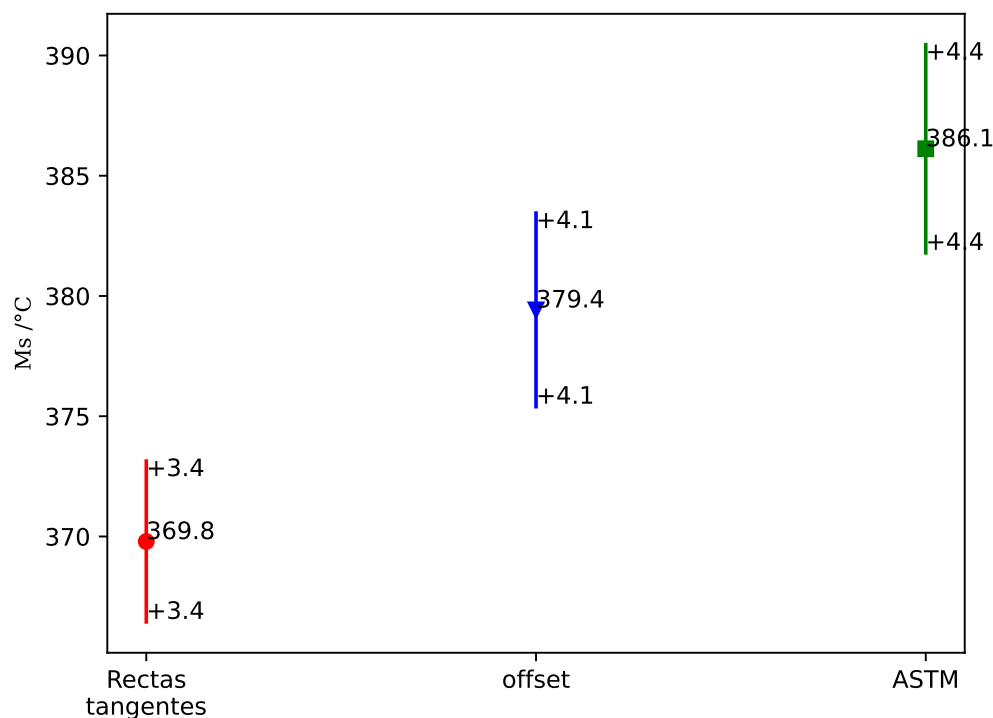


Figura 6.13. Promedio y dispersión de M_s determinada por distintos métodos de análisis, para todas las muestras sometidas al tratamiento térmico 1.

6.4. Modelos aplicados a la determinación de fracción martensítica transformada

En esta sección se presentan los resultados obtenidos a través de los ajustes descritos en el Capítulo 4, sección 4.2; los parámetros fueron ajustados a partir de la aplicación de los algoritmos que figuran en el Anexo. Todos los modelos aplicados tienen como parámetro fundamental a la temperatura M_s ; esta variable, como se detalló en la sección 6.3, puede obtenerse del análisis de las curvas dilatómetricas por alguno de los métodos mencionados o bien utilizando un modelo empírico basado en la composición química del acero. Haciendo uso de estas dos vías, se detallan los resultados obtenidos en las subsecciones siguientes.

6.4.1. Ajustes de parámetros en los modelos de fracción transformada utilizando la M_s obtenida a partir de ensayos dilatómetricos.

Análisis de muestras sometidas al tratamiento térmico 1 (#541, 584, 586, 590)

En primera instancia se promediaron los valores de fracción transformada en función de la temperatura obtenidos para todas las muestras utilizando la regla de la palanca. De esta forma fue posible lograr una única curva para el tratamiento térmico 1 con una desviación estándar experimental asociada. Esta construcción se representa en las Figura 6.14, Figura 6.15 y Figura 6.16, con una línea violeta de trazo continuo y un fondo ensombrecido para la curva promediada y para la dispersión respectivamente. Para el

ajuste de esta curva se ensayaron los modelos de Koistinen y Marburger, Lee-van Tyne y Huyan y colaboradores, utilizando los valores de M_S determinados en las subsecciones 6.3.1, 6.3.2 y 6.3.3 a partir de las curvas dilatométricas por los métodos de rectas tangentes, offset y ASTM respectivamente. En la Tabla 6.7 se resumen los valores de los parámetros obtenidos para el tratamiento térmico 1.

En la Figura 6.14 se observa el ajuste de los modelos utilizando el valor de M_S definido por el método de las rectas tangentes. Para valores menores a esta temperatura los modelos de Koistinen y Marburger y Lee y van Tyne reproducen con muy buena aproximación la curva obtenida a partir de los datos experimentales, sin embargo, por encima de esta temperatura el modelo de Koistinen y Marburger se aleja completamente del curso de la transformación determinado a partir de la curva dilatométrica, dado el carácter exponencial (en particular, no sigmoideal) del mismo.

Por su parte, el modelo de Huyan y colaboradores sigue el comportamiento sigmoideo que exhibe la fracción transformada obtenida por la regla de la palanca, pero se observa un alejamiento por exceso para valores menores al 30 % de fracción transformada y por defecto para valores entre 30 y 80 % de dicha fracción.

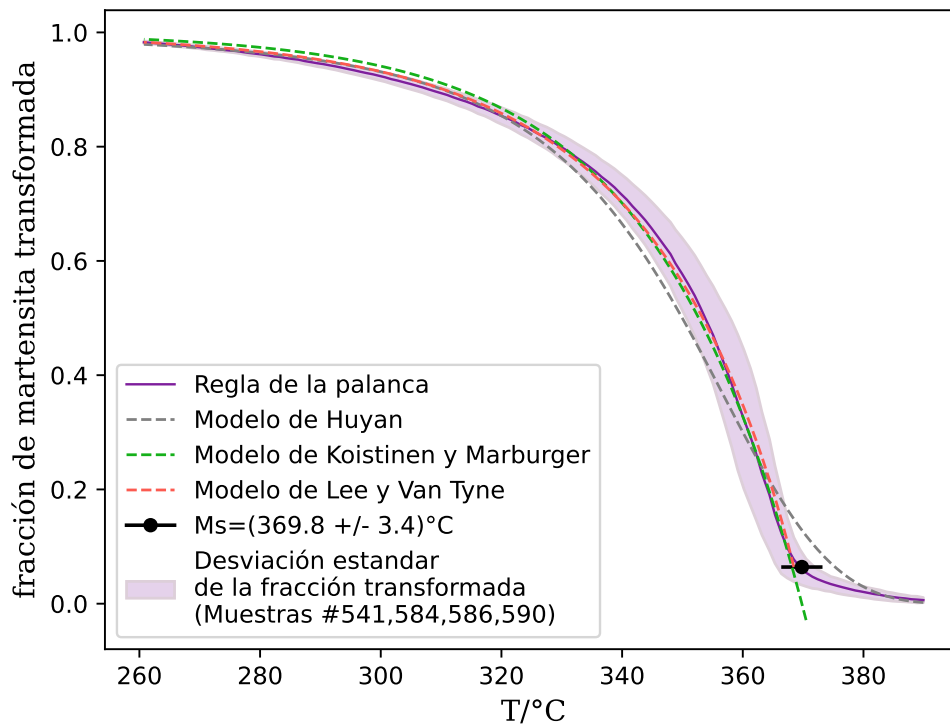
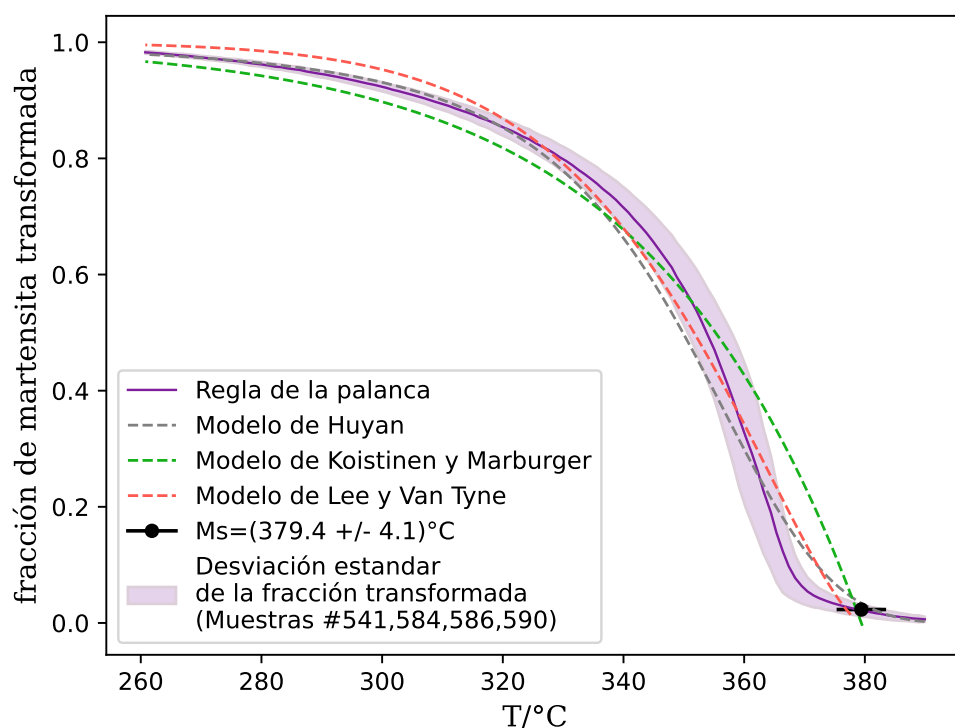


Figura 6.14. Modelos de ajuste a la curva de fracción transformada para M_S definida por el método de las rectas tangentes.

En la Figura 6.15 se observa el ajuste de los modelos utilizando el valor de M_S obtenido por el método de offset. Al igual que en el caso anterior el modelo de Koistinen y Marburger dista mucho de reproducir correctamente los datos experimentales. En cambio, cuando se comparan los modelos de comportamiento sigmoideo se observa que el de Huyan y colaboradores presentó un ajuste más cercano, sobre todo para muy bajas (menores que 0,05) y muy altas (por encima de 0,85) fracciones de martensita transformada.

Tabla 6.7. Resumen de los parámetros determinados para la media de fracción obtenida para las muestras con tratamiento térmico 1.

Modelo	Parámetros ajustados	Rectas tangente	Offset	ASTM
Koistinen y Marburguer	α	0.0401	0.0287	0.0236
Lee y van Tyne	α	0.0508	0.006	0.0013
	β	0.9343	1.4110	1.7628
Huyan	a	-4.5352	-4.3910	-4.2955
	b	133.8064	91.0646	62.8906

**Figura 6.15.** Modelos de ajuste a la curva de fracción transformada para M_S definida por el método offset.

Por último, en la [Figura 6.16](#) se representa el ajuste de la fracción transformada y su desviación estándar asociada, teniendo en cuenta el valor de M_S calculado por el método ASTM. Teniendo presente que la M_S se determina para un valor muy bajo de fracción transformada se puede observar que las curvas ajustadas son similares a las obtenidas a través del método offset. A pesar de esto el modelo de Huyan y colaboradores parece ajustar con mayor cercanía el “inicio suave” percibido en el comienzo de la transformación.

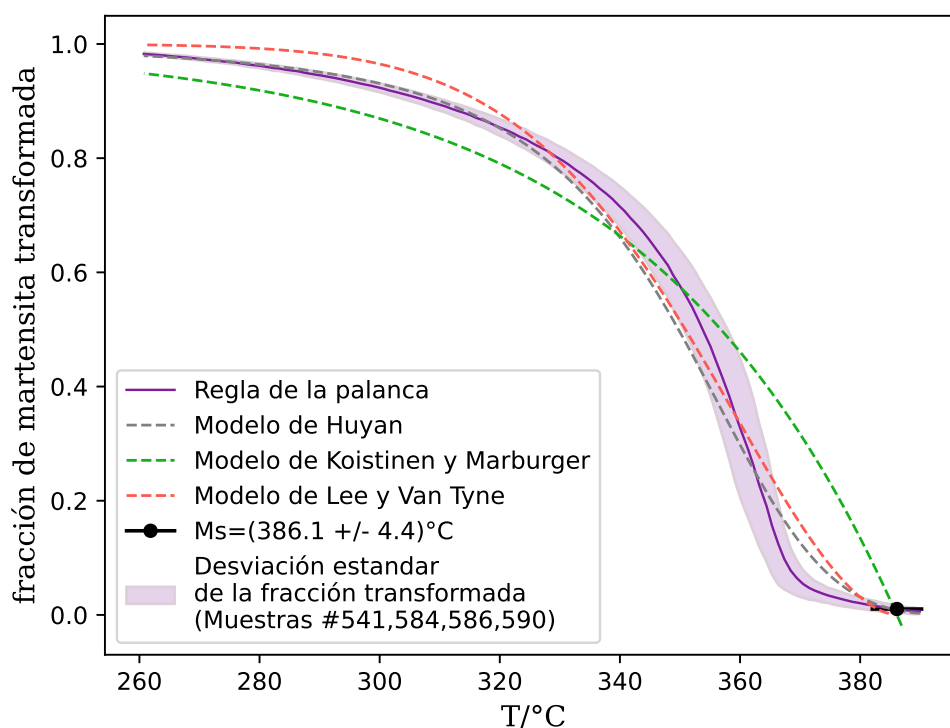


Figura 6.16. Modelos de ajuste a la curva de fracción transformada para M_S definida por el método ASTM

Análisis de muestras sometidas al tratamiento térmico 2 (#587)

En este caso sólo se reprodujo el ciclo térmico en una muestra por lo cual en las [Figura 6.17](#), [Figura 6.18](#) y [Figura 6.19](#) no aparece la parte sombreada que se mostró en los resultados de la subsección anterior; la curva que representa los resultados experimentales de fracción transformada fue la calculada cuando se aplica la regla de la palanca a los datos dilatométricos de la muestra 587. Siguiendo la metodología aplicada para el tratamiento térmico 1, se realizaron los ajustes de los modelos de Koistinen y Marburger, Lee y van Tyne y Huyan y colaboradores usando los valores de M_S obtenidos por los métodos de las rectas tangentes, offset y ASTM. En la [Tabla 6.8](#) se resumen los resultados del ajuste de los parámetros para los modelos mencionados.

Tabla 6.8. Resumen de los parámetros determinados para el ajuste de la curva de fracción transformada obtenida para la muestra #587.

Modelo	Parámetros ajustados	Rectas tangente	Offset	ASTM
Koistinen y Marburguer	α	0.0275	0.0219	0.0194
Lee y van Tyne	α	0.0091	0.0017	0.0006
	β	1.2935	1.6579	1.8794
Huyan	a	-4.1052	-3.9990	-3.9398
	b	106.7480	73.5350	55.0822

En las [Figura 6.17](#), [Figura 6.18](#) y [Figura 6.19](#) se observa claramente que el modelo de Koistinen y Marburger provee un ajuste de menor bondad de los datos experimentales; por el contrario, el modelo de Huyan y colaboradores reproduce satisfactoriamente el

comportamiento de la fracción transformada. También se observa que el modelo de Lee y van Tyne tiene una mayor dependencia de la M_S , es decir que cuanto mayor es el valor de esta temperatura mejor ajusta el comportamiento en todo el rango de la transformación.

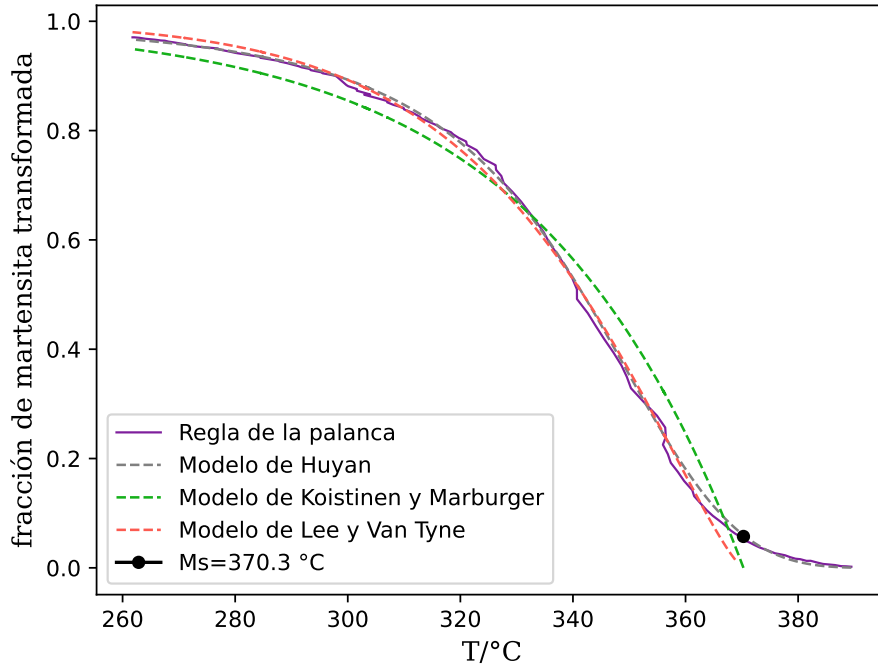


Figura 6.17. Modelos de ajuste de la curva de fracción transformada, para M_S definida por el método de las rectas tangentes y para el tratamiento térmico 2.

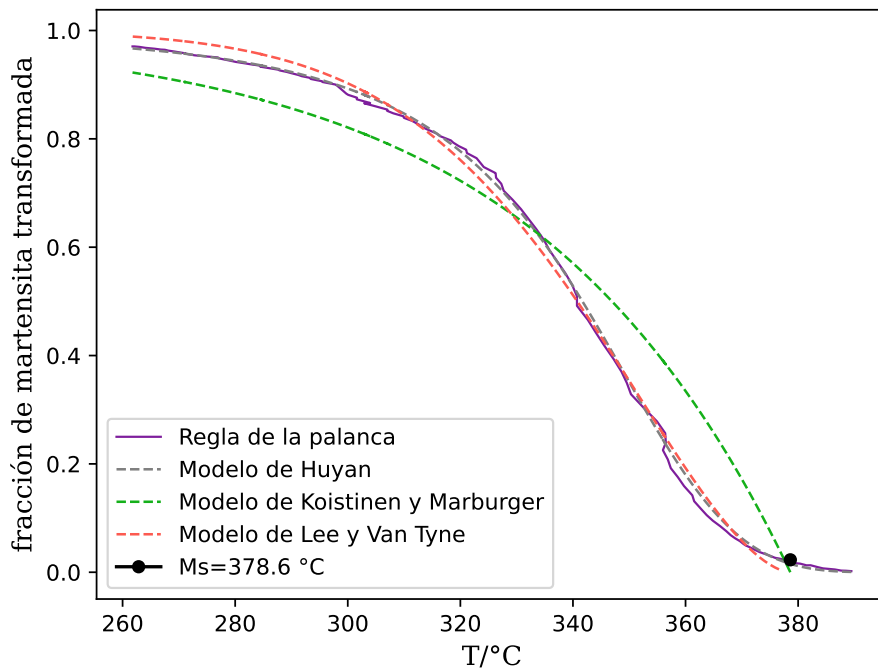


Figura 6.18. Modelos de ajuste de la curva de fracción transformada, para M_S definida por el método de offset y para el tratamiento térmico 2.

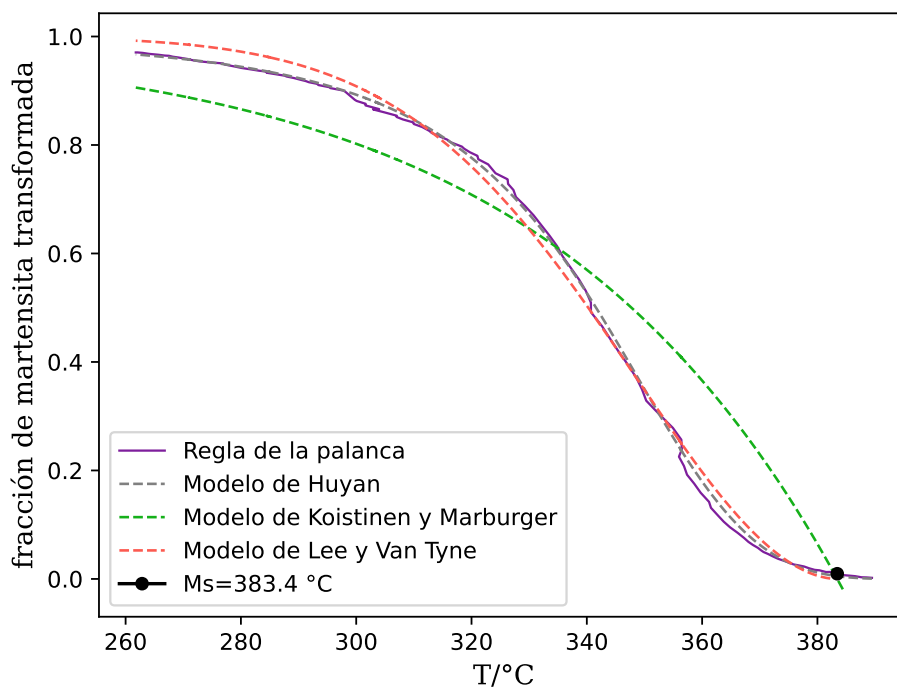


Figura 6.19. Modelos de ajuste de la curva de fracción transformada, para M_S definida por el método ASTM y para el tratamiento térmico 2.

6.4.2. Ajustes de parámetros en los modelos de fracción transformada utilizando la composición química del acero

Cálculo de la M_S a partir de la composición química del acero y ajuste numérico de parámetros de modelo

En el caso del ajuste a partir de la M_S calculada por el método de Finkler y Schirra, se identificó esta temperatura como la de inicio de transformación martensítica y se ajustaron numéricamente los parámetros de los modelos de Koistinen y Marburger, Lee-van Tyne y Huyan. Los valores de los parámetros de ajuste, y el comportamiento de la curva que se exhiben en la [Tabla 6.9](#) y en la [Figura 6.20](#) respectivamente, son similares a los obtenidos a partir del cálculo experimental de la M_S por el método ASTM, dado que el valor de M_S determinado en este último caso es muy próximo al calculado por Finkler y Schirra.

Tabla 6.9. Resumen de parámetros calculados

Finkler y Schirra		$M_S(^{\circ}\text{C})$	384.1
Modelos Ajustados	Koistinen y Marburger	α	0.025
		β	1.658
	Lee y van Tyne	a	-4.263
		b	76.011

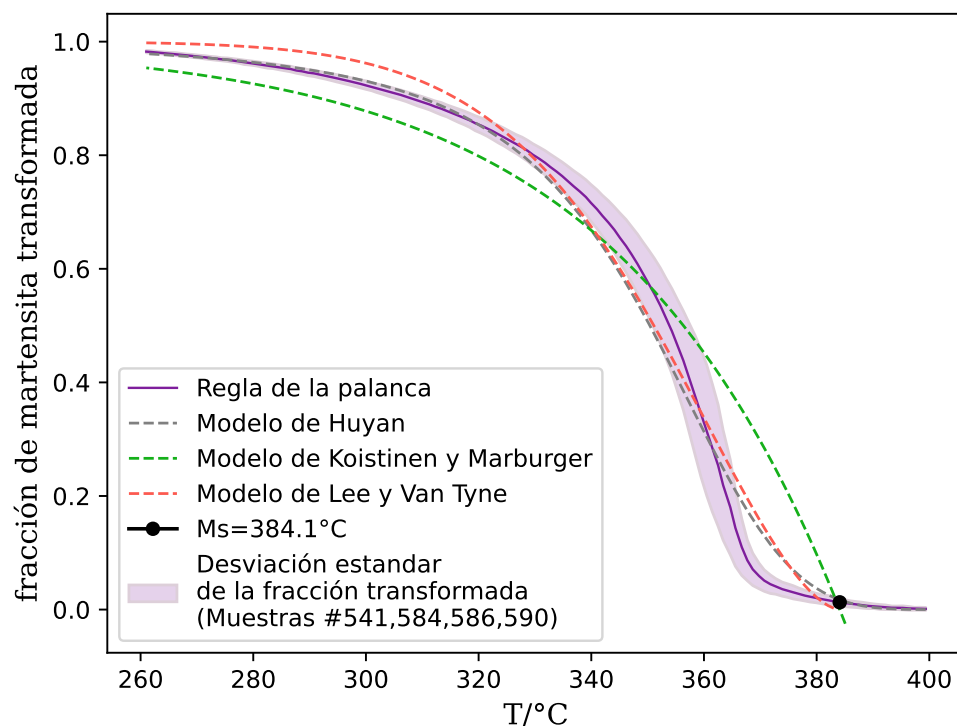


Figura 6.20. Modelos de ajuste aplicados a la curva de fracción transformada para el tratamiento térmico 1 utilizando el valor de M_S calculado por Finkler y Schirra

Cálculo de M_S y de parámetros de modelo a partir de la composición química del acero

De acuerdo a lo mencionado en el Capítulo 4, para ciertos modelos los autores han encontrado una dependencia de los valores de M_S y de los parámetros característicos con la composición química. Así, teniendo en cuenta solamente la composición química del acero ASTM A335 P91, se calcularon los valores de M_S y de los parámetros en los casos correspondientes a los modelos de van Bohemen, van Bohemen y Sietsma y Lee y van Tyne. Los resultados se observan en la [Tabla 6.10](#) y en la [Figura 6.21](#) y [Figura 6.22](#).

Tabla 6.10. Resumen de los parámetros calculados durante el ajuste de la fracción transformada a martensita para los modelos de van Bohemen y Sietsma, van Bohemen y Lee-van Tyne.

Modelo fn. de composición	Magnitudes definidas	Resultados
van Bohemen y Sietsma	M_S (°C)	287.7
	α	0.020
van Bohemen	M_S (°C)	398.8
	α	0.023
Lee y van Tyne	M_S (°C)	369.7
	α	0.067
	β	0.971

En la [Figura 6.21](#) se observa claramente que los modelos de van Bohemen y van Bohemen

y Sietsma no producen un resultado satisfactorio al intentar reproducir la transformación martensítica de un acero ASTM A335 P91, dado que tanto la M_S como el comportamiento de la curva distan mucho de los resultados experimentales obtenidos.

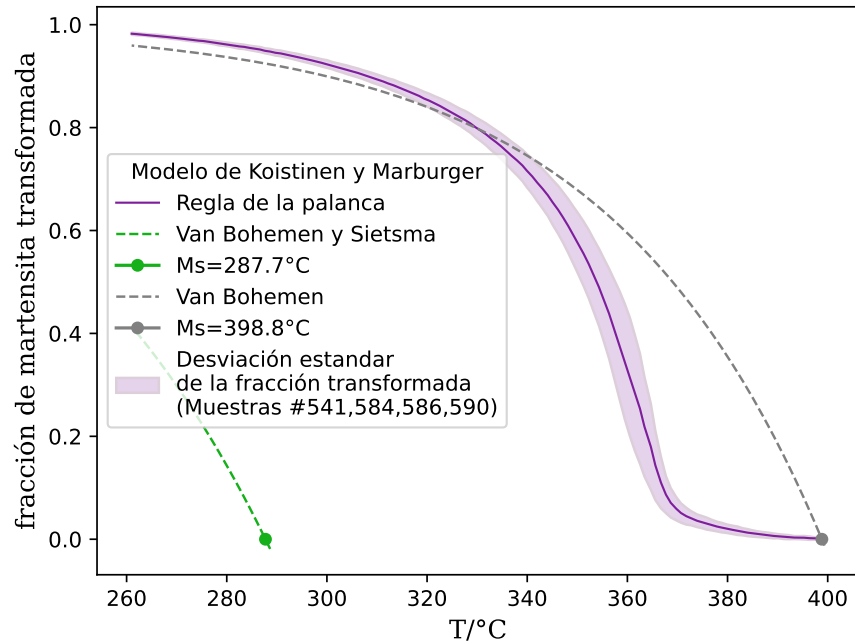


Figura 6.21. Modelos de van Bohemen y van Bohemen - Sietsma aplicados a la curva experimental de fracción transformada obtenida por la regla de la palanca para el tratamiento térmico 1.

En la [Figura 6.22](#) se observa que la curva de fracción transformada a martensita para el modelo de Lee y van Tyne se aleja aún más de los valores experimentales a medida que se progresa a mayores valores de fracción transformada. Cabe señalar que a pesar de que la M_S determinada con la ecuación 4.11 -aplicable a este modelo- es similar a la obtenida por el método de las rectas tangentes no ocurre lo mismo con los parámetros α y β , lo que resulta en curvas de fracción transformada diferentes para ambos casos.

En la [Figura 6.22](#) se observa que la curva de fracción transformada de martensita para el modelo de Lee y van Tyne, se aleja aun más de los valores experimentales a medida que esta avanza para mayores fracciones de transformación. Cabe señalar que a pesar de que la M_S determinada con la ecuación 4.11 -aplicable a este modelo- es similar a la obtenida por el método de las rectas tangentes, no así los parámetros α y β resultando en curvas de fracción transformada diferentes para ambos casos.

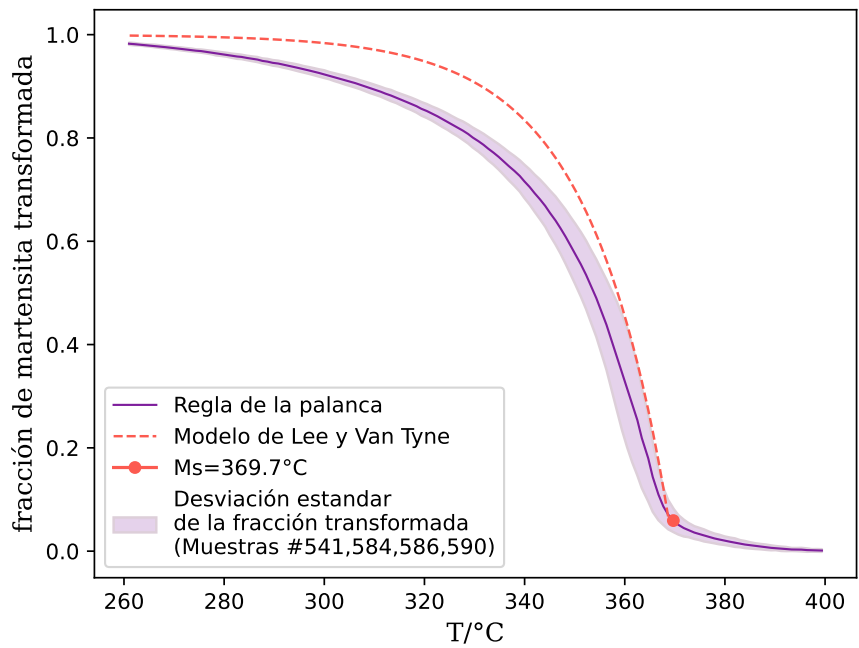


Figura 6.22. Modelo de Lee y van Tyne aplicados a la curva experimental de fracción transformada obtenida por la regla de la palanca para el tratamiento térmico 1.

6.5. Comparación entre determinaciones experimentales y ecuaciones empíricas

En la [Tabla 6.11](#) se muestran los resultados obtenidos para M_S utilizando los métodos experimentales y los basados en ecuaciones empíricas, para los ciclos térmicos 1 y 2. El valor calculado a partir de la expresión de van Bohemen y Sietsma resulta significativamente alejado del resto, probablemente debido a haber sido formulada dicha expresión para aceros de bajo carbono y baja aleación. Por otra parte, se observa que el valor de M_S calculado a través de la expresión de Finkler y Schirra posee un buen acuerdo con los determinados por los métodos experimentales de offset y ASTM.

Tabla 6.11. Resultados de la M_S (°C) obtenida de forma experimental y mediante la aplicación de ecuaciones empíricas.

Métodos experimentales			Modelos de ecuaciones empíricas	
Métodos	Ciclo térmico 1	Ciclo térmico 2		
Rectas tangentes	369.8	370.3	384.1	Finkler y Schirra
			287.7	van Bohemen y Sietsma
offset	379.4	378.6	398.8	van Bohemen
ASTM	386.1	383.4	369.7	Lee y van Tyne

7 Conclusiones

En el presente trabajo se realizó un análisis exploratorio, y un posterior ajuste con distintos modelos metalúrgicos, de datos experimentales concernientes a la transformación austenita- martensita en un acero ASTM 335 grado P91. Dichos datos habían sido obtenidos en el Laboratorio de Técnicas Dinámicas de la Gerencia Materiales de CNEA usando la técnica de dilatometría, con anterioridad al inicio del seminario.

Se realizó una búsqueda bibliográfica detallada relativa al problema de la determinación de la temperatura de inicio de la transformación martensítica a partir de una curva dilatométrica y a los modelos que dan cuenta de la evolución de la fracción transformada a martensita durante un enfriamiento continuo para distintos tipos de aceros. Luego de reunir la información pertinente se estudiaron ambos problemas, aplicando los métodos propuestos en la literatura al caso específico del acero propuesto. Las principales conclusiones extraídas son:

1. Se desarrollaron rutinas de análisis de datos dilatométricos en el lenguaje de programación Python, que permitieron la determinación de:
 - a) temperaturas de inicio de la transformación martensítica por los métodos de rectas tangentes, ASTM y offset
 - b) fracciones transformadas de fase en experimentos de enfriamiento continuo y bajo condiciones compatibles con la aplicación de la regla de la palanca
 - c) ajustes de los modelos de Koistinen y Marburger, van Bohemen, van Bohemen y Sietsma, Lee y Van Tyne y Huyan a las curvas de fracción transformada en función de la temperatura.
2. Se contrastaron los resultados para M_S obtenidos por análisis de las curvas dilatométricas con los determinados a partir de fórmulas semiempíricas que tienen en cuenta la composición química del acero. La expresión desarrollada por Finkler y Schirra para aceros de alto contenido en Cr arrojó valores que presentan un acuerdo satisfactorio con los extraídos de las curvas experimentales.
3. Se ajustaron las curvas experimentales de fracción transformada a martensita en función de la temperatura por medio de los modelos de Koistinen y Marburger, Huyan y col. y Lee y Van Tyne usando una rutina de regresión no lineal sobre los parámetros característicos, para los valores de M_S obtenidos a partir de las curvas dilatométricas por los métodos de rectas tangentes, offset y ASTM. También se utilizó el valor de M_S calculado con el modelo semiempírico de Finkler y Schirra. A partir de ambas metodologías se corroboró que los modelos que proponen una expresión de tipo sigmoidal para dicha fracción transformada observan un mejor ajuste global de las curvas que los modelos de tipo exponencial, dado que permiten reproducir la etapa inicial de la transformación con una bondad de ajuste superior.
4. Usando los modelos semiempíricos de determinación de la temperatura M_S y de fracción transformada de van Bohemen y Sietsma, van Bohemen y Lee y van Tyne, se graficó junto a la curva experimental la fracción transformada a martensita en

función de la temperatura en cada caso. A diferencia de los modelos en los cuales se utilizó el valor de M_S calculado experimentalmente y se ajustaron numéricamente los parámetros característicos, los modelos semiempíricos no mostraron un comportamiento satisfactorio frente a las curvas experimentales de fracción transformada. Una de las posibles razones para este resultado es que los parámetros característicos propuestos por los autores - que dependen de la composición química - no contemplan aceros con un tenor de Cr como el estudiado en el presente trabajo.

5. En el caso del ajuste a través de la expresión de Huyan y colaboradores, la aproximación realizada para evaluar la dependencia de la cantidad ΔG con la temperatura -esto es, la aproximación lineal- se reveló como aceptable, aunque un refinamiento posterior del ajuste ameritaría investigar con más detalle dicha dependencia.
6. Si bien no se contó con información experimental exhaustiva, la comparación preliminar entre los resultados para enfriamientos producidos a 1 °C/s y 10 °C/s no arrojó diferencias significativas en cuanto a los valores determinados para M_S .

8 Anexo

8.1. Macros de Thermocalc

```
@@Florescia A Correa
@@Thermocal_fases_composicion.txt
@@Thermocal: Definicion de fases y composicion
SYS:go p-3
POLY_3:def-mat
Database /TCFE5/: enter
Major element or alloy:fe
Composition input in mass (weight) percent? /Y/: enter
1st alloying element:c .1 cr 8.37 cu .16
Next alloying element:mn .36 mo .88 n .0605
Next alloying element:nb .075 ni .15 si .24
Next alloying element:v .211
Next alloying element: enter
Temperature (C) /1000/: 1050
Reject phase(s) /NONE/: enter
Restore phase(s): /NONE/: enter
OK? /Y/: Y
Should any phase have a miscibility gap check? /N/: N
POLY_3:l-e
OUTPUT TO SCREEN OR FILE /SCREEN/: enter
Options /VWCS/: VWCS
```

8.2. Determinación de M_S por método ASTM

```

1  """Florenxia Antonella Correa
2  ASTM.py
3  Se trabaja solo con el metodo ASTM en la determinacion de Ms
4  Se calcularon los parametros de ajuste de KM
5  Se grafican todos los resultados obtenidos segun la variacion del
   ↪ intervalo
6  Se grafica la dispersión asociada a cada una de las Ms determinadas
   ↪ segun
7  las muestras# 541, 584, 586 y 590
8  """
9  import numpy as np
10 import matplotlib.pyplot as plt
11 import operator
12
13 from Especifico_funciones import *
14 from scipy.optimize import curve_fit
15 from Especifico_funciones import f_KM
16 from pprint import pprint
17 from matplotlib.legend_handler import HandlerLine2D, HandlerTuple
18 from matplotlib import pyplot as plt
19 from matplotlib.legend import _get_legend_handles_labels
20 """Directorio o root:
21 dir_M: carpeta de transformacion martensitica
22 dir_python: donde se encuentran los ejecutables
23 """
24 dir_M='../Datos dilatometria/Transformacion_martensitica/'
25 dir_python='C:/Users/flor/Documents/ITS/Trabajo Final/Python'
26 dir_TC='../Ejemplos Thermocalc/'
27 carpeta='../Informe1/ASTM/'
28 # Listas de archivos:
29 #-----
30
31 nombres_M=['541-besoky-1Cseg-1050C-30min-1Cseg-12.12.17',
32 '584-besoky-1Cseg-1050C-30min-1Cseg-02.21.19',
33 '586-besoky-1Cseg-1050C-30min-1Cseg-02.28.19',
34 '587-besoky-1Cseg-1050C-30min-10Cseg-02.28.19',
35 '590-besoky-1Cseg-1050C-30min-1Cseg-04.16.19']
36 numeros=['541', '584', '586', '587', '590'] #---Saque el 564 que da
   ↪ puros problemas
37 """Tbusqueda: matriz de temperaturas características para el
   ↪ procesamiento, durante el enfriamiento
38 Columnas:
39 0- comienzo de enfriamiento
40 1- cercano al comienzo de la transformacion
41 2- punto durante la transformacion de mayor temperatura
42 3- punto durante la transformacion de menor temperatura
43 Filas corresponde a cada muestra
44 """

```

```

45 Tbusqueda={'541':[1050, 400,370, 350],
46           '584':[1050, 400, 360, 345],
47           '586':[1050, 400, 365, 350],
48           '587':[1050, 400, 350, 320],
49           '590':[1050, 400, 365, 355]}
50 exceptuar={'541':[530,420],
51           '584':[580,455],
52           '586':[550,450],
53           '587':[[],[]],
54           '590':[540,440]}
55
56 Tratamiento_1=['541', '584', '586', '589']
57 # Parametros para datos y guardado:
58 #-----
59 #-----
60 # Generales:
61 n=len(nombres_M)
62 extra=2
63 #-----
64 # GENERAL
65 #-----
66 # Parametros de graficos
67 Colores=['k', 'r', 'b', 'g', 'c', 'y', 'm', 'w']
68 Simbolos=['_', '-', '._', ':', '.', ',', 'o', 'v', 's', 'p', '+', '*',
69           '^', '<', '>', 'x', 'D', 'd']
70 posicion=['upper', 'lower', 'lower', 'lower', 'lower', 'lower']
71 # Formato y titulos
72 Titulo='Comparacion de metodos'
73 Titulosup='Ajuste K-M y definicion de Ms'
74 Formato_titulo={'family':'serif','color':'black','size':12}# 'blue', 'darkred'
75 Formato_titulosup={'family':'serif','color':'black','size':15}
76 Formato_ejes={'family':'serif','color':'black','size':13}
77 Formato_label={'family':'serif','size':9}
78 # Otros parametros
79 epsilon=1.
80 beta0=0.011
81 Ms=[]
82 Metodos=['Tg', 'ASTM', 'offset']
83
84 # Parametros ASTM:
85 m=4# aumento para mayor localizacion del ajuste ASTM
86 mas_ASTM=[]
87 epsilon_ASTM=.01
88 Ms_ASTM=np.zeros(n)
89 par_ASTM=[]
90 dato_1=np.zeros(m)
91 # Fraccion martensitica transformada determinada por metodo ASTM'
92 Titulo_ASTM='Variacion de parametros en ajuste lineal'
93 Titulosup_ASTM='Ajuste K-M y definicion de Ms por metodo ASTM'
94 aux_enc=[]

```

```

94 var_ASTM=[]
95 p_ASTM=[]
96 #-----
97 # COMPOSICIONES
98 #-----
99 Elementos=['C','CU','AL','N','SB','MN','SN','TI','B','PB',
    ↪ 'SI','NI','NB','CA','P','CR','V','O','S','MO','CO','AS',
    ↪ 'FE']
100 Contenido_acero=np.array([0.1,0.16, 0.009, 0.0605,0.001,0.36, 0.008,
    ↪ 0.004, 0.0001, 0.001, 0.24, 0.15, 0.075, 0.0008, 0.014, 8.37,
    ↪ 0.211, .0017, .001, .88, .01, .005, 88.3379 ])
101 FMas_nominal=dict(zip(Elementos, list(Contenido_acero/100.)))
102 FMol_nominal=FraccionMolar(FMas_nominal)
103 PMas_nominal=dict(zip(Elementos, list(Contenido_acero)))
104 # composiciones en equilibrio T=25, 1050°C respectivamente
105 #-----
106 FMas_gama=comp_fase(dir_TC+'f_gamabis.dat', 'FCC_A1#1')
107 FMol_gama=FraccionMolar(FMas_gama[0])
108
109 #-----
110 # CALCULOS PARA METODO ASTM:
111 #-----
112 m=4
113 T_derecha=linspace(750, 600, m)
114 dato_derecho=[0, 0, 0, 0]
115 extra=2
116 aux=[0,0,0,0]
117 ex=[0,0]
118 for i, nombre in enumerate(nombres_M):
119     # extraccion archivo
120     #-----
121     archivo=dir_M+nombre+'.asc'
122     tiempo, deformacion, temperatura=extraccion(archivo)
123     # Puntos relevantes:
124     aux[0]=buscador(Tbusqueda[numeros[i]][0], temperatura, epsilon)
125     aux[1]=buscador(Tbusqueda[numeros[i]][1], temperatura, epsilon)
126     aux[2]=buscador(Tbusqueda[numeros[i]][2], temperatura, epsilon)
127     aux[3]=buscador(Tbusqueda[numeros[i]][3], temperatura, epsilon)
128     # Puntos a exceptuar
129     ex[0]=buscador(exceptuar[numeros[i]][0], temperatura, epsilon)
130     ex[1]=buscador(exceptuar[numeros[i]][1], temperatura, epsilon)
131     if numeros[i] in ['587']:
132         ex[0]=aux[1]
133         ex[1]=aux[1]
134     print(numeros[i])
135     # Algunas variables para guardar datos
136     #-----
137     aux_enc.append(aux)
138     aux_Ms=[]
139     for k, T_der in enumerate(T_derecha):

```

```

140 # Calculo de datos experimental: Regla de la palanca
141 #-----
142     # busco la posicion de la temperatura que define el extremo
143     ↪ derecho del
144     # intervalo
145     j=buscador(T_der, temperatura, epsilon)
146     # se realiza el calculo para los extremos del intervalo
147     ↪ definido y exceptuando
148     # ex
149     dato_1, Ms_1=Met_ASTM(temperatura, deformacion, j, aux[1],
150     ↪ aux[2], aux[3], epsilon_ASTM,ex)
151     # Se define un intervalo de temperatura con dos puntos antes
152     ↪ de donde se
153     # encuentro la Ms
154     T=temperatura[dato_1-extra:-100]
155     # Se define un intervalo de deformacion con dos puntos antes
156     ↪ de donde se
157     # encuentro la Ms
158     defor=deformacion[dato_1-extra:-100]
159     # Se define un intervalo entre los extremos derecho del
160     ↪ intervalo de ajuste y de
161     # puntos a exceptuar
162     intervalo1=np.arange(j,ex[0])
163     # Se define un intervalo entre los extremos izquierdo del
164     ↪ intervalo de ajuste y
165     # de puntos a exceptuar
166     intervalo2=np.arange(ex[1],aux[1])
167     #Se define un intervalo
168     # entre los extremos izquierdo del intervalo de ajuste y de
169     ↪ puntos a exceptuar
170
171     intervalo=np.append(intervalo1,intervalo2)# se unen los
172     ↪ intervalos anteriores
173     # Busqueda de parametros de ajuste el intervalo definido
174     a3,b3=polyfit(temperatura[intervalo],
175     ↪ deformacion[intervalo],1)
176     # Busqueda de parametros de ajuste al final del enfriamiento
177     ↪
178     a4,b4=polyfit(temperatura[-200:], deformacion[-200:],1)
179     p=palanca(T, defor,a3, b3, a4, b4)# Aplicacion de regla de la
180     ↪ palanca
181 # Fiteo de curva
182 #-----
183     # Se encuentra el parametro de ajuste para Koistinen y
184     ↪ Marburger
185     par_1, pcov = curve_fit(f_KM, Ms_1-T, p, beta0)
186     # Guardado de Ms encontrado formando un vector para cada
187     ↪ muestra
188     aux_Ms.append(Ms_1)

```

```

175         # Guardado del parametro de ajuste encontrado formando un
        ↪     vector para cada
176         # muestra
177         par_ASTM.append(par_1)
178     # Graficos de curva experimental y fiteada
179     #-----
180         '''
181         # Definicion de color y simbolo para plot de Ms y f_KM(Ms)
182         color1=Simbolos[6]+':'+Colores[k]
183         # Definicion de color y simbolo para plot de T y f_KM
184         color2=Simbolos[1]+Colores[k]
185         # Definicion de color y simbolo para plot de T y p(fraccion
        ↪     dado por la regla de
186         # la palanca)
187         color3=Simbolos[3]+Colores[k]
188         plt.figure(i)
189         plt.plot(temperatura[dato_1], f_KM(0, par_1), color1,
        ↪     label='Ms= '+str(round(Ms_1, 1)))
190         plt.plot(T, f_KM(Ms_1-T, par_1),color2, label='KM=
        ↪     '+str(round(par_1[0],4)))
191         plt.plot(T, p, color3, label='Intervalo '+str(k)+'':
        ↪     '+str(round(temperatura[j],0))+'-'+str(round(temperatura[aux[1]],0))+°C')
192         plt.legend(frameon=False,
        ↪     prop={'family':'serif',"size":8},title='KM: parametro de ajuste
        ↪     de K-M\n..... curva experimental (regla de la palanca)',
        ↪     title_fontsize=8)
193         plt.xlabel('T/°C',fontdict=Formato_ejes)
194         plt.ylabel('fraccion martensitica
        ↪     transformada',fontdict=Formato_ejes)
195         plt.title(Titulo_ASTM ,fontdict=Formato_titulo)
196         plt.suptitle(numeros[i]+'.' +Titulosup_ASTM,
        ↪     fontdict=Formato_titulosup )
197     # plt.savefig(carpetas+numeros[i]+'ASTM.png')
198     # plt.savefig(carpetas+numeros[i]+'ASTM.pdf')'''
199     #     TauX0.append(temperatura[j])
200     # guardado de Ms cada fila corresponde con una muestra y cada
        ↪     columna con intervalo
201     # diferente
202     Ms.append(aux_Ms)
203
204     # if i==0:
205     #     plt.figure() #---para cuando aplico el filtro
206     #     plt.show()
207     # plt.show()
208     plt.close()
209
210
211     # pprint(np.around(transpose(Ms), decimals=2))
212     #-----
213     # GRAFICOS: Dispersion segun intervalo

```

```

214 #-----
215 #-----
216 # Ms media segun los intervalos, es decir se promedia Ms de todas las
    ↪ muestras
217 # analizadas
218
219 Ms_media=[]
220 TT1=[0,1,2,4]# muestras del tratamiento termico 1 [0,1,2,4]
221 # Ms solo de las muestrass que fueron sometidas a tratamiento termico
    ↪ 1
222 Ms_1=np.array(Ms)[TT1,:]
223 for k, Ms_k in enumerate(Ms_1):
224     Ms_media.append(mean(Ms_1[:,k]))
225 # desviacion estandar de Ms_media segun los intervalos
226 desviacion=[]
227 for k, Ms_m in enumerate(Ms_media):
228     aux=sqrt(sum((np.array(Ms_1[:,k])-Ms_m)**2)/len(TT1))
229     desviacion.append(aux)
230 print(Ms_media, desviacion, sep='\n')
231 # Extremo derecho de intervalo considerado//Redondea las temperatura
    ↪ de los extremos
232 # derecho
233 T_der=np.around(np.array(T_derecha),0)
234 # Extremo izquierdo de intervalo considerado, solo toma las muestras
    ↪ del tratamiento
235 # termico 1 [0,1,2,4]
236 T_iz=mean(convDicc(Tbusqueda)[TT1,1])
237 plt.close('all')
238 plt.figure()
239
240
241 for i, Ms_m in enumerate(Ms_media):
242     color=Simbolos[6+i]+'-'+Colores[i]# formato de simbolos del
    ↪ grafico
243     # label
244     label='Intervalo '+str(i)+' :'+str(round(T_der[i], 0))+'-
    ↪ '+str(round(T_iz,0))+°C'
245     # grafico de intervalo y Ms con correspondiente barra de
    ↪ dispersion
246     plt.errorbar(['Intervalo '+str(i)], Ms_m, yerr=desviacion[i],
    ↪ fmt=color, label=label)
247     # colocar el valor de Ms_media sobre el grafico
248     plt.annotate(str(round(Ms_m,1)), (['Intervalo '+str(i)],
    ↪ Ms_m+0.1))
249     # Poner el valor de la desviacion sobre el extremo superior
250     plt.annotate('+{:.1f}'.format(desviacion[i]), (['Intervalo
    ↪ '+str(i)], Ms_m+ desviacion[i]-.2))
251     # Poner el valor de la desviacion sobre el extremo superior
252     plt.annotate('-{:.1f}'.format(desviacion[i]), (['Intervalo
    ↪ '+str(i)], Ms_m- desviacion[i]+.2))

```

```

253 title= 'Variacion de parametros de ajuste lineal'
254 prop={'family':'serif',"size":8}
255 plt.legend(title=title,title_fontsize=7,prop=prop,
    ↪ ncol=2,frameon=False)
256 plt.ylabel('Ms /°C', fontdict=Formato_ejes)
257 titulo='Metodo ASTM: Ms(media) y desviacion estandar\nMuestras: {:s},
    ↪ {:s}, {:s}, {:s}'.format numeros[0], numeros[1], numeros[2],
    ↪ numeros[4])
258
259 Formato_titulo={'family':'serif','color':'black','size':12}# 'blue', 'darkred'
260 # plt.title(titulo,fontdict=Formato_titulo,loc='left')
261 plt.savefig(carpetas+'ASTMdispint.pdf')
262 plt.show()
263 plt.close()
264
265
266 #-----
267 # GRAFICOS: METODO ASTM
268 #-----
269 # Grafico de las Ms determinadas de todas las muestras
270 #-----
271 plt.close('all')
272 plt.figure()
273 extra0=[.2,.5,0,-.7]
274 extra1=[.3,.3,-.6,-.7]
275 extra2=[.3,-.5,-.5,-.4]
276 extra3=[.2,.2,-.7,-.7]
277 extra4=[-.5,-0.,-.3,.3]
278 extra=[extra0,extra1,extra2,extra3,extra4]
279 # Etiqueta de cada grafico
280 for k in range(m):
281     color4=Simbolos[6+k]+' '+Colores[k]
282     label='Intervalo '+str(k)+' :'+str(round(T_der[k], 0))+'-
    ↪ '+str(round(T_iz,0))+ '°C'
283     plt.plot(numeros, np.array(Ms)[: , k],color4,label=label)
284     for i in range(len(numeros)):
285         plt.annotate(str(round(Ms[i][k],1)), (numeros[i],
    ↪ Ms[i][k]+extra[i][k]))
286 plt.xlabel('# muestra', fontdict=Formato_ejes)
287 plt.ylabel('Ms /°C', fontdict=Formato_ejes)
288 Formato_titulo={'family':'serif','color':'black','size':12}# 'blue', 'darkred'
289 # plt.title('Metodo de definicion de Ms ASTM:\nLocalizacion de
    ↪ ajuste
290 # lineal',fontdict=Formato_titulo,loc='left' )
291
292 plt.legend(frameon=False,
    ↪ prop={'family':'serif',"size":8},title_fontsize=8,title='Intervalos
    ↪ de temperaturas con las\nque se hicieron los ajustes lineales')
293 # plt.savefig('MsASTM.png')
294 carpeta='../Informe1/ASTM/'

```

```
295 plt.savefig(carpeta+'MsASTM.pdf')
296 plt.show()
297 plt.close()
298 print('listo')
```

8.3. Determinación de M_S por método offset

```

1  """Florenxia Antonella Correa
2  offset.py
3  Se trabaja solo con el metodo offset en la determinacion de Ms
4  Se calcularon los parametros de ajuste de KM
5  Se grafican todos los resultados obtenidos segun la variacion del
   ↪ intervalo
6  Se grafica la dispersión asociada a cada una de las Ms determinadas
   ↪ segun
7  las muestras# 541, 584, 586 y 590
8  """
9  import numpy as np
10 import matplotlib.pyplot as plt
11 import operator
12
13 from Especifico_funciones import *
14 from scipy.optimize import curve_fit
15 from Especifico_funciones import f_KM
16 from pprint import pprint
17 from matplotlib.legend_handler import HandlerLine2D, HandlerTuple
18 from matplotlib import pyplot as plt
19 from matplotlib.legend import _get_legend_handles_labels
20 """Directorio o root:
21 dir_M: carpeta de transformacion martensitica
22 dir_python: donde se encuentran los ejecutables
23 """
24 dir_M='../Datos dilatometria/Transformacion_martensitica/'
25 dir_python='C:/Users/flor/Documents/ITS/Trabajo Final/Python'
26 dir_TC='../Ejemplos Thermocalc/'
27 carpeta='../Informe1/offset/'
28
29 # Listas de archivos:
30 #-----
31
32 nombres_M=['541-besoky-1Cseg-1050C-30min-1Cseg-12.12.17',
33 '584-besoky-1Cseg-1050C-30min-1Cseg-02.21.19',
34 '586-besoky-1Cseg-1050C-30min-1Cseg-02.28.19',
35 '587-besoky-1Cseg-1050C-30min-10Cseg-02.28.19',
36 '590-besoky-1Cseg-1050C-30min-1Cseg-04.16.19']
37 numeros=['541', '584', '586', '587', '590' ]
38 """Tbusqueda: matriz de temperaturas caracteristicas para el
   ↪ procesamiento, durante el enfriamiento
39 Columnas:
40 0- comienzo de enfriamiento
41 1- cercano al comienzo de la transformacion
42 2- punto durante la transformacion de mayor temperatura
43 3- punto durante la transformacion de menor temperatura
44 Filas corresponde a cada muestra
45 """

```

```

46
47 # Parametros para datos y guardado:
48 #-----
49 #-----
50 # Generales:
51 n=len(nombres_M)
52 extra=2
53 leyenda1='Ajuste K-M, parámetro de ajuste= '
54 leyenda2='Ms determinadas segun método de las rectas tangentes, ASTM
    ↪ A1033 y offset'
55 guardado2='Ms.png'
56 Colores=['k', 'r', 'b', 'g', 'c', 'y', 'm', 'w']
57 Simbolos=['_', '-', '_.', ':', '.', ',', 'o', 'v', 's', 'p', '+', '*',
    ↪ '^', '<', '>', 'x', 'D', 'd']
58 epsilon=1.
59 beta0=0.011
60 Ms=[]
61 posicion=['upper', 'lower', 'lower', 'lower', 'lower', 'lower']
62 Titulo='Comparación de métodos'
63 Titulosup='Ajuste K-M y definición de Ms'
64 Formato_titulo={'family':'serif','color':'black','size':12}# 'blue', 'darkred'
65 Formato_titulosup={'family':'serif','color':'black','size':15}
66 Formato_ejes={'family':'serif','color':'black','size':13}
67 Formato_label={'family':'serif','size':9}
68 # Parametros offset
69 mas_off=[]
70 epsilon_off=1.
71 Ms_off=np.zeros(n)
72 Ms_off1=np.zeros(n)
73 par_off=[]
74 # 'Fracción martensítica transformada determinada por método offset'
    ↪
75 Titulo_offset='Variación de parámetros en ajuste lineal'
76 Titulosup_offset='Ajuste K-M y definición de Ms por método offset'
77 var_off=[]
78 p_off=[]
79 fV=[0.01, 0.015, 0.02, 3.]
80 V=0.02
81 # Composicion nominal acero
82 #-----
83 Elementos=['C', 'CU', 'AL', 'N', 'SB', 'MN', 'SN', 'TI', 'B', 'PB',
    ↪ 'SI', 'NI', 'NB', 'CA', 'P', 'CR', 'V', 'O', 'S', 'MO', 'CO', 'AS',
    ↪ 'FE']
84 Contenido_acero=np.array([0.1, 0.16, 0.009, 0.0605, 0.001, 0.36, 0.008,
    ↪ 0.004, 0.0001, 0.001, 0.24, 0.15, 0.075, 0.0008, 0.014, 8.37,
    ↪ 0.211, .0017, .001, .88, .01, .005, 88.3379 ])
85 FMas_nominal=dict(zip(Elementos, list(Contenido_acero/100.)))
86 FMol_nominal=FraccionMolar(FMas_nominal)
87 PMas_nominal=dict(zip(Elementos, list(Contenido_acero)))
88 # composiciones en equilibrio T=25, 1050°C respectivamente

```

```

89  #-----
90  FMas_gama=comp_fase(dir_TC+'f_gamabis.dat', 'FCC_A1#1')
91  FMol_gama=FraccionMolar(FMas_gama[0])
92  eta=eta_offset(FMol_nominal, FMas_nominal,V)
93  #-----
94  # CALCULOS PARA METODO OFFSET:
95  #-----
96  Tratamiento_1=['541', '584', '586', '589']
97  Tbusqueda={'541':[1050, 400,370, 350],
98             '584':[1050, 400, 360, 345],
99             '586':[1050, 400, 365, 350],
100            '587':[1050, 400, 350, 320],
101            '590':[1050, 400, 365, 355]}
102  exceptuar={'541':[530,420],
103            '584':[580,455],
104            '586':[550,450],
105            '587':[[],[]],
106            '590':[540,440]}
107  m=4
108  # Temperaturas del extremo derecho del intervalo considerado
109  T_derecha=linspace(750, 600, m)
110  dato_derecho=[0, 0, 0, 0] # Donde se guarda la posición de la
    ↪ temperatura de T_derecha
111  extra=2# utilizado para graficar
112  aux=[0,0,0,0]# las posiciones de Tbusqueda
113  # se utilizara para guardar la posición de los extremos de los puntos
    ↪ a exceptuar
114  ex=[0,0]
115  for i, nombre in enumerate(nombres_M):
116  # extracción archivo
117  #-----
118      archivo=dir_M+nombre+'.asc'# nombre del archivo
119      tiempo, deformacion, temperatura=extraccion(archivo)# extraccion
    ↪ de archivo
120  # Búsqueda de Puntos relevantes en la curva:
121      aux[0]=buscador(Tbusqueda[numeros[i]][0], temperatura, epsilon)
122      aux[1]=buscador(Tbusqueda[numeros[i]][1], temperatura, epsilon)
123      aux[2]=buscador(Tbusqueda[numeros[i]][2], temperatura, epsilon)
124      aux[3]=buscador(Tbusqueda[numeros[i]][3], temperatura, epsilon)
125  # Búsqueda de Puntos a exceptuar
126      ex[0]=buscador(exceptuar[numeros[i]][0], temperatura, epsilon)
127      ex[1]=buscador(exceptuar[numeros[i]][1], temperatura, epsilon)
128      if numeros[i] in ['587']:
129          ex[0]=aux[1]
130          ex[1]=aux[1]
131      print(numeros[i])
132      # reinicio de variable para el posterior guardado de Ms
    ↪ encontradas para una muestra
133      aux_Ms=[]
134  # Calculo de datos experimental: Regla de la palanca

```

```

135 #-----
136 # T_der barre las temperaturas definidas para el extremo derecho
    ↳ del intervalo
137 # considerado
138 for k,T_der in enumerate(T_derecha):
139     j=buscador(T_der, temperatura, epsilon)# busca la posición de
        ↳ dicha T_der
140     # se realiza el calculo para los extremos del intervalo
        ↳ definido y exceptuando
141     # ex
142     dato_1, Ms_1=Met_offset(temperatura, deformacion, eta, j,
        ↳ aux[1], V, epsilon_off,ex)
143     # Se define un intervalo de temperatura con dos puntos antes
        ↳ de donde se
144     # encuentro la Ms
145     T=temperatura[dato_1-extra:-100]
146     # Se define un intervalo de deformación con dos puntos antes
        ↳ de donde se
147     # encuentro la Ms
148     defor=deformacion[dato_1-extra:-100]
149     # Se define un intervalo entre los extremos derecho del
        ↳ intervalo de ajuste y de
150     # puntos a exceptuar
151     intervalo1=np.arange(j,ex[0])
152     # Se define un intervalo entre los extremos izquierdo del
        ↳ intervalo de ajuste y
153     # de puntos a exceptuar
154     intervalo2=np.arange(ex[1],aux[1])
155     # intervalo2=np.arange(ex[1],dato_1)
    ↳ #Se define un
156     # intervalo entre los extremos izquierdo del intervalo de ajuste y de
    ↳ puntos a exceptuar
157
158     intervalo=np.append(intervalo1,intervalo2)# se unen los
        ↳ intervalos anteriores
159     # Busqueda de parametros de ajuste el intervalo definido
160     a3,b3=polyfit(temperatura[intervalo],
        ↳ deformacion[intervalo],1)
161     # Busqueda de parametros de ajuste al final del enfriamiento
        ↳
162     a4,b4=polyfit(temperatura[-200:], deformacion[-200:],1)
163     p=palanca(T, defor,a3, b3, a4, b4)# Aplicacion de regla de la
        ↳ palanca
164     # Se encuentra el parametro de ajuste para Koistinen y
        ↳ Marburger
165     par_1, pcov = curve_fit(f_KM, Ms_1-T, p, beta0)
166 # Graficos de curva experimental y fiteada
167 #-----
168     # Definición de color y simbolo para plot de Ms y f_KM(Ms)
169     color1=Simbolos[6]+'.'+Colores[k]

```

```

170     # Definición de color y simbolo para plot de T y f_KM
171     color2=Simbolos[1]+Colores[k]
172     # Definición de color y simbolo para plot de T y p(fraccion
    ↪     dado por la regla
173     # de la palanca)
174     color3=Simbolos[3]+Colores[k]
175     plt.figure(i)
176     plt.plot(temperatura[dato_1], f_KM(0, par_1), color1,
    ↪     label='Ms= '+str(round(Ms_1, 1)))
177     plt.plot(T, f_KM(Ms_1-T, par_1),color2, label='KM=
    ↪     '+str(round(par_1[0],4)))
178     plt.plot(T, p, color3,label='Intervalo '+str(k)+':
    ↪     '+str(round(temperatura[j],0))+'-'+str(round(temperatura[aux[1]],0))+ '°C
179     plt.legend(frameon=False,
    ↪     prop={'family':'serif',"size":8},title='KM: parámetro de
    ↪     ajuste de curva K-M\n..... curva experimental (regla de la
    ↪     palanca)', title_fontsize=8)
180     plt.xlabel('T/°C',fontdict=Formato_ejes )
181     plt.ylabel('fracción martensítica
    ↪     transformada',fontdict=Formato_ejes )
182     # plt.title( Titulo_offset,fontdict=Formato_titulo )
183     # plt.suptitle(numeros[i]+'.'
    ↪     +Titulosup_offset,fontdict=Formato_titulosup )
184     # plt.savefig(carpetas+numeros[i]+'offset.png')
185     # plt.savefig(carpetas+numeros[i]+'offset.pdf')
186     # Guardado de algunos datos
187     #-----
188     # Guardado de Ms encontrado formando un vector fila para cada
    ↪     muestra y cada
189     # columna es correspondiente al intervalo analizado
190     aux_Ms.append(Ms_1)
191     # Guardado del parametro de ajuste encontrado formando un
    ↪     vector para cada
192     # muestra
193     par_off.append(par_1)
194     Ms.append(aux_Ms)
195     #     if i==0:
196     #         plt.show()
197     # plt.show()
198     plt.close()
199     # pprint(np.around(transpose(Ms), decimals=2))
200
201
202     #-----
203     # GRAFICOS: Dispersión según intervalo
204     #-----
205     #-----
206     # Ms media segun los intervalos, es decir se promedia Ms de todas las
    ↪     muestras
207     # analizadas, es decir se lo toma por columna

```

```

208
209 Ms_media=[]
210 TT1=[0,1,2,4]# muestras del tratamiento termico 1 [0,1,2,4]
211 # Ms solo de las muestrass que fueron sometidas a tratamiento termico
    ↪ 1
212 Ms_1=np.array(Ms)[TT1,:]
213 for k, Ms_k in enumerate(Ms_1):
214     Ms_media.append(mean(Ms_1[:,k]))
215 # desviación estandar de Ms_media segun los intervalos
216 desviacion=[]
217 for k, Ms_m in enumerate(Ms_media):
218     aux=sqrt(sum((np.array(Ms_1[:,k])-Ms_m)**2)/len(TT1))
219     desviacion.append(aux)
220 print(Ms_media, desviacion, sep='\n')
221 # Extremo derecho de intervalo considerado//Redondea las temperatura
    ↪ de los extremos
222 # derecho
223 T_der=np.around(np.array(T_derecha),0)
224 # Extremo izquierdo de intervalo considerado, solo toma las muestras
    ↪ del tratamiento
225 # termico 1 [0,1,2,4]
226 T_iz=mean(convDicc(Tbusqueda)[TT1,1])
227 plt.close('all')
228 plt.figure()
229 for i, Ms_m in enumerate(Ms_media):
230     color=Simbolos[6+i]+'-'+Colores[i]# formato de simbolos del
    ↪ grafico
231     # label
232     label='Intervalo '+str(i)+' :'+str(round(T_der[i], 0))+'-
    ↪ '+str(round(T_iz,0))+°C'
233     # grafico de intervalo y Ms con correspondiente barra de
    ↪ dispersión
234     plt.errorbar(['Intervalo '+str(i)], Ms_m, yerr=desviacion[i],
    ↪ fmt=color, label=label)
235     intervalo='Intervalo {:1d}\n{:.1f}-{:.1f}'.format(i, T_der[i],
    ↪ T_iz)
236     # Poner el valor de la desviación sobre el extremo superior
237     plt.annotate('+{:.1f}'.format(desviacion[i]), (['Intervalo
    ↪ '+str(i)], Ms_m+ desviacion[i]-.5))
238     # Poner el valor de la desviación sobre el extremo superior
239     plt.annotate('-{:.1f}'.format(desviacion[i]), (['Intervalo
    ↪ '+str(i)], Ms_m- desviacion[i]+.2))
240     # colocar el valor de Ms_media sobre el grafico
241     plt.annotate('{:.1f}'.format(Ms_m), (['Intervalo '+str(i)],
    ↪ Ms_m+0.1))
242 # Propiedades para legend()
243 title= 'Variación de parámetros de ajuste lineal'
244 prop={'family':'serif','size':7}
245 plt.legend(title=title,title_fontsize=7,prop=prop,
    ↪ ncol=2,frameon=False)

```

```

246 plt.ylabel('Ms /°C', fontdict=Formato_ejes)
247 titulo='Método offset: Ms(media) y desviación estandar\nMuestras:
    ↪  {:s}, {:s}, {:s}, {:s}'.format(numeros[0], numeros[1], numeros[2],
    ↪  numeros[4])
248 # plt.title(titulo,fontdict=Formato_titulo,loc='left')
249 # plt.savefig(carpetas+'offsetdispint.png')
250 plt.savefig(carpetas+'offsetdispint.pdf')
251 plt.show()
252 plt.close()
253
254
255 #-----
256 # GRAFICOS: METODO OFFSET
257 #-----
258 #-----
259 plt.close('all')
260 plt.figure()
261 extra0=[.2,.2,0,-.7]
262 extra1=[.3,.3,.6,-.7]
263 extra2=[.3,-.5,-.5,-.4]
264 extra3=[.1,.1,-.6,.1]
265 extra4=[-.5,.1,-.35,.3]
266 extra=[extra0,extra1,extra2,extra3,extra4]
267 for k in range(m):
268     color4=Simbolos[6+k]+' '+Colores[k]
269     label='Intervalo '+str(k)+' :'+str(round(T_der[k], 0))+'-
    ↪  '+str(round(T_iz,0))+°C'
270     plt.plot(numeros, np.array(Ms)[: , k],color4,label=label)
271     for i in range(len(numeros)):
272         plt.annotate(str(round(Ms[i][k],1)), (numeros[i],
    ↪  Ms[i][k]+extra[i][k]))
273 plt.xlabel('# muestra', fontdict=Formato_ejes)
274 plt.ylabel('Ms /°C', fontdict=Formato_ejes)
275 Formato_titulo={'family':'serif','color':'black','size':12}# 'blue', 'darkred'
276 titulo='Método de definición de Ms offset:\nLocalización de ajuste
    ↪  lineal'
277 # plt.title(titulo,fontdict=Formato_titulo,loc='left' )
278 tit_leyenda='Temperaturas aprox. con las \nque se hicieron los ajustes
    ↪  lineales'
279 plt.legend(frameon=False,prop={'family':'serif','size':8},title_fontsize=8,title=tit_leyenda)
280 # plt.savefig(carpetas+'Msoffset.png')
281 plt.savefig(carpetas+'Msoffset.pdf')
282 plt.show()
283 plt.close()
284 print('listo')

```

8.4. Determinación de M_S por todos los métodos

```

1  """Florenxia Antonella Correa
2  Definir_Ms.py
3  Se trabaja con todos los metodos de determinacion de Ms
4  Se guardan la Ms y los parametros de ajuste lineal para la
   ↪ determinación de la regla de la palanca
5     y definicion de Ms en un archivo de texto
6  Se grafica todos los resultados obtenidos.
7  """
8  import numpy as np
9  import matplotlib.pyplot as plt
10 import operator
11 # import copy
12
13 from Especifico_funciones import *# todas las funciones creadas en
   ↪ Especifico_funciones
14 from scipy.optimize import curve_fit
15 from Especifico_funciones import f_KM
16 from pprint import pprint
17 from matplotlib.legend_handler import HandlerLine2D, HandlerTuple
18 from matplotlib import pyplot as plt
19 from matplotlib.legend import _get_legend_handles_labels
20 from copy import copy,deepcopy
21
22
23
24 # Notas sobre importaciones:
25 '''si uso
26 import copy      --->    copy.copy(...) o copy.deepcopy(...)
27 from copy import copy, deepcopy      --->    copy(...) o
   ↪ deepcopy(...)'''
28
29 """Directorio o root:
30 dir_M: carpeta de transformacion martensitica
31 dir_python: donde se encuentran los ejecutables
32 dir_TC: donde se encuentra el archivo con las composiciones devuelto
   ↪ por thermocal.
33 """
34 dir_M='../Datos dilatometria/Transformacion_martensitica/'
35 dir_python='C:/Users/flor/Documents/ITS/Trabajo Final/Python'
36 dir_TC='../Ejemplos Thermocalc/'
37 carpeta='../Informe1/Especifico/'
38 # Listas de archivos:
39 #-----
40 nombres_M=['541-besoky-1Cseg-1050C-30min-1Cseg-12.12.17',
41 '584-besoky-1Cseg-1050C-30min-1Cseg-02.21.19',
42 '586-besoky-1Cseg-1050C-30min-1Cseg-02.28.19',
43 '587-besoky-1Cseg-1050C-30min-10Cseg-02.28.19',
44 '590-besoky-1Cseg-1050C-30min-1Cseg-04.16.19']

```

```

45 numeros=['541', '584','586', '587', '590' ]
46 Tratamiento_1=['541', '584', '586', '590']
47
48 """Tbusqueda:      matriz de temperaturas características para el
   ↪ procesamiento, durante el enfriamiento
49 Columnas:
50 0- comienzo de enfriamiento
51 1- cercano al comienzo de la transformación
52 2- punto durante la transformación de mayor temperatura
53 3- punto durante la transformación de menor temperatura
54 Filas      corresponde a cada muestra
55 """
56 #-----
57 # GENERAL
58 #-----
59 # Parametros de graficos
60 Colores=['k', 'r', 'b', 'g', 'c', 'y', 'm', 'w']
61 Simbolos=['_', '-', '_.', ':', '.', ',', 'o', 'v', 's', 'p', '+', '*',
   ↪ '^', '<', '>', 'x', 'D', 'd']
62 posicion=['upper', 'lower','lower', 'lower','lower','lower']
63 # Formato y titulos
64 Titulo='Comparación de métodos'
65 Titulosup='Ajuste K-M y definición de Ms'
66 Formato_titulo={'family':'serif','color':'black','size':12}#'blue', 'darkred'
67 Formato_titulosup={'family':'serif','color':'black','size':15}
68 Formato_ejes={'family':'serif','color':'black','size':13}
69 Formato_label={'family':'serif','size':9}
70 # Otros parametros
71 n=len(numeros)
72 extra=2
73 epsilon=1.
74 beta0=0.011
75 Ms=[]
76 Metodos=['Tg', 'ASTM', 'offset']
77 #-----
78 # TANGENTE
79 #-----
80 Ms_tg=np.zeros(n)
81 epsilon_tg=1.
82 var_tg=[]
83 p_tg=[]
84
85 #-----
86 # ASTM
87 #-----
88 m=4# aumento para mayor localización del ajuste ASTM
89 epsilon_ASTM=.01
90 Ms_ASTM=np.zeros(n)
91 par_ASTM=[]
92 dato_1=np.zeros(m)

```

```

93  # Fracción martensítica transformada determinada por método ASTM'
94  Titulo_ASTM='Variación de parámetros en ajuste lineal'
95  Titulosup_ASTM='Ajuste K-M y definición de Ms por método ASTM'
96  aux_enc=[]
97  Taux0=[]
98  T0=[]
99  var_ASTM=[]
100  p_ASTM=[]
101
102  #-----
103  # OFFSET
104  #-----
105  epsilon_off=1.
106  Ms_off=np.zeros(n)
107  Ms_off1=np.zeros(n)
108  par_off=[]
109  # 'Fracción martensítica transformada determinada por método offset'
110  ↪
111  Titulo_offset='Variación de parámetros en ajuste lineal'
112  Titulosup_offset='Ajuste K-M y definición de Ms por método offset'
113  var_off=[]
114  p_off=[]
115  fV=[0.01, 0.015, 0.02, .03]
116  V=0.02
117  #-----
118  # COMPOSICIONES
119  #-----
120  Elementos=['C','CU','AL', 'N', 'SB', 'MN','SN','TI','B', 'PB',
121  ↪ 'SI','NI', 'NB', 'CA', 'P', 'CR','V', 'O', 'S','MO', 'CO', 'AS',
122  ↪ 'FE']
123  Contenido_acero=np.array([0.1,0.16, 0.009, 0.0605,0.001,0.36, 0.008,
124  ↪ 0.004, 0.0001, 0.001, 0.24, 0.15, 0.075, 0.0008, 0.014, 8.37,
125  ↪ 0.211, .0017, .001, .88, .01, .005, 88.3379 ])
126  FMas_nominal=dict(zip(Elementos, list(Contenido_acero/100.)))
127  FMol_nominal=FraccionMolar(FMas_nominal)
128  PMas_nominal=dict(zip(Elementos, list(Contenido_acero)))
129  # composiciones en equilibrio T=25, 1050°C respectivamente
130  FMas_gama=comp_fase(dir_TC+'f_gamabis.dat', 'FCC_A1#1')
131  FMol_gama=FraccionMolar(FMas_gama[0])
132  eta=eta_offset(FMol_nominal, FMas_nominal, V)
133  etaprima=eta_offset(FMol_gama, FMas_gama[0], V)
134
135  # Defini
136  Tbusqueda={'541':[1050, 400,370, 350],
137  ↪ '584':[1050, 400, 360, 345],
138  ↪ '586':[1050, 400, 365, 350],
139  ↪ '587':[1050, 400, 350, 320],
140  ↪ '590':[1050, 400, 365, 355]}
141  exceptuar={'541':[530,420],
142  ↪ '584':[580,455],

```

```

138         '586': [550, 450],
139         '587': [[], []],
140         '590': [540, 440]}
141 m=4
142 T_derecha=linspace(750, 600, m)
143 dato_derecho=[0, 0, 0, 0]
144 inter_ASTM=3
145 inter_offset=3
146 aux=[0,0,0,0]
147 ex=[0,0]
148 aux_Msmediatotal=0
149
150
151 #.....
152 #-----
153 #           Ms POR DISTINTOS METODOS EXPERIMENTALES
154 #-----
155 #-----
156
157 Resultados=[]
158 for i, nombre in enumerate(nombres_M):
159     archivo=dir_M+nombre+'.asc'
160     tiempo, deformacion, temperatura=extraccion(archivo)
161     # Búsqueda de puntos relevantes:
162     aux[0]=buscador(Tbusqueda[numeros[i]][0], temperatura, epsilon)
163     aux[1]=buscador(Tbusqueda[numeros[i]][1], temperatura, epsilon)
164     aux[2]=buscador(Tbusqueda[numeros[i]][2], temperatura, epsilon)
165     aux[3]=buscador(Tbusqueda[numeros[i]][3], temperatura, epsilon)
166     # Búsqueda de puntos a exceptuar
167     ex[0]=buscador(exceptuar[numeros[i]][0], temperatura, epsilon)
168     ex[1]=buscador(exceptuar[numeros[i]][1], temperatura, epsilon)
169     # dado que no posee perturbaciones que interfiera en la
170     ↪ definición de ajuste
171     if numeros[i] in '587':
172         ex[0]=aux[1]
173         ex[1]=aux[1]
174     # print(numeros[i])
175
176 # Calculos y grafico----->tangente
177 #-----
178     # buscador de extremo derecho que se eligio por tener menor
179     ↪ dispersión
180     p1=buscador(T_derecha[3], temperatura, epsilon)
181     # se encuentra Ms segun el metodo de las tangentes
182     dato_tg, Ms_tg[i]=Met_tangente(temperatura, deformacion, aux[0],
183     ↪ aux[1], aux[2], aux[3], epsilon_tg)
184     # intervalo para grafico y calculo de p:fraccion transformada
185     T=temperatura[dato_tg-extra:-100]
186     # intervalo para calculos de p:fraccion transformada
187     deform=deformacion[dato_tg-extra:-100]

```

```

185     # Definicion de intervalo que excluye perturbaciones de la
        ↪ curva...
186     intervalo1=np.arange(p1,ex[0])
187     # desde el extremo derecho hasta el punto donde se encuentre
        ↪ Ms...
188     intervalo2=np.arange(ex[1],aux[1])
189     # con este se determina el ajuste para calculo de p:fraccion
        ↪ transformada
190     intervalo=np.append(intervalo1,intervalo2)
191     # parametros de ajuste para recta de fase austenita
192     a3,b3=polyfit(temperatura[intervalo], deformacion[intervalo],1)
193     # parametros de ajuste para recta de fase martensita
194     a4,b4=polyfit(temperatura[-200:], deformacion[-200:],1)
195     metodo={}
196     metodo['numero']=numeros[i]
197     metodo['metodo']='tangente'
198     metodo['Ms']=Ms_tg[i]
199     metodo['dato de Ms']=dato_tg
200     metodo['extra']=extra
201     metodo['a_gama']=a3
202     metodo['b_gama']=b3
203     metodo['a_alfaprima']=a4
204     metodo['b_alfaprima']=b4
205     Resultados.append(metodo)
206     # Calculos y grafico----->ASTM
207     #------
208
209     # buscador de extremo derecho que se eligio por tener menor
        ↪ dispersión
210     p1=buscador(T_derecha[inter_ASTM], temperatura, epsilon)
211     # determinacion de Ms con con intervalo de menor dispersión de
        ↪ los probados y se
212     # contempla los puntos a excluir
213     dato_ASTM, Ms_ASTM[i]=Met_ASTM(temperatura, deformacion, p1,
        ↪ aux[1], aux[2], aux[3],epsilon_ASTM, ex)
214     # Intervalo para grafico y calculo de p:fraccion transformada
215     T=temperatura[dato_ASTM-extra:-100]
216     # Intervalo para calculo de p:fraccion transformada
217     defor=deformacion[dato_ASTM-extra:-100]
218     # Definicion de intervalo que excluye perturbaciones de la
        ↪ curva...
219     intervalo1=np.arange(p1,ex[0])
220     # desde el extremo derecho hasta el punto donde se encuentre
        ↪ Ms...
221     intervalo2=np.arange(ex[1],aux[1])
222     # intervalo2=np.arange(ex[1],dato_ASTM)           #desde el extremo
        ↪ derecho hasta el punto
223     # donde se encuentre Ms...
224

```

```

225     # con este se determina el ajuste para calculo de p:fraccion
      ↪ transformada
226 intervalo=np.append(intervalo1,intervalo2)
227 # parametros de ajuste para recta de fase austenita
228 a3,b3=polyfit(temperatura[intervalo], deformacion[intervalo],1)
229 # parametros de ajuste para recta de fase martensita
230 a4,b4=polyfit(temperatura[-200:], deformacion[-200:],1)
231 metodo={}
232 metodo['numero']=numeros[i]
233 metodo['metodo']='ASTM'
234 metodo['Ms']=Ms_ASTM[i]
235 metodo['dato de Ms']=dato_ASTM
236 metodo['extra']=extra
237 metodo['a_gama']=a3
238 metodo['b_gama']=b3
239 metodo['a_alfaprima']=a4
240 metodo['b_alfaprima']=b4
241 Resultados.append(metodo)
242 # Calculos y grafico----->offset
243 #-----
244     # buscador de extremo derecho que se eligio por tener menor
      ↪ dispersión
245 p1=buscador(T_derecha[inter_offset], temperatura, epsilon)
246 # determinacion de Ms con con intervalo de menor dispersión de
      ↪ los probados y se
247 # contempla los puntos a excluir
248 dato_off, Ms_off[i]=Met_offset(temperatura, deformacion, eta, p1,
      ↪ aux[1], V, epsilon_off, ex)
249 # Intervalo para grafico y calculo de p:fraccion transformada
250 T=temperatura[dato_off-extra:-100]
251 # Intervalo para calculo de p:fraccion transformada
252 defor=deformacion[dato_off-extra:-100]
253 # Definicion de intervalo que excluye perturbaciones de la
      ↪ curva...
254 intervalo1=np.arange(p1,ex[0])
255 # desde el extremo derecho hasta el punto donde se encuentre
      ↪ Ms...
256 intervalo2=np.arange(ex[1],aux[1])
257 # intervalo2=np.arange(ex[1],dato_off)           #desde el extremo
      ↪ derecho hasta el punto
258 # donde se encuentre Ms...
259
260     # con este se determina el ajuste para calculo de p:fraccion
      ↪ transformada
261 intervalo=np.append(intervalo1,intervalo2)
262 # parametros de ajuste para recta de fase austenita
263 a3,b3=polyfit(temperatura[intervalo], deformacion[intervalo],1)
264 # parametros de ajuste para recta de fase martensita
265 a4,b4=polyfit(temperatura[-200:], deformacion[-200:],1)
266 metodo={}

```

```

267     metodo['numero']=numeros[i]
268     metodo['metodo']='offset'
269     metodo['Ms']=Ms_off[i]
270     metodo['dato de Ms']=dato_off
271     metodo['extra']=extra
272     metodo['a_gama']=a3
273     metodo['b_gama']=b3
274     metodo['a_alfaprima']=a4
275     metodo['b_alfaprima']=b4
276     Resultados.append(metodo)
277
278
279 #-----
280 #-----
281 #                                GUARDADO DE DATOS
282 #-----
283 #-----
284
285 direccion='../Informe1/Especifico/'
286 nombre='Resultados.txt'
287 archivosalida=direccion+nombre
288 Titulo1='Localización de ajuste: Variacion de parametros de ajuste\n'
289 Titulo2='Todos los Metodos\n'
290 # os.system('cp'+nombre+' '+archivosalida)
291 # fo= open (archivosalida,'a')
292 fo= open (archivosalida,'w')
293
294 fo.write(Titulo1)
295 fo.write('{numero:10s} {metodo:10s} {Ms:10s} {dato_Ms:10s} {ag:10s}
↪ {bg:10s} {am:10s}
↪ {bm:10s}\n'.format(numero='numero',metodo='metodo',Ms='Ms',
↪ dato_Ms='dato_Ms',ag='a_gama',bg='b_gama', am='a_alfapri',
↪ bm='b_alfapri'))
296
297 # muestra metodo Ms dato Ms extra ag bg am bm
298 for i, r in enumerate(Resultados):
299     fo.write('{numero:10s} {metodo:10s} {Ms:<10.2f} {dato_Ms:<10d}
↪ {ag:<10.7f} {bg:<10.4f} {am:<10.7f}
↪ {bm:<10.4f}\n'.format(numero=r['numero'],metodo=r['metodo'],Ms=r['Ms'],
↪ dato_Ms=r['dato de Ms'],ag=r['a_gama'],bg=r['b_gama'],
↪ am=r['a_alfaprima'], bm=r['b_alfaprima']))
300
301 fo.close()
302
303
304 #-----
305 # GRAFICOS DE MS DE TODOS LOS METODOS Y MUESTRAS
306 #-----
307
308 plt.close('all')

```

```

309 plt.figure()
310 colores=['# FC5A50','#15B01A','#7E1E9C']
311 plt.plot(numeros, Ms_tg,'o r', label='Rectas tangentes')
312 plt.plot(numeros, Ms_off, 'v b', label='offset')
313 plt.plot(numeros, Ms_ASTM, 's g',label='ASTM')
314
315
316 for i, numero in enumerate(numeros):
317     plt.annotate('{:.1f}'.format(Ms_tg[i]), (numero, Ms_tg[i]-1.2))
318     plt.annotate('{:.1f}'.format(Ms_ASTM[i]), (numero, Ms_ASTM[i]+.3))
319     plt.annotate('{:.1f}'.format(Ms_off[i]), (numero, Ms_off[i]-1.5))
320 plt.xlabel('# muestra', fontdict=Formato_ejes)
321 plt.ylabel('Ms /°C', fontdict=Formato_ejes)
322 Formato_titulo={'family':'serif','color':'black','size':12}# 'blue', 'darkred'
323 # plt.title('Distintos métodos de definición de
324 # Ms:',fontdict=Formato_titulo,loc='left')
325
326 plt.legend(frameon=False,
327     ↪ prop={'family':'serif','size':8},title_fontsize=8,title='Metodos:')
327 carpeta='../Informe1/Especifico/'
328 plt.savefig(carpeta+'Ms.pdf')
329 plt.show()
330 plt.close()

```

8.5. Aplicación de ajustes de modelos de Koistinen-Marburger, Lee-Van Tyne y Huyan

```

1  """Floresencia Antonella Correa
2  Especifico.py
3  Se hacen todos los ajustes correspondientes a los modelos KM, LVT y
   ↪ H
4      para la media de las muestras del tratamiento 1 (muestras
   ↪ 541, 584, 586, 590)
5      y par ala muestra 587 del tratamiento 2
6  Se calcula la dispersion asociada a la medicion de Ms por cada
   ↪ metodo
7  Se mide la dispersion asociada a la fracción transformada para las
   ↪ muestras del
8      tratamiento 2
9  """
10 import numpy as np
11 import matplotlib.pyplot as plt
12 import operator
13 #import copy
14
15 from Especifico_funciones import *
16 #todas las funciones creadas en Especifico_funciones
17 from scipy.optimize import curve_fit
18 from Especifico_funciones import f_KM
19 from pprint import pprint
20 from matplotlib.legend_handler import HandlerLine2D, HandlerTuple
21 from matplotlib import pyplot as plt
22 from matplotlib.legend import _get_legend_handles_labels
23 from copy import copy,deepcopy
24 #Notas sobre importaciones:
25 '''si uso
26 import copy ----> copy.copy(...) o
   ↪ copy.deepcopy(...)
27 from copy import copy, deepcopy ----> copy(...)
   ↪ o deepcopy(...)'''
28
29
30
31 """Directorio o root:
32 dir_M: carpeta de transformacion martensitica
33 dir_python: donde se encuentran los ejecutables
34 dir_TC: donde se encuentra el archivo con las composiciones devuelto
   ↪ por thermocal.
35 """
36 dir_M='../Datos dilatometria/Transformacion_martensitica/'
37 dir_python='C:/Users/flor/Documents/ITS/Trabajo Final/Python'
38 dir_TC='../Ejemplos Thermocalc/'
39 carpeta='../Informe1/Especifico/'
40 #Listas de archivos:

```

```

41 #-----
42 nombres_M=['541-besoky-1Cseg-1050C-30min-1Cseg-12.12.17',
43 '584-besoky-1Cseg-1050C-30min-1Cseg-02.21.19',
44 '586-besoky-1Cseg-1050C-30min-1Cseg-02.28.19',
45 '587-besoky-1Cseg-1050C-30min-10Cseg-02.28.19',
46 '590-besoky-1Cseg-1050C-30min-1Cseg-04.16.19']
47 numeros=['541', '584', '586', '587', '590' ]
48 numeros_TT1=['541', '584', '586', '590']
49 nombres_TT1=['541-besoky-1Cseg-1050C-30min-1Cseg-12.12.17',
50 '584-besoky-1Cseg-1050C-30min-1Cseg-02.21.19',
51 '586-besoky-1Cseg-1050C-30min-1Cseg-02.28.19',
52 '590-besoky-1Cseg-1050C-30min-1Cseg-04.16.19']
53 """Tbusqueda:      matriz de temperaturas características para el
   ↪ procesamiento, durante el enfriamiento
54 Columnas:
55 0- comienzo de enfriamiento
56 1- cercano al comienzo de la transformacion
57 2- punto durante la transformacion de mayor temperatura
58 3- punto durante la transformacion de menor temperatura
59 Filas      corresponde a cada muestra
60 """
61 #-----
62 #GENERAL
63 #-----
64 #Parametros de graficos
65 Colores=['k', 'r', 'b', 'g', 'c', 'y', 'm', 'w']
66 Simbolos=['_', '-', '_.', ':', '.', ',', 'o', 'v', 's', 'p', '+', '*',
   ↪ '^', '<', '>', 'x', 'D', 'd']
67 posicion=['upper', 'lower', 'lower', 'lower', 'lower', 'lower']
68 #Otros parametros
69 n=len(nombres_M)
70 epsilon=1.
71 Ms=[]
72 Metodos=['Tg', 'ASTM', 'offset']
73 #-----
74 #TANGENTE
75 #-----
76 Ms_tg=np.zeros(n)
77 epsilon_tg=1.
78 #-----
79 #ASTM
80 #-----
81 m=4 #aumento para mayor localizacion del ajuste ASTM tome hasta 450
82 epsilon_ASTM=.01
83 Ms_ASTM=np.zeros(n)
84 #-----
85 #OFFSET
86 #-----
87 epsilon_off=1.
88 Ms_off=np.zeros(n)

```

```

89 Ms_off1=np.zeros(n)
90 fV=[0.01, 0.015, 0.02, .03]
91 V=0.02
92 #-----
93 #COMPOSICIONES
94 #-----
95 Elementos=['C','CU','AL','N','SB','MN','SN','TI','B','PB',
↪ 'SI','NI','NB','CA','P','CR','V','O','S','MO','CO','AS',
↪ 'FE']
96 Contenido_acero=np.array([0.1,0.16, 0.009, 0.0605,0.001,0.36, 0.008,
↪ 0.004, 0.0001, 0.001, 0.24, 0.15, 0.075, 0.0008, 0.014, 8.37,
↪ 0.211, .0017, .001, .88, .01, .005, 88.3379 ])
97 FMas_nominal=dict(zip(Elementos, list(Contenido_acero/100.)))
98 FMol_nominal=FraccionMolar(FMas_nominal)
99 PMas_nominal=dict(zip(Elementos, list(Contenido_acero)))
100 #composiciones en equilibrio T=25, 1050°C respectivamente
101 FMas_gama=comp_fase(dir_TC+'f_gamabis.dat', 'FCC_A1#1')
102 FMol_gama=FraccionMolar(FMas_gama[0])
103
104 #Defini
105 Tbusqueda= {'541':[1050, 400,370, 350],
106             '584':[1050, 400, 360, 345],
107             '586':[1050, 400, 365, 350],
108             '587':[1050, 400, 350, 320],
109             '590':[1050, 400, 365, 355]}
110 exceptuar= {'541':[530,420],
111             '584':[580,455],
112             '586':[550,450],
113             '587':[[[]],[[]],
114             '590':[540,440]}
115 aux=[0,0,0,0]
116 ex=[0,0]
117
118
119 #-----
120 #-----
121 #-----MODELOS DETERMINACION DE FRACCIÓN TRANSFORMADA
122 #-----
123 #-----Modelos segun la composicion-----
124 #-----
125
126 #coeficients o parametros semillas para ajuste
127 KMO=0.011
128 LV01=.001
129 LV02= 1.4
130 coef=LV01, LV02
131 A=0.05
132 #Extraccion de datos de las muestras y otros
133 direccion='../Informe1/Especifico/'
134 nombre='Resultados.txt'

```

```

135 Resultados=lectura(direccion+nombre)
136 #Valores para definir intervalo de analisis
137 #Rango=[390,260]
138 Rango=[400,260]
139 dato=[0,0]
140 T_media=[]
141 #Diccionarios donde se guardaran los valores de interes de los
    ↪ distintos metodos para el analisis de las 4 muestras con el mismo
    ↪ tratamiento termico
142 ASTM={'Ms_media':0., 'desv_Ms':0., 'desv_p':0.}
143 tangente={'Ms_media':0., 'desv_Ms':0., 'desv_p':0.}
144 offset={'Ms_media':0., 'desv_Ms':0., 'desv_p':0.}
145
146
147 #-----CALCULO DE MEDIAS-----
148 #-----
149
150 for i, numero in enumerate(numeros_TT1):
151 #Recorre las solo las muestras con el mismo tratamiento termico del
    ↪ total de las muestras analizada
152     archivo= dir_M+ nombres_TT1[i]+ '.asc'
153     tiempo, deformacion, temperatura= extraccion(archivo)
154     #extraccion de datos medidos directamente
155     muestra=
        ↪ Resultados[numero]
156     #muestra: contiene los datos analizados de cada muestra,
        ↪ dados por la definición de Ms
157     print(i, numero, sep= ' ')
158     #Definicion de variables
159     #-----
160     dato[0]= buscador(Rango[0], temperatura,
        ↪ 1.)
161     #da el extremo derecho del intervalo
162     dato[1]= buscador(Rango[1], temperatura,
        ↪ 1.)
163     #da el extremo izquierdo del intervalo
164     T=
        ↪ temperatura[dato[0]:dato[1]]
165     #recordar que T y Def son la misma mismo para todos lo
        ↪ métodos, y son una medida directa, encambio...
166     Def=
        ↪ deformacion[dato[0]:dato[1]]
167     #...p puede cambiar segun los parametros de ajuste para dicho
        ↪ metodo
168     muestra['T']= T
169     muestra['def']= Def
170     #Calculo asociado al metodo ASTM
171     resumen_p(muestra['ASTM'], T, Def)
172     dato_Ms= int(muestra['ASTM']['dato_Ms'])

```

```

173     ASTM['Ms_media'] = ASTM['Ms_media']*(i/(i+ 1))+
        ↪ temperatura[dato_Ms]/(i+ 1)
174     #Calculo asociado al metodo de las rectas tangentes
175     resumen_p(muestra['tangente'],T , Def)
176     dato_Ms= int(muestra['tangente']['dato_Ms'])
177     tangente['Ms_media'] = tangente['Ms_media']*(i/(i+ 1))+
        ↪ temperatura[dato_Ms]/(i+ 1)
178     #Calculo asociado al metodo offset
179     resumen_p(muestra['offset'], T, Def)
180     dato_Ms= int(muestra['offset']['dato_Ms'])
181     offset['Ms_media'] = offset['Ms_media']*(i/(i+ 1))+
        ↪ temperatura[dato_Ms]/(i+ 1)
182     #Calculo de medias
183     #-----
184     long_media= len(T_media)
185     long_T= len(T)
186     if i= =
        ↪ 0:
187         #iniciar las variables
188         ASTM['p_media'] = muestra['ASTM']['p']
189         tangente['p_media'] = muestra['tangente']['p']
190         offset['p_media'] = muestra['offset']['p']
191         T_media= T
192     else:
193         if long_T<=
            ↪ long_media:
194             #Si el valor de la variable donde se va a guardar la
            ↪ media es de un
195                 ASTM['p_media'] =
                    ↪ ASTM['p_media'][:long_T]*(i/(i+ 1))+
                    ↪ muestra['ASTM']['p']/(i+
                    ↪ 1)
196                 #tamaño MAYOR que el valor a sumar, entonces
                    ↪ la media tendra el tamaño del vector de
                    ↪ menor dimension
197                 tangente['p_media'] =
                    ↪ tangente['p_media'][:long_T]*(i/(i+ 1))+
                    ↪ muestra['tangente']['p']/(i+
                    ↪ 1)
198                 offset['p_media'] =
                    ↪ offset['p_media'][:long_T]*(i/(i+ 1))+
                    ↪ muestra['offset']['p']/(i+ 1)
199                 T_media= T_media[:long_T]*(i/(i+ 1))+ T/(i+
                    ↪ 1)
200         elif
            ↪ long_T>long_media:
201             #Si el valor de la variable donde se va a guardar la
            ↪ media es de un tamaño MENOR que el valor a sumar,
            ↪ idem, el caso anterior

```

```

202     ASTM['p_media'] = ASTM['p_media']*(i/(i+ 1))+
        ↳ muestra['ASTM']['p'][:long_media]/(i+
        ↳ 1)
203     tangente['p_media'] =
        ↳ tangente['p_media']*(i/(i+ 1))+
        ↳ muestra['tangente']['p'][:long_media]/(i+
        ↳ 1)
204     offset['p_media'] = offset['p_media']*(i/(i+
        ↳ 1))+
        ↳ muestra['offset']['p'][:long_media]/(i+ 1)
205     T_media = T_media*(i/(i+ 1))+
        ↳ T[:long_media]/(i+ 1)
206
207     print('pasola')
208     # CALCULO DE DISPERSION
209     #-----
210     long_Tmedia = len(T_media)
211     desv_T = 0.
212     for i, numero in enumerate(numeros_TT1):
213         muestra = Resultados[numero]
214         T = muestra['T']
215         desv_T += (T[:long_Tmedia]-T_media)**2
216         ASTM['desv_p'] +=
        ↳ (muestra['ASTM']['p'][:long_Tmedia]-ASTM['p_media'])**2
217         tangente['desv_p'] +=
        ↳ (muestra['tangente']['p'][:long_Tmedia]-tangente['p_media'])**2
218         offset['desv_p'] +=
        ↳ (muestra['offset']['p'][:long_Tmedia]-offset['p_media'])**2
219
220         ASTM['desv_Ms'] += (muestra['ASTM']['Ms']-ASTM['Ms_media'])**2
221         tangente['desv_Ms'] +=
        ↳ (muestra['tangente']['Ms']-tangente['Ms_media'])**2
222         offset['desv_Ms'] +=
        ↳ (muestra['offset']['Ms']-offset['Ms_media'])**2
223
224     ASTM['desv_p'] = sqrt(ASTM['desv_p']/len(numeros_TT1))
225     tangente['desv_p'] = sqrt(tangente['desv_p']/len(numeros_TT1))
226     offset['desv_p'] = sqrt(offset['desv_p']/len(numeros_TT1))
227
228     ASTM['desv_Ms'] = sqrt(ASTM['desv_Ms']/len(numeros_TT1))
229     tangente['desv_Ms'] = sqrt(tangente['desv_Ms']/len(numeros_TT1))
230     offset['desv_Ms'] = sqrt(offset['desv_Ms']/len(numeros_TT1))
231
232     desv_T = sqrt(desv_T/len(numeros_TT1))
233     print('pasola1b')
234
235     #-----
236     # AJUSTES: DISTINTOS MODELOS
237     #-----
238     #-----

```

```

239 #                                     #541, 584, 586, 590
240 #-----
241 #-----
242 KMO=0.0011
243 LV01=.001
244 LV02= 1.4
245 coef=LV01, LV02
246 A=0.05
247
248 #                                     Ms DEFINIDA POR METODO ASTM
249 #-----
250 ' ' '
251 Ms=ASTM['Ms_media']
252 desv_Ms=ASTM['desv_Ms']
253 p_media=ASTM['p_media']
254 desv_p=ASTM['desv_p']
255 #MODELO DE HUYAN
256 B=0.006*(Ms+273.15)-0.1369
257 aux_DeltaG=100.*(p_media/(A*(1.-p_media)))*(1./B)
258 H1, H2=polyfit(T_media-Ms, aux_DeltaG, 1)
259 dato_Msmedia=buscador(Ms, T_media, 1.)
260 print(dato_Msmedia)
261 #MODELO DE KONSTINEN Y MARBURGER
262 KM, cov_KM=curve_fit(f_KM, Ms-T_media[dato_Msmedia-1:],
    ↪ p_media[dato_Msmedia-1:], KMO)
263 #MODELO DE LEE Y VAN TYNE
264 LV, cov_LV = curve_fit(f_LV, Ms-T_media[dato_Msmedia+1:],
    ↪ p_media[dato_Msmedia+1:], coef)
265 #GUARDADO
266 ASTM['KM']=KM[0]
267 ASTM['LV1']=LV[0]
268 ASTM['LV2']=LV[1]
269 ASTM['H1']=H1
270 ASTM['H2']=H2
271 plt.figure()
272 plt.plot(T_media, p_media,color='#7E1E9C', linestyle='-',
    ↪ linewidth=1.,label='Regla de la palanca')
273 leyenda_desv='Desviación estandar\nde la fracción
    ↪ transformada\n(Muestras #541,584,586,590)'
274 plt.fill_between(T_media, p_media+desv_p, p_media-desv_p,alpha=0.2,
    ↪ edgecolor='#580F41', facecolor='#7E1E9C', linewidth=1.15,
    ↪ antialiased=True, label=leyenda_desv)
275 plt.plot(T_media, f_H(Ms-T_media, A, B, H1, H2), color='#808080',
    ↪ linestyle='--', linewidth=1.15, label='Modelo de Huyan y
    ↪ colaboradores')
276 plt.plot(T_media[dato_Msmedia-1:], f_KM(Ms-T_media[dato_Msmedia-1:],
    ↪ KM[0]),color='#15B01A', linestyle='--',
    ↪ linewidth=1.15,label='Modelo de Koistinen y Marburger')

```

```

277 plt.plot(T_media[datos_Msmedia+1:], f_LV(Ms-T_media[datos_Msmedia+1:],
    ↪ *tuple(LV)),color='#FC5A50', linestyle='--', linewidth=1.15,
    ↪ label='Modelo de Lee y Van Tyne')
278 plt.errorbar(Ms, p_media[datos_Msmedia], xerr=desv_Ms,
    ↪ color='#000000')
279 plt.plot(Ms, p_media[datos_Msmedia], color='#000000', linestyle='-',
    ↪ marker='.', markersize=10, label='Ms=({:0.1f} +/-
    ↪ {:0.1f})°C'.format(Ms, desv_Ms))
280 plt.legend()
281 Formato_ejes={'family':'serif','color':'black','size':12}
282 plt.xlabel('T/°C',fontdict=Formato_ejes)
283 plt.ylabel('fracción de martensita
    ↪ transformada',fontdict=Formato_ejes)
284 title='Diferentes modelos de ajuste: Tratamiento térmico 1\nMs
    ↪ definido por metodo ASTM'
285 #plt.title(title,
    ↪ fontdict={'family':'serif','color':'black','size':10},loc='left')
286 plt.savefig('../Informe1/ASTM/Modelos_Ms(ASTM).pdf')
287 #plt.savefig('../Informe1/ASTM/Modelos_Ms(ASTM)zoom.pdf')
288 plt.show()
289 plt.close()
290 print('paso1c')
291 '''
292 #Ms DEFINIDA POR METODO DE LAS RECTAS TANGENTES
293 #-----
294 '''
295 Ms=tangente['Ms_media']
296 desv_Ms=tangente['desv_Ms']
297 p_media=tangente['p_media']
298 desv_p=tangente['desv_p']
299 #MODELO DE HUYAN
300 B=0.006*(Ms+273.15)-0.1369
301 aux_DeltaG=100.*(p_media/(A*(1.-p_media)))*(1./B)
302 H1, H2=polyfit(T_media-Ms, aux_DeltaG, 1)
303 datos_Msmedia=buscador(Ms, T_media, 1.)
304 #MODELO DE KONSTINEN Y MARBURGER
305 KM, cov_KM=curve_fit(f_KM, Ms-T_media[datos_Msmedia-1:],
    ↪ p_media[datos_Msmedia-1:], KMO)
306 #MODELO DE LEE Y VAN TYNE
307 LV, cov_LV = curve_fit(f_LV, Ms-T_media[datos_Msmedia+1:],
    ↪ p_media[datos_Msmedia+1:], coef)
308 #GUARDADO
309 tangente['KM']=KM[0]
310 tangente['LV1']=LV[0]
311 tangente['LV2']=LV[1]
312 tangente['H1']=H1
313 tangente['H2']=H2
314 plt.figure()
315 plt.plot(T_media, p_media,color='#7E1E9C', linestyle='-',
    ↪ linewidth=1.,label='Regla de la palanca')

```

```

316  leyenda_desv='Desviación estandar\nde la fracción
    ↪ transformada\n(Muestras #541,584,586,590)'
317  plt.fill_between(T_media, p_media+desv_p, p_media-desv_p, alpha=0.2,
    ↪ edgecolor='#580F41', facecolor='#7E1E9C', linewidth=1.15,
    ↪ antialiased=True, label=leyenda_desv)
318  plt.plot(T_media, f_H(Ms-T_media, A, B, H1, H2), color='#808080',
    ↪ linestyle='--', linewidth=1.15, label='Modelo de Huyan y
    ↪ colaboradores')
319  plt.plot(T_media[dato_Msmedia-1:], f_KM(Ms-T_media[dato_Msmedia-1:],
    ↪ KM[0]), color='#15B01A', linestyle='--', linewidth=1.15,
    ↪ label='Modelo de Koistinen y Marburger')
320  plt.plot(T_media[dato_Msmedia+1:], f_LV(Ms-T_media[dato_Msmedia+1:],
    ↪ *tuple(LV)), color='#FC5A50', linestyle='--', linewidth=1.15,
    ↪ label='Modelo de Lee y Van Tyne')
321  plt.plot(Ms, p_media[dato_Msmedia], color='#000000', linestyle='-',
    ↪ marker='.', markersize=10, label='Ms=({:0.1f} +/-
    ↪ {:0.1f})°C'.format(Ms, desv_Ms))
322  plt.errorbar(Ms, p_media[dato_Msmedia], xerr=desv_Ms,
    ↪ color='#000000')
323  plt.legend()
324  Formato_ejes={'family':'serif','color':'black','size':12}
325  plt.xlabel('T/°C',fontdict=Formato_ejes)
326  plt.ylabel('fracción de martensita
    ↪ transformada',fontdict=Formato_ejes)
327  title='Diferentes modelos de ajuste: Tratamiento térmico 1\nMs
    ↪ definido por metodo de las rectas tangentes'
328  #plt.title(title,
    ↪ fontdict={'family':'serif','color':'black','size':10},loc='left')
329  plt.savefig('../Informe1/tg/Modelos_Ms(tg).pdf')
330  #plt.savefig('../Informe1/tg/Modelos_Ms(tg)zoom.pdf')
331  #-----Desplazamiento, solo para el modelo de Huyan:
332  #fprima=f_Hprima(Ms-T_media, A, B, H1, H2, p_media[dato_Msmedia])
333  #plt.plot(T_media, fprima, color='#808080', linestyle=':',
    ↪ label='Modelo de Huyan desplazada')
334  #plt.plot(T_media[dato_Msmedia+1:],
    ↪ f_LVprima(Ms-T_media[dato_Msmedia+1:], LV[0], LV[1],
    ↪ p_media[dato_Msmedia]), color='#D2691E', linestyle=':',
    ↪ label='Modelo de Lee y Van Tyne')
335  #plt.savefig('../Informe1/tg/MetHuyan_Ms(tg).pdf')
336  plt.show()
337  plt.close()
338  print('pasold')
339  '''
340  #                               Ms DEFINIDA POR METODO OFFSET
341  #-----
342  '''
343  Ms=offset['Ms_media']
344  desv_Ms=offset['desv_Ms']
345  p_media=offset['p_media']
346  desv_p=offset['desv_p']

```

```

347 #MODELO DE HUYAN
348 B=0.006*(Ms+273.15)-0.1369
349 aux_DeltaG=100.*(p_media/(A*(1.-p_media)))*(1./B)
350 H1, H2=polyfit(T_media-Ms, aux_DeltaG, 1)
351 dato_Msmedia=buscador(Ms, T_media, 1.)
352 #MODELO DE KONSTINEN Y MARBURGER
353 KM, cov_KM=curve_fit(f_KM, Ms-T_media[dato_Msmedia-1:],
    ↪ p_media[dato_Msmedia-1:], KMO)
354 #MODELO DE LEE Y VAN TYNE
355 LV, cov_LV = curve_fit(f_LV, Ms-T_media[dato_Msmedia+1:],
    ↪ p_media[dato_Msmedia+1:], coef)
356 #GUARDADO
357 offset['KM']=KM[0]
358 offset['LV1']=LV[0]
359 offset['LV2']=LV[1]
360 offset['H1']=H1
361 offset['H2']=H2
362 plt.figure()
363 plt.plot(T_media, p_media,color='#7E1E9C', linestyle='-',
    ↪ linewidth=1.,label='Regla de la palanca')
364 leyenda_desv='Desviación estandar\nde la fracción
    ↪ transformada\n(Muestras #541,584,586,590)'
365 plt.fill_between(T_media, p_media+desv_p, p_media-desv_p, alpha=0.2,
    ↪ edgecolor='#580F41', facecolor='#7E1E9C', linewidth=1.15,
    ↪ antialiased=True, label=leyenda_desv)
366 plt.plot(T_media,f_H(Ms-T_media, A, B, H1, H2), color='#808080',
    ↪ linestyle='--', linewidth=1.15, label='Modelo de Huyan y
    ↪ colaboradores')
367 plt.plot(T_media[dato_Msmedia-1:], f_KM(Ms-T_media[dato_Msmedia-1:],
    ↪ KM[0]),color='#15B01A', linestyle='--', linewidth=1.15,
    ↪ label='Modelo de Koistinen y Marburger')
368 plt.plot(T_media[dato_Msmedia+1:], f_LV(Ms-T_media[dato_Msmedia+1:],
    ↪ *tuple(LV)),color='#FC5A50', linestyle='--', linewidth=1.15,
    ↪ label='Modelo de Lee y Van Tyne')
369 plt.plot(Ms, p_media[dato_Msmedia], color='#000000', linestyle='-',
    ↪ marker='.', markersize=10, label='Ms={:0.1f} +/-
    ↪ {:0.1f})°C'.format(Ms, desv_Ms))
370 plt.errorbar(Ms, p_media[dato_Msmedia], xerr=desv_Ms,
    ↪ color='#000000')
371 plt.legend()
372 Formato_ejes={'family':'serif','color':'black','size':12}
373 plt.xlabel('T/°C',fontdict=Formato_ejes)
374 plt.ylabel('fracción de martensita
    ↪ transformada',fontdict=Formato_ejes)
375 title='Diferentes modelos de ajuste: Tratamiento térmico 1\nMs
    ↪ definido por metodo offset'
376 #plt.title(title,
    ↪ fontdict={'family':'serif','color':'black','size':10},loc='left')
377 plt.savefig('../Informe1/offset/Modelos_Ms(offset).pdf')
378 #plt.savefig('../Informe1/offset/Modelos_Ms(offset)zoom.pdf')

```

```

379 plt.show()
380 plt.close()
381 print('pasoid')
382 '''
383
384 #PRINTEO
385 '''
386 print('tangente', 'offset', 'ASMT')
387 print('KM', tangente['KM'], offset['KM'], ASTM['KM'], sep=' & ')
388 print('LV1', tangente['LV1'], offset['LV1'], ASTM['LV1'], sep=' & ')
389 print('LV2', tangente['LV2'], offset['LV2'], ASTM['LV2'], sep=' & ')
390 print('H1', tangente['H1'], offset['H1'], ASTM['H1'], sep=' & ')
391 print('H2', tangente['H2'], offset['H2'], ASTM['H2'], sep=' & ')
392 '''
393
394
395 #                                AJUSTES: plot de media
396 #-----
397 #-----
398 '''
399 metodos_array=['Rectas\ntangentes','offset', 'ASTM']
400 Ms_mediaarray=[tangente['Ms_media'], offset['Ms_media'],
401               ↪ ASTM['Ms_media']]
402 desv_Msarray=[tangente['desv_Ms'], offset['desv_Ms'],
403               ↪ ASTM['desv_Ms']]
404 #colores=['#FC5A50', '#15B01A', '#7E1E9C']
405 plt.figure()
406
407 for i, metodo in enumerate(metodos_array):
408     plt.errorbar([metodo],
409     ↪ Ms_mediaarray[i], yerr=desv_Msarray[i], color=Colores[i+1],
410     ↪ marker=Simbolos[6+i])
411     plt.annotate('{:.1f}'.format(Ms_mediaarray[i]), ([metodo],
412     ↪ Ms_mediaarray[i]+0.1))
413     #colocar el valor de Ms_media sobre el grafico
414     plt.annotate('+{:.1f}'.format(desv_Msarray[i]), ([metodo],
415     ↪ Ms_mediaarray[i]+desv_Msarray[i]-0.7))
416     #colocar el valor de Ms_media sobre el grafico
417     plt.annotate('+{:.1f}'.format(desv_Msarray[i]), ([metodo],
418     ↪ Ms_mediaarray[i]-desv_Msarray[i]+0.2))
419     #colocar el valor de Ms_media sobre el grafico
420
421 #title= #'Tratamiento térmico 1:\n1Cseg-1050C-30min-1Cseg\nMuestras:
422     ↪ {s}, {s}, {s}, {s}'.format(numeros[0], numeros[1],
423     ↪ numeros[2], numeros[3])
424 #prop={'family':'serif', "size":8}
425 #plt.legend(title=title, title_fontsize=7, prop=prop,
426     ↪ ncol=2, frameon=False)
427
428 Formato_ejes={'family':'serif', 'color':'black', 'size':9}

```

```

419 plt.ylabel('Ms /°C', fontdict=Formato_ejes)
420 titulo='Ms(media): distintos métodos\nMuestras: #{:s}, {:s}, {:s},
↳   {:s}'.format( numeros[0], numeros[1], numeros[2], numeros[4])
421 Formato_titulo={'family':'serif','color':'black','size':12}
↳   #'blue','darkred'
422 #plt.title(titulo,fontdict=Formato_titulo,loc='left')
423 carpeta='../Informe1/Especifico/'
424 plt.savefig(carpeta+'Ms_mediaydis.pdf')
425 plt.show()
426 plt.close()
427
428 '''
429
430 #                                     587
431 #-----
432 #-----
433 '''
434 numero='587'
435 archivo=dir_M+'587-besoky-1Cseg-1050C-30min-10Cseg-02.28.19'+'.asc'
436 tiempo, deformacion, temperatura=extraccion(archivo)
437 #extraccion de datos medidos directamente
438 muestra=Resultados[numero]
439 #muestra: contiene los datos analizados de cada muestra, dados por la
↳   definición de Ms
440 #Definicion de variables
441 #-----
442 dato[0]=buscador(Rango[0], temperatura,
↳   1.)
443 #da el extremo derecho del intervalo
444 dato[1]=buscador(Rango[1], temperatura,
↳   1.)
445 #da el extremo izquierdo del intervalo
446 T=temperatura[dato[0]:dato[1]]
447 #recordar que T y Def son la misma mismo para todos lo métodos, y
↳   son una medida directa, encambio...
448 Def=deformacion[dato[0]:dato[1]]
449 #p puede cambiar segun los parametros de ajuste para dicho metodo
450 muestra['T']=T
451 muestra['def']=Def
452 '''
453 #                                     Ms DEFINIDA POR METODO ASTM
454 #-----
455 '''
456 resumen_p(muestra['ASTM'], T, Def)
457 dato_Ms=int(muestra['ASTM']['dato_Ms'])
458 Ms=temperatura[dato_Ms]
459 p=muestra['ASTM']['p']
460 #MODELO DE HUYAN
461 B=0.006*(Ms+273.15)-0.1369
462 aux_DeltaG=100.*(p/(A*(1.-p)))*(1./B)

```

```

463 H1, H2=polyfit(T-Ms, aux_DeltaG, 1)
464 dato_Msprima=buscador(Ms, T, 1.)
465 print(dato_Msprima)
466 #MODELO DE KONSTINEN Y MARBURGER
467 KM, cov_KM=curve_fit(f_KM, Ms-T[dato_Msprima-1:], p[dato_Msprima-1:],
    ↪ KMO)
468 #MODELO DE LEE Y VAN TYNE
469 LV, cov_LV = curve_fit(f_LV, Ms-T[dato_Msprima+1:],
    ↪ p[dato_Msprima+1:], coef)
470 muestra['ASTM']['KM']=KM[0]
471 muestra['ASTM']['LV1']=LV[0]
472 muestra['ASTM']['LV2']=LV[1]
473 muestra['ASTM']['H1']=H1
474 muestra['ASTM']['H2']=H2
475 plt.figure()
476 plt.plot(T, p,color='#7E1E9C', linestyle='-',
    ↪ linewidth=1.,label='Regla de la palanca')
477 plt.plot(T, f_H(Ms-T, A, B, H1, H2), color='#808080', linestyle='--',
    ↪ linewidth=1.15, label='Modelo de Huyan y colaboradores')
478 plt.plot(T[dato_Msprima-1:], f_KM(Ms-T[dato_Msprima-1:],
    ↪ KM[0]),color='#15B01A', linestyle='--',
    ↪ linewidth=1.15,label='Modelo de Koistinen y Marburger')
479 plt.plot(T[dato_Msprima+1:], f_LV(Ms-T[dato_Msprima+1:],
    ↪ *tuple(LV)),color='#FC5A50', linestyle='--', linewidth=1.15,
    ↪ label='Modelo de Lee y Van Tyne')
480 plt.plot(Ms, p[dato_Msprima], color='#000000', linestyle='-',
    ↪ marker='.' , markersize=10, label='Ms={:0.1f} °C'.format(Ms))
481 plt.legend()
482 Formato_ejes={'family':'serif','color':'black','size':12}
483 plt.xlabel('T/°C',fontdict=Formato_ejes)
484 plt.ylabel('fracción de martensita
    ↪ transformada',fontdict=Formato_ejes)
485 title=numero+'. Diferentes modelos de ajuste: Tratamiento térmico
    ↪ 2\nMs definido por metodo ASTM'
486 #plt.title(title,
    ↪ fontdict={'family':'serif','color':'black','size':10},loc='left')
487 plt.savefig('../Informe1/ASTM/{:s}Modelos_Ms(ASTM).pdf'.format(numero))
488 #plt.savefig('../Informe1/ASTM/Metodos_Ms(ASTM)zoom.pdf')
489 plt.show()
490 plt.close()
491 print('paso2a')
492 '''
493 # Ms DEFINIDA POR METODO OFFSET
494 #-----
495 '''
496 resumen_p(muestra['offset'], T, Def)
497 dato_Ms=int(muestra['offset']['dato_Ms'])
498 Ms=temperatura[dato_Ms]
499 p=muestra['offset']['p']
500 #MODELO DE HUYAN

```

```

501 B=0.006*(Ms+273.15)-0.1369
502 aux_DeltaG=100.*(p/(A*(1.-p)))*(1./B)
503 H1, H2=polyfit(T-Ms, aux_DeltaG, 1)
504 dato_Msprima=buscador(Ms, T, 1.)
505
506 #MODELO DE KONSTINEN Y MARBURGER
507 KM, cov_KM=curve_fit(f_KM, Ms-T[dato_Msprima-1:], p[dato_Msprima-1:],
    ↪ KM0)
508 #MODELO DE LEE Y VAN TYNE
509 LV, cov_LV = curve_fit(f_LV, Ms-T[dato_Msprima+1:],
    ↪ p[dato_Msprima+1:], coef)
510
511 muestra['offset']['KM']=KM[0]
512 muestra['offset']['LV1']=LV[0]
513 muestra['offset']['LV2']=LV[1]
514 muestra['offset']['H1']=H1
515 muestra['offset']['H2']=H2
516
517 plt.figure()
518 plt.plot(T, p,color='#7E1E9C', linestyle='-',
    ↪ linewidth=1.,label='Regla de la palanca')
519 plt.plot(T, f_H(Ms-T, A, B, H1, H2), color='#808080', linestyle='--',
    ↪ linewidth=1.15, label='Modelo de Huyan y colaboradores')
520 plt.plot(T[dato_Msprima-1:], f_KM(Ms-T[dato_Msprima-1:],
    ↪ KM[0]),color='#15B01A', linestyle='--',
    ↪ linewidth=1.15,label='Modelo de Koistinen y Marburger')
521 plt.plot(T[dato_Msprima+1:], f_LV(Ms-T[dato_Msprima+1:],
    ↪ *tuple(LV)),color='#FC5A50', linestyle='--', linewidth=1.15,
    ↪ label='Modelo de Lee y Van Tyne')
522 plt.plot(Ms, p[dato_Msprima], color='#000000', linestyle='-',
    ↪ marker='.' , markersize=10, label='Ms={:0.1f} °C'.format(Ms))
523 plt.legend()
524 Formato_ejes={'family':'serif','color':'black','size':12}
525 plt.xlabel('T/°C',fontdict=Formato_ejes)
526 plt.ylabel('fracción de martensita
    ↪ transformada',fontdict=Formato_ejes)
527 title=numero+'. Diferentes modelos de ajuste: Tratamiento térmico
    ↪ 2\nMs definido por metodo offset'
528 #plt.title(title,
    ↪ fontdict={'family':'serif','color':'black','size':10},loc='left')
529 plt.savefig('../Informe1/offset/{:s}Modelos_Ms(offset).pdf'.format(numero))
530 plt.show()
531 plt.close()
532 print('paso2b')
533 '''
534 #                               Ms DEFINIDA POR METODO RECTAS TG
535 #-----
536 '''
537 resumen_p(muestra['tangente'], T, Def)
538 dato_Ms=int(muestra['tangente']['dato_Ms'])

```

```

539 Ms=temperatura[dato_Ms]
540 p=muestra['tangente']['p']
541 #MODELO DE HUYAN
542 B=0.006*(Ms+273.15)-0.1369
543 aux_DeltaG=100.*(p/(A*(1.-p)))*(1./B)
544 H1, H2=polyfit(T-Ms, aux_DeltaG, 1)
545 dato_Msprima=buscador(Ms, T, 1.)
546 #MODELO DE KONSTINEN Y MARBURGER
547 KM, cov_KM=curve_fit(f_KM, Ms-T[dato_Msprima-1:], p[dato_Msprima-1:],
    ↪ KMO)
548 #MODELO DE LEE Y VAN TYNE
549 LV, cov_LV = curve_fit(f_LV, Ms-T[dato_Msprima+1:],
    ↪ p[dato_Msprima+1:], coef)
550 muestra['tangente']['KM']=KM[0]
551 muestra['tangente']['LV1']=LV[0]
552 muestra['tangente']['LV2']=LV[1]
553 muestra['tangente']['H1']=H1
554 muestra['tangente']['H2']=H2
555
556 plt.figure()
557 plt.plot(T, p,color='#7E1E9C', linestyle='-',
    ↪ linewidth=1.,label='Regla de la palanca')
558 plt.plot(T, f_H(Ms-T, A, B, H1, H2), color='#808080', linestyle='--',
    ↪ linewidth=1.15, label='Modelo de Huyan y colaboradores')
559 plt.plot(T[dato_Msprima-1:], f_KM(Ms-T[dato_Msprima-1:],
    ↪ KM[0]),color='#15B01A', linestyle='--',
    ↪ linewidth=1.15,label='Modelo de Koistinen y Marburger')
560 plt.plot(T[dato_Msprima+1:], f_LV(Ms-T[dato_Msprima+1:],
    ↪ *tuple(LV)),color='#FC5A50', linestyle='--', linewidth=1.15,
    ↪ label='Modelo de Lee y Van Tyne')
561 plt.plot(Ms, p[dato_Msprima], color='#000000', linestyle='-',
    ↪ marker='.' , markersize=10, label='Ms={:0.1f} °C'.format(Ms))
562 plt.legend()
563 Formato_ejes={'family':'serif','color':'black','size':12}
564 plt.xlabel('T/°C',fontdict=Formato_ejes)
565 plt.ylabel('fracción de martensita
    ↪ transformada',fontdict=Formato_ejes)
566 title=numero+'. Diferentes modelos de ajuste: Tratamiento térmico
    ↪ 2\nMs definido por metodo de las rectas tangentes'
567 #plt.title(title,
    ↪ fontdict={'family':'serif','color':'black','size':10},loc='left')
568 plt.savefig('../Informe1/tg/{:s}Modelos_Ms(tg).pdf'.format(numero))
569 plt.show()
570 plt.close()
571 print('paso2c')
572 '''
573 #PRINTEO
574 '''
575 print('tangente', 'offset', 'ASMT')

```

```

576 print('KM', muestra['tangente']['KM'], muestra['offset']['KM'],
      ↪ muestra['ASTM']['KM'], sep=' & ')
577 print('LV1',muestra['tangente']['LV1'], muestra['offset']['LV1'],
      ↪ muestra['ASTM']['LV1'], sep=' & ')
578 print('LV2', muestra['tangente']['LV2'], muestra['offset']['LV2'],
      ↪ muestra['ASTM']['LV2'], sep=' & ')
579 print('H1', muestra['tangente']['H1'], muestra['offset']['H1'],
      ↪ muestra['ASTM']['H1'], sep=' & ')
580 print('H2',muestra['tangente']['H2'], muestra['offset']['H2'],
      ↪ muestra['ASTM']['H2'], sep=' & ')
581
582 '''
583
584 #                               SEGUN LA COMPOSICION 1
585 #-----
586 '''
587 parametros=parametros_composicion(PMas_nominal)
588 #pprint(parametros)
589 #Definicion de variables
590 #-----
591 numero='586'
592 archivo=dir_M+'586-besoky-1Cseg-1050C-30min-1Cseg-02.28.19'+'.asc'
593 tiempo, deformacion, temperatura=extraccion(archivo)
594 #extraccion de datos medidos directamente
595 muestra=Resultados[numero]
596 #muestra: contiene los datos analizados de cada muestra, dados por la
      ↪ definición de Ms
597 Rango=[410,260]
598 dato[0]=buscador(Rango[0], temperatura,
      ↪ 1.)
599 #da el extremo derecho del intervalo
600 dato[1]=buscador(Rango[1], temperatura,
      ↪ 1.)
601 #da el extremo izquierdo del intervalo
602 T=temperatura[dato[0]:dato[1]]
603 #recordar que T y Def son la misma mismo para todos lo métodos, y
      ↪ son una medida directa, encambio...
604 Def=deformacion[dato[0]:dato[1]]
605 #...p puede cambiar segun los parametros de ajuste para dicho metodo
606 muestra['T']=T
607 muestra['def']=Def
608 resumen_p(muestra['ASTM'], T, Def)
609 p=muestra['ASTM']['p']
610 '''
611 #MODELO DE KONSTINEN Y MARBURGER:
612 #-----
613 '''
614 plt.figure()
615 plt.plot(T, p,color='#7E1E9C', linestyle='-', linewidth=1.,
      ↪ label='Regla de la palanca')

```

```

616 #van Bohemen
617 #-----
618 Modelo=parametros['vanBohemen']
619 Ms=Modelo['Ms']
620 KM=Modelo['parametro1']
621 dato_Msprima=buscador(Ms, T, 1.)
622 plt.plot(T[dato_Msprima-1:], f_KM(Ms-T[dato_Msprima-1:],
    ↪ KM),color='#9ACD32', linestyle='--',
    ↪ linewidth=1.15,label='Parametros van Bohemen')
623 plt.plot(Ms,0., 'o', color='#15B01A' , label='Ms={:0.1f}
    ↪ °C'.format(Ms))
624 #van Bohemen y Sietsma
625 #-----
626 Modelo=parametros['vanBohemen_Sietsma']
627 Ms=Modelo['Ms']
628 KM=Modelo['parametro1']
629 dato_Msprima=buscador(Ms, T, 1.)
630 plt.plot(T[dato_Msprima-1:], f_KM(Ms-T[dato_Msprima-1:], KM),
    ↪ color='#808080', linestyle='--',
    ↪ linewidth=1.15,label='Parametros van Bohemen y Sietsma')
631 plt.plot(Ms,0., 'o' ,color='#000000',label='Ms={:0.1f}
    ↪ °C'.format(Ms))
632 #Detalles del grafico
633 plt.legend(frameon=False,prop={'family':'serif',"size":7})
634 title='Modelos de Koistinen y Marburger\nParámetros calculados en
    ↪ función de la composición'
635 plt.title(title,
    ↪ fontdict={'family':'serif','color':'black','size':10},loc='left')
636 plt.savefig('../Informe1/Especifico/Composicion_KM.pdf')
637 plt.show()
638 plt.close()
639 #MODELO DE LEE Y VAN TYNE:
640 #-----
641 plt.figure()
642 plt.plot(T, p,color='#7E1E9C', linestyle='-', linewidth=1.,
    ↪ label='Regla de la palanca')
643 Modelo=parametros['Lee_VanTyne']
644 Ms=Modelo['Ms']
645 LV=Modelo['parametro1'], Modelo['parametro2']
646 dato_Msprima=buscador(Ms, T, 1.)
647 plt.plot(T[dato_Msprima:], f_LV(Ms-T[dato_Msprima:],
    ↪ *LV),color='#9ACD32', linestyle='--', linewidth=1.15,
    ↪ label='Modelo de Lee y Van Tyne')
648 #Detalles del grafico
649 plt.plot(Ms,0., 'o', color='#15B01A', label='Ms={:0.1f}
    ↪ °C'.format(Ms))
650 plt.legend(frameon=False,prop={'family':'serif',"size":7})
651 title='Modelos de Lee y Van Tyne\nParámetros calculados en función de
    ↪ la composición'

```

```

652 plt.title(title,
    ↪ fontdict={'family':'serif','color':'black','size':10},loc='left')
653 plt.savefig('../Informe1/Especifico/Composicion_LV.pdf')
654 plt.show()
655 plt.close()
656
657 #FINKLE Y sCHIRRA:
658 #-----
659 Modelo=parametros['Finkler_Schirra']
660 Ms=Modelo['Ms']
661 dato_Msprima=buscador(Ms, T, 1.)
662 #MODELO DE HUYAN
663 A=0.05
664 B=0.006*(Ms+273.15)-0.1369
665 aux_DeltaG=100.*(p/(A*(1.-p)))*(1./B)
666 H1, H2=polyfit(T-Ms, aux_DeltaG, 1)
667 #MODELO DE KONSTINEN Y MARBURGER
668 KM, cov_KM=curve_fit(f_KM, Ms-T[dato_Msprima-1:], p[dato_Msprima-1:],
    ↪ KM0)
669 #MODELO DE LEE Y VAN TYNE
670 LV, cov_LV = curve_fit(f_LV, Ms-T[dato_Msprima+1:],
    ↪ p[dato_Msprima+1:], coef)
671 plt.figure()
672 plt.plot(T, p,color='#7E1E9C', linestyle='-',
    ↪ linewidth=1.,label='Regla de la palanca')
673 plt.plot(T, f_H(Ms-T, A, B, H1, H2), color='#808080', linestyle='--',
    ↪ linewidth=1.15, label='Modelo de Huyan y colaboradores')
674 plt.plot(T[dato_Msprima-1:], f_KM(Ms-T[dato_Msprima-1:],
    ↪ KM[0]),color='#15B01A', linestyle='--',
    ↪ linewidth=1.15,label='Modelo de Koistinen y Marburger')
675 plt.plot(T[dato_Msprima+1:], f_LV(Ms-T[dato_Msprima+1:],
    ↪ *tuple(LV)),color='#FC5A50', linestyle='--', linewidth=1.15,
    ↪ label='Modelo de Lee y Van Tyne')
676 plt.plot(Ms, p[dato_Msprima], color='#000000', linestyle='-',
    ↪ marker='.', markersize=10, label='Ms={:0.1f} °C'.format(Ms))
677 plt.legend()
678 title='Diferentes modelos de ajuste:\nMs definida según Finkler y
    ↪ Schirra'
679 plt.title(title,
    ↪ fontdict={'family':'serif','color':'black','size':10},loc='left')
680 plt.savefig('../Informe1/Especifico/Composicion_FS.pdf')
681 plt.show()
682 plt.close()
683 '''
684
685
686 '''
687 {'Finkler_Schirra': {'Ms': 384.09473},
688  'Lee_VanTyne': {'Ms': 369.70000000000005,
689                  'parametro1': 0.066859,

```

```

690         'parametro2': 0.970658},
691     'vanBohemen': {'Ms': 398.83840964122373,
692                   'parametro1': 0.023262571969346165},
693     'vanBohemen_Sietsma': {'Ms': 287.73,
694                             'parametro1': 0.0199781}}
695     '''
696     #          SEGUN LA COMPOSICION 2
697     #-----
698
699     parametros=parametros_composicion(PMas_nominal)
700     p_media=ASTM['p_media']
701     desv_p=ASTM['desv_p']
702
703
704     '''
705     #MODELO DE VAN BOHEMEN Y SIETSMA:
706     #-----
707     Modelo=parametros['vanBohemen_Sietsma']
708     Ms=Modelo['Ms']
709     KM=Modelo['parametro1']
710     dato_Msmedia=buscador(Ms, T_media, 1.)
711     plt.figure()
712     plt.plot(T_media, p_media,color='#7E1E9C', linestyle='-',
713              ↳ linewidth=1.,label='Regla de la palanca')
714     leyenda_desv='Desviación estandar\nde la fracción
715     ↳ transformada\n(Muestras #541,584,586,590)'
716     plt.fill_between(T_media, p_media+desv_p, p_media-desv_p,alpha=0.2,
717     ↳ edgecolor='#580F41', facecolor='#7E1E9C', linewidth=1.15,
718     ↳ antialiased=True, label=leyenda_desv)
719     plt.plot(T_media[dato_Msmedia-1:], f_KM(Ms-T_media[dato_Msmedia-1:],
720     ↳ KM),color='#15B01A', linestyle='--', linewidth=1.15,label='Van
721     ↳ Bohemen y Sietsma')
722     plt.plot(Ms, 0., color='#15B01A', linestyle='-', marker='.' ,
723     ↳ markersize=10, label='Ms={:0.1f}°C'.format(Ms))
724     '''
725     #MODELO DE VAN BOHEMEN
726     #-----
727     '''
728     Modelo=parametros['vanBohemen']
729     Ms=Modelo['Ms']
730     KM=Modelo['parametro1']
731     dato_Msmedia=buscador(Ms, T, 1.)
732     plt.plot(T_media[dato_Msmedia-1:], f_KM(Ms-T_media[dato_Msmedia-1:],
733     ↳ KM),color='#808080', linestyle='--', linewidth=1.15,label='Van
734     ↳ Bohemen')
735     plt.plot(Ms, 0., color='#808080', linestyle='-', marker='.' ,
736     ↳ markersize=10, label='Ms={:0.1f}°C'.format(Ms))
737     plt.legend(title='Modelo de Koistinen y Marburger')
738     Formato_ejes={'family':'serif','color':'black','size':12}
739     plt.xlabel('T/°C',fontdict=Formato_ejes)

```

```

730 plt.ylabel('fracción de martensita
    ↳ transformada',fontdict=Formato_ejes)
731 title='Modelos de Koistinen y Marburger\nParámetros calculados en
    ↳ función de la composición'
732 #plt.title(title,
    ↳ fontdict={'family':'serif','color':'black','size':10},loc='left')
733 plt.savefig('../Informe1/Especifico/Composicion_KM.pdf')
734 plt.show()
735 plt.close()
736 '''
737
738
739 #MODELO DE LEE Y VAN TYNE:
740 #-----
741 '''
742 Modelo=parametros['Lee_VanTyne']
743 Ms=Modelo['Ms']
744 LV=Modelo['parametro1'], Modelo['parametro2']
745 dato_Msmedia=buscador(Ms, T_media, 1.)
746 plt.figure()
747 plt.plot(T_media, p_media,color='#7E1E9C', linestyle='-',
    ↳ linewidth=1.,label='Regla de la palanca')
748 leyenda_desv='Desviación estandar\nde la fracción
    ↳ transformada\n(Muestras #541,584,586,590)'
749 plt.fill_between(T_media, p_media+desv_p, p_media-desv_p,alpha=0.2,
    ↳ edgecolor='#580F41', facecolor='#7E1E9C', linewidth=1.15,
    ↳ antialiased=True, label=leyenda_desv)
750 plt.plot(T_media[dato_Msmedia+1:], f_LV(Ms-T_media[dato_Msmedia+1:],
    ↳ *LV),color='#FC5A50', linestyle='--', linewidth=1.15,
    ↳ label='Modelo de Lee y Van Tyne')
751 plt.plot(Ms, p_media[dato_Msmedia], color='#FC5A50', linestyle='-',
    ↳ marker='.' , markersize=10, label='Ms={:0.1f}°C'.format(Ms))
752 plt.legend()
753 Formato_ejes={'family':'serif','color':'black','size':12}
754 plt.xlabel('T/°C',fontdict=Formato_ejes)
755 plt.ylabel('fracción de martensita
    ↳ transformada',fontdict=Formato_ejes)
756 title='Modelos de Lee y Van Tyne\nParámetros calculados en función de
    ↳ la composición'
757 #plt.title(title,
    ↳ fontdict={'family':'serif','color':'black','size':10},loc='left')
758 plt.savefig('../Informe1/Especifico/Composicion_LV.pdf')
759 #plt.savefig('../Informe1/ASTM/Modelos_Ms(ASTM)zoom.pdf')
760 plt.show()
761 plt.close()
762 '''
763 #FINKLE Y SCHIRRA:
764 #-----
765
766 Modelo=parametros['Finkler_Schirra']

```

```

767 Ms=Modelo['Ms']
768 #MODELO DE HUYAN
769 B=0.006*(Ms+273.15)-0.1369
770 aux_DeltaG=100.*(p_media/(A*(1.-p_media)))*(1./B)
771 H1, H2=polyfit(T_media-Ms, aux_DeltaG, 1)
772 dato_Msmedia=buscador(Ms, T_media, 1.)
773 #MODELO DE KONSTINEN Y MARBURGER
774 KM, cov_KM=curve_fit(f_KM, Ms-T_media[dato_Msmedia-1:],
    ↪ p_media[dato_Msmedia-1:], KM0)
775 #MODELO DE LEE Y VAN TYNE
776 LV, cov_LV = curve_fit(f_LV, Ms-T_media[dato_Msmedia+1:],
    ↪ p_media[dato_Msmedia+1:], coef)
777 plt.figure()
778 plt.plot(T_media, p_media,color='#7E1E9C', linestyle='-',
    ↪ linewidth=1.,label='Regla de la palanca')
779 leyenda_desv='Desviación estandar\nde la fracción
    ↪ transformada\n(Muestras #541,584,586,590)'
780 plt.fill_between(T_media, p_media+desv_p, p_media-desv_p,alpha=0.2,
    ↪ edgecolor='#580F41', facecolor='#7E1E9C', linewidth=1.15,
    ↪ antialiased=True, label=leyenda_desv)
781 plt.plot(T_media, f_H(Ms-T_media, A, B, H1, H2), color='#808080',
    ↪ linestyle='--', linewidth=1.15, label='Modelo de Huyan y
    ↪ colaboradores')
782 plt.plot(T_media[dato_Msmedia-1:], f_KM(Ms-T_media[dato_Msmedia-1:],
    ↪ KM[0]),color='#15B01A', linestyle='--',
    ↪ linewidth=1.15,label='Modelo de Koistinen y Marburger')
783 plt.plot(T_media[dato_Msmedia+1:], f_LV(Ms-T_media[dato_Msmedia+1:],
    ↪ *tuple(LV)),color='#FC5A50', linestyle='--', linewidth=1.15,
    ↪ label='Modelo de Lee y Van Tyne')
784 plt.plot(Ms, p_media[dato_Msmedia], color='#000000', linestyle='-',
    ↪ marker='.', markersize=10, label='Ms={:0.1f}°C'.format(Ms))
785 plt.legend()
786 Formato_ejes={'family':'serif','color':'black','size':12}
787 plt.xlabel('T/°C',fontdict=Formato_ejes)
788 plt.ylabel('fracción de martensita
    ↪ transformada',fontdict=Formato_ejes)
789 title='Diferentes modelos de ajuste:\nMs definida según Finkler y
    ↪ Schirra'
790 #plt.title(title,
    ↪ fontdict={'family':'serif','color':'black','size':10},loc='left')
791 plt.savefig('../Informe1/Especifico/Composicion_FS.pdf')
792 #plt.savefig('../Informe1/ASTM/Modelos_Ms(ASTM)zoom.pdf')
793 plt.show()
794 plt.close()
795
796 '''
797 print('tangente', 'offset', 'ASMT')
798 print('KM', KM[0], sep=' & ')
799 print('LV1', LV[0], sep=' & ')
800 print('LV2', LV[1], sep=' & ')

```

```
801 print('H1', H1, sep=' & ')
802 print('H2', H2, sep=' & ')
803 '''
804
805 print('listo')
```

8.6. Curvas de dilatometría y tratamientos térmicos

```

1  """Florenxia Antonella Correa
2  dilatometria_TT.py
3  Graficos de curvas dilatometricas y tratamientos termicos
4  """
5  import numpy as np
6  import matplotlib.pyplot as plt
7  import operator
8  # import copy
9
10 from Especifico_funciones import ## todas las funciones creadas en
    ↪ Especifico_funciones
11 from scipy.optimize import curve_fit
12 from Especifico_funciones import f_KM
13 from pprint import pprint
14 from matplotlib.legend_handler import HandlerLine2D, HandlerTuple
15 from matplotlib import pyplot as plt
16 from matplotlib.legend import _get_legend_handles_labels
17 from copy import copy, deepcopy
18
19
20
21 # Notas sobre importaciones:
22 '''si uso
23 import copy          --->    copy.copy(...) o copy.deepcopy(...)
24 from copy import copy, deepcopy      --->    copy(...) o
    ↪ deepcopy(...)'''
25
26 """Directorio o root:
27 dir_M: carpeta de transformacion martensitica
28 dir_python: donde se encuentran los ejecutables
29 dir_TC: donde se encuentra el archivo con las composiciones devuelto
    ↪ por thermocalc.
30 """
31 dir_M='../Datos dilatometria/Transformacion_martensitica/'
32 dir_python='C:/Users/flor/Documents/ITS/Trabajo Final/Python'
33 dir_TC='../Ejemplos Thermocalc/'
34 carpeta='../Informe1/Especifico/'
35 # Listas de archivos:
36 #-----
37 nombres_M=['541-besoky-1Cseg-1050C-30min-1Cseg-12.12.17',
38 '584-besoky-1Cseg-1050C-30min-1Cseg-02.21.19',
39 '586-besoky-1Cseg-1050C-30min-1Cseg-02.28.19',
40 '587-besoky-1Cseg-1050C-30min-10Cseg-02.28.19',
41 '590-besoky-1Cseg-1050C-30min-1Cseg-04.16.19']
42 numeros=['541', '584', '586', '587', '590']
43 Tratamiento_1=['541', '584', '586', '589']
44

```

```

45  """Tbusqueda:      matriz de temperaturas características para el
    ↪ procesamiento, durante el enfriamiento
46  Columnas:
47  0- comienzo de enfriamiento
48  1- cercano al comienzo de la transformación
49  2- punto durante la transformación de mayor temperatura
50  3- punto durante la transformación de menor temperatura
51  Filas              corresponde a cada muestra
52  """
53  #-----
54  # GENERAL
55  #-----
56  # Parametros de graficos
57  Colores=['k', 'r', 'b', 'g', 'c', 'y', 'm', 'w']
58  Simbolos=['_', '-', '._', ':', '.', ',', 'o', 'v', 's', 'p', '+', '*',
    ↪ '^', '<', '>', 'x', 'D', 'd']
59  posicion=['upper', 'lower', 'lower', 'lower', 'lower', 'lower']
60  # Formato y titulos
61  Titulo='Comparación de métodos'
62  Titulosup='Ajuste K-M y definición de Ms'
63  Formato_titulo={'family':'serif','color':'black','size':12}# 'blue', 'darkred'
64  Formato_titulosup={'family':'serif','color':'black','size':15}
65  Formato_ejes={'family':'serif','color':'black','size':13}
66  Formato_label={'family':'serif','size':9}
67  # Otros parametros
68  n=len(nombres_M)
69  extra=2
70  epsilon=1.
71  beta0=0.011
72  Ms=[]
73  Metodos=['Tg', 'ASTM', 'offset']
74  #-----
75  # TANGENTE
76  #-----
77  Ms_tg=np.zeros(n)
78  epsilon_tg=1.
79  var_tg=[]
80  p_tg=[]
81
82  #-----
83  # ASTM
84  #-----
85  m=4# aumento para mayor localización del ajuste ASTM tome hasta 450
86  epsilon_ASTM=.01
87  Ms_ASTM=np.zeros(n)
88  par_ASTM=[]
89  dato_1=np.zeros(m)
90  # Fracción martensítica transformada determinada por método ASTM'
91  Titulo_ASTM='Variación de parámetros en ajuste lineal'
92  Titulosup_ASTM='Ajuste K-M y definición de Ms por método ASTM'

```

```

93 aux_enc=[]
94 Taux0=[]
95 T0=[]
96 var_ASTM=[]
97 p_ASTM=[]
98
99 #-----
100 # OFFSET
101 #-----
102 epsilon_off=1.
103 Ms_off=np.zeros(n)
104 Ms_off1=np.zeros(n)
105 par_off=[]
106 # 'Fracción martensítica transformada determinada por método offset'
    ↪
107 Titulo_offset='Variación de parámetros en ajuste lineal'
108 Titulosup_offset='Ajuste K-M y definición de Ms por método offset'
109 var_off=[]
110 p_off=[]
111 fV=[0.01, 0.015, 0.02, .03]
112 V=0.02
113 #-----
114 # COMPOSICIONES
115 #-----
116 Elementos=['C','CU','AL','N','SB','MN','SN','TI','B','PB',
    ↪ 'SI','NI','NB','CA','P','CR','V','O','S','MO','CO','AS',
    ↪ 'FE']
117 Contenido_acero=np.array([0.1,0.16, 0.009, 0.0605,0.001,0.36, 0.008,
    ↪ 0.004, 0.0001, 0.001, 0.24, 0.15, 0.075, 0.0008, 0.014, 8.37,
    ↪ 0.211, .0017, .001, .88, .01, .005, 88.3379 ])
118 FMas_nominal=dict(zip(Elementos, list(Contenido_acero/100.)))
119 FMol_nominal=FraccionMolar(FMas_nominal)
120 PMas_nominal=dict(zip(Elementos, list(Contenido_acero)))
121 # composiciones en equilibrio T=25, 1050°C respectivamente
122 FMas_gama=comp_fase(dir_TC+'f_gamabis.dat', 'FCC_A1# 1')
123 FMol_gama=FraccionMolar(FMas_gama[0])
124 eta=eta_offset(FMol_nominal, FMas_nominal, V)
125 etaprima=eta_offset(FMol_gama, FMas_gama[0], V)
126
127 # Defini
128 Tbusqueda={'541':[1050, 400,370, 350],
129            '584':[1050, 400, 360, 345],
130            '586':[1050, 400, 365, 350],
131            '587':[1050, 400, 350, 320],
132            '590':[1050, 400, 365, 355]}
133 exceptuar={'541':[530,420],
134            '584':[580,455],
135            '586':[550,450],
136            '587':[],[]},
137            '590':[540,440]}

```

```

138 m=4
139 T_derecha=linspace(750, 600, m)
140 dato_derecho=[0, 0, 0, 0]
141 inter_ASTM=2
142 inter_offset=0
143 aux=[0,0,0,0]
144 ex=[0,0]
145 #.....
146 #-----
147 #-----Graficos de tratamiento térmico y dilatométrica-----
148 #-----
149 #-----
150
151 for i, nombre in enumerate(nombres_M):
152     print(numeros[i])
153     archivo=dir_M+nombre+'.asc'# apertura de archivo
154     tiempo, deformacion, temperatura=extraccion(archivo)# extraccion
155 # Busqueda de puntos relevantes:
156 aux[0]=buscador(Tbusqueda[numeros[i]][0], temperatura, epsilon)
157 aux[1]=buscador(Tbusqueda[numeros[i]][1], temperatura, epsilon)
158 aux[2]=buscador(Tbusqueda[numeros[i]][2], temperatura, epsilon)
159 aux[3]=buscador(Tbusqueda[numeros[i]][3], temperatura, epsilon)
160 # Busqueda de puntos a exceptuar
161 ex[0]=buscador(exceptuar[numeros[i]][0], temperatura, epsilon)
162 ex[1]=buscador(exceptuar[numeros[i]][1], temperatura, epsilon)
163 # dado que no posee perturbaciones que interfiera en la
    ↪ definición de ajuste
164 if numeros[i] in '587':
165     ex[0]=aux[1]
166     ex[1]=aux[1]
167 # Graficos Ciclos termicos y de dilatométrica
168 #-----
169 fig=plt.figure()
170 for k, T_der in enumerate(T_derecha):
171     color=Simbolos[6+k]+' '+Colores[k]
172     # busqueda de puntos del extremo derecho del intervalo de
        ↪ ajuste
173 dato_der=buscador(T_der, temperatura, epsilon)
174 label='{ :1d}: { :1.1f}-{:1.1f}'
        ↪ '°C'.format(k,T_derecha[k],Tbusqueda[numeros[i]][1])
175 # marca de extremo derecho del ontervalo de ajuste
176 plt.plot(temperatura[dato_der], deformacion[dato_der], color,
        ↪ label=label)
177 # coloca etiqueta de extremo der de los intervalos
        ↪ considerados1,2, 3,4
178 plt.annotate(str(k), (temperatura[dato_der]-20,
        ↪ deformacion[dato_der]+2))
179 # coloca etiqueta de extremo izq de los intervalos
        ↪ considerados 1,2, 3,4

```

```

180         plt.annotate('{:.0f}'.format(T_der), (T_der-10,
        ↪         deformacion[dato_der]-7))
181
182     plt.plot(temperatura[aux[1]], deformacion[aux[1]], '* y', label=
        ↪     'Extremo izq. de intervalo', markersize=8.5)
183     # coloca etiqueta de extremo izq de los intervalos considerados
        ↪     1,2, 3,4
184     plt.annotate(Tbusqueda[numeros[i]][1], (temperatura[aux[1]]+10,
        ↪     deformacion[aux[1]]-2))
185     # parte1 de curva de enfriamiento
186     plt.plot(temperatura[aux[0]:ex[0]], deformacion[aux[0]:ex[0]],
        ↪     '-b', label='Curva de enfriamiento')
187     # parte2 de la curva de enfriamiento y ademas intervalo excluido
188     plt.plot(temperatura[ex[0]:ex[1]], deformacion[ex[0]:ex[1]],
        ↪     '--b', label='Intervalo no considerado')
189     # parte3 de la curva de enfriamiento
190     plt.plot(temperatura[ex[1]:], deformacion[ex[1]:], '-b')
191     plt.plot(temperatura[-1], deformacion[-1], 'd m', label='Intervalo
        ↪     ajuste: fase martensita')
192     plt.plot(temperatura[-150], deformacion[-150], 'd m')
193     # REpetición solo para que el simbolo quede arriba
194     for k, T_der in enumerate(T_derecha):
195         color=Simbolos[6+k]+' '+Colores[k]
196         # busqueda de puntos del extremo derecho del intervalo de
        ↪         ajuste
197         dato_der=buscador(T_der, temperatura, epsilon)
198         # marca de extremo derecho del ontervalo de ajuste
199         plt.plot(temperatura[dato_der], deformacion[dato_der], color)
200     # print(dato_der,T_der,sep=' ')
201     # ,label= 'Extremo izq. de intervalo')
202     plt.plot(temperatura[aux[1]], deformacion[aux[1]], '* y')
203     # Parametros del grafico
204     plt.xlabel('T/°C', fontdict=Formato_ejes)
205     plt.ylabel('Deformación', fontdict=Formato_ejes)
206     # plt.title(numeros[i]+' . Curva de
        ↪     dilatometría', fontdict=Formato_titulosup, loc='left')
207     #
208
209     plt.legend(frameon=False, prop={'family': 'serif', "size": 6.5},
        ↪     loc='lower right', title_fontsize=6.5, title='Extremo der. de
        ↪     intervalo:')
210     # Parametros para el subgrafico
211     ax=fig.add_subplot(2, 2, 1)
212     ax.yaxis.set_ticks_position('right') # Coloca el eje a la derecha
213     plt.gca().yaxis.set_label_position("right") # Coloca el label a la
        ↪     derecha
214     plt.ylabel('Deformacion', fontdict=Formato_ejes) # ylabel
215     plt.plot(temperatura[:aux[0]], deformacion[:aux[0]],
        ↪     '-r', label='Curva de calentamiento')
216     plt.plot(temperatura[aux[0]:], deformacion[aux[0]:], '-b')

```

```

217     plt.savefig(carpeta+numeros[i]+'d.pdf')
218 # plt.savefig(carpeta+numeros[i]+'d.png')
219
220 # if numeros[i]=='541':
221 #     plt.show()
222     plt.show()
223     plt.close()
224 #----tratamiento termico
225 #-----
226     '''Formato_ejes={'family':'serif','color':'black','size':13}
227     plt.plot(tiempo, temperatura)
228     plt.ylabel('T/°C',fontdict=Formato_ejes)
229     plt.xlabel('tiempo/s',fontdict=Formato_ejes)
230 # plt.title(numeros[i]+' . Tratamiento
    ↪     termico',fontdict=Formato_titulosup,loc='left')
231     if i==3:
232 # plt.title('Tratamiento térmico 1: \nMuestras: #541, 584, 586,
    ↪     590',fontdict=Formato_titulosup,loc='left')
233 # plt.savefig(carpeta+numeros[i]+'TT1.pdf')
234 # plt.title('Tratamiento térmico 2: \nMuestras:
235 # #587',fontdict=Formato_titulosup,loc='left')
236
237     plt.savefig(carpeta+numeros[i]+'TT2.pdf')
238     plt.show()
239 # plt.savefig(carpeta+numeros[i]+'TT.pdf')
240 # plt.savefig(carpeta+numeros[i]+'TT.png')
241 # if i==0:
242 #     plt.show()
243     plt.close()
244     '''

```

8.7. Funciones

```

1  """Florenxia Antonella Correa
2  Especifico_funciones
3  Se desenvuelven las funciones pque se aplican en los otros archivos.
4  """
5  import numpy as np
6  import matplotlib.pyplot as plt
7  import math
8  import re
9  from numpy import *
10 from scipy.optimize import curve_fit
11 from pprint import pprint
12 from copy import copy,deepcopy
13
14 #-----
15 # FUNCION:      .
16 # ENTRADA:      .
17 #              .
18 #              .
19 # SALIDA:       .
20 #              .
21 #-----
22
23 #-----
24 #-----
25 #   MODELOS DE DETERMINACION DE FRACCION TRANSFORMADA
26 #-----
27 #-----
28 def f_KM(T_Ms, alfa):
29     ---Koistinen - Marburguer
30     -----
31     # FUNCION:      .f_KM(), funcion de koistinen y Marburguer (la forma es
32     ↪ dada para luego
33     # aplicar curve_fit(), debe ir la variable y luego parametro de
34     ↪ ajuste).
35
36     # ENTRADA:      .T_Ms: es (Ms-T)...Ms determinada por métodos
37     ↪ experimentales o modelos
38     # previos
39
40     #              .alfa: coeficiente de KM
41     # SALIDA:      .devuelve la evaluacion de la función
42     #-----
43
44     return 1-np.exp(-alfa*T_Ms)
45
46 def ff_KM(T_Ms, *parametros):
47     ---Koistinen - Marburguer, desplazado y se considera gama fuera de
48     ↪ parentesis

```

```

45 #-----
46 # FUNCION:      .f_KM(), funcion de koistinen y Marburguer (la forma es
    ↪ dada para luego
47 # aplicar curve_fit(), debe ir la variable y luego parametro de
    ↪ ajuste).
48
49 # ENTRADA:      .T_Ms: es (Ms-T)...Ms determinada por métodos
    ↪ experimentales o modelos
50 # previos
51
52 #              .alfa: coeficiente de KM
53 #              .gama: desplazamiento para coincidencia con fraccion
    ↪ transformada a Ms
54 # SALIDA:      .devuelve la evaluacion de la función
55 #-----
56     alfa, gama=parametros
57     return 1-np.exp(-alfa*T_Ms+gama)
58
59 def fff_KM(T_Ms,*parametros):
60 #---Koistinen - Marburguer, desplazado y se considera gama dentro de
    ↪ parentesis
61 #-----
62 # FUNCION:      .f_KM(), funcion de koistinen y Marburguer (la forma es
    ↪ dada para luego
63 # aplicar curve_fit(), debe ir la variable y luego parametro de
    ↪ ajuste).
64
65 # ENTRADA:      .T_Ms: es (Ms-T)...Ms determinada por métodos
    ↪ experimentales o modelos
66 # previos
67
68 #              .alfa: coeficiente de KM
69 #              .gama: desplazamiento para coincidencia con fraccion
    ↪ transformada a Ms
70 # SALIDA:      .devuelve la evaluacion de la función
71 #-----
72     alfa=parametros[0]
73     gama=parametros[1]
74     return 1-np.exp(-alfa*(T_Ms+gama))
75
76
77 def f_LV(T_Ms, *parametros):
78 #---Lee - Van Tyne
79 #-----
80 # FUNCION:      .f_LV(), funcion de kLee y Van Tyne (la forma es dada
    ↪ para luego aplicar
81 # curve_fit(), debe ir la variable y luego parametro de ajuste).
82
83 # ENTRADA:      .T_Ms: es (Ms-T)...Ms determinada por métodos
    ↪ experimentales o modelos

```

```

84 # previos
85
86 #             .alfa: coeficiente 1
87 #             .beta: coeficiente 2
88 # SALIDA:     .devuelve la evaluacion de la función
89 #-----
90     alfa, beta=parametros
91     return 1-np.exp(-alfa*(T_Ms**beta))
92
93 def f_LVprima(T_Ms, *parametros):
94     ---Lee - Van Tyne, desplazado y se considera gama fuera de
95     ↪ parentesis
96     -----
97     # FUNCION:     .f_LV(), funcion de kLee y Van Tyne (la forma es dada
98     ↪ para luego aplicar
99     # curve_fit(), debe ir la variable y luego parametro de ajuste).
100
101     # ENTRADA:     .T_Ms: es (Ms-T)...Ms determinada por métodos
102     ↪ experimentales o modelos
103     # previos
104
105     #             .alfa: coeficiente 1
106     #             .beta: coeficiente 2
107     #             .gama: desplazamiento para coincidencia con fraccion
108     ↪ transformada a Ms
109     # SALIDA:     .devuelve la evaluacion de la función
110     #-----
111     alfa, beta, po=parametros
112     gama=np.log(1-po)
113     return 1-np.exp(-alfa*(T_Ms**beta)+gama)
114
115 def fff_LV(T_Ms, *parametros):
116     ---Lee - Van Tyne, desplazado y se considera gama dentro de
117     ↪ parentesis
118     -----
119     # FUNCION:     .f_LV(), funcion de kLee y Van Tyne (la forma es dada
120     ↪ para luego aplicar
121     # curve_fit(), debe ir la variable y luego parametro de ajuste).
122
123     # ENTRADA:     .T_Ms: es (Ms-T)...Ms determinada por métodos
124     ↪ experimentales o modelos
125     # previos
126
127     #             .alfa: coeficiente 1
128     #             .beta: coeficiente 2
129     #             .gama: desplazamiento para coincidencia con fraccion
130     ↪ transformada a Ms
131     # SALIDA:     .devuelve la evaluacion de la función
132     #-----
133     alfa, beta, gama=parametros

```

```

126         return 1-np.exp(-alfa*((T_Ms+gama)**beta))
127
128 def f_H(T_Ms, *parametros):
129     #---Huyan y otros
130     #-----
131     # FUNCION:      .f_H(), funcion de modelo planteado por Huyan y otros
132     ↪ (la forma es dada
133     # para luego aplicar curve_fit(), debe ir la variable y luego
134     ↪ parametro de ajuste).
135
136     # ENTRADA:      .T_Ms: es (Ms-T)...Ms determinada por métodos
137     ↪ experimentales o modelos
138     # previos
139
140     #
141     #          .alfa: coeficiente 1
142     #          .beta: coeficiente 2
143     # SALIDA:      .devuelve la evaluacion de la función
144     #-----
145     A, B, H1, H2=parametros
146     DeltaG=-H1*T_Ms+H2
147     return 1./(1.+1./(A*(DeltaG/100.))**B))
148
149 def f_Hprima(T_Ms, *parametros):
150     #---Huyan y otros
151     #-----
152     # FUNCION:      .f_Hprima(), funcion de modelo planteado por Huyan y
153     ↪ otros, tiene presente
154     # un desplazamiento dado que a Ms la fracción de transformada no es
155     ↪ nula(la forma es
156     # dada para luego aplicar curve_fit(), debe ir la variable y luego
157     ↪ parametro de ajuste).
158
159     # ENTRADA:      .T_Ms: es (Ms-T)...Ms determinada por métodos
160     ↪ experimentales o modelos
161     # previos
162
163     #
164     #          .alfa: coeficiente 1
165     #          .beta: coeficiente 2
166     # SALIDA:      .devuelve la evaluacion de la función
167     #-----
168     A, B, H1, H2, po=parametros
169     DeltaG=-H1*T_Ms+H2
170     H3=100.*(po/(A*(1.-po)))**((1./B) - H2
171
172     return 1./(1.+1./(A*((DeltaG+H3)/100.))**B))
173
174 def parametros_composicion(pwt):
175     # FUNCION:      .parametros_composicion()

```

```

168 # ENTRADA:      .pwt: composición del acero en porcentaje en peso
    ↪   atómico, tipo
169 # diccionario
170
171 # SALIDA:      .diccionario con los parametros determinado segun
    ↪   distintos autores
172 #              .Ms_vBS: temperatura "teorica" de inicio de
    ↪   transformación martensítica o
173 # temperatura K-M
174
175 #              .vBS: parametro del exponenete (multiplica)
176 #              .Ms_FS: Temperatura de inicio de transformación (para
    ↪   aceros con
177 # 8-14%Cr), segun Finkle y Schirra
178
179 #              .Ms_vB: Ms segun van Bohemen
180 #              .vB: parametro del exponenete (multiplica) segun van
    ↪   Bohemen
181 #              .LV1: parametro del exponenete (multiplica) segun Lee y
    ↪   Van Tyne
182 #              .LV2: parametro del exponenete (eleva) segun Lee y Van
    ↪   Tyne
183     parametros={}
184 # calculos de van Bohemen - Sietsma
185     Ms_vBs= 462- 273*pwt.get('C', 0) -26*pwt.get('MN',0)-
    ↪   16*pwt.get('NI',0)-13*pwt.get('CR',0)-30*pwt.get('MO',0)
186     vBS=.0224 - .0107*pwt.get('C',0) - .0007*pwt.get('MN',0) -
    ↪   .00005*pwt.get('NI',0)- .00012*pwt.get('CR',0) -
    ↪   .0001*pwt.get('MO',0)
187     parametros['vanBohemen_Sietsma']={}
188     parametros['vanBohemen_Sietsma']['Ms']=Ms_vBs
189     parametros['vanBohemen_Sietsma']['parametro1']=vBS
190 # calculos de Finkler y Schirra:
191     Ms_FS=635. - 474.*(pwt.get('C',0.))+
    ↪   0.86*(pwt.get('N',0.)-0.15*(pwt.get('NB',0.)+pwt.get('ZR',0.)))
    ↪   - 0.066*(pwt.get('TA',0.) + pwt.get('HF',0.)) -
    ↪   (17*pwt.get('CR',0.)+33*pwt.get('MN',0.)+21*pwt.get('MO',0.)+17*pwt.get(
192     parametros['Finkler_Schirra']={}
193     parametros['Finkler_Schirra']['Ms']=Ms_FS
194 # calculo de van Bohemen(vB)
195
    ↪   Ms_vB=565-600*(1-np.exp(-.96*pwt.get('C',0)))-31*pwt.get('MN',0)-13*pwt.
196     vB=27.2-
    ↪   .14*pwt.get('MN',0)-.21*pwt.get('SI',0)-.11*pwt.get('CR',0)-.08*pwt.get(
197     vB=vB*1e-3
198     parametros['vanBohemen']={}
199     parametros['vanBohemen']['Ms']=Ms_vB
200     parametros['vanBohemen']['parametro1']=vB
201
202 # calculo de Lee y Van Tyne(LV):

```

203

```
↪ Ms_LV=764.2-302.6*pwt.get('C',0)-30.6*pwt.get('MN',0)-14.5*pwt.get('SI',0)-1
```

204

```
↪ LV1=.02321-.0105*pwt.get('C',0)-.0017*pwt.get('NI',0)+.0074*pwt.get('CR',0)-
```

205

```
↪ LV2=1.4304-1.1836*pwt.get('C',0)+.7527*pwt.get('C',0)**2-.0258*pwt.get('NI',0)
```

206

```
parametros['Lee_VanTyne']={}

```

207

```
parametros['Lee_VanTyne']['Ms']=Ms_LV-273.15

```

208

```
parametros['Lee_VanTyne']['parametro1']=LV1

```

209

```
parametros['Lee_VanTyne']['parametro2']=LV2

```

210

```
return parametros

```

211

212

```
213 def comparacion_parametros(Ms,dato, funciones,T, p, semillas,extra):

```

214

```
    # FUNCION:      .comparacion_parametros()

```

215

```
    # ENTRADA:

```

216

```
    #              .Ms: martensita start

```

217

```
    #              .funciones: funciones del modelo a comparar, en lista.

```

218

```
    #              .T: datos que son la variable de la funcion,

```

```
    ↪ temperatura

```

219

```
    #              .p: datos a ajustar, fraccion transformada calculada

```

```
    ↪ por regla de la

```

220

```
    # palanca

```

221

222

```
    #              .semillas:      valores semillas para el calculo de los

```

```
    ↪ parametros de

```

223

```
    # ajuste, en lista y del tamaño de las funciones.

```

224

225

```
    #              Las funciones que requieren más de un

```

```
    ↪ parametro deben ponerse

```

226

```
    # en tuplas

```

227

228

```
    # SALIDA:      .KMs: Los parametros calculados sgun los fiteos

```

229

```
    #              .desviacion asociada al fiteo

```

230

```
    #-----

```

231

```
    parametrosfiteo=[] # para el guardado de los parametros calculados

```

232

```
    desviacion=[] # para guardar datos de la deviancion estandar dado

```

```
    ↪ por el fiteo

```

233

```
    # plt.figure()          #para guardado de graficos

```

234

```
    for j, fn in enumerate(funciones):

```

235

```
        par,cov=curve_fit(fn, Ms-T[extra[j]:], p[extra[j]:],

```

```
        ↪ semillas[j])

```

236

```
        parametrosfiteo.append(list(par))

```

237

```
        desviacion.append(list(np.sqrt(np.diag(cov))))

```

238

```
    return parametrosfiteo, desviacion

```

239

240

```
def graficoscomparacion_parametros(T, p,

```

```
    ↪ Ms,posicion_Ms,funciones,detalles,extra,parametros, simb_color):

```

241

```
    # FUNCION:      .graficoscomparacionparametros()

```

242

```
    # ENTRADA:      .poriscion_Ms en el vector T

```

```

243 # .Ms: martensita start
244 # .funciones: funciones del modelo a comparar, en lista.
245 # .T: temperatura
246 # .p: fraccion calculada pro regla de la palanca
247 # .extra: desde donde quiero evaluar funcion, evitar
    ↪ las
248 # indeterminaciones
249
250 # .parametros: los parametros de ajuste
251 # .detalles: lista con referencia al grafico:
252 # detalles[0]: titulo grafico
253 # detalles[1]: titulo de la leyenda
254 # detalles[2]: lista con los label de los
    ↪ graficos
255 # detalles[:]:
256 # SALIDA: .graficos de fraccion transformada segun los parametros
    ↪ de ajuste
257 Colores=['k', 'r', 'b', 'g', 'c', 'y', 'm', 'w']
258 Simbolos=['-', '--', '-.', ':'] # colores para el grafico
259 # formato de leyenda de graficos
260 Formato_ejes={'family':'serif','color':'black','size':9}
261 leyendas=[]
262 aux=''
263
264 # detalles[2]=[['forma1','LV1=', ' LV2'],['forma2','LV1=', '
    ↪ LV2'],['forma3','LV1=',
265 # ', LV2']]
266
267 for i, det in enumerate(detalles[2]):
268     aux=aux+det[0]
269     for j, parte in enumerate(det[1:]):
270         aux=aux+parte+'%0.4f' %parametros[i][j]
271     leyendas.append(aux)
272     aux=''
273
274 plt.plot(T, p, '-k', label='Curva experimental: Regla de la
    ↪ palanca')
275 for j, fn in enumerate(funciones):
276     # fraccion de transformacion por ajuste K-M
277     plt.plot(T[extra[j]:],
        ↪ fn(Ms-T[extra[j]:],*tuple(parametros[j])),simb_color[j],label=leyend
278 # Ms encontrada experimentalmente por Metodo ASTM, graficada
    ↪ sobre p
279 # #fraccion de transformacion por regla de la palanca
    ↪
280 plt.plot(Ms,p[posicion_Ms] , 'o k',
    ↪ label='Ms={:0.1f}°C'.format(Ms))
281 plt.legend(frameon=False, title_fontsize=7,
    ↪ prop={'family':'serif','size':7}, title=detalles[1])
282 plt.xlabel('T/°C',fontdict=Formato_ejes)

```

```

283     plt.ylabel('fraccion de martensita
        ↳ transformada',fontdict=Formato_ejes)
284     plt.title(detalles[0],
        ↳ fontdict={'family':'serif','color':'black','size':10},loc='left')
285
286     #-----
287     #-----
288     #             METODOS DE DETERMINACION DE Ms
289     #-----
290     #-----
291     def Met_tangente(temperatura, deformacion,p1, p2, p3, p4, epsilon):
292     #-----
293     # FUNCION:      .Met_tangente()
294     # ENTRADA:      .temperatura: temperaturas medidas, del tipo
        ↳ np.array(n) (donde n es la
295     # longitud del array)
296
297     #             .deformacion: defromaciones medidas, del tipo
        ↳ np.array(n) (donde n es la
298     # longitud del array)
299
300     #             .p1, p2: posiciones de temperatura caracteristicos de
        ↳ la curva
301     # dilatometrica con las cuales se usara para definir la recta1
302
303     #             .epsilon: cota de evaluación
304     #             .p3, p4: posiciones de temperatura caracteristicos de
        ↳ la curva
305     # dilatometrica con las cuales se usara para definir la recta2
306
307     # SALIDA:      .dato_tg: la posicion de la Ms medida en el np.array()
        ↳ de temperatura
308     #             .Ms_tg: Ms medida en el np.array() de temperatura
309     #-----
310
311     # Definicion
312     #-----
313     desp_1=400# Desplazamiento del punto caracteristico p1
314     desp_2=0# Desplazamiento del punto caracteristico p2
315     desp_3=0# Desplazamiento del punto caracteristico p3
316     desp_4=0# Desplazamiento del punto caracteristico p4
317     # parametros de ajuste en gama
318
        ↳ a1,b1=polyfit(np.array([temperatura[p1+desp_1],temperatura[p2+desp_2]]),
        ↳ np.array([deformacion[p1+desp_1],deformacion[p2+desp_2]]),1)
319     # parametro de ajuste durante la transformación
320
        ↳ a2,b2=polyfit(np.array([temperatura[p3+desp_3],temperatura[p4+desp_4]]),
        ↳ np.array([deformacion[p3+desp_3],deformacion[p4+desp_4]]),1)
321     # Evaluacion:

```

```

322 #-----
323     eval_1=-50
324     eval_2=50
325     recta1=a1*temperatura[p2+eval_1:p4+eval_2]+b1
326     recta2=a2*temperatura[p2+eval_1:p4+eval_2]+b2
327     diferencial1=recta1-recta2
328     for i, dato in enumerate(np.arange(p2+eval_1,p4+eval_2)):
329         if abs(diferencial1[i])<epsilon:
330             Ms_tg=temperatura[dato]
331             dato_tg=int(dato)
332     return dato_tg, Ms_tg
333
334 def Met_ASTM(temperatura, deformacion, p1, p2, p3, p4, epsilon,
335     ↪ opcion):
336     # FUNCION:      .Met_ASTM()
337     # ENTRADA:      .temperatura: temperaturas medidas, del tipo
338     ↪ np.array(n) (donde n es la
339     # longitud del array)
340     #
341     .deformacion: defromaciones medidas, del tipo
342     ↪ np.array(n) (donde n es la
343     # longitud del array)
344     #
345     .p1, p2: posiciones de temperatura caracteristicos de
346     ↪ la curva
347     # dilatométrica con las cuales se usara para definir la recta1
348     #
349     .epsilon: cota de evaluación
350     #
351     .opcion: consideración sobre puntos a exceptuar
352     # SALIDA:      .dato_ASTM: la posicion de la Ms medida en el np.array()
353     ↪ de temperatura
354     #
355     .Ms_ASTM: Ms medida en el np.array() de temperatura
356     #-----
357     # Definicion
358     #-----
359     intervalo1=np.arange(p1,opcion[0])
360     intervalo2=np.arange(opcion[1],p2)
361     intervalo=np.append(intervalo1,intervalo2)
362     a3,b3=polyfit(temperatura[intervalo], deformacion[intervalo],1)
363     a4,b4=polyfit(temperatura[-150:], deformacion[-150:],1)
364     # Correcciones de segmento excluido para la evaluación:
365     #-----
366     #
367     ↪ deformacion[opcion[0]:opcion[1]]=a3*temperatura[opcion[0]:opcion[1]]+b3
368     ↪
369     # #reemplazar el los datos de deformacion el intervalo excluido en el
370     ↪ ajuste por la
371     # recta obtenido por el ajuste
372     #
373     # Evaluacion

```

```

364 #-----
365     evaluacion=range(p2-200,p4)# generación de intervalo de
        ↪     evaluación
366     recta3=a3*temperatura[evaluacion]+b3
367     recta4=a4*temperatura[evaluacion]+b4
368     diferencia2=(deformacion[evaluacion]-recta3)/(recta4-recta3)
369     for i, dato in enumerate(evaluacion):
370         if abs(diferencia2[i])<epsilon:
371             Ms_ASTM=temperatura[dato]
372             dato_ASTM=int(dato)
373 # Grafiquitos de fracción
374 #-----
375 # inicio=p2-100
376 # fin=p2+50
377 # numerador=deformacion[dato_ASTM]-a3*temperatura[dato_ASTM]-b3
378 # denominador=(a4-a3)*temperatura[dato_ASTM]+b4-b3
379 # f=numerador/denominador
380 #
        ↪     numerador_ff=(deformacion[inicio:fin]-a3*temperatura[inicio:fin]-b3)
381 # denominador_ff=((a4-a3)*temperatura[inicio:fin]+b4-b3)
382 # ff=numerador_ff/denominador_ff
383 # plt.figure()
384 # plt.plot(temperatura[inicio:fin], ff , 'o-g')
385 # plt.plot(temperatura[inicio:fin], ff , 'o-b')
386 # plt.plot(temperatura[dato_ASTM], f, 's r')
387 # plt.show()
388 # plt.close()
389 # Grafico de rectas:
390 #-----
391 # plt.plot(temperatura[p2-200:p4], a3*temperatura[p2-200:p4]+b3)
392 # plt.plot(temperatura[-300:], a4*temperatura[-300:]+b4)
393 # plt.plot(temperatura[p2-50:], deformacion[p2-50:])
394 # plt.plot(temperatura[p2], deformacion[p2], 'o b')
395 # plt.plot(temperatura[dato_ASTM], deformacion[dato_ASTM], 's r')
396 # plt.plot(temperatura[-150], deformacion[-150], 'o b')
397 # plt.plot(temperatura[-100], deformacion[-100], 'o k')
398 # plt.plot(temperatura[-50], deformacion[-50], 'p y')
399     return dato_ASTM, Ms_ASTM
400
401 def Met_offset(temperatura, deformacion, eta0, p1, p2, V,epsilon,
        ↪     opcion):
402     # FUNCION:         .Met_offset()
403     # ENTRADA:         .temperatura: temperaturas medidas, del tipo
        ↪     np.array(n) (donde n es la
404     # longitud del array)
405
406     #
        ↪     .deformacion: defromaciones medidas, del tipo
        ↪     np.array(n) (donde n es la
407     # longitud del array)
408

```

```

409 #             .eta0: offset calculado por segun la composición.
410 #             .p1, p2: posiciones de temperatura caracteristicos de
    ↪ la curva
411 # dilatometrica con las cuales se usara para definir la recta1
412
413 #             .epsilon: cota de evaluación
414 #             .opcion: consideración sobre puntos a exceptuar
415 # SALIDA:      .dato_off: la posicion de la Ms medida en el np.array()
    ↪ de temperatura
416 #             .Ms_off: Ms medida en el np.array() de temperatura
417 #-----
418 # Definicion
419 #-----
420     evaluacion=range(p2-300,p2+75)
421     intervalo1=np.arange(p1,opcion[0])
422     intervalo2=np.arange(opcion[1],p2)
423     intervalo=np.append(intervalo1,intervalo2)
424     a,b=polyfit(temperatura[intervalo], deformacion[intervalo],1)
425 # EValuación
426 #-----
427
    ↪ diferencia=(deformacion[evaluacion]-(a*temperatura[evaluacion]+b+eta0))
428 for i, dato in enumerate(evaluacion):
429     if abs(diferencia[i])<epsilon:
430         Ms_off=temperatura[dato]
431         dato_off=int(dato)
432 return dato_off, Ms_off
433
434
435 def eta_offset(x_alfa, wt_gama,V):
436 # FUNCION:      .eta_offset()
437 # ENTRADA:      .x_alfa: fraccion molar de los componentes de la fase
    ↪ alfa
438 #             .wt_gama: fraccion en peso de los componentes de la fase
    ↪ gama
439 #             .V: fraccion en volumen de martensita transformada
440 # SALIDA:      .eta0: parametro para metodo offset
441 #-----
442
443     a_Fe=0.28664
444     num1=(a_Fe -0.0279*x_alfa.get('C',0))**2
445     num2=a_Fe+0.2496*x_alfa.get('C',0)
446     den=3*a_Fe**2
447     term=(num1*num2-a_Fe**3)/den
448
    ↪ a_alfa=0.28664+term-0.003*x_alfa.get('SI',0)+0.006*x_alfa.get('MN',0)+0.
449
    ↪ a_gama=0.3573+0.33*wt_gama.get('C',0)+0.0095*wt_gama.get('MN',0)-0.002*w
450     eta0=np.cbrt(a_gama*(-3)*(2*V*a_alfa**3+(1-V)*a_gama**3))-1
451 return eta0

```

```

452
453 #-----
454 #-----
455 #          FUNCIONES CONVERSIÓN Y EXPERIMENTAL
456 #-----
457 #-----
458 def FraccionMolar(FMasica):
459 # FUNCION:      .FraccionMolar()
460 # ENTRADA:      .FMasica: diccionario de composición en fraccion masica
461 ↪
462 # SALIDA:      .Fmolar: diccionario de composicion en fracción molar
463 #-----
464 # Elementos y su masa atomica por mol
465 #-----
466 Elementos=['C','CU','AL','N','SB','MN','SN','TI','B','PB',
467 ↪ 'SI','NI','NB','CA','P','CR','V','O','S','MO','CO',
468 ↪ 'AS','FE']
469 MA=[12.01,63.55, 26.98, 14.01 ,121.76,54.94, 118.71, 47.87, 10.81,
470 ↪ 207.2, 28.09, 58.69, 92.91, 40.08, 30.97, 52., 50.94, 16.,
471 ↪ 32.06, 95.95, 58.93, 74.92, 55.85 ]
472 Elem_MA=dict(zip(Elementos, MA))
473 # Valores y claves del acero:
474 #-----
475 v_FMasica=FMasica.values()# valores
476 c_FMasica=list(FMasica)# claves
477 # Definicion de diccionario con claves de FMasica
478 FMolar=dict(zip(c_FMasica,list(np.zeros(len(v_FMasica)))))
479 aux=0.
480 #  $x_i = n_i/N = m_i/(P A_i \cdot N) = (M/N) \cdot (w_i/P A_i)$ 
481 #  $w_i = m_i/M \Rightarrow m_i = M \cdot w_i$ 
482 #  $N = \sum(n_i) = \sum(m_i/P A_i) = M \cdot \sum((m_i/M) \cdot (1/P A_i)) = M \cdot \sum(w_i/P A_i) \Rightarrow$ 
483 ↪  $M/N = 1/\sum(w_i/P A_i)$ 
484 # por lo tanto:  $x_i = (1/\sum(w_i/P A_i)) \cdot (w_i/P A_i)$ 
485 for i,s in enumerate(FMasica):
486     if s in Elementos:
487         aux+=FMasica[s]/Elem_MA[s]
488 M_N=1./aux
489 for i,s in enumerate(FMasica):
490     if s in Elementos:
491         FMolar[s]=M_N*FMasica[s]/Elem_MA[s]
492     else:
493         FMolar[s]=FMasica[s]
494 return(FMolar)
495
496 def palanca(temperatura, deformacion, am, bm,ah, bh):
497 # FUNCION:      .palanca()
498 # ENTRADA:      .temperatura: np.array(n)
499 #               .deformación: np.array(n)
500 #               .am, bm: parametros de ajuste lineal de fase madre, am:
501 ↪ pendiente, bm:

```

```

495 # ordenada al origen
496
497 # .ah, bh: parametros de ajuste lineal de fase hija, ah:
    ↪ pendiente, bh:
498 # ordenada al origen
499
500 # .Aclaraciones: .n es el tamaño del array, 1xn.
501 # SALIDA: .p: fraccion en volumen transformada, np.array(n)
502 #-----
503     p=(deformacion-am*temperatura-bm)/((ah-am)*temperatura+(bh-bm))
504     return p
505
506 def resumen_p(diccionario, T, d):
507     # FUNCION: .resumen_p()
508     # ENTRADA: .diccionario con los datos asociados a cierta muestra y
    ↪ metodo de
509     # determinacion de Ms.
510
511     # .T: np.array(n)
512     # .d: np.array(n)
513     # SALIDA: .El mismo diccionario con la p calculada
514     #-----
515     dato_Ms=int(diccionario['dato_Ms'])
516     a3=diccionario['a_gama']
517     b3=diccionario['b_gama']
518     a4=diccionario['a_alfapri']
519     b4=diccionario['b_alfapri']
520     p=palanca(T, d,a3, b3, a4, b4)
521     diccionario['p']=p
522
523     #-----
524     #-----
525     # LECTURA DE ARCHIVOS Y EXTRACCIONES
526     #-----
527     #-----
528     def extraccion(nombre):
529         # FUNCION: .extraccion()
530         # ENTRADA: .nombre: archivo
531         # SALIDA: .tiempo: np.array(n)
532         # .deformacion: np.array(n)
533         # .temperatura: np.array(n)
534         # .Aclaraciones: .n es el tamaño del array, 1xn.
535         #-----
536
537         algo=open(nombre, 'rt')
538         # salta las primeras 4 lineas quizas se puede corregir con un try y
    ↪ except
539         for i in range(4):
540             line = algo.readline()
541         tiempo=[]

```

```

542     deformacion=[]
543     temperatura=[]
544
545     while line:
546         try:
547             line = algo.readline()
548             aux= np.fromstring(line.strip(),dtype=float ,sep=" ")
549             tiempo.append(aux[1])
550             deformacion.append(aux[2])
551             temperatura.append(aux[3])
552         except (ValueError,IndexError):
553             ''
554     algo.close
555     return np.array(tiempo), np.array(deformacion),
556         ↪ np.array(temperatura)
557
558 def lectura(nombre):
559     # FUNCION:      .extraccion()
560     # ENTRADA:      .nombre: archivo
561     # SALIDA:      .tiempo: np.array(n)
562     #               .deformacion: np.array(n)
563     #               .temperatura: np.array(n)
564     #               .Aclaraciones:      .n es el tamaño del array, 1xn.
565     #-----
566     algo=open(nombre,'rt')
567     line = algo.readline()
568     line = algo.readline()
569     titulo=line.split()
570     llaves=titulo[2:]
571     metodo={}
572     muestras={}
573     while line:
574         try:
575             line = algo.readline()
576             aux=line.strip('\n').split()
577             dr=dict(zip(llaves,map(float,aux[2:])))
578             metodo[aux[1]]=dr.copy()
579             muestras[aux[0]]=metodo.copy()
580             dr={}
581         except (ValueError,IndexError):
582             ''
583     algo.close
584
585     return muestras
586
587 def comp_fase(nombre_archivo, fase):
588     # FUNCION:      .comp_fase()
589     # ENTRADA:      .nombre_archivo: devuelto para Thermocalc
590     #               .fase: de busqueda

```

```

590 # .Aclaraciones: .Esta pensado para un archivo
    ↳ devuelto por
591 # Thermocalc.
592
593 # .el archivo de Thermocalc hace los
    ↳ calculos para un
594 # mol de
595
596 # SALIDA: .FMasica: lista de diccionarios de composicion en
    ↳ fracción masica de la
597 # fase
598
599 #-----
600
601 algo=open(nombre_archivo,'rt')# apertura de archivo
602 elementos=[] # va a acumular las claves del diccionario, elementos
    ↳ de la composicion
603 fraccion=[] # va a tener los valores del diccionario
604 line = algo.readline()# lectura de linea
605 comp_lista=[] # inicializa la lista de diccionarios de composicion
606 while line:
607     try:
608         line=algo.readline()
609         aux=line.strip('\n').split()
610         comp_dic={}# inicializa diccionarios de composicion
611         if fase in line:
612             comp_dic['fase']=aux[0]
613             line=algo.readline()
614             aux=line.strip('\n').split()
615             comp_dic['N(moles)']=float(aux[3].strip(','))
616             comp_dic['M(masa)']=float(aux[5])
617             i=0
618             while i!=1:
619                 if line=='\n':
620                     i+=1
621                     break
622                 line=algo.readline()
623                 aux=line.strip('\n').split()
624 # lista[start:stop:step]
625                 elementos.extend(aux[0::2])
626 # map(int, z)
627                 # map:converta a flotantes los valores de la
                    ↳ lista
628                 fraccion.extend(map(float,aux[1::2]))
629                 comp_dic.update(dict(zip(elementos, fraccion)))
630                 comp_lista.append(comp_dic)
631             except (ValueError,IndexError):
632                 ''
633 algo.close
634 FMasica=comp_lista

```

```

635     return FMasica
636
637 #-----
638 #-----
639 #          GRAFICADORES, AJUSTES Y BUSQUEDA
640 #-----
641 #-----
642 def PromedioMovil(ydatos, numprom, tipo):
643     # FUNCION:      .PromedioMovil()
644     # ENTRADA:      .ydatos: datos de ordenada que van a ser filtrados,
        ↪ np.array(n)
645     #              .numprom: numero de puntos a tomar para el pomedio
646     #              .tipo: acara donde se va a colocar el promedio
647     #              . 'c': centrado
648     #              . 'i' (izquierda): punto extremo izq
649     #              . 'd' (derecha): punto extremo der
650     # SALIDA:      .y: datos de ordenada filtrados, np.array(n)
651     #              .Aclaracion:      .Los punto sobrentes de los
        ↪ extremos sin promediar se
652     # rellenan con los mismos puntos
653
654 #-----
655 #(Este caso es centrado pero faltaria agregar para izq y der)
656     desp=int((numprom-1)/2)
657     inicio=desp
658     final=int(len(ydatos)-desp)
659     ydatos=list(ydatos)
660     y=ydatos[:desp]
661     for i in range(desp,final):
662         aux=sum(ydatos[i-desp:i+desp+1])/numprom
663         y.append(aux)
664     for i in ydatos[-desp:]:
665         y.append(i)
666     return np.array(y)
667
668 def convDicc(diccionario):
669     # FUNCION:      .convDicc()
670     # ENTRADA:      .diccionario
671     # SALIDA:      .lista de los valores del diccionario
672     #-----
673     lista=[]
674     for s in diccionario:
675         lista.append(diccionario[s])
676     return np.array(lista)
677
678 def buscador(buscar, datos, epsilon):
679     # FUNCION:      .buscador()
680     # ENTRADA:      .datos: donde se desea encontrar la posición de dicho
        ↪ valor.
681     #              .epsilon: cota de evaluacion

```

```

682 # SALIDA:      .p1: posicion de dicho valor
683 #-----
684     if np.size(buscar)==0:
685         p1=[]
686     else:
687 # for i in range(np.size(datos)):
688         for i, dato in enumerate(datos):
689             if abs(dato-buscar)<epsilon:
690 #         if abs(datos[i]-buscar)<epsilon:
691                 p1=int(i)
692     return p1
693
694 def extender(lista_de_listas):
695 # FUNCION:      .extender()
696 # ENTRADA:      .lista_de_listas: listas que se quiere unir como listas
697 #               ↪ de listas.
698 # SALIDA:      .lista_ext: lista extendida
699     lista_ext=[]
700     for l in lista_de_listas:
701         lista_ext.extend(l)
702 # lista_ext.append(l)
703     return lista_ext
704
705
706 def extender_apendizar(lista1, lista2):
707 # FUNCION:      .extender_apendizar()
708 # ENTRADA:      .lista1: lista o dato que se quiere extender con una de
709 #               ↪ lista 2
710 #               .lista2: dvalores a extender a la lista1
711 # SALIDA:      .lista: lista con los valores i de la lista 1 y 2
712 #               ↪ extendido formanco una
713 # listaa apendizado
714     lista=[]
715     lista1=deepcopy(lista1)
716     for i, l1 in enumerate(lista1):
717         l1.extend(lista2[i])
718         lista.append(l1)
719     return lista

```


9 Referencias

1. Honeycombe, R. W. K. y Bhadeshia, H. K. D. H., *Steels, Microstructure and Properties*; Butterworth-Heinemann: 2006.
2. Baker, H.; American Society for Metals; ASM International y Okamoto, *ASM Handbook, Volume 03 - Alloy Phase Diagrams*; ASM International.: 1985; vol. 3.
3. Ando, T.; Gokan, K. y Arase, K. *Tetsu-to-Hagane* **1957**, *43*, 127-132.
4. Porter, D. A. y Easterling, K. E., *Phase Transformations in Metals and Alloys*; Springer Publishing: New York, Estados Unidos, 1992.
5. Andrews, K. W. *J. Iron Steel Inst.* **1965**, *July*, 721-727.
6. Speer, J.; Matlock, D.; De Cooman, B. y Schroth, J. *Acta Materialia* **2003**, *51*, 2611-2622.
7. DOE, U. <https://www.gen-4.org/gif/upload/docs/application/pdf/2014-03/gif-tru2014.pdf> **2014**.
8. Abram, T. y Ion, S. *Energy Policy* **2008**, *36*, 4323-4330.
9. Murty, K. y Charit, I. *Journal of Nuclear Materials* **2008**, *383*, 189-195.
10. Klueh, R L. *International Materials Reviews* **2005**, *50*, 287-310.
11. Yvon, P. y Carré, F. *Journal of Nuclear Materials* **2009**, *385*, 217-222.
12. Klueh, R. y Nelson, A. *Journal of Nuclear Materials* **2007**, *371*, 37-52.
13. ASM International Materials Park, *ASM Speciality Handbook: Heat Resistant Materials*, 1985.
14. A. Czyrska-Filemonowicz; A. Zielińska-Lipiec y P.J. Ennis *Journal of Achievements in Materials and Manufacturing Engineering, PhD.* **2006**, *19*, 43-48.
15. Yardley, Victoria Anne; University of Cambridge y Department of Materials Science and Metallurgy *Magnetic Detection of Microstructural Change in Power Plant Steels, PhD. Thesis*; 2003.
16. N. Zavaleta Gutiérrez; ITJS y CNEA/UNSAM *Estudio del proceso de crecimiento heterogéneo de grano austenítico en acerosmartensíticos 9 % Cr, Tesis Doctoral*; 2010.
17. Hald, J. *International Journal of Pressure Vessels and Piping* **2008**, *85*, 30-37.
18. Haarmann, K.; Vaillant, J. y Vandenberghe, B. *The T91/P91 Book* **2002**, 62.
19. M. L. Garibaldi; ITSJ y CNEA/UNSAM *Comportamiento en transformación y caracterización de aceros de alta temperatura para reactores nucleares avanzados: Aceros ASTM A335P91, Tesis de Maestría*; 2016.
20. Hunkel, M.; Surm, H. y Steinbacher, M. en *Handbook of Thermal Analysis and Calorimetry*; Elsevier: 2018; vol. 6, págs. 103-129.
21. D. A. Arcone.; ITJS y CNEA/UNSAM *Influencia de la velocidad de calentamiento a la austenita sobre el comportamiento en transformación en aceros de alta temperatura para reactores nucleares avanzados, Tesis de Maestría*; 2016.

22. Garai, J. *Calphad* **2006**, *30*, 354-356.
23. Van Bohemen, S. *Scripta Materialia* **2013**, *69*, 315-318.
24. Alghamdi, T. y Liu, S. *Welding Journal* **2014**, *93*, 243-252.
25. Finkler, H. y Schirra, M. *Steel research* **1996**, *67*, 328-342.
26. Haynes, A. y Steven, W. *J. Iron Steel Inst* **1956**, *183*, 349-359.
27. *ASTM A1033-10, Standard Practice for Quantitative Measurement and Reporting of Hypoeutectoid Carbon and Low-Alloy Steel Phase Transformations*; 2013.
28. Yang, H.-S. y Bhadeshia, H. K. D. H. *Materials Science and Technology* **2007**, *23*, 556-560.
29. Kamyabi-Gol, A.; Herath, D. y Mendez, P. *Canadian Metallurgical Quarterly* **2017**, *56*, 85-93.
30. Koistinen, D. y Marburger *Acta Metallurgica* **1959**, *7*, 59-60.
31. Magee, C. L. : *The nucleation of martensite: in: H. I. Aaronson, Phase Transformations: ASM International, Materials Park, Ohio, USA* **1970**, 115-156.
32. Van Bohemen, S. M. C. y Sietsma, J. *Materials Science and Technology* **2009**, *25*, 1009-1012.
33. Van Bohemen, S. M. C. *Materials Science and Technology* **2012**, *28*, 487-495.
34. Lee, S.-J. y van Tyne *Metallurgical and Materials Transactions A* **2012**.
35. Capdevila, C.; Caballero F, G. y Garcia, A. *ISIJ International* **2002**, *42*, 894-902.
36. Huyan, F.; Hedström, P.; Höglund, L. y Borgenstam, A. *Metallurgical and Materials Transactions A* **2016**, *47*, 4404-4410.
37. Czichos, H.; Saito, T. y Smith, L. E., *Springer handbook of metrology and testing*; Springer Science & Business Media: 2011.
38. Sundman, B.; Lukas, H. y Fries, S., *Computational thermodynamics: the Calphad method*; Cambridge University Press New York: 2007.