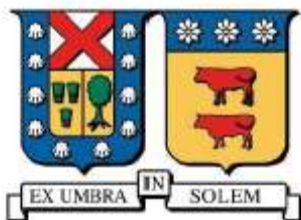


UNIVERSIDAD TÉCNICA FEDERICO SANTA MARÍA
DEPARTAMENTO DE MECÁNICA

VALPARAÍSO – CHILE



**“MODELACIÓN NUMÉRICA DE UN REACTOR
DE WATER-SPLITTING TERMOQUÍMICO DE
DOS ETAPAS BASADO EN UN CICLO
REDUCCIÓN-OXIDACIÓN DE ÓXIDO DE
CERIO”**

NICOLÁS ESTEBAN ALEGRÍA GALETOVIC

**MEMORIA DE TITULACIÓN PARA OPTAR AL TÍTULO DE INGENIERO CIVIL
MECÁNICO**

PROFESOR GUÍA : DR. ING. MARIO TOLEDO T.

PROFESOR CORREFERENTE : DR. ING. FRANCISCO GRACIA C.

MAYO – 2021

Agradecimientos

En primer lugar agradecer a mi madre Ivanica y a mi padre Fernando; su apoyo, esfuerzo y amor incondicional hicieron que fuera quien soy, les debo todo a ellos.

Agradecer a mi hermano Felipe, mi mejor amigo y quien me acompañó durante gran parte de esta aventura en Valparaíso.

Agradecer también a mis abuelos y abuelas: Jorge, Lucila, Mario y Elvira. Me hicieron sentir como si Punta Arenas no estuviera tan lejos. Gracias también a mis tías, tíos, primas, primos y toda la familia que me apoyó en este camino.

Agradecer a mis amigos y amigas. A Vecines, Pichicatas, (Mec+2)*peras, SEAD, Obra en proceso, entre tantos. Algunos vínculos se mantienen hasta el día de hoy, otros cumplieron su tiempo, pero aún así gracias a todas esas personitas que dejaron algo de ellas en mi ser.

Finalmente, agradecer a las personas que fueron parte de este trabajo. A mi profesor guía Mario Toledo, a mi co-referente Francisco Gracia, a mi amigo Luca Paniagua, con el que iniciamos este camino antes de que cambiara el mundo. Mención y agradecimiento especial a Miguel Godoy, que con un tip me ahorró varias horas de iteración.

Resumen

El presente trabajo se trata del desarrollo e implementación de un modelo computacional que simule el proceso termoquímico de reducción y oxidación para la producción de hidrógeno mediante Water-Splitting. Esta iniciativa nace en el marco del proyecto FONDAP “Solar Energy Research Center (SERC-Chile)”, específicamente en la línea de desarrollo e investigación de almacenamiento de energía y química-solar; bajo este proyecto se realizó el diseño y puesta en marcha de un reactor de Water-Splitting y es en base al funcionamiento de este reactor que se desarrolla el modelo.

Para desarrollar el modelo se planteó en primer lugar la situación física a estudiar. Luego de una revisión bibliográfica se decidió tratar el reactante como medio poroso. Se utilizó un modelo de combustión en medios porosos [19] para los balances de energía y masa, por otro lado, se utilizaron ecuaciones de variación de concentración de vacancias ([11] y [10]) para determinar la producción de oxígeno e hidrógeno en la respectiva reducción/oxidación.

Además del modelo del reactor de Water-Splitting se realizó un modelo de comparación para evaluar el comportamiento de la componente cinética del modelo y determinar el grado de cercanía del modelo con datos experimentales. Para esto se utilizó la situación física descrita por Arifin [11] y se desarrolló un modelo simple para obtener moles de oxígeno e hidrógeno de acuerdo al procedimiento descrito en dicha publicación. A partir del modelo de comparación, se determinó una cercanía de los valores para la oxidación mas no para la reducción. Esta diferencia en la reducción se le atribuye a fenómenos de difusión de especies y diferencias entre la situación física real y la modelada.

Respecto al modelo del reactor de Water-Splitting, se realizó un análisis de sensibilidad de variables. Este análisis determinó que las variables más relevantes eran temperaturas en la condición de borde y tiempo de reacción, en particular la oxidación resultó ser más sensible que la oxidación según los casos planteados. Sin embargo, para que el modelo planteado sea válido, es preciso realizar experimentos con el reactor de Water-Splitting para así ajustar los parámetros y obtener resultados lo más cercanos a la realidad posible.

Se espera que este estudio sirva de motivación para seguir el desarrollo investigativo de ciclos termoquímicos, considerando el gran potencial de energía solar disponible en Chile, la utilización de ciclos termoquímicos podría ser una alternativa viable para la producción de hidrógeno en el futuro.

Palabras clave: Water-Splitting, ciclos termoquímicos, óxido de cerio, medios porosos, modelación computacional.

Abstract

The present work is about the development and implementation of a computational model that simulates the thermochemical process of reduction and oxidation for the production of hydrogen through Water-Splitting. This initiative was born within FONDAP project “Solar Energy Research Center (SERC-Chile)”, specifically in the line of development and research of energy storage and solar-chemistry; as part of this project, the design and start-up of a Water-Splitting reactor was carried out; the model developed here is based on this reactor’s operation.

To develop the model, the physical situation of the problem was first analysed and defined. After a bibliographic review, it was decided to treat the reactant as a porous medium. A porous media combustion model [19] was used for energy and mass balances, on the other hand, vacancy concentration variation equations ([11] and [10]) were used to determine the production of oxygen and hydrogen in the respective reduction / oxidation.

In addition to the Water-Splitting reactor model, a comparison model was made to evaluate the behavior of the kinetic component of the model and determine the degree of closeness of the model with experimental data. For this, the physical situation described by Arifin in [11] was used and a simple model was developed to obtain moles of oxygen and hydrogen according to the procedure described in [11]. From the comparison model, results were similar in oxidation but different in the reduction step. This difference is attributed to species diffusion phenomena and differences between the real and modeled physical situation and thermal behaviour of ceria.

Regarding the Water-Splitting reactor model, a sensitivity analysis of variables was carried out. This analysis determined that the most relevant variables were border temperatures and reaction time. Oxidation was more sensitive than reduction according to the cases presented. However, for the proposed model to be valid, it is necessary to carry out experiments with the Water-Splitting reactor in order to adjust the parameters and obtain results as close to reality as possible.

It is expected that this study will serve as motivation to continue the research of thermochemical cycles, considering the great potential of solar energy available in Chile, the use of thermochemical cycles could be a viable alternative for the production of hydrogen in the future.

Key words: Water-Splitting, thermochemical cycles, cerium oxide, porous media, computational modeling.

Glosario

Listado de símbolos y siglas

$[x]$	Concentración de x
δ	Concentración de vacancias
k	Constante de velocidad
R	Constante universal de los gases
E	Energía de activación
Δh°	Entalpía molar
A	Factor pre-exponencial o de frecuencia
α	Grado de completación de una reacción
γ	Orden de la reacción
e	Porosidad
<i>redox</i>	Referencia al ciclo reducción-oxidación
T	Temperatura

Especies químicas

CeO_2	Óxido de cerio (ceria)
$\text{CeO}_{2-\delta}$	Ceria no-estequiométrica
CO_2	Dióxido de carbono
CO	Monóxido de carbono
H_2	Hidrógeno molecular
H_2O	Agua
O_2	Oxígeno molecular

Sub-índices

<i>ce</i>	Calentador eléctrico
<i>borde, r</i>	Condición de borde de reducción
<i>borde, o</i>	Condición de borde de oxidación
<i>g</i>	Gas
<i>gas</i>	Gaseoso
<i>max</i>	Máximo
<i>ox</i>	Oxidación
<i>red</i>	Reducción
<i>s</i>	Sólido
<i>vac</i>	Vacancias

Índice

Introducción	XII
Capítulo 1: Marco Teórico	1
1.1. Hidrógeno	1
1.1.1. Métodos de producción	1
1.1.2. Almacenamiento	2
1.1.3. Aplicaciones	3
1.2. Hidrógeno Verde	4
1.2.1. Electrólisis	5
1.2.2. Termólisis	6
1.2.2.1. Termólisis directa	6
1.2.2.2. Ciclos termoquímicos	6
1.3. Ciclos termoquímicos de dos etapas	7
1.4. Ciclo CeO_2 : Cinética de reacción	8
1.4.1. Modelo de Bulfin	9
1.4.2. Entalpía de reacción	12
1.4.3. Modelo de Arifin	14
1.5. Modelación computacional de Water-Splitting	15
Capítulo 2: Modelo Matemático	18
2.1. Situación Física	18
2.2. Ecuaciones gobernantes	20
2.2.1. Continuidad de la mezcla	21
2.2.2. Energía de la fase gaseosa	21
2.2.3. Energía de la fase sólida	22
2.2.4. Ecuación de reacción química	24
2.2.5. Ecuación de conservación de especies químicas	25
2.2.6. Condiciones de borde	26
2.2.6.1. Reducción(endotérmica)	26
2.2.6.2. Oxidación(exotérmica)	27
2.2.7. Consideraciones del modelo	28
2.2.7.1. Temperatura inicial del sólido (T_s) para la oxidación	30

2.3.	Propiedades físicas del medio poroso y del gas	31
2.3.1.	Medio poroso (CeO_2)	31
2.3.2.	Gas argón	31
2.3.3.	Vapor de agua	32
Capítulo 3:	Desarrollo Numérico	33
3.1.	Algoritmo de solución	33
3.2.	Discretización del modelo	34
3.2.1.	Ecuación de reacción química	34
3.2.2.	Ecuación de conservación de energía de la fase sólida.	35
3.2.3.	Ecuación de conservación de energía de la fase gaseosa.	38
3.2.4.	Determinación del tamaño de malla y paso de tiempo óptimo	39
Capítulo 4:	Modelo de comparación	47
4.1.	Introducción	47
4.2.	Situación Física	47
4.3.	Ecuaciones gobernantes	48
4.4.	Consideraciones	49
4.5.	Algoritmo de solución	50
Capítulo 5:	Resultados y Análisis	51
5.1.	Modelo de comparación	51
5.1.1.	Parámetros	51
5.1.2.	Resultados	52
5.1.2.1.	Cinética Bulfin & Arifin	52
5.1.2.2.	Cinética Arifin	54
5.1.2.3.	Variación $n_{red}=1.65$	56
5.2.	Modelo 1-D	58
5.2.1.	Parámetros	59
5.2.2.	Parámetros a sensibilizar	60
5.2.3.	Producción total y casos extremos	61
5.2.4.	Sensibilización	63
5.2.5.	Casos Extremos: Perfiles de T_s , T_g , y δ	67
5.2.5.1.	Caso 1: Condiciones más favorables	68
5.2.5.2.	Caso 5: Condiciones menos favorables	74

Conclusiones	80
Bibliografía	82
Anexos	85

Índice de figuras

1.1. Métodos de producción de H_2 [2].	2
1.2. Aplicaciones del hidrógeno [6].	4
1.3. Ejemplo de celda electrolítica. Departamento de energía de EE.UU.	5
1.4. Cambio en la entalpía en la reducción vs concentración de vacancias δ . El ajuste logarítmico para $CeO_{2-\delta}$ tiene la forma $\Delta h^\circ = 395 - 31.4 \log(\delta)$ [kJ/mol] [13].	13
1.5. Cantidad total de H_2 y O_2 producido en Water-Splitting a distintas temperaturas de horno (temperatura de oxidación) [11].	15
1.6. Esquema reactor CR5 [15].	16
1.7. Arreglo piramidal de tubos de tungsteno y distribución de temperatura [18].	17
2.1. Representación de la situación física de la etapa de reducción.	19
2.2. Representación de la situación física de la etapa de oxidación.	20
3.1. Perfiles de temperatura T_s después de 10 minutos de reducción con diferentes valores de Δt	40
3.2. Perfiles de temperatura T_g después de 10 minutos de reducción con diferentes valores de Δt	40
3.3. Perfiles de concentración de vacancias δ después de 10 minutos de reducción con diferentes valores de Δt	41
3.4. Perfiles de temperatura T_s después de 10 minutos de oxidación con diferentes valores de Δt	41
3.5. Perfiles de temperatura T_g después de 10 minutos de oxidación con diferentes valores de Δt	42
3.6. Perfiles de concentración de vacancias δ después de 10 minutos de oxidación con diferentes valores de Δt	42
3.7. Perfiles de temperatura T_s después de 5 minutos de reducción con diferentes valores de n_x	43
3.8. Perfiles de temperatura T_g después de 5 minutos de reducción con diferentes valores de n_x	44
3.9. Perfiles de concentración de vacancias δ después de 5 minutos de reducción con diferentes valores de n_x	44
3.10. Perfiles de temperatura T_s después de 5 minutos de oxidación con diferentes valores de n_x	45

3.11. Perfiles de temperatura T_g después de 5 minutos de oxidación con diferentes valores de n_x	45
3.12. Perfiles de concentración de vacancias δ después de 5 minutos de oxidación con diferentes valores de n_x	46
5.1. Producción de oxígeno para el modelo de comparación considerando las cinéticas de Bulfin y Arifin para la reducción y oxidación respectivamente. . .	53
5.2. Producción de hidrógeno para el modelo de comparación considerando las cinéticas de Bulfin y Arifin para la reducción y oxidación respectivamente. . .	53
5.3. Producción de hidrógeno para el modelo de comparación utilizando resultados experimentales de reducción como condición inicial de oxidación. . . .	55
5.4. Producción de oxígeno para el modelo de comparación considerando $n_{red} = 1.65$	57
5.5. Producción de hidrógeno para el modelo de comparación considerando $n_{red} = 1.65$	58
5.6. Tabla de parámetros a sensibilizar.	60
5.7. Producción de O_2 y H_2 para casos extremos y casos intercalados de reducción (R) y oxidación (O). El signo + representa el Caso 1 (más favorable) y el signo - el Caso 5 (menos favorable).	62
5.8. Producción de hidrógeno para cada caso y variable de sensibilización. . . .	63
5.9. Sensibilización por pares de variables (parámetros de reducción y oxidación sensibilizados al mismo tiempo).	65
5.10. Producción de H_2 de acuerdo a la sensibilización de aporte de calor volumétrico Q_{ce}	66
5.11. Perfil de temperatura del sólido para la reducción considerando Caso 1 (más favorable).	68
5.12. Perfil de temperatura del gas para la reducción considerando Caso 1 (más favorable).	69
5.13. Distribución de vacancias para la reducción considerando Caso 1 (más favorable).	70
5.14. Perfil de temperatura del sólido para la oxidación considerando Caso 1 (más favorable).	71
5.15. Perfil de temperatura del gas para la oxidación considerando Caso 1 (más favorable).	72

5.16. Distribución de vacancias para la reducción considerando Caso 1 (más favorable).	73
5.17. Perfil de temperatura del sólido para la reducción considerando Caso 5 (menos favorable).	74
5.18. Perfil de temperatura del gas para la reducción considerando Caso 5 (menos favorable).	75
5.19. Distribución de vacancias del sólido para la reducción considerando Caso 5 (menos favorable).	76
5.20. Perfil de temperatura del sólido para la oxidación considerando Caso 5 (menos favorable).	77
5.21. Perfil de temperatura del gas para la oxidación considerando Caso 5 (menos favorable).	78
5.22. Distribución de vacancias del sólido para la oxidación considerando Caso 5 (menos favorable).	79
7.1. Sensibilización temperatura de borde.	85
7.2. Sensibilización velocidad del gas.	85
7.3. Sensibilización temperatura inicial (oxidación).	86
7.4. Sensibilización tiempo de reacción.	86
7.5. Sensibilización aporte de calor volumétrico.	86

Índice de cuadros

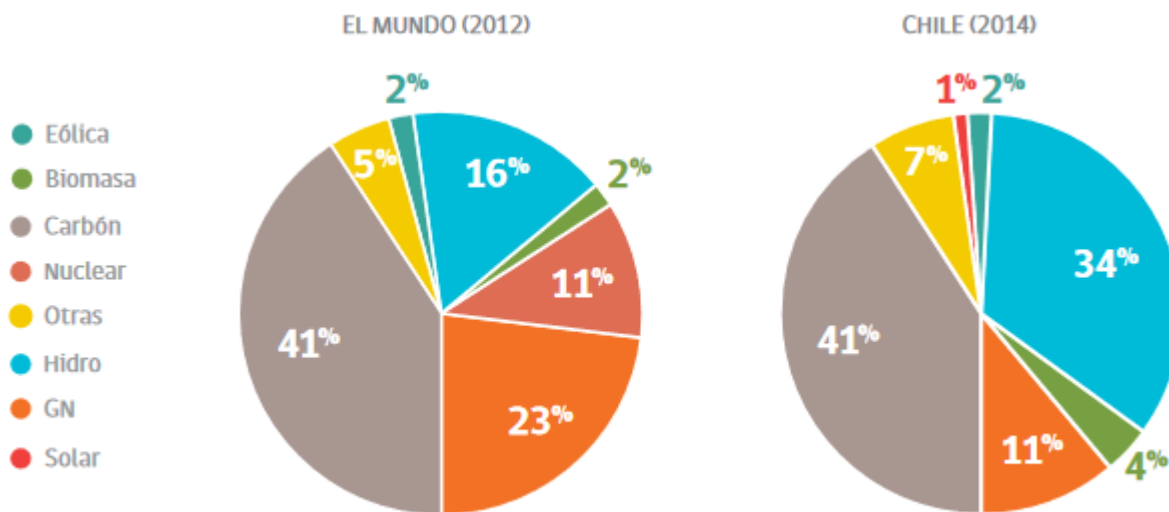
1.1. Parámetros de Arrhenius para la cinética de reacción definida por Bulfin [10].	11
1.2. Parámetros de Arrhenius para la cinética de reacción de oxidación definida por Arifin [11].	14
5.1. Tabla de parámetros utilizados en la etapa de reducción.	51
5.2. Tabla de parámetros utilizados en la etapa de oxidación.	51
5.3. Tabla de parámetros utilizados en ambas etapas.	51
5.4. Concentraciones de vacancias final (para cálculo de $f(\alpha)$) obtenida mediante proceso iterativo.	52
5.5. Concentración de vacancias final. Para la reducción, el valor de δ es equivalente a la producción de O_2 reportada por Arifin.	54
5.6. Concentración de vacancias final considerando un $n_{red} = 1.65$	57
5.7. Tabla de parámetros utilizados en la etapa de reducción.	59
5.8. Tabla de parámetros utilizados en la etapa de oxidación.	59
5.9. Tabla de propiedades del medio poroso utilizadas para la simulación y obtención de resultados.	59
5.10. Parámetros generales utilizados en la simulación.	59
5.11. Aporte de calor volumétrico y aporte de calor total equivalente.	67

Introducción

Motivación

El cuidado del planeta ha sido uno de los principales desafíos que se ha planteado la civilización humana en los últimos 40 años. En estos años se han buscado diversas soluciones a los problemas asociados al uso de combustibles fósiles y derivados del petróleo, problemas que podemos constatar al observar el entorno y darnos cuenta de lo alteradas que se encuentran las estaciones -debido al cambio climático por ejemplo- o al ver la contaminación atmosférica presente en varias ciudades al rededor del planeta.

Actualmente, la matriz energética chilena está compuesta en su mayoría por combustibles fósiles, sin embargo, se proyecta que en 2050 el 70 % de la energía eléctrica que se consuma en Chile será de origen solar o eólico, lo cual se pretende lograr en paralelo con el Plan de Descarbonización [1], esto gracias a las ventajas comparativas y a la disponibilidad del recurso solar y eólico en el país.



Matriz energética de Chile y mundial. Política energética de Chile: Energía 2050.

Una de las desventajas de las energías renovables no convencionales es que, a diferencia de los combustibles fósiles, no se pueden almacenar; esto implica que la energía no puede ser suministrada cuando se requiera, sino que cuando haya sol o viento por ejemplo. En este contexto, el hidrógeno producido mediante ERNC (denominado 'hidrógeno verde') aparece como un vector energético importante para servir como medio para almacenar energía generalmente solar o eólica.

En este trabajo en particular, cuyo espíritu es ser una contribución para la ciencia en lo relacionado al hidrógeno verde, se estudiará la producción de hidrógeno mediante un método llamado Water-Splitting, en el cual es posible -a través de un proceso termoquímico- obtener hidrógeno utilizando energía solar concentrada, vapor de agua y un óxido metálico utilizado como reactante.

El estudio de este proceso termoquímico se realizará de manera teórica empleando modelos matemáticos ya definidos, acoplándolos y resolviéndolos mediante métodos numéricos. Como herramienta de solución numérica se utiliza Python 3, en específico la biblioteca NumPy.

Reactor de Water-Splitting

El modelo unidimensional de Water-Splitting a desarrollar es una simplificación del funcionamiento de un reactor diseñado en la Universidad Técnica Federico Santa María. . La energía necesaria para la reacción es principalmente de origen solar; se utiliza un espejo parabólico para concentrar la energía de los rayos del sol y así transferirla al reactante ubicado al interior del reactor. Otra parte de la energía utilizada proviene de un alambre enrollado al rededor del reactor el cual es utilizado como un calentador eléctrico, brindando así un aporte extra de energía al sistema.

El reactor consta de una estructura metálica con una cámara de reacción en su interior, donde se coloca una probeta de cuarzo; en esta probeta se centra el foco del espejo parabólico para así absorber la energía solar y transferirla al reactante. El reactor también tiene aberturas para la entrada y salida de los gases que participan en el proceso termoquímico.

Marco de referencia.

Dentro del marco del proyecto FONDAP “Solar Energy Research Center (SERC-Chile)”, específicamente en la línea de desarrollo e investigación de almacenamiento de energía y química-solar.

Objetivo general.

Desarrollo e implementación de un modelo numérico unidimensional para la simulación computacional de un reactor de Water-Splitting termoquímico de dos etapas.

Objetivos específicos.

1. Definir contexto, situación física y ecuaciones gobernantes del problema.
2. Discretizar ecuaciones e implementar un código en Python para la resolución de las ecuaciones, siguiendo el algoritmo de Thomas (método de matriz tri-diagonal TDMA).
3. Comparar modelo cinético utilizado con resultados experimentales documentados considerando la respectiva situación física.
4. Analizar resultados obtenidos y concluir.

Capítulo 1: Marco Teórico

1.1. Hidrógeno

El hidrógeno es el primer y más liviano elemento de la tabla periódica. Es estable en forma molecular -H₂- sin embargo, y a pesar de ser uno de los elementos más abundantes en la tierra, no se encuentra en su forma atómica en la naturaleza, se encuentra combinado con otros elementos, como carbono y oxígeno (dando paso a compuestos orgánicos y agua, por ejemplo). Es por esta razón que es preciso obtenerlo mediante alguno de los diversos métodos existentes. Se ha convertido últimamente en un vector energético interesante, lo que ha motivado a la comunidad científica a desarrollar investigación sobre su producción, almacenamiento y aplicaciones.

1.1.1. Métodos de producción

Existen diversos métodos de producción de hidrógeno. La fuente de la energía utilizada para la producción determinará si el H₂ es de origen fósil, renovable o híbrido. Al existir formas tan diversas de producción, es posible obtener hidrógeno a niveles industriales en grandes plantas productoras, o a nivel más pequeño en instalaciones cercanas al punto de uso.

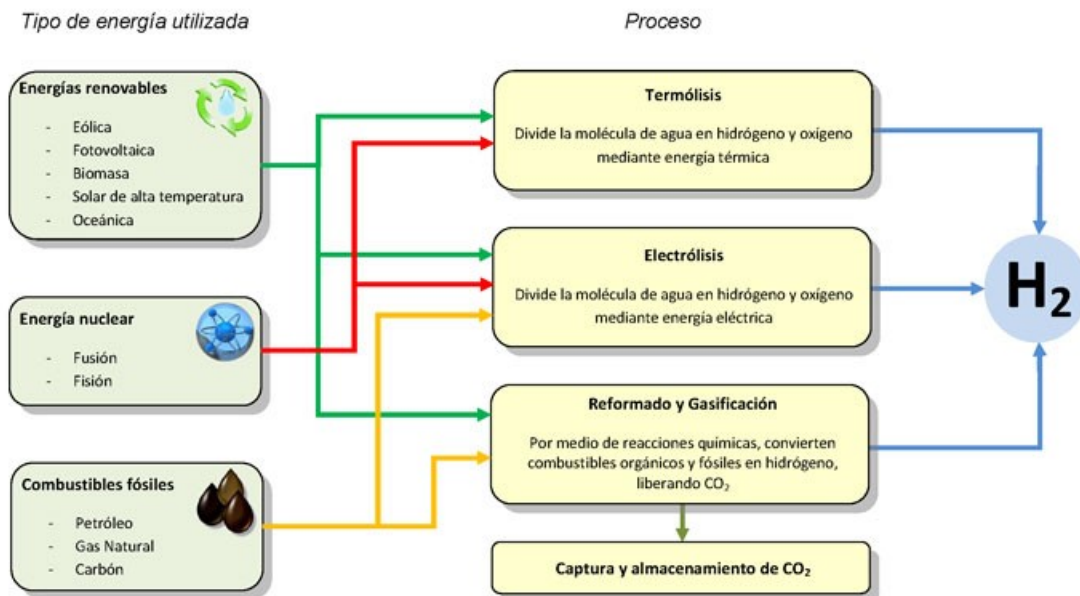


Figura 1.1: Métodos de producción de H₂ [2].

Existe una denominación de colores para el hidrógeno. El color es determinado por la cantidad de CO₂ liberado en la producción de hidrógeno, es decir, determina que tan limpio es este hidrógeno [3].

- Hidrógeno negro/marrón: Producto de la gasificación de carbón.
- Hidrógeno gris: Reformado de metano a partir de gas natural y vapor.
- Hidrógeno azul: Producido como hidrógeno gris pero cuyas emisiones son capturadas mediante un sistema de captura y almacenamiento de carbono (CCUS: carbon capture, utilization and storage).
- Hidrógeno verde: Producido de fuentes 100 % renovables, por lo que no se emite CO₂ en su producción.

1.1.2. Almacenamiento

Una de las características del hidrógeno es su alta densidad energética por unidad de masa, sin embargo, también tiene la característica de requerir un mayor volumen para almacenar

una cantidad de energía equivalente en comparación a otros combustibles. Debido a esto es que existen distintos métodos para el almacenamiento de H_2 , muchos de los cuales siguen en fase de desarrollo e investigación.

Estos métodos de almacenamiento se pueden dividir en:

- **Gas comprimido:** Es el método más tradicional y difundido. Las presiones de almacenamiento más utilizadas suelen ser de entre 200 y 350 [bar]. Existen también depósitos de 700 [bar] que cuentan con ensayos y certificación, sin embargo su uso es principalmente demostrativo.
- **Hidrógeno líquido:** El hidrógeno licuado (LH_2) es utilizado principalmente en aplicaciones de transporte aeroespacial, en estas condiciones el hidrógeno alcanza una densidad de energía volumétrica de $\sim 8.4 [MJ/l]$. Para llevar H_2 a estado líquido es necesario que alcance temperaturas de entre 14 y 20 [K], esta baja temperatura representa una de las principales dificultades de este método.
- **Hidruros metálicos:** Son compuestos formados a partir de una reacción entre hidrógeno y un elemento o compuesto metálico. Al controlar la temperatura y presión se puede manejar el sentido de la reacción química, pudiendo así almacenar hidrógeno. Un ejemplo de un hidruro metálico estudiado para almacenamiento a gran escala es el hidruro de magnesio [5].
- **Hidruros no-metálico:** Análogo a los hidruros metálicos, solo cambia el tipo de elemento que reacciona con H_2 . Dentro de los hidruros no-metálicos, uno de los más utilizados para almacenar hidrógeno es el amoníaco (NH_3).
- **Estructuras de carbono:** Existen estudios sobre estructuras que tienen una alta capacidad para almacenar hidrógeno. En particular, los nanotubos de carbono están siendo especialmente estudiados para este fin.

1.1.3. Aplicaciones

El hidrógeno ha sido utilizado a través de varias décadas en diversas industrias. De acuerdo con datos de 2007, el consumo de hidrógeno se distribuye entre producción de amoníaco

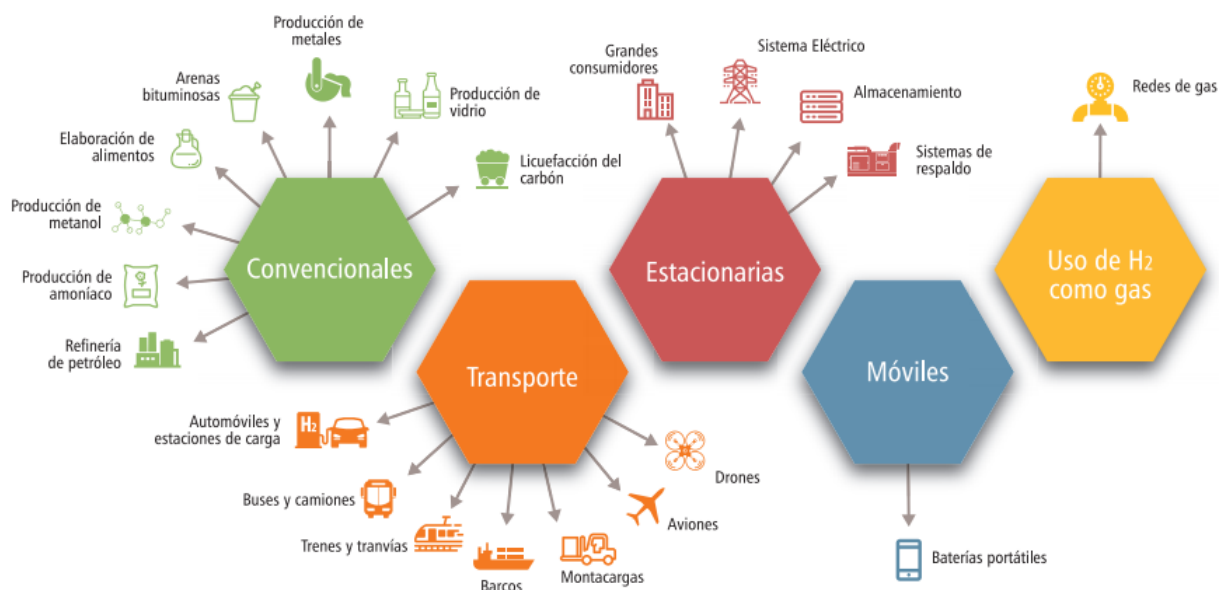


Figura 1.2: Aplicaciones del hidrógeno [6].

(54 % del consumo global de H_2), industria química/refinería (35 %), industria electrónica (6 %), industria del metal/vidrio (3 %) e industria alimentaria (2 %) [6].

Desde el punto de vista energético, el hidrógeno puede ser utilizado como combustible para transporte o para generar energía eléctrica mediante celdas de combustible (fuel cells). A partir de los poderes caloríficos se puede establecer comparaciones entre el hidrógeno y otros combustibles: 1[kg] de H_2 equivale a 2.78[kg] de gasolina, 2.80[kg] de gasóleo y ~ 3 [kg] de gas natural[7]. El elevado poder calorífico, junto con el hecho de que al combustión solamente se genera H_2O como producto, hacen al hidrógeno un candidato natural para reemplazar los combustibles fósiles en el futuro cercano.

1.2. Hidrógeno Verde

El hidrógeno verde ha tomado mucha relevancia en el último tiempo debido a la posibilidad que abre de poder producir/almacenar energía con cero emisiones ambientales. En Chile, esta relevancia ha sido incluso más grande debido a la gran disponibilidad del recurso solar y eólico, lo que lo vuelve al país sudamericano un candidato importante a ser líder en la producción mundial de hidrógeno verde. Dentro de los métodos de producción de H_2 verde se destaca la electrólisis y la termólisis.

1.2.1. Electrólisis

La electrólisis es uno de los métodos más utilizados para la producción de hidrógeno verde, por ser ya ampliamente investigada, limpia y relativamente simple. Consiste básicamente en hacer circular una corriente eléctrica continua a través de dos electrodos (ánodo y cátodo) en contacto con agua, lo que genera la ruptura de la molécula de agua (Water-Splitting) en hidrógeno y oxígeno. Cuando la fuente de la energía eléctrica es de origen renovable, el hidrógeno producido toma la categoría de hidrógeno verde.

En la producción de H_2 , los electrodos sumergidos en el electrolito acuoso están separados por una membrana (también llamada diafragma) que evita la recombinación con el oxígeno. El agua utilizada en el proceso debe ser tratada para así evitar la deposición de minerales en el sistema.

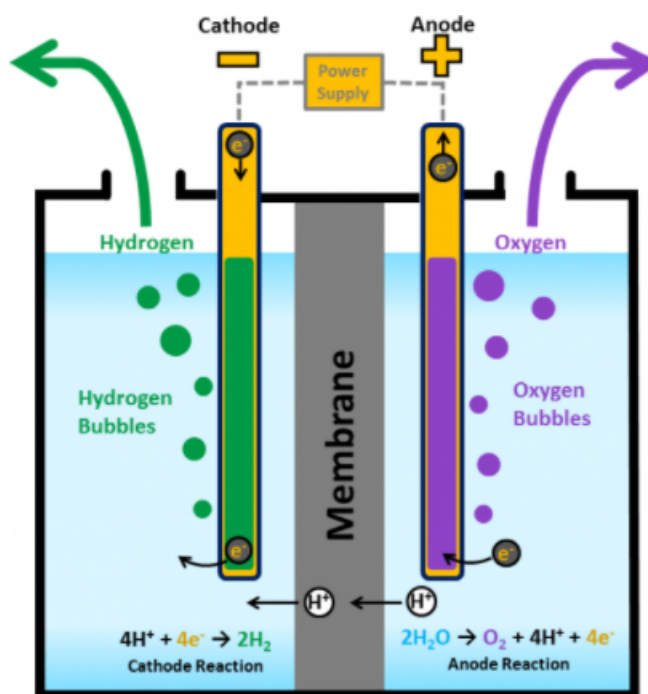


Figura 1.3: Ejemplo de celda electrolítica. Departamento de energía de EE.UU.

El conjunto de electrodos, el diafragma y electrolito componen un elemento básico denominado celda electrolítica. Un conjunto de celdas es el que compone al equipo de producción de hidrógeno llamado electrolizador. Se tienen distintos tipos de electrolizadores de acuerdo

a sus elementos y forma de funcionamiento, estos son: alcalinos, membrana de intercambio de protones y de estado sólido [6].

1.2.2. Termólisis

Tal como la electrólisis, la termólisis también es una forma de Water-Splitting, con la diferencia de que la energía utilizada para lograr el splitting es de carácter térmico. Para lograr la disociación del agua es necesario alcanzar altas temperaturas, es por ello que el uso de energía solar concentrada es idóneo. Inicialmente la fuente principal de energía de estos procesos era nuclear, sin embargo, debido a las características constructivas y físicas de los reactores su aplicación no es recomendable.

La termólisis puede ser un proceso directo o por etapas (mediante la utilización de ciclos termoquímicos).

1.2.2.1 Termólisis directa

La termólisis directa es la separación en un sólo paso de H_2O mediante la aplicación de energía térmica. Este proceso tiene dos grandes inconvenientes. Por un lado se necesitan temperaturas de entre 2000 y 3000[K] [8], esto se traduce en problemas de resistencia térmica de materiales y de re-radiación, lo que dificulta la absorción de energía. Por otro lado es preciso utilizar una técnica para separar el hidrógeno del oxígeno para así evitar la mezcla explosiva de estos elementos.

1.2.2.2 Ciclos termoquímicos

La utilización de ciclos termoquímicos para Water-Splitting soluciona los dos inconvenientes que se presentan en el caso de termólisis directa. El proceso consiste en la reducción y posterior oxidación de un compuesto. Los reactantes utilizados para realizar esta reacción son de diversa naturaleza: óxidos metálicos, cloruros de hierro, sulfuros, entre otros [9].

La temperatura requerida para la reducción depende del compuesto en el que se trabaje, sin embargo, es la más alta del proceso, y por lo general es del orden de 2000[K]; en este sentido, el proceso en sí utiliza temperaturas en promedio menores a la temperatura de la

termólisis directa. Por otro lado, al realizarse la reducción y oxidación en dos etapas distintas se evita la recombinación de oxígeno e hidrógeno.

Estos ciclos pueden ser de varias etapas. En [9] se realizó una revisión de ciclos termoquímicos de 2 hasta 6 etapas, sin embargo, cada etapa del ciclo está asociada a pérdidas de calor y de separación de los productos, por ende las ineficiencias son inherentes y aumentan con el número de etapas. De acuerdo con esto, los ciclos de dos etapas son aquellos que resultan más interesantes para su estudio.

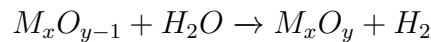
1.3. Ciclos termoquímicos de dos etapas

Al ser un ciclo de dos etapas, la reducción y posterior oxidación se hacen en etapas consecutivas de acuerdo a la siguientes ecuaciones, correspondientes a una reducción y oxidación genéricas para cualquier óxido metálico.

Reducción:



Oxidación:



La primera etapa consiste en una reacción endotérmica en donde se realiza la reducción del reactante. La energía requerida para esta reacción puede provenir de distintas fuentes, pero al ser las temperaturas del orden de los 2000[K] se opta por sistemas de concentración solar. Al llegar al rango de temperaturas de reducción, los átomos de oxígeno se desprenden del material; este rango varía según el material que se utilice para el ciclo *redox*. Se suele utilizar un gas inerte como gas de arrastre para reducir la recombinación del O_2 que queda depositado en la superficie del material.

Una vez que se remueve el O_2 se da paso a la reacción exotérmica de oxidación. Para producir esta reacción se expone el reactante reducido a vapor de agua. Tanto las vacancias de oxígeno existentes en el material como energía térmica aplicada al proceso permiten alcanzar

la energía y entropía de formación del agua, generando la ruptura de la molécula de H_2O . El oxígeno es capturado por las vacancias mientras que el hidrógeno es expulsado con la corriente de vapor que no alcanza a reaccionar.

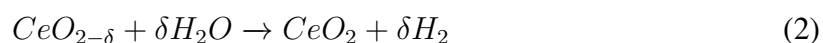
Existe una amplia gama de reactantes que pueden ser utilizados para realizar Water-Splitting termoquímico de dos etapas. Estos reactantes poseen distintas características y rendimientos, también pueden ser calificados como volátiles y no volátiles. Estos últimos son particularmente interesantes debido a que resulta más simple el funcionamiento físico al no existir cambios de fase de por medio. Uno de los reactantes no volátiles más estudiado recientemente es el óxido de cerio, CeO_2 o también denominado *ceria*.

1.4. Ciclo CeO_2 : Cinética de reacción

La cinética de reacción de la reducción y oxidación de la ceria no-estequiométrica ha sido recientemente estudiada y definida empíricamente. Al ser expuesta a alta temperatura, este compuesto se reduce, liberando oxígeno.



La ceria reducida puede ser utilizada para separar H_2O o CO_2 .



En el año 2013, B. Bulfin y un grupo de investigadores de Dublin desarrollaron un modelo analítico de la oxidación y reducción de CeO_2 [10]. Mediante este modelo es posible determinar la variación de la concentración de vacancias (δ) de una muestra de CeO_2 al ser reducida u oxidada según sean las condiciones. Por otro lado, en 2018, D. Arifin y A. Weimer publicaron su investigación sobre la cinética de reacción de la producción de H_2 y CO solar-termoquímico mediante la oxidación de CeO_2 reducido [11]. Si bien el estudio de Bulfin involucra la cinética de tanto la reducción como la oxidación, este último proceso se realiza

en una atmósfera oxigenada, diferenciándose de la oxidación por Water-Splitting realizada por Arifin.

Ley de Arrhenius

Los modelos presentados en los trabajos citados anteriormente están basados en la Ley de Arrhenius, la cual determina la dependencia de la temperatura en la velocidad de reacción de un proceso.

Para varias reacciones químicas, la velocidad de reacción se puede escribir como el producto de un término dependiente de la temperatura y otro dependiente de la composición. Esto, de acuerdo con [12], se representa matemáticamente como:

$$\begin{aligned} r_i &= f_1(\text{temperature}) \cdot f_2(\text{composition}) \\ &= k \cdot f_2(\text{composition}) \end{aligned} \quad (4)$$

$$k = k_0 e^{\frac{-E}{RT}} \quad (5)$$

La Ley de Arrhenius se puede escribir como (5), donde k_0 corresponde al término pre-exponencial o de frecuencia, E a la energía de activación, R la constante universal de los gases y T la temperatura.

1.4.1. Modelo de Bulfin

El modelo desarrollado en el trabajo de Bulfin [10] fue determinado a partir de datos presentes en la literatura y experimentos desarrollados en el mismo contexto de la investigación. En el experimento realizado para validar y corregir el modelo se utilizaron pellets de óxido de cerio, los cuales fueron reducidos a alta temperatura y baja presión parcial de oxígeno para luego ser re-oxidados en una atmósfera oxigenada.

A partir de los datos experimentales y bibliográficos se elaboró un modelo que permite predecir la composición de equilibrio del CeO_2 en un amplio rango de presiones ($2 \cdot 10^{-4}$ y $2 \cdot 10^{-8}$ [bar]) y temperaturas (1000 y 1900 [°C]).

La reacción no se produce hasta la total descomposición del CeO_2 , sino que se llega hasta un valor máximo de δ denominado δ_{max} ¹. La reacción de reducción depende de la concentración de oxígeno removible, mientras que la oxidación depende de la concentración de vacancias y de la concentración de oxígeno gaseoso (se desprecia la difusión de oxígeno por el momento). En este sentido, la variación en la concentración de vacancias será la tasa a la que el oxígeno sale del CeO_2 menos la tasa a la cual éste se re-combina:

$$\frac{d[O_{vac}]}{dt} = [O_{Ce}]k_{red} - [O_{vac}][O_{gas}]^n k_{ox} \quad (6)$$

Donde k_{red} y k_{ox} son las constantes de velocidad, las cuales tienen la forma de Arrhenius.

Al dividir (6) por la concentración de cerio $[Ce]$ se tiene:

$$\frac{1}{[Ce]} \frac{d[O_{vac}]}{dt} = \frac{[O_{Ce}]}{[Ce]} k_{red} - \frac{[O_{vac}]}{[Ce]} [O_{gas}]^n k_{ox} \quad (7)$$

Los valores $[O_{Ce}]/[Ce]$ y $[O_{vac}]/[Ce]$ pueden ser obtenidos a través de la ecuación estequiométrica (1), considerando un grado de reducción de δ o de δ_{max} :

$$\begin{aligned} \frac{[O_{Ce}]}{[Ce]} &= \delta_{max} - \delta \\ \frac{[O_{vac}]}{[Ce]} &= \delta \end{aligned}$$

Reemplazando en (7) resulta:

$$\frac{d\delta}{dt} = (\delta_{max} - \delta)k_{red} - (\delta)[O_{gas}]^n k_{ox} \quad (8)$$

Al representar las constantes de velocidad por sus formas de Arrhenius y la concentración de oxígeno gaseoso como una función de la presión parcial de oxígeno resulta la ecuación siguiente:

$$\frac{\partial \delta}{\partial t} = (\delta_{max} - \delta) \cdot A_{red} \cdot \exp\left(\frac{-E_{red}}{R \cdot T_s}\right) - \delta \cdot P_{O_2}^n \cdot A_{ox} \cdot \exp\left(\frac{-E_{ox}}{R \cdot T_s}\right) \quad (9)$$

¹En [10] se denomina a δ_{max} como x .

Donde A representa el factor pre-exponencial para la reducción y oxidación respectivamente, tal como E representa las energías de activación. P_{O_2} es la presión parcial de oxígeno.

Para la ecuación (9) se definieron los siguientes parámetros:

d_{max}	0.35
n	0.218 ± 0.0013
E_{red}	232 ± 5 [kJ/mol]
E_{ox}	36 ± 4 [kJ/mol]
A_{red}	720000 ± 360000 [1/s]
A_{ox}	82 ± 41 [1/(s·bar)]

Cuadro 1.1: Parámetros de Arrhenius para la cinética de reacción definida por Bulfin [10].

Para considerar la difusión de oxígeno a través del material se aplica una restricción a la velocidad de reacción definida en la ecuación (9):

$$\frac{d\delta}{dt} = tasa * (1 - \alpha)^{1/3} \quad (10)$$

Donde “tasa” es igual al valor obtenido de (9). El valor de α representa el grado de completación de la reacción, es decir la cantidad de oxígeno liberado/absorbido hasta un tiempo determinado dividido la cantidad total de oxígeno liberado/absorbido. Este valor es definido matemáticamente por Arifin [11] y se muestra en la sección subsiguiente.

De acuerdo a lo señalado por los autores, este modelo se puede acoplar fácilmente a simulaciones de flujo de calor. Solamente se debería considerar el término correspondiente al aporte o pérdida de calor debido a la reacción química en el balance de energía:

$$Q = \frac{d\delta}{dt} \Delta H$$

Es importante señalar que la tasa obtenida de (9) fue derivada dividiendo los términos por la concentración de cerio (7), en este sentido, para obtener el valor absoluto es necesario multiplicar el valor obtenido por la concentración de cerio [Ce].

Si se trata el reactante como un medio poroso, [Ce] corresponderá a los moles de Ce por metro cúbico de medio poroso, lo que también se puede representar como la concentración de cerio en fase sólida multiplicado por el volumen que corresponde al sólido $(1 - e)$, donde

e corresponde al espacio vacío del medio poroso:

$$[Ce] = (1 - e)[Ce]_{sólido}$$

La fase sólida de este medio poroso puede ser considerada como una solución sólida de cerio y oxígeno. En este sentido, mediante la densidad y el peso molecular se pueden obtener los moles de CeO_2 por cada metro cúbico de sólido:

$$[CeO_2]_{sólido} = \frac{\rho_{CeO_2}}{PM_{CeO_2}} = \frac{7.22[g/cm^3]}{172.115[g/mol]} \cdot \frac{10^6[cm^3/m^3]}{10^3[mol/kmol]} = 41.9487[kmol/m^3]$$

A partir de los pesos moleculares se puede obtener la fracción molar de Ce en el CeO_2 :

$$\Psi_{Ce} = \frac{PM_{Ce}}{PM_{CeO_2}} = 0.8237[-]$$

Multiplicando la fracción molar de Ce por $[CeO_2]$ se obtendrá la concentración de cerio en el sólido:

$$[Ce]_{sólido} = [CeO_2]_{sólido} \cdot \Psi_{Ce} = 34.5531[kmol/m^3]$$

Finalmente, la concentración de Ce en el medio poroso según la estequiometría es:

$$[Ce] = (1 - e) \cdot 34.5531[kmol/m^3]$$

La concentración de cerio en el sólido a utilizar en el modelo podría ser modificada de acuerdo a la concentración reportada por el fabricante del reactante a utilizar.

1.4.2. Entalpía de reacción

Como se mencionó con anterioridad, las etapas de reducción y oxidación que componen el ciclo termoquímico corresponden a reacciones endotérmica y exotérmica respectivamente. En este sentido hay que considerar que las entalpías de reacción ΔH (o Δh° cuando se trata de entalpía por unidad de mol) debieran ser de signo negativo y positivo respectivamente.

A medida que concentración de vacancias (δ) aumenta o disminuye, la entalpía de reacción también se modifica. Este fenómeno también fue objeto de estudio por B. Bulfin [13]. En dicha investigación se contrastan datos de equilibrio termogravimétrico obtenidos experimentalmente con aquellos disponibles en la literatura.

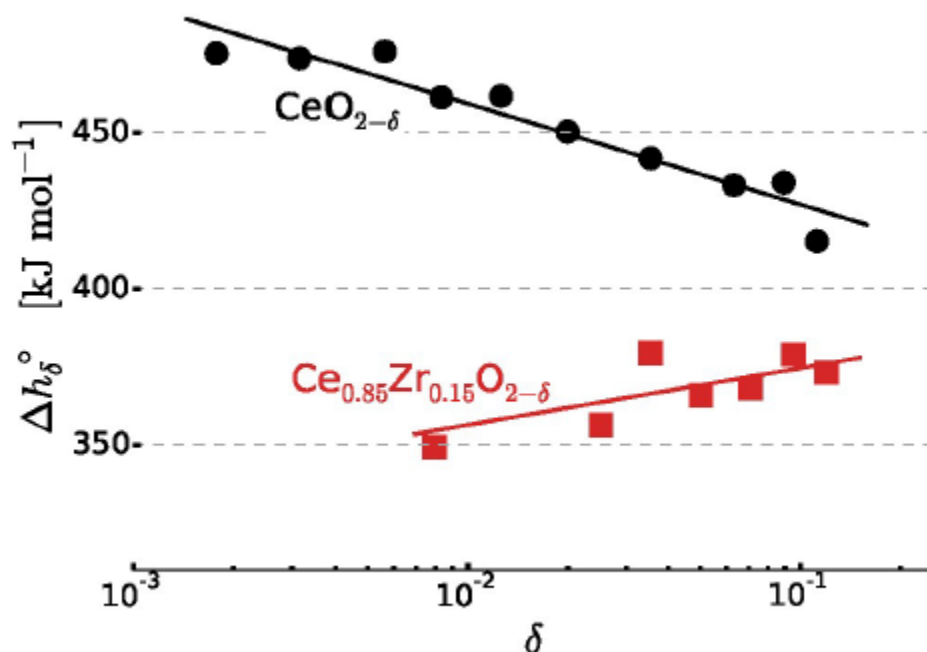


Figura 1.4: Cambio en la entalpía en la reducción vs concentración de vacancias δ . El ajuste logarítmico para $\text{CeO}_{2-\delta}$ tiene la forma $\Delta h^\circ = 395 - 31.4 \log(\delta)$ [kJ/ mol] [13].

Si consideramos el ciclo *redox* como reversible, las entalpías de reacción de la reducción y oxidación toman el mismo valor numérico con signos opuestos. Sin embargo, en el caso particular de la oxidación por Water-Splitting se tiene que considerar además la entalpía de reacción debido a la ruptura de la molécula de agua, constante y de valor $\Delta h^\circ = -285.85$ [kJ/ mol] [14]. Al sumar la entalpía del *splitting* con la entalpía de la oxidación de la ceria se obtiene un valor $\Delta h^\circ < 0 \forall \delta$, por lo que la reacción globalmente es exotérmica.

1.4.3. Modelo de Arifin

En [11] se estudia en particular la reacción de oxidación de los procesos de Water-Splitting y CO₂-Splitting. A partir de experimentos desarrollados es que se define la cinética de reacción. Para el caso de Water-Splitting, la reacción se realiza en los rangos de 750 - 950 [°C] y una fracción volumétrica de H₂O de entre 20 - 40 %.

Al igual que en el modelo desarrollado por Bulfin, la constante de velocidad tiene la forma de Arrhenius:

$$rate\left[\frac{1}{s}\right] = A_0 e^{\left(\frac{E_0}{RT}\right)} [Y_{oxid}(t)]^\gamma f(\alpha) \quad (11)$$

$$\alpha = \int_0^t \frac{H_2(t)}{H_{2,total}} dt \vee \alpha = \int_0^t \frac{CO(t)}{CO_{total}} dt$$

Donde A₀ y E₀ corresponden a los parámetros de Arrhenius de la reacción, γ el orden de la reacción de la fracción molar de oxidante y f(α) es una función que describe el progreso de la reacción.

A partir de los resultados experimentales, los parámetros de (11) de la tabla siguiente son aquellos que mejor se ajustan al modelo:

Reacción	A ₀ [1/s]	E ₀ [kJ/mol]	γ	f(α) = (1 - α) ⁿ	Temperatura [°C]
Water-Splitting	1.0	29	0.89	n=1.0	750-950

Cuadro 1.2: Parámetros de Arrhenius para la cinética de reacción de oxidación definida por Arifin [11].

En el trabajo de Arifin también se reportan datos de producción de hidrógeno y oxígeno en función de la temperatura de oxidación. Esta temperatura es además la temperatura inicial del proceso de reducción (desde esta temperatura se comienza a calentar el material).

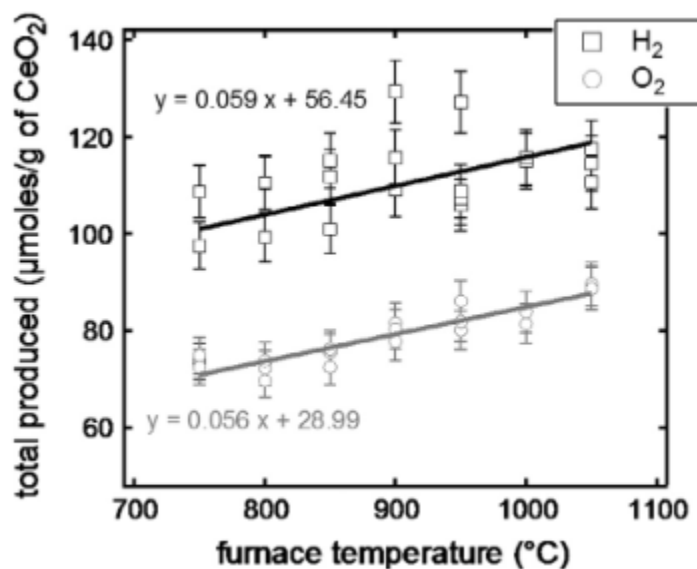


Figura 1.5: Cantidad total de H₂ y O₂ producido en Water-Splitting a distintas temperaturas de horno (temperatura de oxidación) [11].

1.5. Modelación computacional de Water-Splitting

No existen muchas publicaciones al respecto de la modelación computacional de Water-Splitting. Sin embargo, y respecto a la modelación del proceso utilizando óxido de cerio como reactante, se encuentran en la literatura los trabajos de Darryl James en 2006 [15] y de R. E Hogan en 2012 [16] en los cuales se desarrolla el modelo computacional de un reactor de Water-Splitting llamado "CR5" (Counter-Rotating-Ring Receiver/Reactor/Recuperator). En este reactor, alimentado por energía solar concentrada, el reactante se encuentra en forma de anillos, donde suceden las etapas de reducción y oxidación al mismo tiempo.

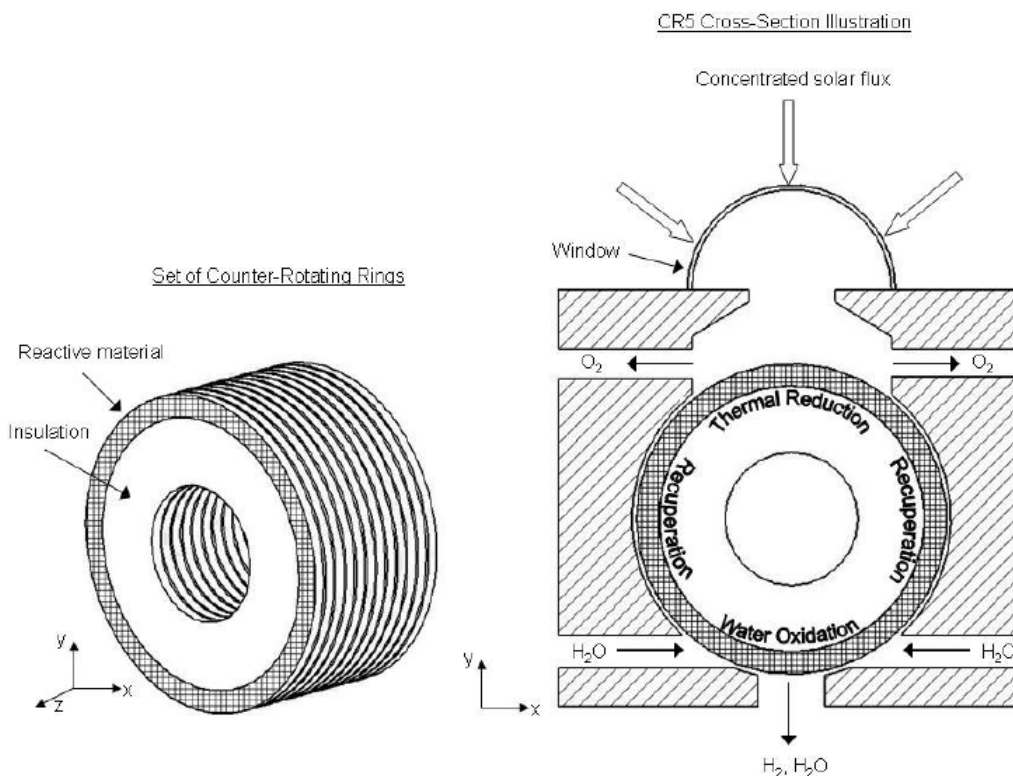


Figura 1.6: Esquema reactor CR5 [15].

Por otro lado, Valle-Hernandez en 2016 ([17] y [18]) también realiza un modelo computacional de un reactor de *splitting*. En este modelo la ceria se encuentra dentro de un arreglo piramidal de tubos de tungsteno, los cuales son calentados mediante energía solar concentrada. En el modelo la ceria se considera un medio poroso. La investigación reporta perfiles de temperatura y velocidad dentro del tubo de tungsteno, así como también perfiles de concentración de oxígeno e hidrógeno. Para la modelación se resolvieron las ecuaciones de mecánica de fluidos (utilizando ecuaciones de Brinkman[18]), transferencia de masa y energía en el tubo de tungsteno. Para la resolución se utilizó un software de CFD no especificado.

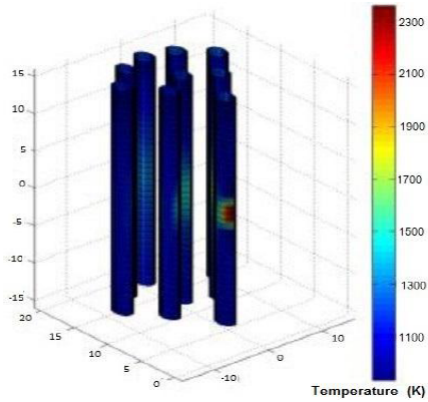


Figura 1.7: Arreglo piramidal de tubos de tungsteno y distribución de temperatura [18].

Respecto al considerar el reactante como un medio poroso, existen en la literatura diversas investigaciones sobre fenómenos de combustión en medios porosos. En investigaciones desarrollada por M. Toledo en 2005 [19] y 2009 [20] se presentan un modelo unidimensional de combustión en medios porosos, el cual se basa en el método de volúmenes promedio y se resuelven ecuaciones de conservación de la energía, masa y especies químicas.

El sistema de ecuaciones que resulta al plantear el modelo en el medio poroso discretizado forma una matriz tri-diagonal. El Algoritmo de Thomas (también llamado TDMA: Tri-Diagonal-Matrix-Algorithm) es idóneo para la resolución de este tipo de sistemas [21].

Capítulo 2: Modelo Matemático

2.1. Situación Física

Para producir la reacción de Water-Splitting en primer lugar se calienta el óxido metálico al interior del reactor (descrito en la Introducción) para así llevarlo a la temperatura de reducción. Durante este proceso se inyecta un gas inerte el cual funciona como gas de arrastre para remover el oxígeno que se libera. Luego de reducir el material se da inicio a la etapa de oxidación, en la cual se deja de suministrar la energía solar (debido a que se requiere menor temperatura) y se inyecta vapor de agua para así obtener a la salida del reactor una mezcla de vapor de agua e hidrógeno.

El modelo a proponer consta en una simplificación unidimensional del reactor de Water-Splitting descrito. Se utilizará óxido de cerio como reactante, el cual será modelado como un medio poroso. El aporte de calor suministrado por el calentador eléctrico será considerado como un aporte volumétrico de calor, mientras que el aporte de calor proveniente de la energía solar concentrada se considerará como una condición de borde en modelo. Debido a que el proceso consta de dos reacciones en serie a continuación se presentan diagramas representativos de las situaciones físicas de los procesos de reducción y oxidación respectivamente:

Reducción (endotérmica)



La reducción en un solo paso de la ceria se puede describir mediante la ecuación (12). Debido a la naturaleza inerte del gas de arrastre este no reacciona con los elementos y es por ese motivo que no se considera en la ecuación de reacción.

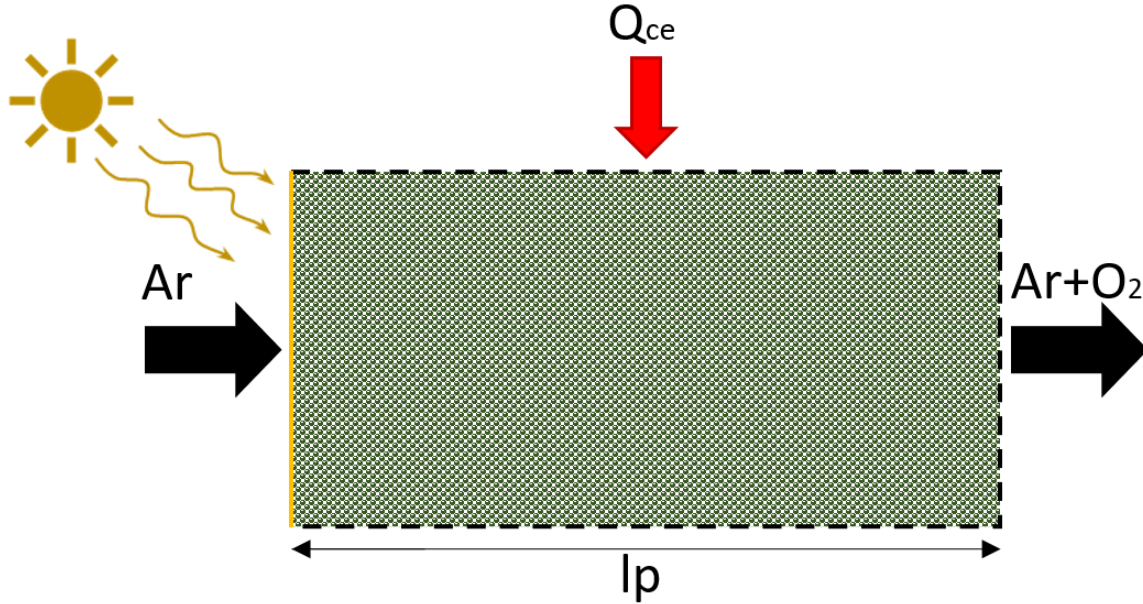
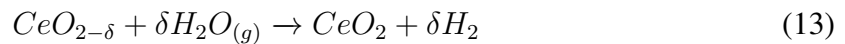


Figura 2.1: Representación de la situación física de la etapa de reducción.

Se considerará la zona de integración la longitud de material poroso l_p . El aporte de energía solar será incluido como condición de borde en el extremo izquierdo. A medida que procede la reacción, se genera una transferencia de calor a través de las fases sólidas y gaseosas del medio poroso, generando perfiles de temperaturas $T_g(x, t)$ y $T_s(x, t)$. La generación de vacancias en la ceria (fase sólida) producto de la reducción será representada por el parámetro $\delta(x, t)$, parámetro que va aumentando a medida que ocurre la reacción de reducción. A la salida se obtienen como productos argón y el oxígeno desprendido debido a la reducción.

Oxidación (exotérmica)



La oxidación en un solo paso de la ceria se puede describir mediante la ecuación (13).

Dentro de esta ecuación está considerada la ruptura de la molécula de agua según:



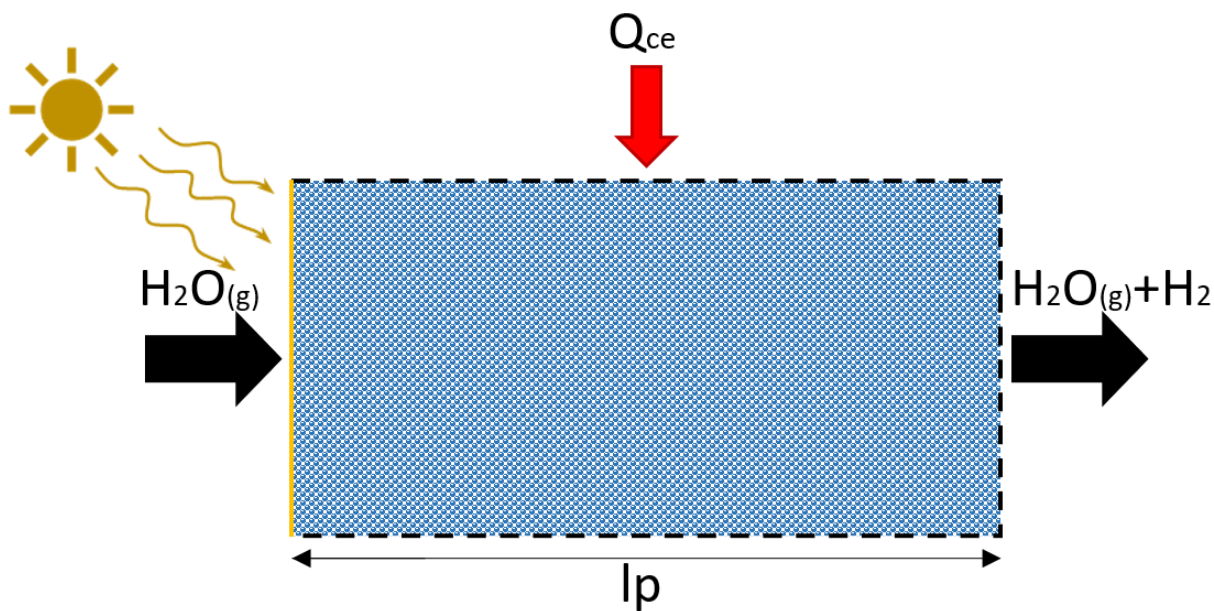


Figura 2.2: Representación de la situación física de la etapa de oxidación.

La situación física es geoméricamente idéntica a la de la etapa de oxidación, la gran diferencia radica en los elementos de entrada y salida, así como también en la condición de borde del lado izquierdo, la cual se detallará en la sección de condiciones de borde. A medida que procede la reacción, la temperatura va decayendo debido a la inyección de vapor y a la condición de borde de temperatura, el material se va re-oxidando por lo que el parámetro δ va disminuyendo en todo el material hasta llegar idealmente a un valor de cero y completar la re-oxidación de la ceria. Se libera H_2 como consecuencia del *splitting* de agua y de la absorción del oxígeno que se produce debido a las vacancias generadas en la etapa anterior (reducción). Debido a que se inyecta agua en exceso para producir la reacción es que se obtiene como producto de salida una mezcla de vapor de agua con hidrógeno gaseoso.

2.2. Ecuaciones gobernantes

El modelo unidimensional de la situación física se desarrollará acoplando el balance de masa y energía utilizado en modelos de combustión en medios porosos (específicamente aquel utilizado por Toledo [19]) con la cinética de reacción descritas por Bulfin [10] y Arifin [11].

2.2.1. Continuidad de la mezcla

Esta ecuación representa la conservación de masa de la fase gaseosa (la fase sólida al ser estática no necesita ecuaciones de conservación).

$$\frac{\partial}{\partial x}(\rho_g \cdot u_g) = 0 \quad (15)$$

Donde:

- $\rho_g[\text{kg/m}^3]$: Densidad de la fase gaseosa.
- $u_g[\text{m/s}]$: Velocidad de la fase gaseosa.

2.2.2. Energía de la fase gaseosa

Esta ecuación representa la conservación de energía para la fase gaseosa.

$$\varepsilon \cdot (\rho \cdot C_p)_g \left(\frac{\partial T_g}{\partial t} + u_g \frac{\partial T_g}{\partial x} \right) = \frac{\partial}{\partial x} (\varepsilon \cdot \lambda_g \frac{\partial T_g}{\partial x}) - a \cdot (T_g - T_s) \quad (16)$$

Donde:

- $\varepsilon[-]$: Porosidad entre partículas.
- $(\rho \cdot C_p)_g [\text{J/m}^3\text{K}]$: Densidad y calor específico del gas.
- $\lambda_g[\text{W/mK}]$: Coeficiente de conductividad térmica del gas.
- $a [\text{W/m}^3\text{K}]$: Coeficiente intersticial de intercambio de calor. El término que contiene este parámetro representa la transferencia de calor entre las dos fases presentes.
- $T_g[\text{K}]$: Temperatura del gas.
- $T_s[\text{K}]$: Temperatura del sólido.
- $Q_{ce}[\text{W/m}^3]$: Flujo de calor aportado por el calentador eléctrico.

Se puede apreciar que las unidades de todos los términos de la ecuación corresponden a potencia por unidad de volumen $[\text{W/m}^3]$.

2.2.3. Energía de la fase sólida

Esta ecuación representa la conservación de energía para la fase sólida, la cual corresponde a la ceria modelada como un medio poroso.

$$(1 - \varepsilon)(\rho \cdot C_p)_s \frac{\partial T_s}{\partial t} = \frac{\partial}{\partial x} \left(\lambda_{ef} \frac{\partial T_s}{\partial x} \right) + a \cdot (T_g - T_s) + Q_{ce} \pm \frac{\partial \delta}{\partial t} [Ce] \cdot \Delta h^\circ \quad (17)$$

Donde:

- $(\rho \cdot C_p)_s [\text{J/m}^3\text{K}]$: Densidad y calor específico del sólido.
- $\lambda_{ef} [\text{W/mK}]$: Coeficiente de conductividad térmica efectivo del medio poroso.
- $[Ce] [\text{kmol/m}^3]$: Concentración de cerio.
- $\frac{\partial \delta}{\partial t} [Ce] \Delta h^\circ [\text{W/m}^3]$: Aporte o pérdida de calor debido a la reacción química. $\delta [-] \cdot [Ce] [\text{kmol/m}^3]$ representa la concentración de vacancias de oxígeno, por ende $\frac{\partial \delta}{\partial t} [Ce] [\text{kmol/m}^3\text{s}]$ es una representación del llenado/vaciado de vacancias en el tiempo. Por otro lado $\Delta h^\circ [\text{J/kmol}]$ representa la entalpía de la reacción.

Respecto al último término de la ecuación de conservación es necesario ser preciso en algunos aspectos. Este término debe ser negativo cuando se modele la reacción de reducción y positivo cuando se modele la oxidación, para así ser consistente con la naturaleza endotérmica y exotérmica de estos procesos. Tanto para la reducción como para la oxidación, la multiplicación de la entalpía de reacción Δh° por la razón de cambio de concentración de vacancias $\frac{d\delta}{dt}$ tiene signo positivo, es debido a esto que se utiliza un signo « $\pm$ » en la ecuación. De acuerdo a todo lo señalado, para la reducción el signo de « $\pm$ » será negativo mientras que para la oxidación será positivo.

Al igual que la ecuación de energía para la fase gaseosa, en esta ecuación todos sus términos tienen unidades de potencia por unidad de volumen.

Coefficiente intersticial de intercambio de calor a [W/m³K]

De acuerdo con [19] la transferencia de calor entre las fases sólida y gaseosa se evalúa utilizando el coeficiente intersticial de intercambio de calor a , determinado por la ecuación siguiente:

$$a = a_{gs} \cdot h_{gs} \quad (18)$$

El parámetro a_{gs} es la superficie específica interna del medio poroso, mientras que h_{gs} corresponde al coeficiente convectivo interfásico. Estos parámetros se obtienen mediante las siguientes relaciones [19]:

$$a_{gs} = \frac{6 \cdot (1 - \varepsilon)}{d_p} \quad (19)$$

$$h_{gs} = \frac{\lambda_g}{d_p} [2.0 + 1.1 \cdot Pr^{1/3} \cdot Re^{0.6}] \quad (20)$$

Los términos Pr y Re corresponden a los números adimensional de Prandtl y Reynolds respectivamente. El número de Prandtl es proporcional al cociente entre la difusión de momento y la difusividad térmica mientras que el número de Reynolds corresponde al ratio entre las fuerzas inerciales y las fuerzas viscosas de un fluido. Están dados por las siguientes ecuaciones [19]:

$$Pr = \frac{\nu \cdot \rho_g \cdot Cp_g}{\lambda_g}$$

$$Re = \frac{e \cdot u_g \cdot dp}{\nu}$$

Donde u_g es la velocidad del gas, dp la longitud característica de la partícula del medio poroso, ν_g la viscosidad cinemática del gas y λ_g la conductividad del gas.

Coeficiente de conductividad térmica efectivo del medio poroso λ_{ef} [W/mK]

De acuerdo con [19] se puede representar la conducción y radiación a través de un medio poroso en un solo término conductivo utilizando el denominado coeficiente de conductividad térmica efectivo. Este parámetro se calcula de la manera siguiente:

$$\lambda_{ef} = (1 - \varepsilon) \cdot \lambda_s + \frac{32 \cdot \sigma \cdot d_p \cdot \varepsilon \cdot T_s^3}{9 \cdot (1 - \varepsilon)} \quad (21)$$

Como se puede dar cuenta, el segundo término de 21 es aquel que considera la radiación dentro del medio poroso, donde σ corresponde a la constante de Stephan-Boltzmann.

2.2.4. Ecuación de reacción química

Esta última ecuación del modelo representa la variación de vacancias que se van generando y llenando en los procesos de reducción y oxidación de CeO_2 respectivamente. La derivación de esta ecuación proviene del estudio realizado por Bulfin [10] (ec. 20) sin embargo el término relacionado con la oxidación es modificado al considerar como velocidad de reacción de oxidación aquella determinada por Arifin [11] (ec. 3 y 4). Debido a que la velocidad de oxidación de Arifin ya considera en su interior un término que representa la acción del oxidante la ecuación de Bulfin se ajusta eliminando el término relacionado con la presión parcial de oxígeno en el ambiente.

$$\frac{\partial \delta}{\partial t} = (\delta_{max} - \delta) \cdot f_{red}(\alpha) \cdot A_{red} \cdot \exp\left(\frac{-E_{red}}{R \cdot T_s}\right) - \delta \cdot [Y_{ox}(t)]^\gamma \cdot f_{ox}(\alpha) \cdot A_{ox} \cdot \exp\left(\frac{-E_{ox}}{R \cdot T_s}\right) \quad (22)$$

Donde:

- $\delta[-]$: Concentración de vacancias de oxígeno por concentración de cerio $\frac{[O_{vac}]}{[Ce]}$.
- $\delta_{max}[-]$: Valor máximo de δ , que representa el máximo nivel de oxidación de la ceria que se puede obtener sin alterar la estructura metálica. Es un valor constante que se puede encontrar en [10] denotado como x .

- $A_{red}[1/s]$: Factor pre-exponencial o de frecuencia de la expresión de Arrhenius para la reacción de reducción.
- $E_{red}[kJ/mol]$: Energía de activación para la reacción de reducción.
- $[Y_{ox}(t)][-]$: Fracción molar de oxidante $-H_2O-$ en función del tiempo.
- $\gamma[-]$: Orden de la reacción de la fracción molar de oxidante.
- $\alpha[-]$: Extensión normalizada de la reacción. Se define matemáticamente como $\int_0^t \frac{H_2(t)}{H_{2,total}} dt$ o $\int_0^t \frac{O_2(t)}{O_{2,total}} dt$ dependiendo si se trata del proceso de reducción u oxidación.
- $f(\alpha)$: Es una función extraída a partir de la teoría cinética de estado sólido (SS kinetic theory) que describe la tasa de avance del proceso de oxidación. Esta función tiene la forma $f(\alpha) = (1 - \alpha)^n$ tanto para la reducción como para la oxidación, lo que se modifica es el valor del exponente. Para la reducción, Bulfin utiliza un valor $n_{red} = 1/3$, mientras que para la oxidación (Arifin) se usa un valor de $n_{ox} = 1$.
- $A_{ox}[1/s]$: Factor pre-exponencial o de frecuencia de la expresión de Arrhenius para la reacción de oxidación.
- $E_{ox}[kJ/mol]$: Energía de activación para la reacción de oxidación.
- $R [kJ/molK]$: Constante universal de los gases.

Los términos de la ecuación de reacción química tienen unidades de $[1/s]$.

2.2.5. Ecuación de conservación de especies químicas

La ecuación de conservación de especies químicas representa la variación de las concentraciones de las distintas especies químicas que se tienen en el problema estudiado a través del medio poroso.

$$\frac{\partial \bar{w}}{\partial t} + u_g \frac{\partial \bar{w}}{\partial x} = \frac{1}{\rho_g} \frac{\partial}{\partial x} \left(D \cdot \rho_g \frac{\partial \bar{w}}{\partial x} \right) - K \cdot \bar{w} \cdot e^{\frac{-E}{R \cdot T_s}} \quad (23)$$

Donde $\bar{w}$ corresponde a la fracción molar normalizada de la especie química que se desee calcular.

La única especie química cuya fracción molar es un dato de entrada para las ecuaciones de conservación de energía, en particular para el proceso de oxidación exotérmica, es la fracción molar de oxidante Y_{ox} . Si bien la concentración de oxidante va disminuyendo a medida que la reacción se produce (mientras aumenta la concentración de H_2) esta variación en la concentración es podrías despreciables si la cantidad de moles de hidrógeno obtenido es baja respecto a los moles de vapor de agua. De ser así, no sería necesaria la obtención de la concentración de oxidante a través del reactor.

2.2.6. Condiciones de borde

Se tienen condiciones de borde distintas para cada una de las dos etapas del proceso. Algunas condiciones de borde serán parámetros a sensibilizar posteriormente, por lo que no tienen valor único, en esos casos se expresa simplemente como un valor constante. En el Capítulo 5: Resultados y Análisis se mostrarán los valores que se le asignaron a estos parámetros.

2.2.6.1 Reducción(endotérmica)

Condiciones iniciales ($t = 0$):

- $\delta = 0$
- $T_g = T_s = T_0 = 300[\text{K}]$

Condiciones de borde:

- $x = 0$
- $T_g = T_{Ar} = 300[\text{K}]$
- $T_s = T_{borde,r}$
- $u_g = u_{Ar}$

$$\blacksquare \delta = \delta_{max} = 0.35$$

$$x \rightarrow \infty$$

$$\blacksquare \frac{\partial T_g}{\partial x} = \frac{\partial T_s}{\partial x} = \frac{\partial \delta}{\partial x} = 0$$

2.2.6.2 Oxidación(exotérmica)

La concentración de vacancias inicial de la oxidación será la distribución obtenida al final del proceso de reducción.

Para el caso particular de la oxidación, también se sensibilizará la temperatura inicial del medio poroso.

Condiciones iniciales ($t = 0$):

$$\blacksquare \delta = \delta_{final-reducción}$$

$$\blacksquare T_g = 300[\text{K}]$$

$$\blacksquare T_s = T_{inicial}$$

$$\blacksquare Y_{ox} = 0$$

Condiciones de borde:

$$x = 0:$$

$$\blacksquare T_g = 400[\text{K}]$$

$$\blacksquare T_s = T_{borde,o}$$

$$\blacksquare u_g = u_{H_2O}$$

$$\blacksquare \delta = 0$$

$$\blacksquare Y_{ox} = 1[-]$$

$$x \rightarrow \infty$$

$$\blacksquare \frac{\partial T_g}{\partial x} = \frac{\partial T_s}{\partial x} = \frac{\partial \delta}{\partial x} = \frac{\partial Y_{ox}}{\partial x} = 0$$

2.2.7. Consideraciones del modelo

Proceso unidimensional

La primera consideración del modelo -ya mencionada anteriormente- es que el proceso es considerado unidimensional. Esta simplificación es realizada debido a la alta complejidad que conllevaría el plasmar un modelo de dos o más dimensiones que involucre transferencia de calor y masa. En este sentido, mediante el estudio de una sola dimensión física del problema se puede obtener conocimiento sobre el comportamiento del proceso en sí y así tener un acercamiento a lo que sería el funcionamiento del reactor con un bajo costo de cálculo computacional.

Exceso de Ar/H₂O en las etapas

Tanto el argón utilizado en la reducción y el vapor de agua utilizado en la oxidación serán los gases predominantes del volumen de control, el oxígeno e hidrógeno liberados en ambas etapas será considerablemente menor. En este sentido, las fracciones molares de O₂ y H₂ serán cercanas a cero. Debido a esta consideración no se utilizará la ecuación de conservación de especie química para obtener la producción de las especies, sino que se obtendrán mediante estequiometría.

Producción de O₂ y H₂ estequiométrica

Tomando en consideración lo mencionado el exceso de Ar y H₂O, la producción de oxígeno e hidrógeno en los procesos se determinará a través de las ecuaciones estequiométricas (12) y (13). La variable que determinará la producción será entonces la concentración de vacancias δ .

Cinética de reacción

Para la cinética de reacción en particular se tendrán las siguientes consideraciones:

- Cinética de reacción de un solo paso.

- Se estudia principalmente la dependencia de la temperatura en el proceso, por lo tanto la presión no se considerará en la cinética y para fines de cálculo de propiedades se considerará como presión ambiental constante.
- Se considerará que el proceso eventualmente llegará a la reducción/oxidación total del material. Esto definirá $O_{2,total}$ y $H_{2,total}$ que se utilizarán para el cálculo de α .

En base a las consideraciones expuestas hasta ahora, es posible obtener una expresión para α en función de δ :

$$\alpha_{red} = \int_0^t \frac{O_2(t)}{O_{2,total}} dt = \frac{\delta(t)}{\delta_{max}} \quad (24)$$

La cantidad de O_2 total producido será igual al valor máximo de reducción δ_{max} . Por otro lado, la cantidad producida de O_2 hasta el tiempo t está dada por $\delta(t)$.

$$\alpha_{ox} = \int_0^t \frac{H_2(t)}{H_{2,total}} dt = \frac{\delta_{inicial} - \delta(t)}{\delta_{inicial}} \quad (25)$$

Para la oxidación, la cantidad total de H_2 producido será igual al valor de $\delta_{inicial}$, recordemos que se considera como estado final la oxidación/reducción completa. Por otro lado, la producción de hidrógeno hasta t será igual a la diferencia entre $\delta_{inicial}$ y $\delta(t)$.

Según el proceso que se quiera modelar, uno de los dos términos de la ecuación de reacción química puede ser despreciado. En el caso de la reducción, debido a que el único gas que se suministra al volumen de control es argón, se podría decir que la fracción molar de oxidante tiende a ser cero, por lo que para dicho caso, la ecuación de reacción química sería de la siguiente forma:

$$\begin{aligned} \frac{\partial \delta}{\partial t} &= (\delta_{max} - \delta) \cdot k_{red} \\ &= (\delta_{max} - \delta) \cdot f_{red}(\alpha) \cdot A_{red} \cdot \exp\left(\frac{-E_{red}}{R \cdot T_s}\right) \\ &= (\delta_{max} - \delta) \cdot \left(1 - \frac{\delta}{\delta_{max}}\right)^{n_{red}} \cdot A_{red} \cdot \exp\left(\frac{-E_{red}}{R \cdot T_s}\right) \end{aligned} \quad (26)$$

Debido a las energías de activación, el término de la reducción se vuelve cerca de 5 or-

denes de magnitud menor al término de oxidación cuando nos encontramos en el rango de temperaturas de dicha etapa (700-900[°C]), por ende podríamos despreciar el termino correspondiente a la reducción. Esta acción también es realizada por Bulfin en su procedimiento para determinar la cinética de la ceria [10]. La ecuación de reacción para la etapa de oxidación tendría la siguiente forma:

$$\begin{aligned}
 \frac{\partial \delta}{\partial t} &= -\delta \cdot k_{ox} \\
 &= -\delta \cdot [Y_{ox}(t)]^\gamma \cdot f_{ox}(\alpha) \cdot A_{ox} \cdot \exp\left(\frac{-E_{ox}}{R \cdot T_s}\right) \\
 &= -\delta \cdot [Y_{ox}(t)]^\gamma \cdot \left(1 - \frac{\delta_{inicial} - \delta}{\delta_{inicial}}\right)^{n_{ox}} \cdot A_{ox} \cdot \exp\left(\frac{-E_{ox}}{R \cdot T_s}\right)
 \end{aligned} \tag{27}$$

Otra consideración que se realiza para el proceso de oxidación es de considerar la fracción molar de oxidante Y_{ox} como una constante en el tiempo y espacio. Esto ya que de acuerdo a los resultados de Arifin [11] la fracción molar de H_2 es del orden de 10^{-4} , por ende al solo estar inyectando vapor de agua al sistema la fracción molar de H_2O sería prácticamente 1 durante todo el proceso.

Observando las ecuaciones del modelo, podemos dar cuenta que la concentración de vacancias δ varía en el tiempo en función del cambio de temperatura del sólido, en este sentido las vacancias se consideran estáticas.

2.2.7.1 Temperatura inicial del sólido (T_s) para la oxidación

A diferencia de la T_s inicial para la reducción -la cual tiene un valor fijo de temperatura ambiente- el valor de T_s inicial para la oxidación será una variable sensibilizar.

Inicialmente se pensó en considerar como T_s inicial de oxidación el valor de T_s obtenido al final de la reducción, ya que inicialmente las reacciones serían en serie y prácticamente una seguida de la otra.

La temperatura del sólido al final de la reducción puede llegar a ser particularmente alta, si bien esto incrementaría la velocidad de oxidación también representa un peligro ya que la reducción no sería despreciable (se tendría una segunda etapa con reducción+oxidación). Al

tener una reducción (liberación de oxígeno) junto a una oxidación (liberación de hidrógeno) se genera la combustión del hidrógeno, lo que podría ocasionar graves daños tanto para el reactor como para los operadores. Es por esto que se debe evitar que el sólido llegue a temperaturas mayores a 1400[K] en la oxidación.

2.3. Propiedades físicas del medio poroso y del gas

2.3.1. Medio poroso (CeO_2)

Para las propiedades del medio poroso (porosidad y diámetro de partícula) se utilizarán los datos de pellets utilizados por Bulfin en [10]. Por el lado de las propiedades termofísicas del CeO_2 se utilizarán las propiedades de la ceria estequiométrica, las cuales serán en función de la temperatura del sólido.

- Porosidad: $e = 0.65[-]$
- Diámetro de partícula: $dp = 0.01 \text{ [m]}$
- Calor específico: $Cp_s(T_s) = (0.0599 \ln(T_s) + 0.0390) \cdot 10^3 \text{ [J/kgK]}$
- Densidad: $\rho_s(T_s) = (-0.0002T_s + 6.8561) \cdot 10^3 \text{ [kg/m}^3\text{]}$
- Conductividad térmica: $\lambda_s(T_s) = (6.7760 \cdot 10^{-2} + (2.7930 \cdot 10^{-4}))^{-1} \text{ [W/mK]}$

Las ecuaciones para el calor específico y densidad fueron obtenidas a partir de ajustes realizados a los datos reportados por Nelson en [22]. Por otro lado, el coeficiente de conducción de calor también fue obtenido en dicho trabajo, sin embargo en este caso la ecuación de ajuste si es presentada en el texto.

2.3.2. Gas argón

Las propiedades del argón también se determinarán en función de la temperatura de la fase gaseosa.

- Calor específico: $Cp_g(T_g) = (A + B \cdot T_g + C \cdot T_g^2 + D \cdot T_g^3 + E \cdot T_g^{-2}) \cdot 10^3 / 39.948 \text{ [J/kgK]}$

Donde $A = 20.7860$, $B = 2.8259 \cdot 10^{-7}$, $C = 1.4642 \cdot 10^{-7}$, $D = 1.0921 \cdot 10^{-8}$ y $3.6614 \cdot 10^{-8}$.

- Densidad: $\rho_g(T_g) = 1.6030 \cdot 300/T_g \text{ [kg/m}^3\text{]}$
- Conductividad térmica: $\lambda_g(T_g) = (0.0442 \cdot T_g + 4.7827) \cdot 10^{-3} \text{ [W/mK]}$
- Viscosidad dinámica: $\mu(T_g) = (0.0566 \cdot T_g + 6.0645) \cdot 10^{-6} \text{ [Pa}\cdot\text{s]}$

El ajuste para el calor específico se obtuvo desde el Libro Web de Química del NIST [23]. La densidad se obtiene a partir de la ley de los gases ideales. Los ajustes para la conductividad térmica y viscosidad dinámica son obtenidos a partir de los datos publicados por Younglove [24] considerando presión atmosférica.

2.3.3. Vapor de agua

Debido a la poca variación en la temperatura del vapor de agua a lo largo del proceso de reducción se considerarán las propiedades termofísicas como constantes. En este sentido se utilizarán las propiedades del vapor saturado a $100[^\circ\text{C}]$, disponibles en [25]:

- Calor específico: $C_{p_g} = 2.01 \cdot 10^3 \text{ [J/kgK]}$
- Densidad: $\rho_g = 1/1.673 = 0.5977 \text{ [kg/m}^3\text{]}$
- Conductividad térmica: $\lambda_g = 24.8 \cdot 10^{-3} \text{ [W/mK]}$
- Viscosidad dinámica: $\mu = 12 \cdot 10^{-6} \text{ [Pa}\cdot\text{s]}$

Capítulo 3: Desarrollo Numérico

Las ecuaciones presentadas en el capítulo anterior se resolverán numéricamente. Se realizará una discretización en diferencias finitas implícita del modelo matemático (conservación de masa, energía y ecuación de concentración de vacancias) considerando propiedades variables. Se utilizan diferencias adelantadas para las derivadas temporales, diferencias atrasadas para las derivadas espaciales de primer orden y diferencias centradas para las derivadas de segundo orden. Las propiedades son evaluadas en el paso de tiempo n , que correspondería al paso de tiempo “anterior”; en este sentido, los valores a calcular por cada iteración (paso de tiempo “actual”) corresponden al paso de tiempo $n + 1$.

3.1. Algoritmo de solución

De acuerdo a la naturaleza de las ecuaciones se determinó utilizar el siguiente algoritmo de solución:

1. Se calculan las concentraciones de vacancias para cada nodo de acuerdo con (26).
2. Se resuelven las ecuaciones de energía para el sólido (17) y para el gas (16) de manera simultánea .
3. Una vez calculadas las temperaturas se procede a calcular la concentración de vacancias para el paso de tiempo siguiente, repitiendo el ciclo. El proceso de cálculo se realiza hasta que se completa algún criterio de término, el cual puede ser cierto grado de avance en la reacción o cierta cantidad de tiempo de reacción.
4. Ya finalizada la etapa de reducción comienzan los cálculos para la etapa de oxidación. Análogamente a la reducción se calcula en primera instancia la concentración de vacancias con la ecuación (27).
5. Posteriormente se calculan las temperaturas de fase sólida y gaseosa en simultáneo. Los pasos 4 y 5 se repiten hasta completar el criterio de término.

3.2. Discretización del modelo

3.2.1. Ecuación de reacción química

Las ecuaciones de reacción química para las etapas de reducción y oxidación corresponden a ecuaciones diferenciales de primer orden, es por esto que aplicar el esquema implícito a las ecuaciones (26) y (27) se puede obtener mediante operaciones algebraicas elementales el valor de δ actual (δ^{n+1}) a partir de los valores anteriores de δ y temperatura del sólido (δ^n y T_s^n).

Reducción

Discretizando (26) de acuerdo al esquema descrito y factorizando se tiene:

$$\begin{aligned}\frac{\delta^{n+1} - \delta^n}{\Delta t} &= (\delta_{max} - \delta^{n+1}) \cdot k_{red}^n \\ \rightarrow \delta^{n+1} &= \frac{\Delta t \cdot \delta_{max} \cdot k_{red}^n + \delta^n}{1 + \Delta t \cdot k_{red}^n} \\ &= \frac{\delta^n / \Delta t + \delta_{max} \cdot k_{red}^n}{1 / \Delta t + k_{red}^n}\end{aligned}\tag{28}$$

Donde:

$$k_{red}^n = \left(1 - \frac{\delta^n}{\delta_{max}}\right)^{n_{red}} \cdot A_{red} \cdot \exp\left(\frac{-E_{red}}{R \cdot T_s^n}\right)$$

Oxidación

Discretizando (27) de acuerdo al esquema descrito y factorizando se tiene:

$$\begin{aligned}\frac{\delta^{n+1} - \delta^n}{\Delta t} &= -\delta^{n+1} \cdot k_{ox}^n \\ \rightarrow \delta^{n+1} &= \frac{\delta^n}{1 + \Delta t \cdot k_{ox}^n} \\ &= \frac{\delta^n / \Delta t}{1 / \Delta t + k_{ox}^n}\end{aligned}\tag{29}$$

Donde:

$$k_{ox}^n = [Y_{ox}^n]^\gamma \cdot \left(1 - \frac{\delta_{inicial} - \delta^n}{\delta_{inicial}}\right)^{n_{ox}} \cdot A_{ox} \cdot \exp\left(\frac{-E_{ox}}{R \cdot T_s^n}\right)$$

3.2.2. Ecuación de conservación de energía de la fase sólida.

Al discretizar la ecuación (17) mediante esquema implícito y propiedades variables se tiene:

$$\begin{aligned} (1 - \varepsilon)(\rho \cdot C_p)_{s,i}^n \frac{T_{s,i}^{n+1} - T_{s,i}^n}{\Delta t} = \frac{1}{2 \cdot \Delta x^2} \left[(\lambda_{ef,i+1}^n + \lambda_{ef,i}^n)(T_{s,i+1}^{n+1} - T_{s,i}^{n+1}) - \right. \\ \left. - (\lambda_{ef,i}^n + \lambda_{ef,i-1}^n)(T_{s,i}^{n+1} - T_{s,i-1}^{n+1}) \right] + a_i^n (T_{g,i}^n - T_{s,i}^{n+1}) + Q_{ce,i} \pm \frac{\delta_i^{n+1} - \delta_i^n}{\Delta t} [Ce_i] \cdot \Delta h^{on} \end{aligned} \quad (30)$$

Factorizando de acuerdo a las temperaturas en el tiempo $n + 1$ se tiene:

$$\begin{aligned} T_{s,i-1}^{n+1} \left[\frac{\lambda_{ef,i}^n + \lambda_{ef,i-1}^n}{2 \cdot \Delta x^2} \right] - T_{s,i}^{n+1} \left[\frac{(1 - \varepsilon)(\rho \cdot C_p)_{s,i}^n}{\Delta t} + a_i^n + \frac{1}{2 \Delta x^2} [\lambda_{ef,i-1}^n + 2 \cdot \lambda_{ef,i}^n + \lambda_{ef,i+1}^n] \right] + \\ + T_{s,i+1}^{n+1} \left[\frac{\lambda_{ef,i+1}^n + \lambda_{ef,i}^n}{2 \cdot \Delta x^2} \right] = - \left[\frac{T_{s,i}^n}{\Delta t} (1 - \varepsilon)(\rho \cdot C_p)_{s,i}^n + a_i^n \cdot T_{g,i}^n + Q_{ce,i} \pm \frac{\delta_i^{n+1} - \delta_i^n}{\Delta t} [Ce_i] \Delta h^{on} \right] \end{aligned} \quad (31)$$

Agrupando los términos en una sola variable se tiene:

$$\begin{aligned} A_i &= \frac{\lambda_{ef,i}^n + \lambda_{ef,i-1}^n}{2 \cdot \Delta x^2} \\ B_i &= \frac{\lambda_{ef,i+1}^n + \lambda_{ef,i}^n}{2 \cdot \Delta x^2} \\ C_i &= \frac{(1 - \varepsilon)(\rho \cdot C_p)_{s,i}^n}{\Delta t} + a_i^n + A_i + B_i \end{aligned}$$

$$F_i = \frac{T_{s,i}^n}{\Delta t} (1 - \varepsilon)(\rho \cdot C_p)_{s,i}^n + a_i^n \cdot T_{g,i}^n + Q_{ce,i} \pm \frac{\delta_i^{n+1} - \delta_i^n}{\Delta t} [Ce_i] \Delta h^{on}$$

De esta manera queda conformado el siguiente sistema de ecuaciones lineales:

$$T_{s,i-1}^{n+1} \cdot A_i - T_{s,i}^{n+1} \cdot C_i + T_{s,i+1}^{n+1} \cdot B_i = -F_i \quad (32)$$

Donde $i = 2, \dots, k - 1$; corresponden a los subíndices de las posiciones nodales para un determinado instante de tiempo, siendo k el último nodo de la malla. El sistema de ecuaciones resuelve las temperaturas para todos los nodos excepto aquellos de los extremos, para los cuales se obtienen los valores de acuerdo a las condiciones de borde.

La solución del sistema de ecuaciones se realiza a través del algoritmo para matrices tri-diagonales (TDMA: Tri-Diagonal Matrix Algorithm), también denominado Algoritmo de Thomas. Este procedimiento establece como hipótesis que existe una relación lineal entre dos nodos vecinos cualesquiera de la malla en un instante de tiempo determinado, siendo ésta hipótesis representada mediante la siguiente ecuación lineal [19]:

$$T_{s,i-1}^{n+1} = \alpha_i \cdot T_{s,i}^{n+1} + \beta_i \quad (33)$$

La ecuación (33) permite expresar el sistema de ecuaciones lineales (32) en términos de las temperaturas en los nodos i e $i + 1$:

$$A_i \cdot (\alpha_i \cdot T_{s,i}^{n+1} + \beta_i) - C_i \cdot T_{s,i}^{n+1} + B_i \cdot T_{s,i+1}^{n+1} = -F_i \quad (34)$$

Despejando para la temperatura en el nodo i se tiene:

$$T_{s,i}^{n+1} = \frac{B_i}{C_i - A_i \cdot \alpha_i} T_{s,i+1}^{n+1} + \frac{F_i + A_i \cdot \beta_i}{C_i - A_i \cdot \alpha_i} \quad (35)$$

A partir de la hipótesis planteada como la ecuación (33) también se tiene:

$$T_{s,i}^{n+1} = \alpha_{i+1} \cdot T_{s,i+1}^{n+1} + \beta_{i+1} \quad (36)$$

Reemplazando (36) en (35) se tiene:

$$\alpha_{i+1} = \frac{B_i}{C_i - A_i \cdot \alpha_i} \quad \beta_{i+1} = \frac{F_i + A_i \cdot \beta_i}{C_i - A_i \cdot \alpha_i} \quad (37)$$

Como condición de borde del primer nodo se tiene que :

$$T_{s,1} = T_{borde} \quad (38)$$

Donde T_{borde} corresponde a un valor a sensibilizar tanto para la reducción como para la oxidación, como se explicó con anterioridad.

Según la hipótesis (36) aplicada a dicho nodo se tiene que:

$$T_{s,1} = \alpha_2 \cdot T_{s,2} + \beta_2 \quad (39)$$

Despejando se pueden obtener los valores de los coeficientes para el primer nodo:

$$\begin{aligned} \alpha_2 &= 0 \\ \beta_2 &= T_{borde} \end{aligned} \quad (40)$$

La condición de borde del último nodo indica que la propiedad se ha desarrollado, es decir que llega a un estado estacionario en el que no varía, por lo que el valor en el último nodo de -por ejemplo- la temperatura, será igual al valor de dicha propiedad en el penúltimo nodo.

Para la temperatura en el sólido se tiene:

$$\begin{aligned} \frac{\partial T_s}{\partial x} = 0 &\rightarrow \frac{T_{s,k} - T_{s,k-1}}{\Delta x} = 0 \\ &\rightarrow T_{s,k} = T_{s,k-1} \end{aligned} \quad (41)$$

Reemplazando en la hipótesis se tiene:

$$\begin{aligned} T_{s,k}^{n+1} &= \alpha_k \cdot T_{s,k}^{n+1} + \beta_k \\ &\rightarrow T_{s,k}^{n+1} = \frac{\beta_k}{1 - \alpha_k} \end{aligned} \quad (42)$$

De esta expresión se desprende que necesariamente $\alpha_k \neq 1$. Esto se verifica y demuestra al considerar la forma de los coeficientes y que de acuerdo a la condición del método $C > A+B$ se tiene que $\alpha < 1$.

Mediante (37) y (40) es posible obtener los parámetros α y β para valores nodales desde $i = 2$ hasta $i = k$, haciendo un barrido ascendente a través de la malla. Una vez conocidos estos parámetros es posible obtener mediante (42) y (36) los valores nodales para las temperaturas del sólido desde los nodos $i = k$ hasta $i = 1$, recorriendo la malla de manera descendente.

3.2.3. Ecuación de conservación de energía de la fase gaseosa.

El proceso de resolución de para la ecuación de energía de la fase gaseosa es análogo al descrito para su par de la fase sólida. Discretizando (16) se tiene:

$$\varepsilon \cdot (\rho \cdot C_p)_{g,i}^n \left(\frac{T_{g,i}^{n+1} - T_{g,i}^n}{\Delta t} + u_{g,i}^n \frac{T_{g,i}^{n+1} - T_{g,i-1}^{n+1}}{\Delta x} \right) = \frac{1}{2 \cdot \Delta x^2} \left[(\lambda_{g,i+1}^n + \lambda_{g,i}^n)(T_{g,i+1}^{n+1} - T_{g,i}^{n+1}) - (\lambda_{g,i}^n + \lambda_{g,i-1}^n)(T_{g,i}^{n+1} - T_{g,i-1}^{n+1}) \right] + \frac{a_i^n}{\varepsilon} (T_{s,i}^n - T_{g,i}^{n+1}) \quad (43)$$

Factorizando para las temperaturas de fase gaseosa en $n + 1$ se tiene:

$$T_{g,i-1}^{n+1} \left[\frac{u_{g,i}^n}{\Delta x} (\rho \cdot C_p)_{g,i}^n + \frac{\lambda_{g,i-1}^n + \lambda_{g,i}^n}{2 \cdot \Delta x^2} \right] - T_{g,i}^{n+1} \left[\frac{(\rho \cdot C_p)_{g,i}^n}{\Delta t} + \frac{u_{g,i}^n}{\Delta x} (\rho \cdot C_p)_{g,i}^n + \frac{1}{2 \Delta x^2} [\lambda_{g,i-1}^n + 2 \cdot \lambda_{g,i}^n + \lambda_{g,i+1}^n] + \frac{a_i^n}{\varepsilon} \right] + T_{g,i+1}^{n+1} \left[\frac{\lambda_{g,i+1}^n + \lambda_{g,i}^n}{2 \cdot \Delta x^2} \right] = - \left[\frac{T_{g,i}^n}{\Delta t} (\rho \cdot C_p)_{g,i}^n + \frac{a_i^n}{\varepsilon} \cdot T_{s,i}^n \right] \quad (44)$$

Agrupando los términos en una sola variable se tiene:

$$A_i = \frac{u_{g,i}^n}{\Delta x} (\rho \cdot C_p)_{g,i}^n + \frac{\lambda_{g,i-1}^n + \lambda_{g,i}^n}{2 \cdot \Delta x^2}$$

$$B_i = \frac{\lambda_{g,i+1}^n + \lambda_{g,i}^n}{2 \cdot \Delta x^2}$$

$$C_i = \frac{(\rho \cdot C_p)_{g,i}^n}{\Delta t} + \frac{a_i^n}{\varepsilon} + A_i + B_i$$

$$F_i = \frac{T_{g,i}^n}{\Delta t} (\rho \cdot C_p)_{g,i}^n + \frac{a_i^n}{\varepsilon} \cdot T_{s,i}^n$$

Lo que forma el siguiente sistema de ecuaciones lineales para las temperaturas nodales del gas:

$$T_{g,i-1}^{n+1} \cdot A_i - T_{g,i}^{n+1} \cdot C_i + T_{g,i+1}^{n+1} \cdot B_i = -F_i \quad (45)$$

Este sistema se resuelve de manera análoga al sistema obtenido a partir de la discretización de la ecuación de energía de la fase sólida (ecuación (32)).

Es importante agregar al algoritmo las ecuaciones de conservación de masa y la ley de gas perfecto para el cálculo de densidad y velocidad del gas:

$$\rho_i^{n+1} = \frac{\rho_0 \cdot T_0}{T_{g,i}^{n+1}} \quad u_i^{n+1} = \frac{\rho_0 \cdot u_0}{\rho_i^{n+1}} \quad (46)$$

3.2.4. Determinación del tamaño de malla y paso de tiempo óptimo

Para determinar el valor de Δt óptimo se probó el algoritmo en Jupyter, un entorno de desarrollo para programar en Python. Se obtienen los perfiles de temperatura de la fase sólida y concentración de vacancias para la reducción y oxidación una vez transcurridos 2.5, 5, 7.5 y 10 minutos de reacción y probando 4 distintos pasos de tiempo $\Delta t = 2, 0.2, 0.02$ y 0.002 segundos.

Para la reducción se tienen los siguientes perfiles:

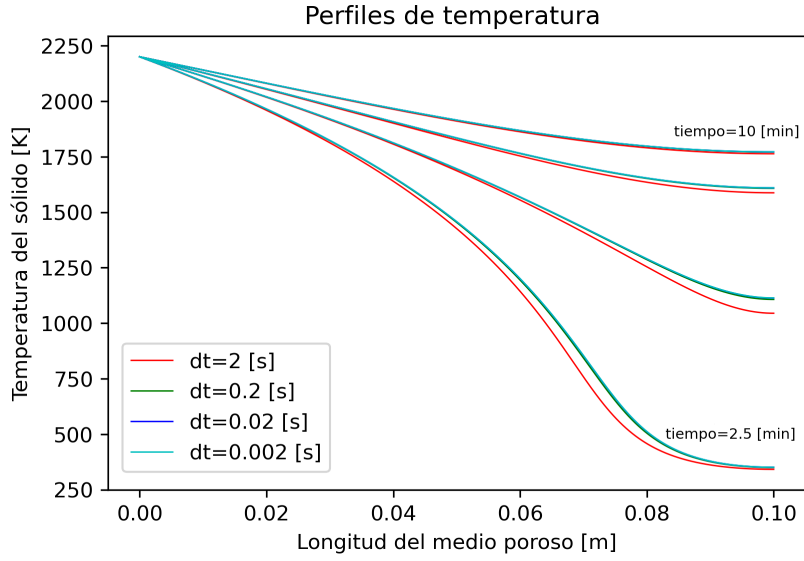


Figura 3.1: Perfiles de temperatura T_s después de 10 minutos de reducción con diferentes valores de Δt .

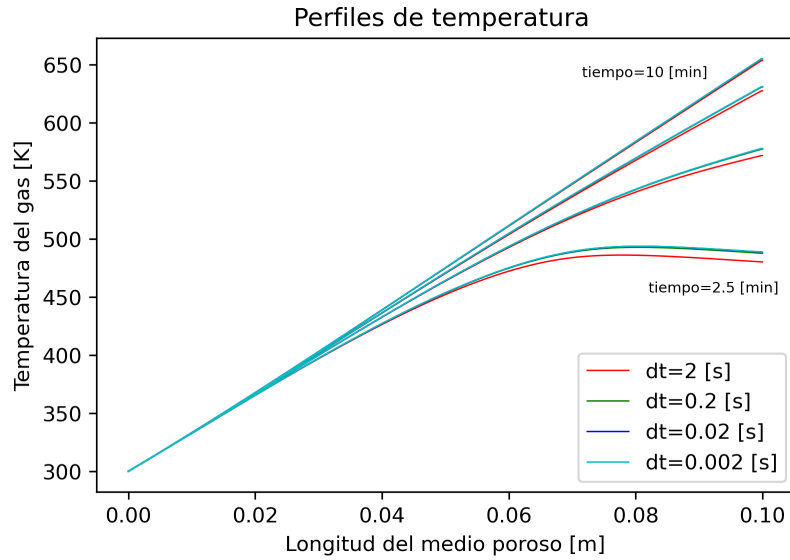


Figura 3.2: Perfiles de temperatura T_g después de 10 minutos de reducción con diferentes valores de Δt .

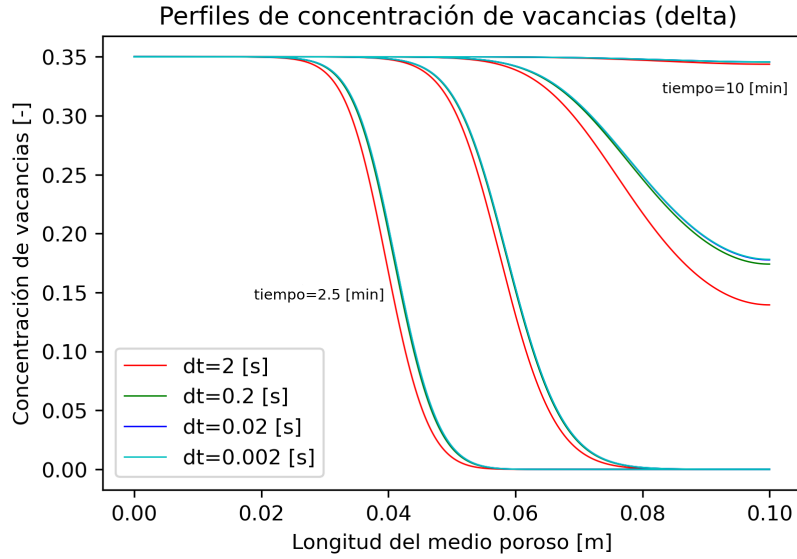


Figura 3.3: Perfiles de concentración de vacancias δ después de 10 minutos de reducción con diferentes valores de Δt .

Para esta simulación se utilizaron los parámetros: $u_g = 1$ [m/s], $T_{borde,r} = 2200$ [K], $Q_{ce} = 10^6$ [W/m³] y $n_x = 800$ [nodos].

Por el lado de la oxidación, se tienen los siguientes perfiles:

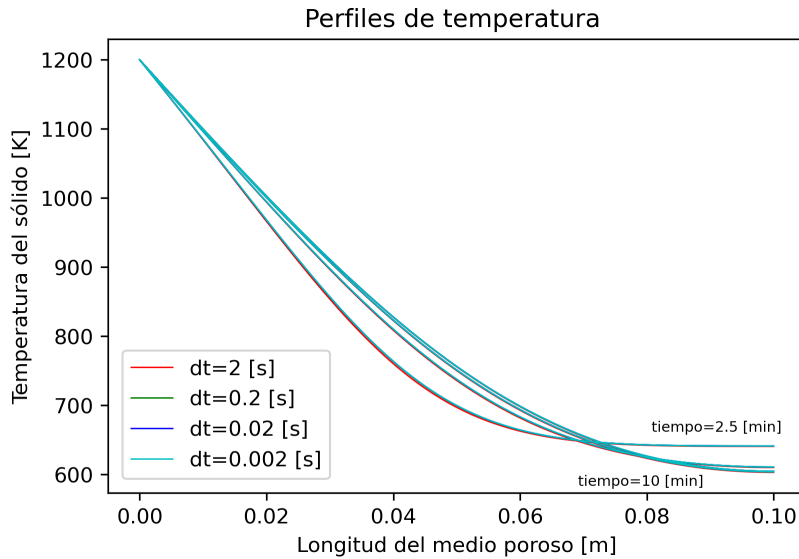


Figura 3.4: Perfiles de temperatura T_s después de 10 minutos de oxidación con diferentes valores de Δt .

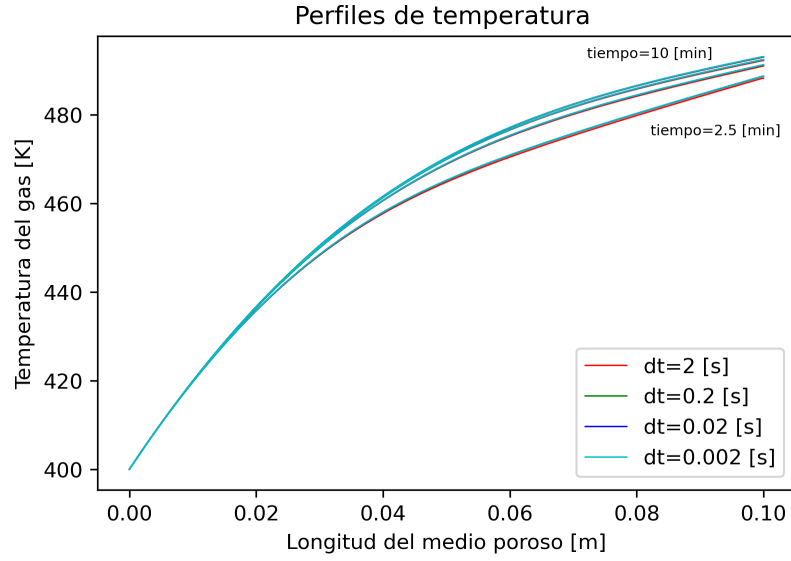


Figura 3.5: Perfiles de temperatura T_g después de 10 minutos de oxidación con diferentes valores de Δt .

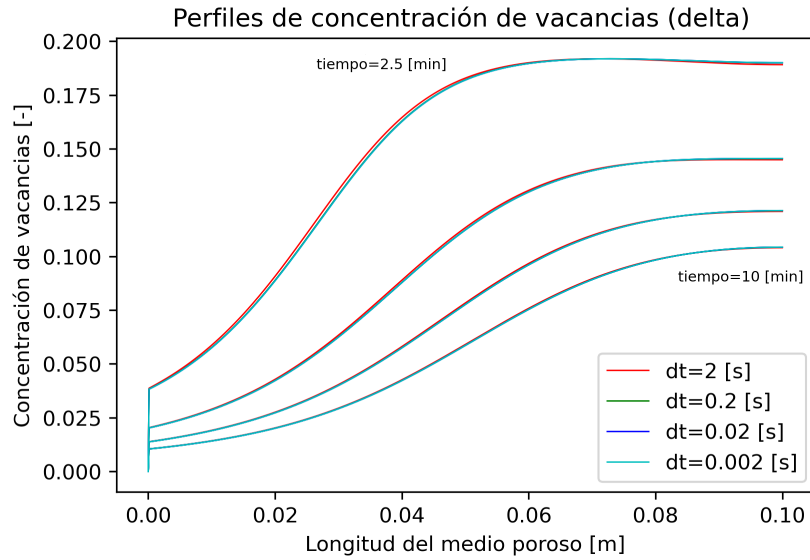


Figura 3.6: Perfiles de concentración de vacancias δ después de 10 minutos de oxidación con diferentes valores de Δt .

Para esta simulación se utilizaron los parámetros: $u_g = 1$ [m/s], $T_{borde,o} = 1200$ [K], $T_{inicial} = 700$ [K] $Q_{ce} = 10^6$ [W/m³] y $n_x = 800$ [nodos].

Se puede observar una superposición en las curvas con los pasos de tiempo $\Delta t = 0.02$ y $\Delta t = 0.002$ en todas las situaciones. En este sentido se decide utilizar un paso de tiempo de $\Delta t = 0.02$ para las simulaciones a llevar a cabo.

Para determinar el tamaño de malla óptimo se realiza un procedimiento análogo a la determinación del paso de tiempo. En este caso, se considerarán tamaños de malla de $n_x = 200, 400, 600, 800$ y 1000 nodos junto con un tiempo de reacción de 5 minutos y un paso de tiempo de $\Delta t = 0.02$ [s]. El resto de los parámetros utilizados para la determinación de tamaño de malla son iguales a aquellos utilizados para determinar el paso de tiempo.

Para la reducción se tienen los siguientes perfiles:

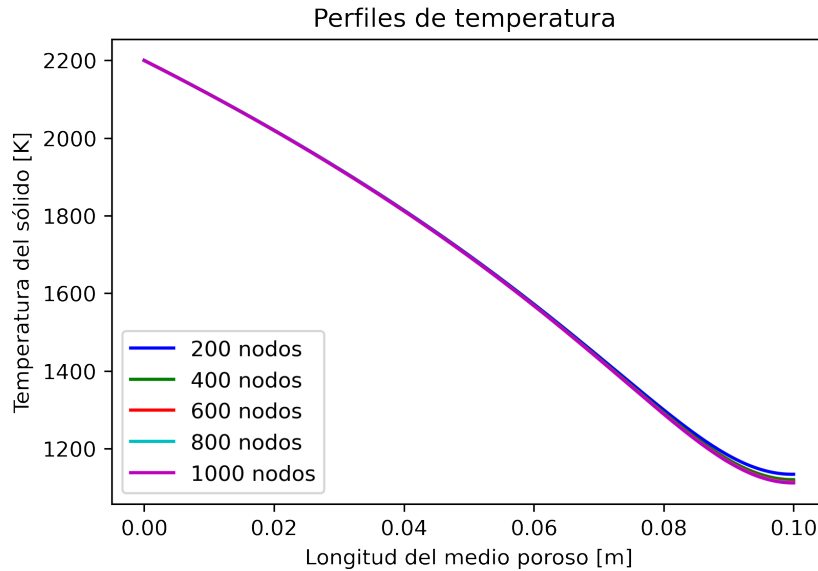


Figura 3.7: Perfiles de temperatura T_s después de 5 minutos de reducción con diferentes valores de n_x .

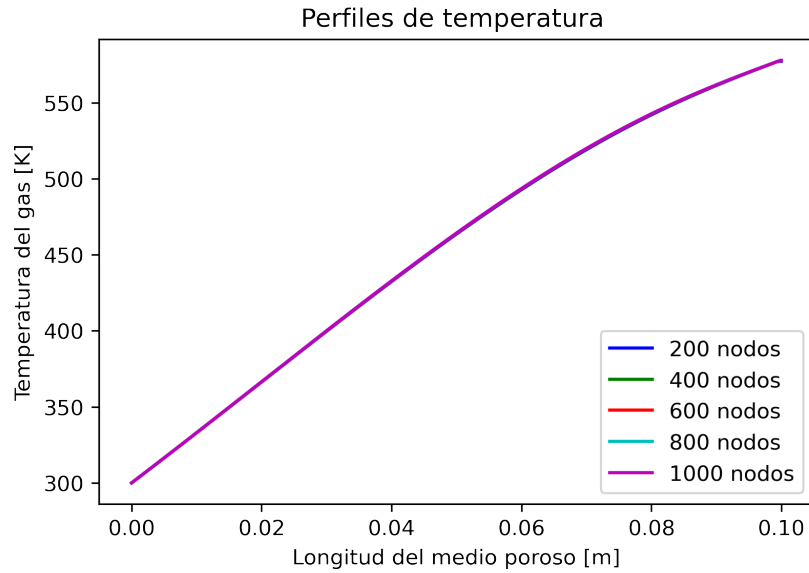


Figura 3.8: Perfiles de temperatura T_g después de 5 minutos de reducción con diferentes valores de n_x .

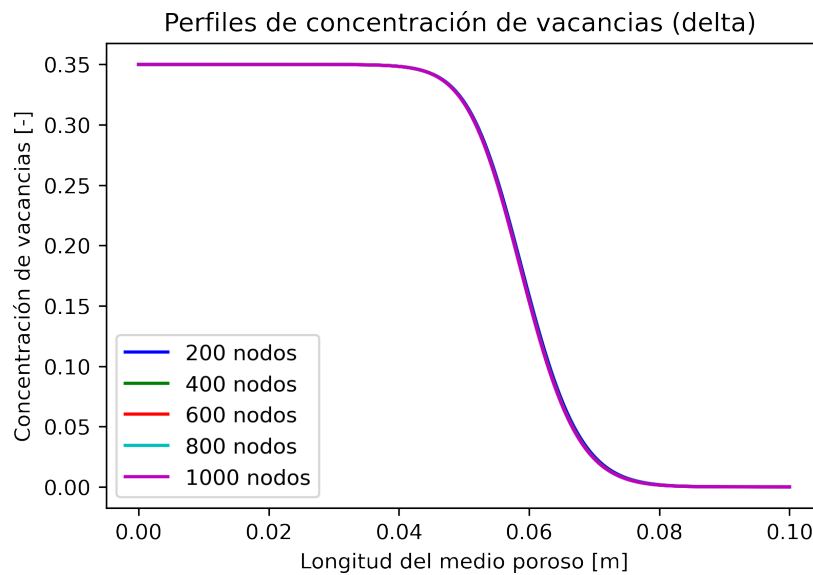


Figura 3.9: Perfiles de concentración de vacancias δ después de 5 minutos de reducción con diferentes valores de n_x .

Para la oxidación se tienen los siguientes perfiles:

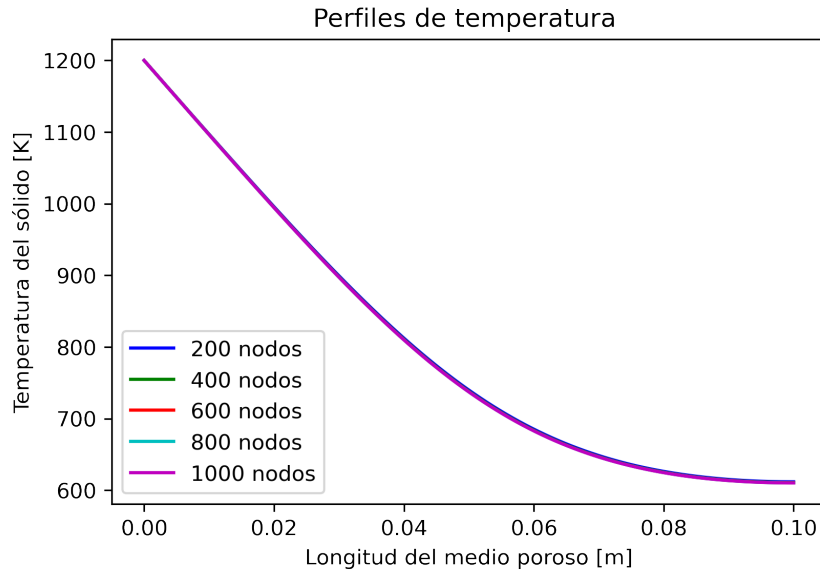


Figura 3.10: Perfiles de temperatura T_s después de 5 minutos de oxidación con diferentes valores de n_x .

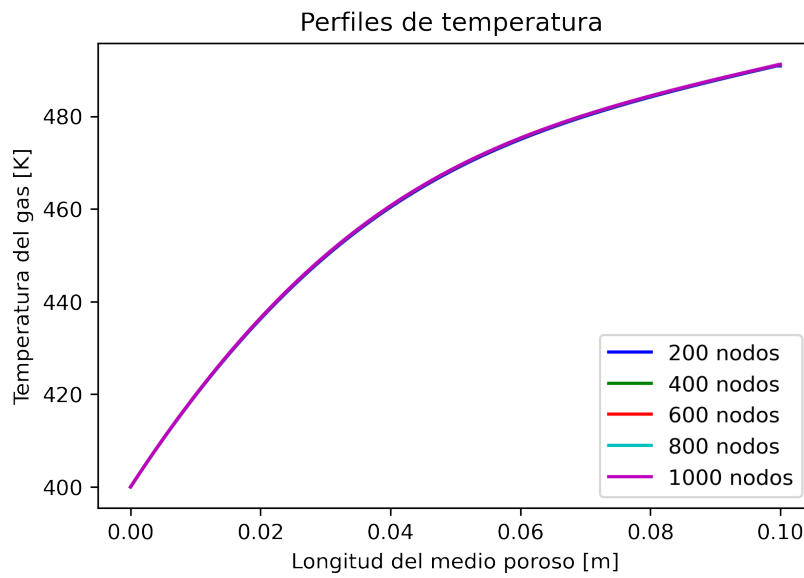


Figura 3.11: Perfiles de temperatura T_g después de 5 minutos de oxidación con diferentes valores de n_x .

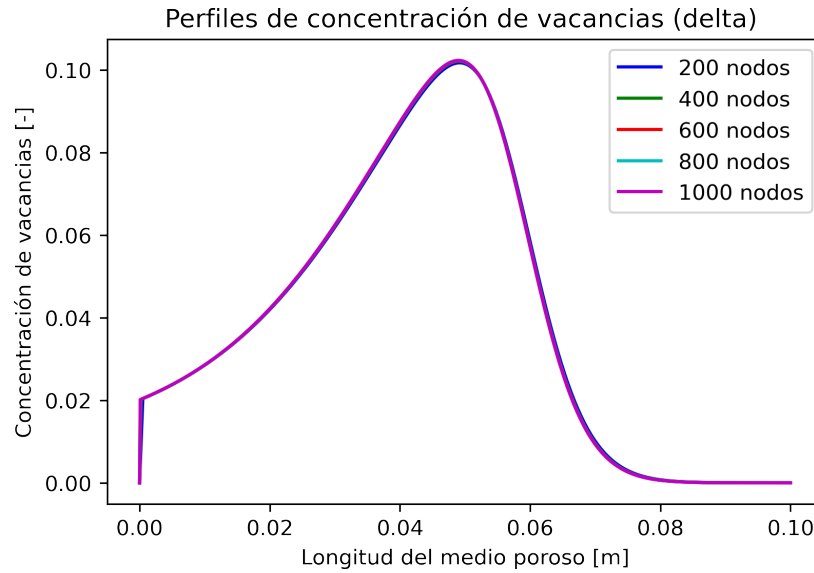


Figura 3.12: Perfiles de concentración de vacancias δ después de 5 minutos de oxidación con diferentes valores de n_x .

Observando las distintas gráficas se puede observar una superposición de los perfiles para distintos valores nodales en todo el dominio de estudio. En este sentido, se decide utilizar un valor de $n_x = 800$ [nodos] para las simulaciones a realizar.

Capítulo 4: Modelo de comparación cinética

4.1. Introducción

Los modelos computacionales pueden ser herramientas muy útiles a la hora de estudiar los fenómenos físicos. Permiten realizar modificaciones a los parámetros operacionales y predecir el comportamiento del objeto de estudio sin necesidad de hacer experimentos en laboratorio, lo que se traduce en un ahorro tanto de tiempo como de recursos diversos. Sin embargo, el desarrollo de modelos no está ajeno a la experimentación, de hecho todo modelo computacional debe ser comparado y validado mediante resultados experimentales.

Debido a la imposibilidad de realizar experimentos en el reactor de Water-Splitting para validar el modelo desarrollado en los capítulos anteriores surgió como opción el comparar la componente cinética del modelo con resultados experimentales disponibles en la literatura, en específico aquellos reportados por Arifin [11]. En este sentido, a partir de la situación física del experimento desarrollado por Arifin se desarrollará un modelo computacional para así comparar los resultados y determinar que tan cercanos a la realidad resulta el modelo cinético.

4.2. Situación Física

La muestra utilizada en los experimentos consiste en óxido de cerio comercial en polvo. La reacción se lleva a cabo en un reactor idealizado de flujo de estancamiento (SFR: Stagnation Flow Reactor).

El ciclo *redox* consistió en mantener la temperatura de la cámara de reacción a la temperatura de oxidación T_{ox} mientras se suministra helio de alta pureza sobre la muestra, con un caudal de 500 [cm³/s] estándar.

Durante la reducción térmica, el polvo de ceria fue calentado desde T_{ox} hasta 1500[°C] a

una tasa de $17[^\circ\text{C}]$ por segundo. Se utilizó un láser de espectro infrarojo cercano de $500[\text{W}]$ para calentar localmente la muestra, esto mientras la temperatura de la cámara de reacción se mantuvo a T_{ox} . El material se mantuvo a $1500[^\circ\text{C}]$ por 240 segundos.

Por otro lado, la oxidación se realizó de manera isotérmica a T_{ox} agregando H_2O al flujo de He durante 600 segundos. El flujo total se mantuvo en $500 [\text{cm}^3/\text{s}]$. Se realizaron experimentos en un rango de temperaturas de entre $750[^\circ\text{C}]$ y $950[^\circ\text{C}]$, con fracciones volumétricas de H_2O de entre 20 % y 40 %.

4.3. Ecuaciones gobernantes

Para el desarrollo del modelo de comparación se definirá un perfil de temperatura de sólido de acuerdo a la descripción que hace Arifin sobre la variación de la temperatura, de acuerdo con esto solamente se resolverán las ecuaciones de reacción química para la reducción y oxidación.

El perfil de temperatura del sólido para la reducción se puede expresar mediante la siguiente función:

$$Ts(t) = \begin{cases} T_{ox} + t \cdot 17 & ; \quad t < \frac{1500 - T_{ox}}{17} \\ 1500 & ; \quad \frac{1500 - T_{ox}}{17} < t < \frac{1500 - T_{ox}}{17} + 240 \end{cases} \quad (47)$$

Donde el tiempo está expresado en segundo y las temperaturas en grados celsius. La temperatura del sólido aumenta en una tasa de $17[^\circ\text{C}]$ por segundo hasta llegar a los $1500[^\circ\text{C}]$. La temperatura se mantiene en ese nivel durante 240 segundos.

Para la oxidación, el perfil de temperaturas se define como:

$$Ts(t) = T_{ox} \quad ; \quad t < 600 \quad (48)$$

Para la reducción se utilizará la ecuación (26) con modificaciones:

$$\frac{\partial \delta}{\partial t} = (\delta_{max} - \delta) \cdot \left(1 - \frac{\delta}{\delta_{final}}\right)^{n_{red}} \cdot A_{red} \cdot \exp\left(\frac{-E_{red}}{R \cdot T_s}\right) \quad (49)$$

Mientras que para la oxidación se utilizará (27) con modificaciones:

$$\frac{\partial \delta}{\partial t} = -\delta \cdot [Y_{ox}(t)]^\gamma \cdot \left(1 - \frac{\delta_{inicial} - \delta}{\delta_{inicial} - \delta_{final}}\right)^{n_{ox}} \cdot A_{ox} \cdot \exp\left(\frac{-E_{ox}}{R \cdot T_s}\right) \quad (50)$$

Es importante señalar que de utilizar la temperatura en grados celsius es necesario o utilizar la constante universal de los gases R en las unidades respectivas o hacer la conversión de unidades de temperatura correspondiente.

4.4. Consideraciones

El modelo no contempla la mecánica de fluidos involucrada en el problema, esto debido a que -como la ceria utilizada esta en forma de un polvo- se estima una baja penetración del gas de arrastre helio en la muestra. En este sentido se considerará el reactante como un sistema concentrado, es decir, la temperatura variará uniformemente en el material de acuerdo a la función descrita con anterioridad.

Por otro lado, se tienen prácticamente las mismas consideraciones que modelo del reactor de Water-Splitting unidimensional respecto a la cinética de reacción y a la producción estequiométrica de O_2 y H_2 .

A diferencia del modelo unidimensional, se considera que los procesos llegan a un estado de equilibrio diferente a la reducción/oxidación completa del material. En este sentido, el valor de α en (24) y (25) se recalculará cada vez que se realice la simulación del proceso (en estricto rigor, es δ_{final} en (49) y (50) el valor que se va recalculando). Esta consideración se realiza ya que debido a la baja penetración del gas de arrastre es probable que exista mayor re-oxidación del material, cambiando el punto de equilibrio de la reacción.

4.5. Algoritmo de solución

El algoritmo es similar al utilizado para el reactor unidimensional, con la salvedad de que en este caso se realiza una iteración del valor α debido a lo señalado en el apartado de consideraciones.

1. Se establece un valor de concentración de vacancias final δ_{final} para la reducción como hipótesis. Este valor calculará α con la ecuación (24).
2. A partir de (47) se obtiene la temperatura del sólido T_s para el primer paso de tiempo.
3. Se calcula la concentración de vacancias δ con la ecuación (49).
4. Se vuelven a repetir los pasos 2. y 3. hasta que se completa el tiempo de reacción determinado.
5. Se compara δ_{final} calculado con la hipótesis inicial. Si la diferencia entre los valores es menor a 10^{-5} se toma este valor de δ obtenido como la nueva hipótesis y se repite el algoritmo desde el paso 1. Si se cumple el criterio se toma el valor de δ calculado como el valor final.

El algoritmo expuesto corresponde al proceso de reducción, sin embargo para la oxidación el procedimiento es análogo, considerando las ecuaciones (48), (50) y (25) para los cálculos de T_s , δ y α respectivamente.

Capítulo 5: Resultados y Análisis

A continuación se presentan los resultados obtenidos y su respectivo análisis para tanto en el modelo de comparación como en el modelo unidimensional del reactor de Water-Splitting. Se presentan además los parámetros utilizados en las simulaciones

5.1. Modelo de comparación

5.1.1. Parámetros

Los parámetros que se mostrarán en las tablas siguientes corresponden a aquellos utilizados para resolver las ecuaciones de concentración de vacancias (49) y (50). Recordemos que que el comportamiento de la temperatura se define por las ecuaciones (47) y (48).

δ_{max} [-]	A_{red} [1/s]	E_{red} [kJ/mol]	Y_{ox} [-]	n_{red} [-]
0.35	720000	232	0	1/3

Cuadro 5.1: Tabla de parámetros utilizados en la etapa de reducción.

A_{ox} [1/s]	E_{ox} [kJ/mol]	Y_{ox} [-]	γ [-]	n_{ox} [-]
1	29	0.3	0.89	1

Cuadro 5.2: Tabla de parámetros utilizados en la etapa de oxidación.

Δt [s]	R [kJ/molK]
0.02	8.3145

Cuadro 5.3: Tabla de parámetros utilizados en ambas etapas.

5.1.2. Resultados

5.1.2.1 Cinética Bulfin & Arifin

Los valores de δ_{final} utilizados para calcular $f(\alpha)$ fueron obtenidos mediante un proceso iterativo y sus valores se muestran a continuación para cada etapa y cada caso definido por la temperatura de oxidación:

Temperatura de oxidación T_{ox} [°C]	δ_{final} [-] Reducción	δ_{final} [-] Oxidación
750	0.34933276	0.19972006
800	0.34933308	0.18600724
850	0.34933334	0.17338739
900	0.34933371	0.16185165
950	0.34933402	0.1513547
1000	0.3493343	0.14182989
1100	0.34933472	0.12538496

Cuadro 5.4: Concentraciones de vacancias final (para cálculo de $f(\alpha)$) obtenida mediante proceso iterativo.

A partir del valor de δ obtenidos al final de las simulaciones y de la ecuación estequiométrica se obtuvieron la producción de oxígeno e hidrógeno. En las gráficas siguientes se muestran los valores obtenidos junto con la producción de oxígeno e hidrógeno que reporta Arifin:

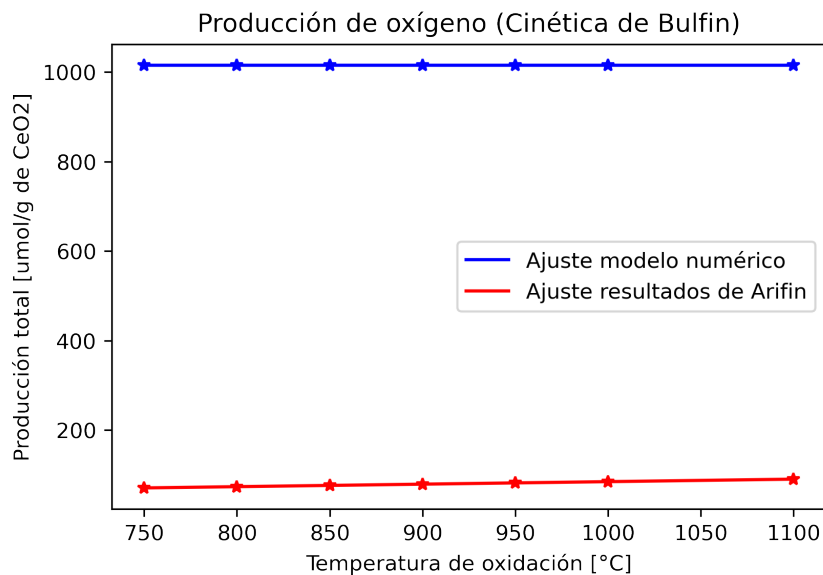


Figura 5.1: Producción de oxígeno para el modelo de comparación considerando las cinéticas de Bulfin y Arifin para la reducción y oxidación respectivamente.

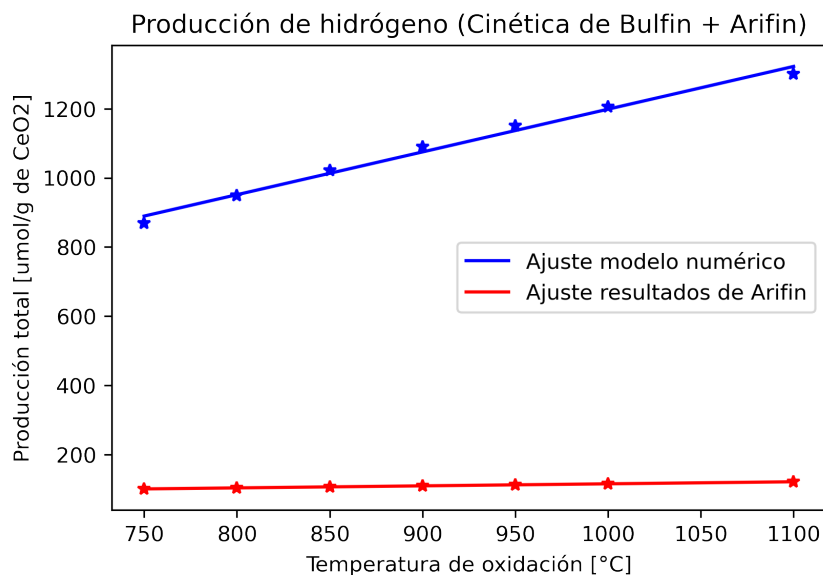


Figura 5.2: Producción de hidrógeno para el modelo de comparación considerando las cinéticas de Bulfin y Arifin para la reducción y oxidación respectivamente.

Podemos observar una diferencia de casi 3 órdenes de magnitud entre la producción de O_2 y H_2 reportada por Arifin y la calculada por el modelo computacional. Teniendo en consi-

deración que cantidad de material oxidado es directamente proporcional al nivel de reducción alcanzado en la primera etapa, es de esperar que la considerable diferencia en la producción de oxígeno también se refleja en la producción de hidrógeno. En este sentido, de tener una reducción más acotada sería posible obtener resultados del modelo que se ajusten de mejor manera al experimento.

5.1.2.2 Cinética Arifin

Tomando en consideración lo señalado anteriormente, se decidió calcular la producción de hidrógeno realizando la simulación solo del proceso de oxidación, para la cual las condiciones iniciales de concentración de vacancias $\delta(0)$ aquella con la cual se obtendría una producción de oxígeno igual a la obtenida por Arifin. A partir de la producción de oxígeno del experimento se realiza el cálculo estequiométrico de manera inversa, en donde se obtienen los valores de δ , correspondientes al δ final de reducción e inicial de oxidación. Estos valores son mostrados en la tabla 5.5 junto con los valores δ_{final} para la oxidación a ser utilizados como hipótesis para el cálculo de $f(\alpha)$.

Temperatura de oxidación T_{ox} [°C]	δ_{final} [-] Reducción	δ_{final} [-] Oxidación
750	0.02443689	0.01192768
800	0.02540073	0.01139412
850	0.02636458	0.01089149
900	0.02732842	0.01042392
950	0.02829226	0.00999278
1000	0.02925611	0.00959759
1100	0.0311838	0.00890827

Cuadro 5.5: Concentración de vacancias final. Para la reducción, el valor de δ es equivalente a la producción de O_2 reportada por Arifin.

A partir de los valores de δ mostrados en la tabla anterior se realiza la simulación del proceso de oxidación. Los resultados de producción de H_2 se muestran en la figura siguiente:

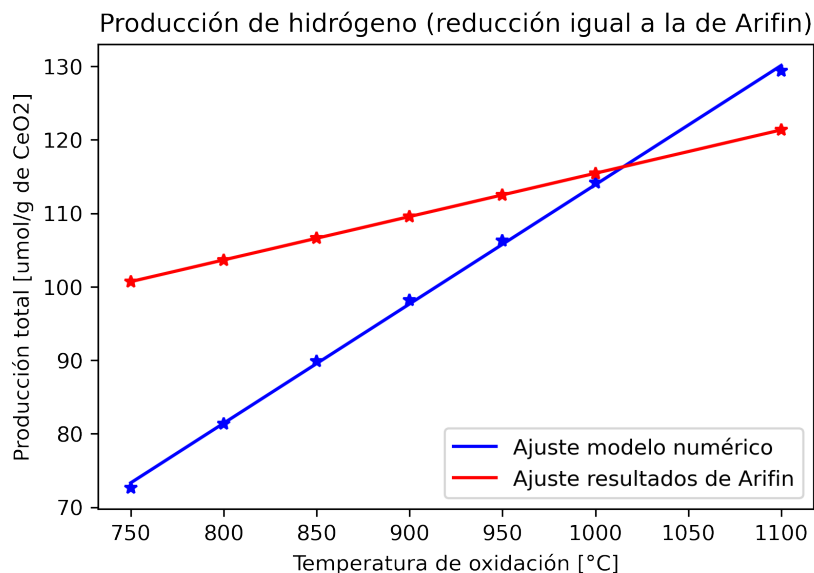


Figura 5.3: Producción de hidrógeno para el modelo de comparación utilizando resultados experimentales de reducción como condición inicial de oxidación.

A diferencia de lo que se puede observar en la figura 5.2, la producción de hidrógeno se acerca considerablemente a los resultados experimentales al cambiar la condición inicial de δ de oxidación. Esto confirma la hipótesis de que la alta producción de hidrógeno presentada en la figura 5.2 es consecuencia de la alta producción de oxígeno y por ende alto grado de reducción logrado en la primera etapa del proceso.

El alto grado de reducción que se obtiene en una primera instancia -resultados mostrados en la sección 5.1.2.1- puede deberse a varios factores:

- **Temperatura del sólido:** La definición de la temperatura del sólido fue realizada basándose en la descripción realizada por Arifin del comportamiento de esta temperatura, sin embargo, es altamente probable que esta descripción no se ajuste a la realidad, y que -debido a pérdidas de calor por radiación principalmente- la temperatura real del sólido haya sido menor a la temperatura descrita. Esto afectaría directamente la velocidad de reacción del proceso y por ende el grado de reducción.
- **Re-oxidación en la reducción:** Una temperatura de reducción que no sea lo suficientemente alta sumado a una baja penetración del gas de arrastre podría implicar que exista

re-oxidación del material durante la reducción. Una re-oxidación relevante implicaría una menor concentración de vacancias al final de la etapa y por ende menor producción de oxígeno.

- Producción estequiométrica: La cantidad de producto obtenida es determinada a partir de la ecuación estequiométrica considerando los valores de concentración de vacancias δ . Esta corresponde a una idealización, por lo que en la realidad la producción de O_2 y H_2 al no seguir la estequiometría debería ser distinta.
- Cinética de reacción y difusión: Debido a que la cinética de reacción utilizada para la reducción es aquella descrita por Bulfin, las diferencias entre las situaciones físicas de ambos experimentos podrían implicar diferencias en el factor de frecuencia de expresión de Arrhenius para determinar la constante de velocidad.

5.1.2.3 Variación $n_{red}=1.65$

Una manera de manipular la cinética de reacción de la reducción es variando el valor del exponente del término $f(\alpha) = (1 - \alpha)^n$. Éste término representa la tasa de avance de la reacción, al aumentar el valor de n el valor de $f(\alpha)$ será menor (ya que la base de la exponencial siempre será menor a 1). En este sentido, la reacción tendrá el mismo comportamiento en lo relativo a la temperatura, sin embargo, la constante de velocidad de la reacción será menor. Luego de un proceso iterativo, se determinó que con un valor de $n_{red} = 1.65$ se obtienen niveles de producción de hidrógeno más cercanos a los resultados de Arifin.

Para la modelación de este nuevo caso planteado, se utilizaron los siguientes valores de δ_{final} para las dos etapas del proceso:

Temperatura de oxidación T_{ox} [°C]	δ_{final} [-] Reducción	δ_{final} [-] Oxidación
750	0.0298415	0.01476997
800	0.02984592	0.01359096
850	0.02985032	0.0125308
900	0.02985469	0.0115819
950	0.02985896	0.01073468
1000	0.02986298	0.00997889
1100	0.02986875	0.0087017

Cuadro 5.6: Concentración de vacancias final considerando un $n_{red} = 1.65$.

Al realizar la modelación de los procesos con un valor de $n_{red} = 1.65$ se obtienen los siguientes resultados:

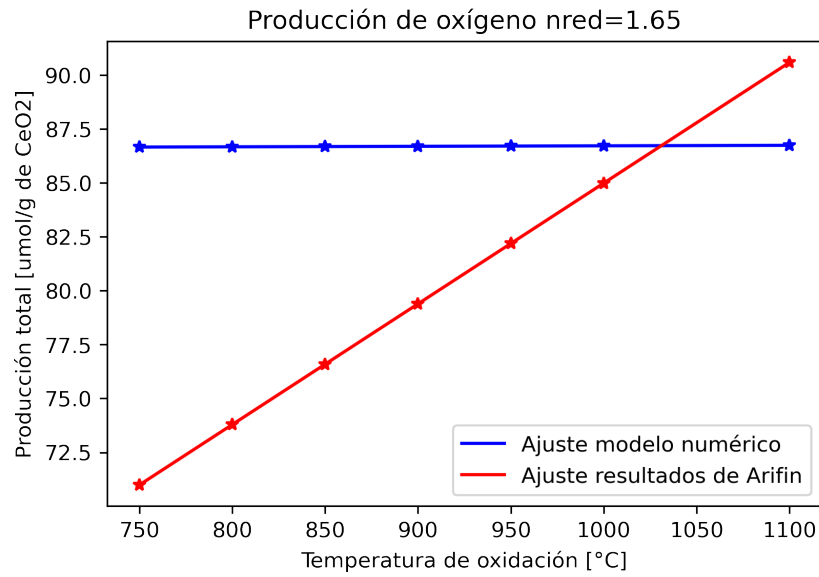


Figura 5.4: Producción de oxígeno para el modelo de comparación considerando $n_{red} = 1.65$.

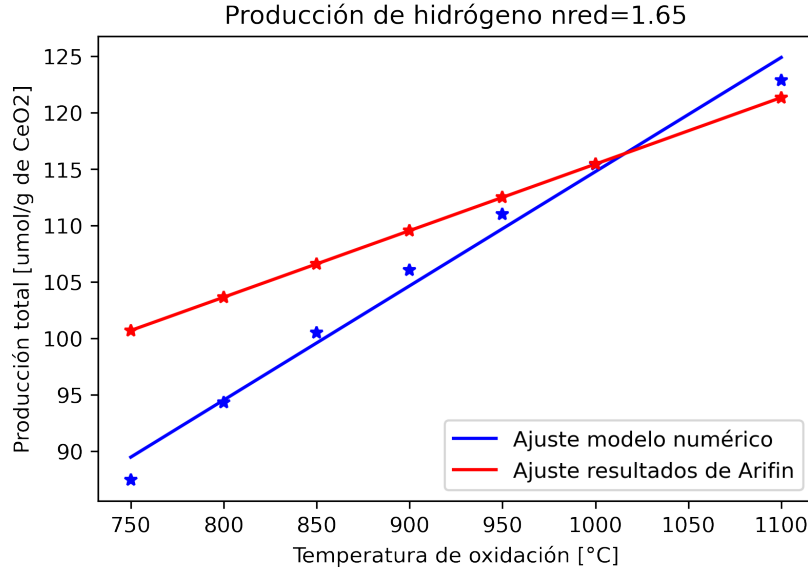


Figura 5.5: Producción de hidrógeno para el modelo de comparación considerando $n_{red} = 1.65$.

Podemos observar en la figura 5.4 que la producción de oxígeno para las distintas temperaturas de oxidación mantiene el mismo comportamiento exhibido con anterioridad al utilizar la cinética de Bulfin sin modificar para la reducción. De acuerdo con esto, es posible señalar que si bien al modificar el valor de n_{red} se logró disminuir considerablemente la velocidad de reducción, todavía existe una discrepancia con los resultados experimentales Arifin. Es probable que estas discrepancias estén relacionadas con la temperatura del sólido, debido al comportamiento constante que muestra el modelo en la reducción. En este sentido, la diferencia podría encontrarse en la manera en la que se define el perfil de temperatura.

Por otro lado, y a pesar de las diferencias en la reducción, la producción de hidrógeno tiene un comportamiento muy similar, obteniendo valores prácticamente idénticos al experimento cuando la temperatura de oxidación se encuentra en torno a los 1000[°C].

5.2. Modelo 1-D

Para la modelación del reactor de Water-Splitting -de situación física descrita en el Capítulo 2- se decidió realizar un análisis de sensibilidad; de esta manera, se puede determinar qué

variables resultan más y menos relevantes respecto a la cantidad de hidrógeno que producirá el reactor.

Los resultados obtenidos a partir del análisis de sensibilidad pueden permitir un mejor conocimiento del fenómeno tanto a la hora de realizar futuros experimentos en el reactor como con motivo de tener un conocimiento integral respecto a la incidencia de tanto la etapa de reducción como de oxidación en la producción final de hidrógeno.

5.2.1. Parámetros

En primer lugar se mostrarán en las tablas siguientes aquellos parámetros utilizados para resolver las ecuaciones del modelo presentado en el Capítulo 2.

δ_{max} [-]	A_{red} [1/s]	E_{red} [kJ/mol]	Y_{ox} [-]	n_{red} [-]
0.35	720000	232	0	1/3

Cuadro 5.7: Tabla de parámetros utilizados en la etapa de reducción.

A_{ox} [1/s]	E_{ox} [kJ/mol]	Y_{ox} [-]	γ [-]	n_{ox} [-]
1	29	1	0.89	1

Cuadro 5.8: Tabla de parámetros utilizados en la etapa de oxidación.

e [-]	lp [m]	dp [m]	$[Ce]$ [kmol/m ³]	mol_{CeO_2} [mol]	Vol [m ³]
0.65	0.1	0.01	12.0925	10.7179	$7.3 \cdot 10^{-4}$

Cuadro 5.9: Tabla de propiedades del medio poroso utilizadas para la simulación y obtención de resultados.

Δt [s]	nx [nodos]	R [kJ/molK]	σ [W/m ² K ⁴]
0.02	800	$8.3145 \cdot 10^{-3}$	$5.67 \cdot 10^{-8}$

Cuadro 5.10: Parámetros generales utilizados en la simulación.

5.2.2. Parámetros a sensibilizar

Dentro de las distintas variables involucradas en el problema a estudiar se realizó una selección de variables a sensibilizar. El criterio de selección utilizado fue el grado de control que se tiene sobre estas variables, por lo que su sensibilización si resulta relevante y podría contrastarse con experimentos.

Tanto para la reducción como para la oxidación las variables seleccionadas son: temperatura de borde T_{borde} , velocidad del flujo de gas u_g , aporte de calor externo Q_{ce} y tiempo de reacción t_r . Para el caso particular de la oxidación se sensibilizará también la temperatura inicial del sólido $T_{inicial}$. Los valores utilizados para cada variable se pueden encontrar en la tabla mostrada en la figura 5.6.

Tabla de parámetros a sensibilizar						
Condiciones:	Casos:	Caso 1	Caso 2	Caso 3	Caso 4	Caso 5
Reducción	Aporte de calor volumétrico [W/m ³]	1000000 (10 ⁶)	100000 (10 ⁵)	10000 (10 ⁴)	1000 (10 ³)	100 (10 ²)
	Temperatura de borde [K]	2200	2100	2000	1900	1800
	Velocidad del gas [m/s]	1	1,5	2	2,5	3
	Tiempo de reacción [min]	50	40	30	20	10
Oxidación	Aporte de calor volumétrico [W/m ³]	1000000	100000	10000	1000	100
	Temperatura de borde [K]	1200	1100	1000	900	800
	Velocidad del vapor [m/s]	1	1,5	2	2,5	3
	Temperatura inicial [K]	700	600	500	400	300
	Tiempo de reacción [min]	50	40	30	20	10
Caso base						

Figura 5.6: Tabla de parámetros a sensibilizar.

Como se puede apreciar en la tabla, se asignaron 5 casos para cada variable. Los casos se agruparon desde el más favorable (Caso 1) hasta el menos favorable (Caso 5). Este último caso será utilizado como caso base para el análisis.

Para determinar los casos más y menos favorables se realizó un pequeño análisis del comportamiento cualitativo del modelo: considerando que la reacción química depende principalmente de la temperatura, es de esperar que mayor temperatura implique mayor producción de hidrógeno. En este sentido, un mayor aporte de calor volumétrico implicaría también mayor producción de hidrógeno. Para el caso del tiempo de reacción también se tiene espera una proporcionalidad directa con la producción de H₂. Por otro lado, una mayor velocidad de gas

implicaría mayor intercambio de calor con la fase gaseosa y por lo tanto menor temperatura del sólido, por lo que al aumentar la velocidad de gas se espera una menor producción de hidrógeno.

5.2.3. Producción total y casos extremos

De acuerdo con la estequiometría (ecuaciones (12) y (13)) la cantidad máxima de moles de oxígeno e hidrógeno producida por cada etapa será igual a $\delta/2$ y δ respectivamente por cada mol de CeO_2 . Al tomar en consideración que $\delta_{max} = 0.35$ y la cantidad de CeO_2 en el reactor señalada en la tabla 5.9 se pueden calcular los valores máximo teóricos de producción de O_2 y H_2 :

- Moles máximo de O_2 : 1.8756 [mol]
- Moles máximos de H_2 : 3.7513 [mol]

Estos valores serán de utilidad para poder comparar la cantidad de productos calculada para el caso más favorable (Caso 1 de la figura 5.6) en particular.

Para la primera toma de resultados se determinó calcular la producción total para los casos extremos (más y menos favorable) junto con casos intercalados, es decir una reducción favorable junto con una oxidación desfavorable y viceversa. Estos resultados se muestran en la gráfica de la figura 5.7

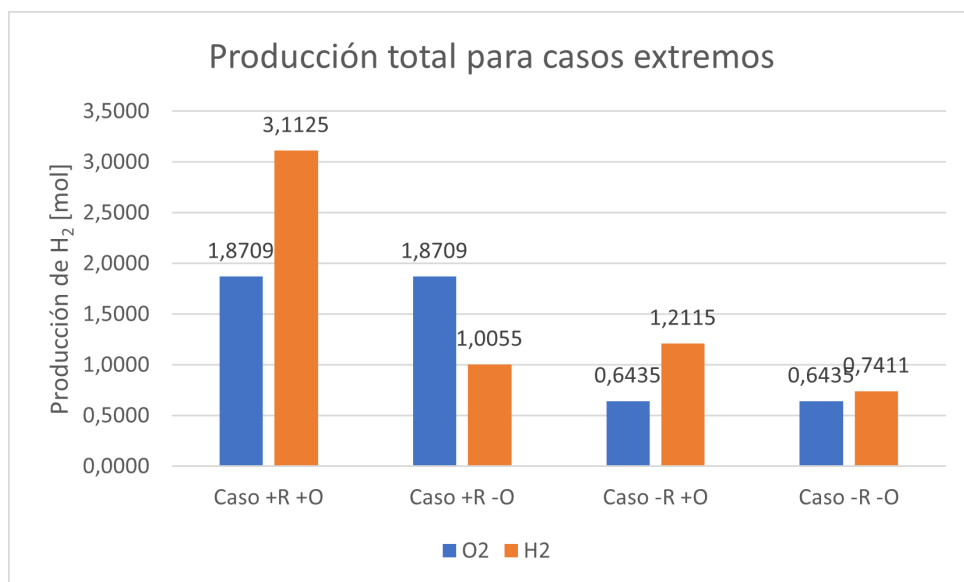


Figura 5.7: Producción de O₂ y H₂ para casos extremos y casos intercalados de reducción (R) y oxidación (O). El signo + representa el Caso 1 (más favorable) y el signo - el Caso 5 (menos favorable).

A partir de estos primeros resultados se desprenden varios aspectos interesantes respecto al proceso termoquímico simulado. En primer lugar podemos observar que la etapa reducción más favorable (+R) entrega una producción de oxígeno muy cercana a la producción máxima (diferencia de 0.25 %), mientras que la producción de hidrógeno -para el mismo caso (+R+O)- tiene una diferencia un tanto mayor con la producción máxima, diferencia de aproximadamente 0.6 [mol] (correspondiente a un 17 %).

Otro aspecto interesante la diferencia de producción de hidrógeno en los casos intercalados (+R-O y -R+O). Si bien ambos producen una cantidad de H₂ casi un tercio menor al caso más favorable, para el caso -R+O se tiene una producción de H₂ mayor en 0,2 [mol] (~20 %) respecto al caso +R-O. Esto implica que a priori es el proceso de oxidación el que toma más relevancia en la producción de hidrógeno en desmedro de la reducción. Esta última aseveración va ligada de acuerdo a las condiciones establecidas para cada caso mostradas en la figura 5.6.

En tercer lugar, es importante considerar los moles de H₂ producido en el cuarto grupo de columnas (Caso -R-O) ya que este será el caso que se tomará como base para el análisis

de sensibilizar a realizar a continuación. En este sentido, la producción mínima de hidrógeno será de 0.7411 [mol].

5.2.4. Sensibilización

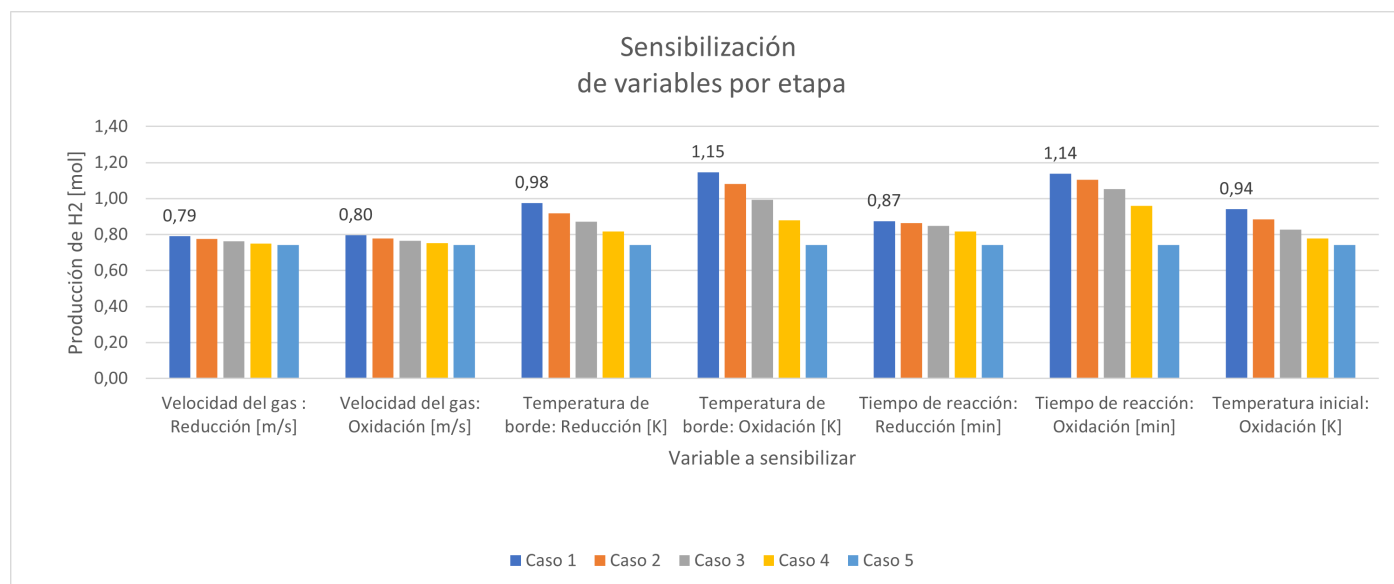


Figura 5.8: Producción de hidrógeno para cada caso y variable de sensibilización.

En la figura 5.8 se muestra la producción de hidrógeno obtenida a partir de la sensibilización de cada variable de acuerdo a los parámetros mostrados en la tabla de la figura 5.6. Los números sobre las columnas representan los moles de H_2 que son producidos en el Caso 1 (más favorable) para cada variable.

Se puede observar que la velocidad de entrada del gas no es una variable que incida significativamente en la producción de H_2 . La incidencia de esta variable tiene relación principalmente con la transferencia de calor desde el gas hacia el sólido. Debido a la considerable diferencia de temperaturas entre ambas fases, una mayor velocidad de flujo de gas implica un mayor coeficiente de convección y por ende mayor transferencia de calor desde la fase sólida a la fase gaseosa, por lo que la temperatura del sólido -y por consecuencia la producción de H_2 - es menor. La diferencia en las temperaturas para cada fase se evidenciará en la sección 5.2.5, donde se muestran los perfiles de temperatura para los casos extremos

La temperatura de borde es una variable muy sensible de acuerdo a los parámetros establecidos. En particular para el caso de oxidación, se tiene un aumento de aproximadamente el 55 % en los moles producto de H_2 para más favorable. Por el lado de la temperatura de borde de reducción, si bien su influencia es menor a su contra parte de oxidación, el aumento en la producción si es considerable, de hecho es la tercera variable que mayor incremento porcentual presenta ($\sim 32\%$).

La temperatura inicial para la oxidación tiene una influencia similar a la temperatura de borde de reducción, llegando a un valor máximo de 0.94[mol] de hidrógeno producido para una temperatura inicial de oxidación de 700[K]. De acuerdo a los modelos utilizados, la reacción química depende principalmente de la temperatura del sólido; es consistente entonces que cualquier variación que mejore el comportamiento térmico del reactor implicará una mayor producción de H_2 .

El último grupo de variables señaladas en la figura 5.8 corresponde al tiempo de reacción. Se aprecia un mayor efecto de la variable para el caso de la oxidación en desmedro de la reducción. Esto sigue la lógica observada tanto en la sensibilización de las temperaturas como en los resultados mostrados en la figura 5.7 en el sentido de que, de acuerdo a los casos planteados, la oxidación es más relevante que la reducción en lo que se refiere a la sensibilidad de los parámetros que la componen. Para el caso de la oxidación, se tiene un incremento aproximado del 53 % en la producción de H_2 , mientras que la reducción presenta un incremento ligeramente mayor ($\sim 18\%$) al observado en la sensibilización de la velocidad inicial de los gases.

Si bien en el análisis de sensibilidad mostrado en la figura 5.8 se eligió sensibilizar una variable a la vez, el programa desarrollado para el modelo del reactor permite poder sensibilizar varias variables al mismo tiempo y así estudiar la sinergia existente entre dos o más variables.

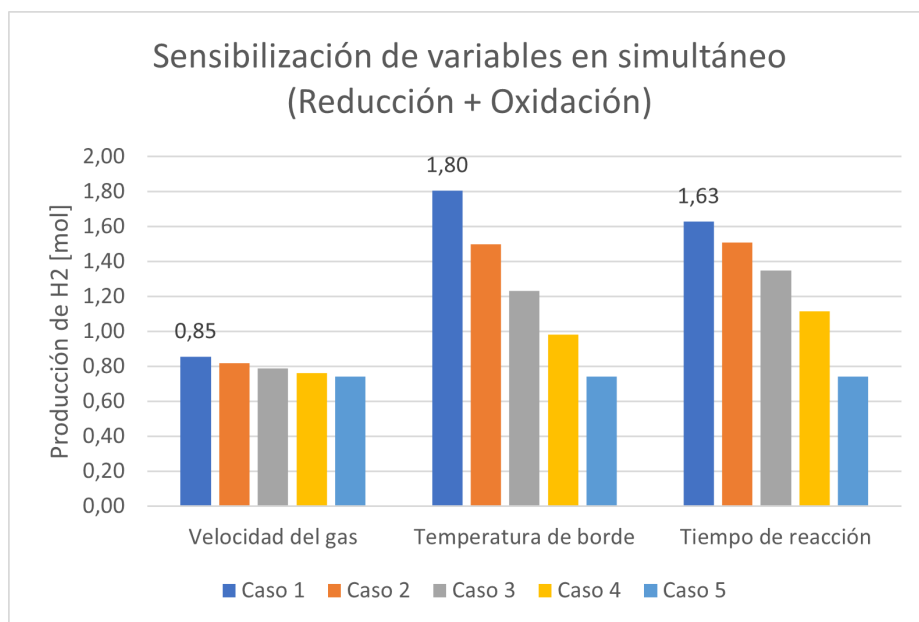


Figura 5.9: Sensibilización por pares de variables (parámetros de reducción y oxidación sensibilizados al mismo tiempo).

En la figura 5.9 se muestra la sensibilización realizada por pares de variables. En este caso, se decidió sensibilizar en simultáneo la misma variable de cada etapa (velocidad para reducción y oxidación, temperaturas para reducción y oxidación, etc.). Los resultados se conciben con aquellos mostrados en la sensibilización individual. Se observa un gran incremento por cada caso al variar la temperatura de borde de ambas etapas, seguido por una variación también considerable al aumentar los tiempos de reacción. Por el lado de la velocidad del gas, solo se tiene un incremento de aproximadamente 0.05[mol] en comparación al valor obtenido sensibilizando la velocidad de gas de solo una etapa.

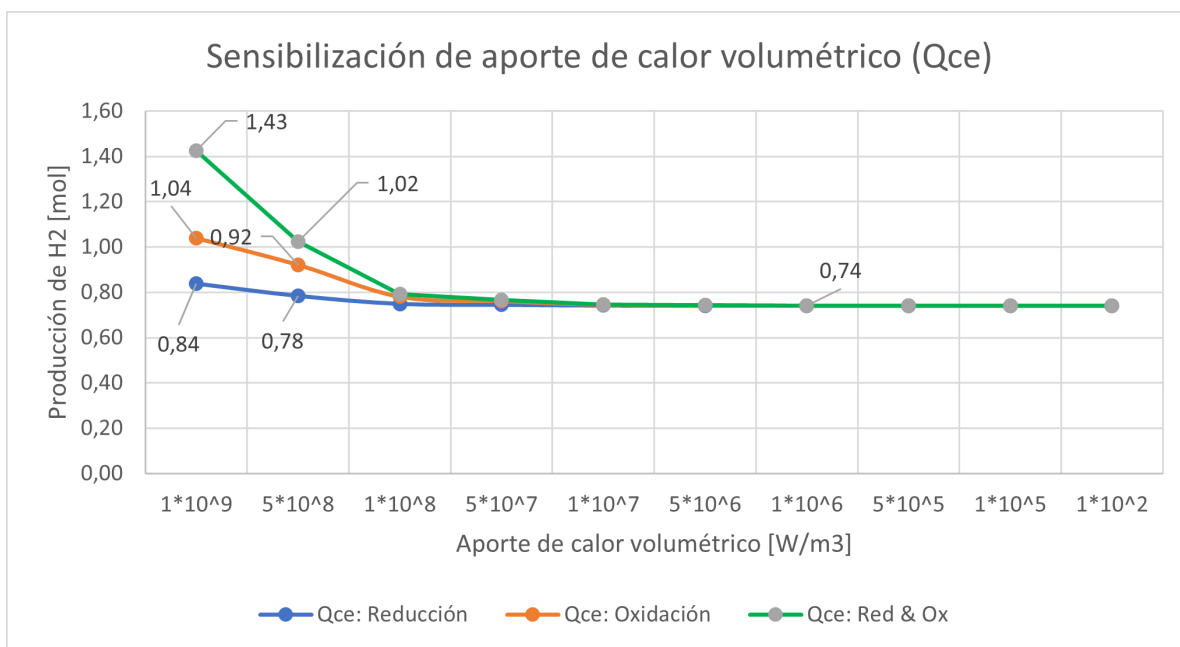


Figura 5.10: Producción de H_2 de acuerdo a la sensibilización de aporte de calor volumétrico Q_{ce} .

En los resultados mostrados en las figuras 5.8 y 5.9 no se incluyen los resultados de sensibilización del aporte de calor volumétrico Q_{ce} debido a que la variable resultado no ser sensible en lo absoluto de acuerdo al utilizar valores de entre 10^2 y 10^6 [W/m^3]. En este sentido, se decidió graficar individualmente este parámetro y utilizar valores de Q_{ce} de mayor magnitud.

En la figura 5.10 se muestra la sensibilización de la variable Q_{ce} de reducción, oxidación y de ambas etapas (análogo a los resultados de la figura 5.9). Al utilizar un valores de Q_{ce} mayores a 10^8 recién se observan incrementos en la producción de hidrógeno. Para un $Q_{ce} = 10^9$ (oxidación) se tiene un incremento de aproximadamente el 40 % respecto al caso base. Al igual que en las variables observadas en los análisis anteriores, para la oxidación se tiene un incremento mayor en la producción de H_2 en comparación a la reducción (~ 13 %).

En el cuadro 5.11 se muestra el aporte de calor volumétrico y el calor suministrado. Este último valor corresponde a Q_{ce} multiplicado por el volumen aproximado del reactor. De esta manera resulta más sencillo comparar cuanta es la potencia del calentador eléctrico externo.

Q_{ce} [W/m ³]	Q total [W]	Q total
10^9	730000	0,73 [MW]
10^8	73000	73[kW]
10^7	7300	7,3[kW]
10^6	730	730[W]
10^5	73	73[W]
10^4	7,3	7,3[W]

Cuadro 5.11: Aporte de calor volumétrico y aporte de calor total equivalente.

5.2.5. Casos Extremos: Perfiles de T_s , T_g , y δ

En esta sección se presentarán los perfiles de temperatura de fase sólida (T_s) y gaseosa (T_g) junto con la distribución de vacancias (δ) de las etapas de reducción y oxidación utilizando los parámetros correspondientes a los casos 1 y 5 mostrados en la figura 5.6 (casos extremos). El tiempo de reacción utilizado para las dos etapas del proceso es de 10 minutos, en las gráficas se mostrarán los perfiles en intervalos de 1 minuto, a modo de observar la evolución de la variable tanto en el tiempo como en el espacio.

5.2.5.1 Caso 1: Condiciones más favorables

Reducción

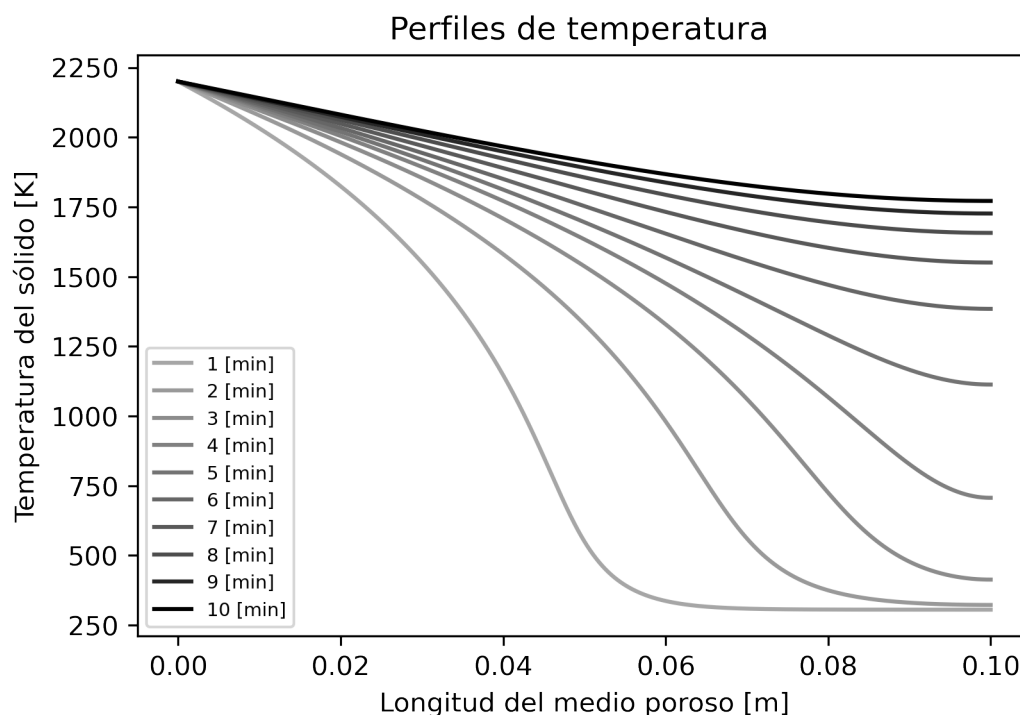


Figura 5.11: Perfil de temperatura del sólido para la reducción considerando Caso 1 (más favorable).

Las condiciones de borde impuestas se ven claramente reflejadas en este perfil de temperatura, en donde para todos los tiempos se tiene una temperatura fija en el extremo izquierdo (correspondiente al aporte solar) y una temperatura constante en los últimos nodos del extremo derecho. Las separaciones más pronunciadas entre cada perfil graficado indican variaciones más rápidas en la temperatura. De acuerdo con esto podemos deducir que durante los primeros 5 el aumento de temperatura es más rápido que en los 5 minutos posteriores. Además se puede observar -por ejemplo- como la temperatura en el extremo derecho crece rápidamente (desde los 3 minutos aproximadamente) hasta llegar a un valor de ~ 1900 [K].

Luego de 10 minutos de reducción, el medio poroso en su totalidad tiene una temperatura promedio de aproximadamente 2000[K]; una temperatura óptima para una reducción rápida

y efectiva.

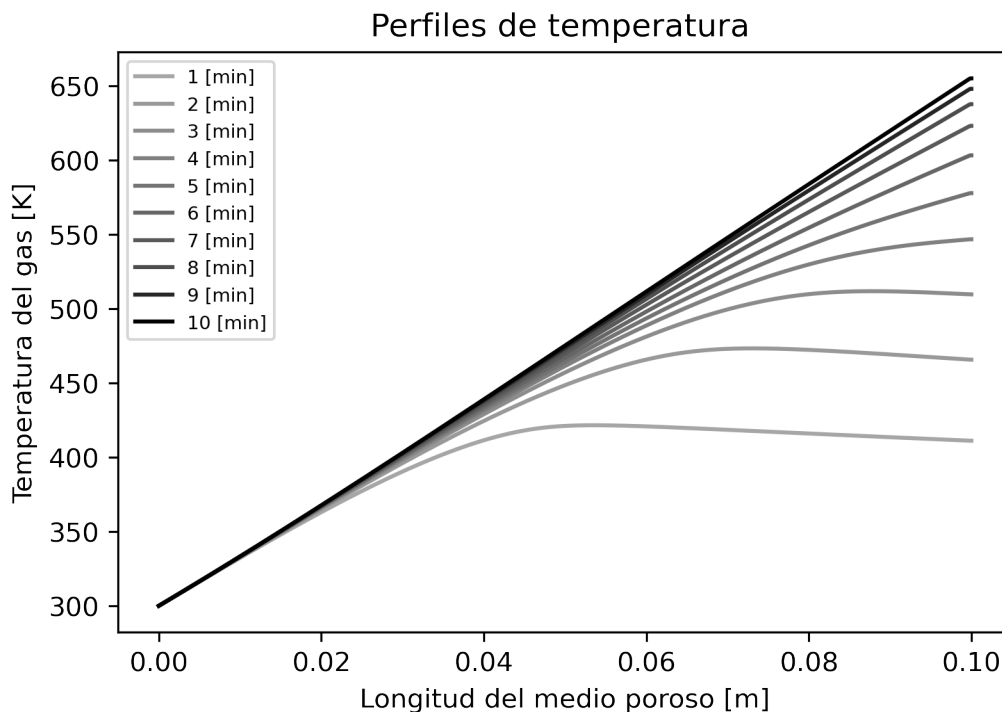


Figura 5.12: Perfil de temperatura del gas para la reducción considerando Caso 1 (más favorable).

La temperatura de fase gaseosa muestra un comportamiento similar en lo que respecta a la separación de los perfiles graficados por cada minuto de reacción. Luego de 10 minutos de reacción se observa un comportamiento convergente de la temperatura, obteniéndose un perfil lineal que va desde la temperatura de borde (300[K]) hasta una temperatura aproximada de 650[K].

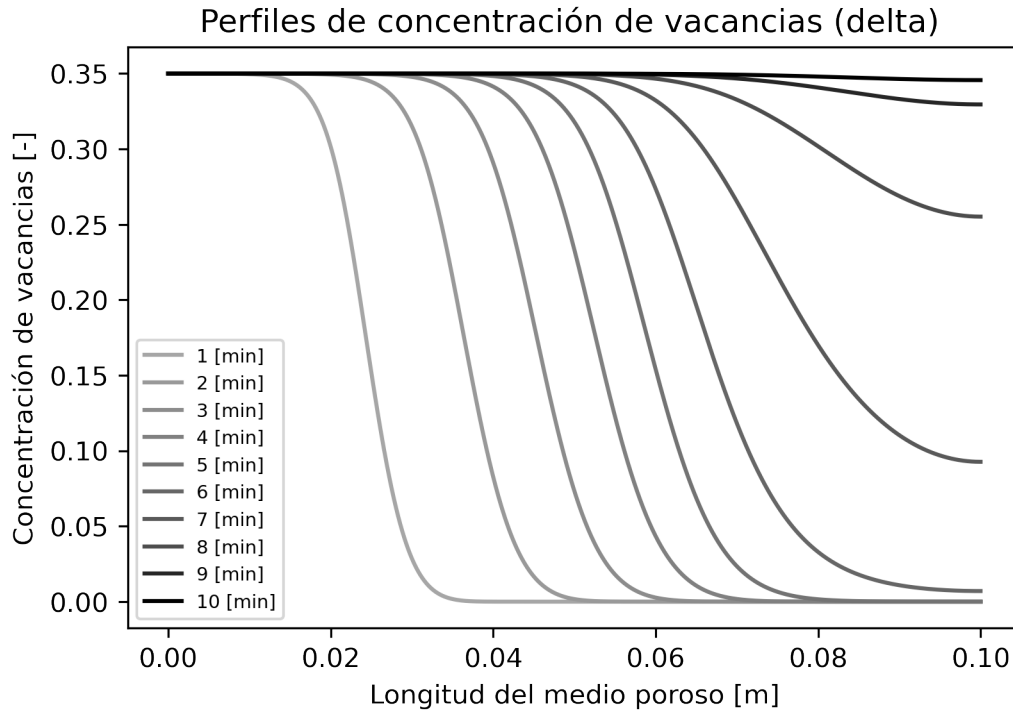


Figura 5.13: Distribución de vacancias para la reducción considerando Caso 1 (más favorable).

Tal como en las gráficas anteriores podemos ver claramente las condiciones de borde reflejadas en la gráfica para el perfil de concentración de vacancias δ .

Se tiene una reducción prácticamente completa una vez transcurridos 10 minutos de reacción. Esto es consistente con la cantidad de oxígeno calculada y mostrada en la figura 5.7.

Se observa que desde aproximadamente el minuto 6 la reducción de la vecindad del extremo derecho comienza ser apreciable. Al observar la figura 5.11 podemos dar cuenta que la temperatura en esa zona para dicho tiempo de reacción es de aproximadamente 1500[K].

Oxidación

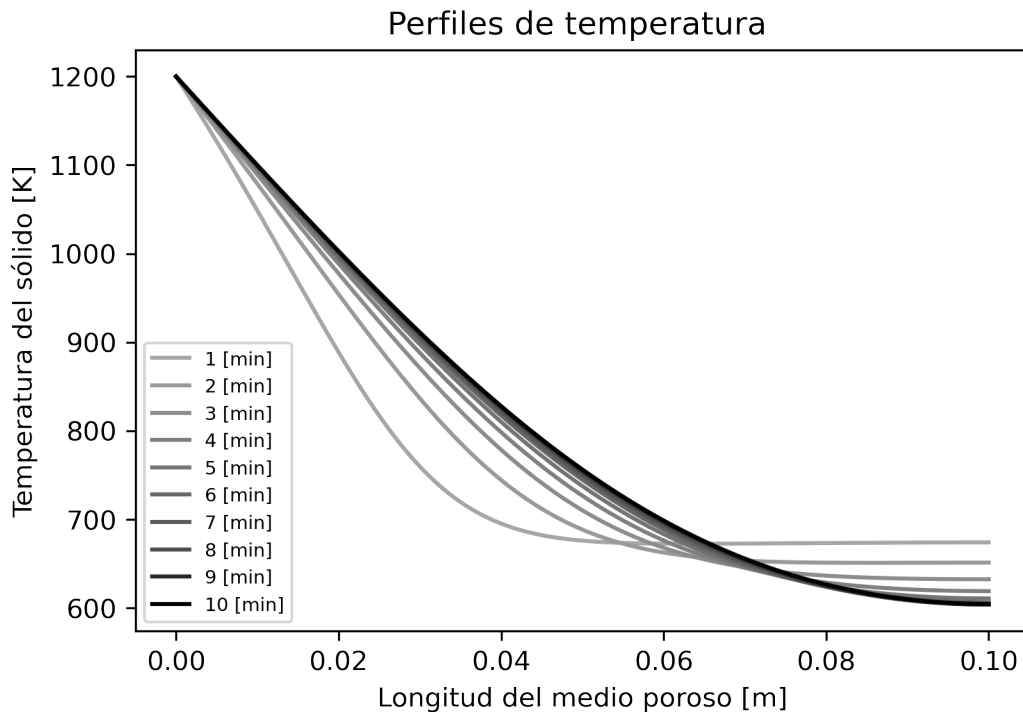


Figura 5.14: Perfil de temperatura del sólido para la oxidación considerando Caso 1 (más favorable).

La forma de los perfiles de temperatura del sólido para la oxidación es similar a la reducción (comportamiento exponencial), la temperatura en promedio es de menor magnitud debido a la condición de borde de oxidación.

Una diferencia significativa es que en el primer 60 % del medio poroso (desde la longitud 0 hasta 0.06[m]) se observa un aumento de temperatura en el tiempo, mientras que la sección restante muestra una disminución de la temperatura. Este último fenómeno es causado debido a que la transferencia de calor entre la fase sólida y gaseosa es tal que la temperatura de equilibrio resulta ser menor a la temperatura inicial (fijada en 700[K] para este caso).

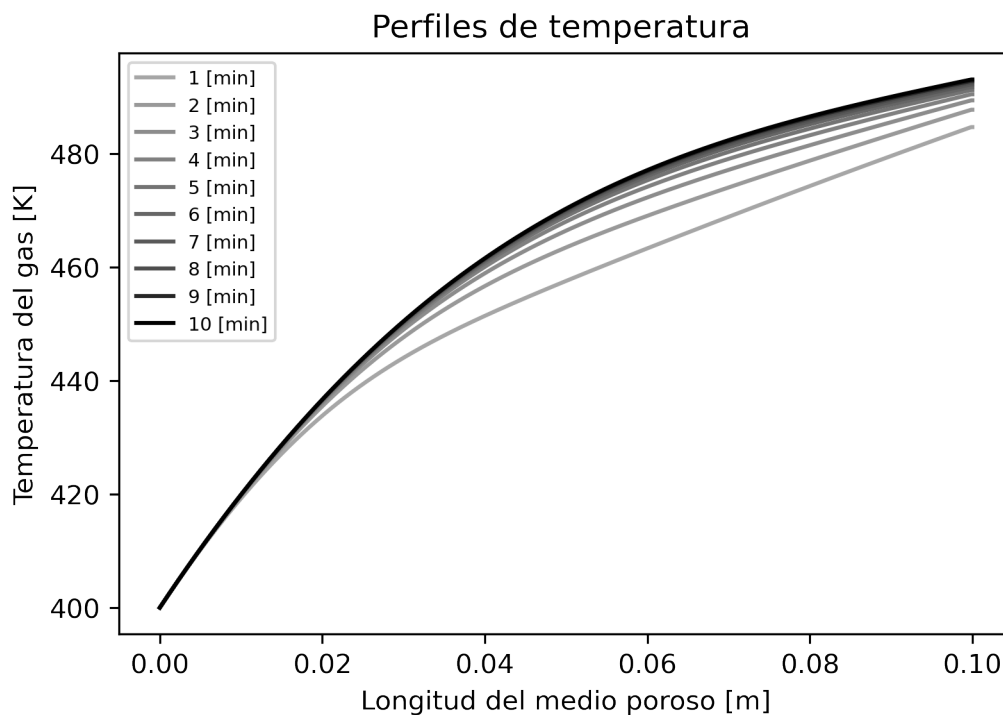


Figura 5.15: Perfil de temperatura del gas para la oxidación considerando Caso 1 (más favorable).

Se observa menor temperatura de fase gaseosa respecto a la reducción. Esto puede ser causado tanto por el menor gradiente de temperatura de sólido en comparación a la reducción (donde se tiene casi 1000 K de temperatura promedio de diferencia) como por el mayor calor específico del vapor en comparación del argón, que implica una mayor necesidad energética para cambiar en 1[K] la temperatura.

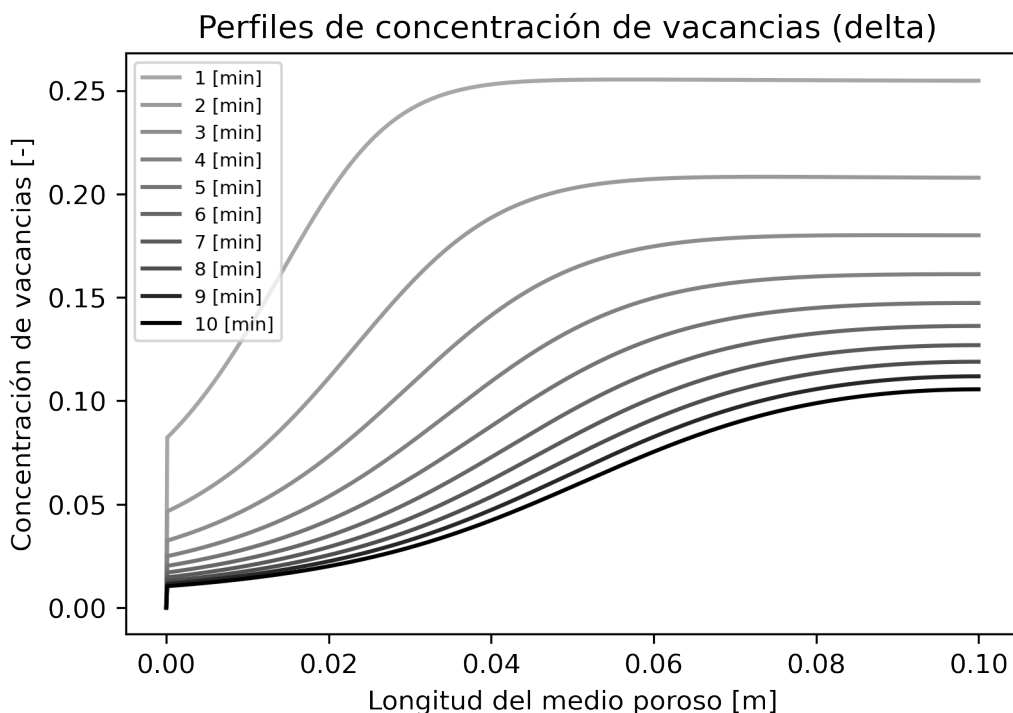


Figura 5.16: Distribución de vacancias para la reducción considerando Caso 1 (más favorable).

De la figura 5.16 se desprende que solo el primer 20 % se oxida completamente. Desde los 0.06[m] en adelante se oxida hasta un valor máximo de $\delta=0.1$ aproximadamente. Al ver la temperatura del sólido según la figura 5.14 (específicamente en el sector de medio poroso que no se reduce completamente) podemos dar cuenta de que la temperatura converge a un valor de aproximadamente 600K, temperatura que genera una oxidación muy lenta.

La separación de las curvas correspondientes a los 9 y 10 minutos de reacción es mayor en el tramo derecho del medio poroso, y al observar este sector podemos dar cuenta de una separación constante desde aproximadamente los 6 minutos en adelante; en este sentido el material podría seguir oxidándose al prolongar la reacción aunque a una velocidad lenta, debido a que la temperatura debiera mantenerse constante.

5.2.5.2 Caso 5: Condiciones menos favorables

Reducción

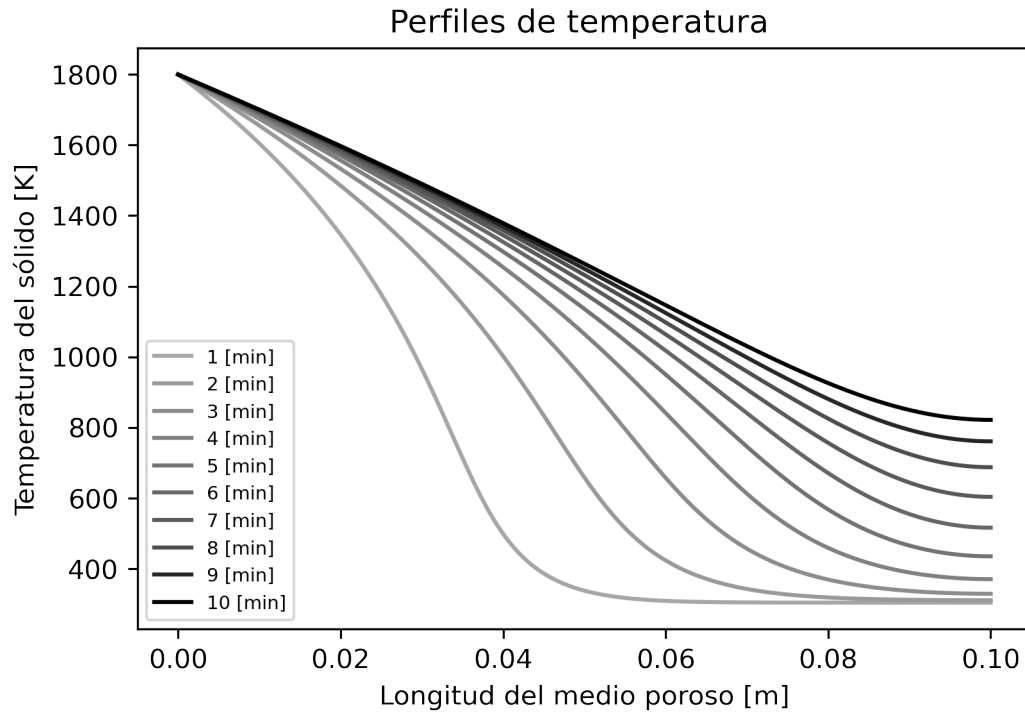


Figura 5.17: Perfil de temperatura del sólido para la reducción considerando Caso 5 (menos favorable).

El comportamiento de la temperatura del sólido dadas las condiciones del Caso 5 (figura 5.17) es muy similar a su par del Caso 1 (figura 5.11). Las principales diferencias observadas radican en la magnitud de las temperaturas y en la velocidad de propagación de calor (aumento de temperatura), la cual es una consecuencia directa del menor gradiente de temperatura. Importante señalar que entre la temperatura máxima y la temperatura mínima se tiene una diferencia de aproximadamente 800[K], casi 4 veces mayor que en el Caso 1.

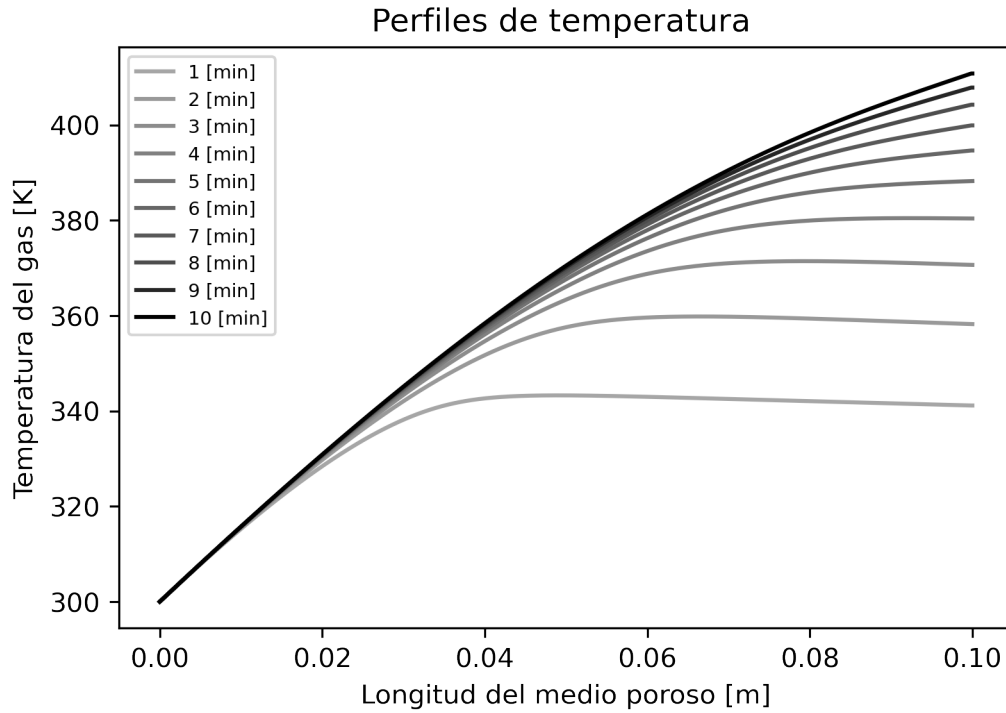


Figura 5.18: Perfil de temperatura del gas para la reducción considerando Caso 5 (menos favorable).

Para el caso de la fase gaseosa, se presentan las mismas diferencias con las condiciones más favorables que se tuvieron al comparar la temperatura del sólido (magnitud y velocidad de propagación). Se observa una diferencia de 100[K] entre cada extremo; una diferencia bastante menor que los casi 350[K] que se observan en la figura 5.12 correspondiente al caso más favorable.

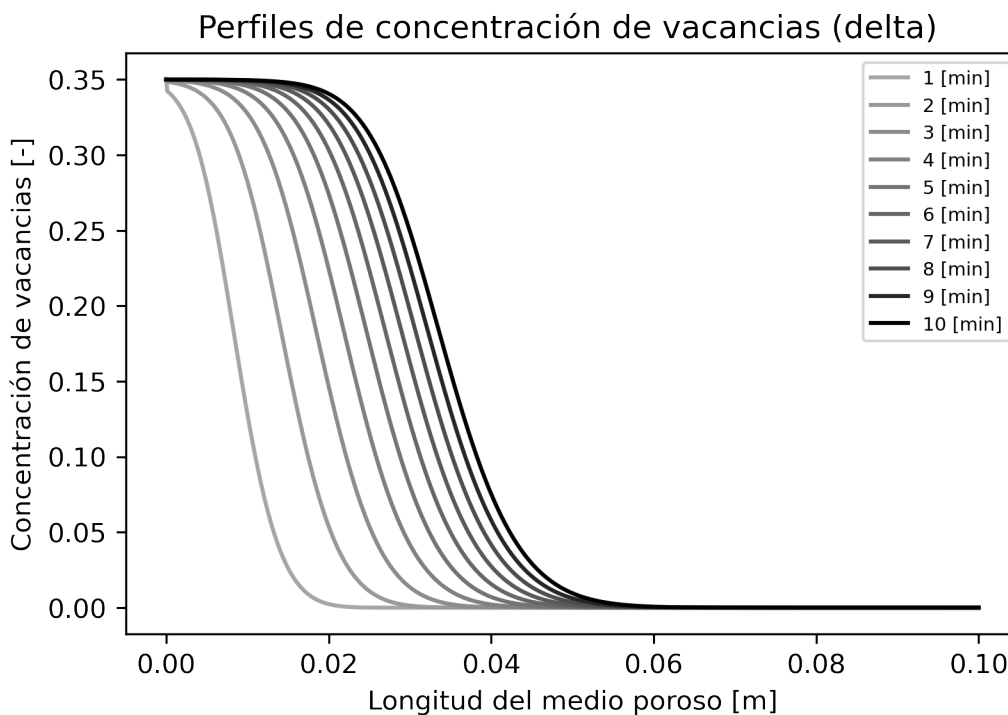


Figura 5.19: Distribución de vacancias del sólido para la reducción considerando Caso 5 (menos favorable).

El perfil de concentración de vacancias muestra que para este caso cerca de un 25 % del reactante llega hasta el grado máximo de reducción de 0.35[-]. Además, casi la mitad del medio poroso no se reduce en absoluto, lo cual explica la gran diferencia entre las producciones de oxígeno de los casos más y menos favorables, y además, implicará una baja producción de hidrógeno debido a la poca disponibilidad de material reducido.

Oxidación

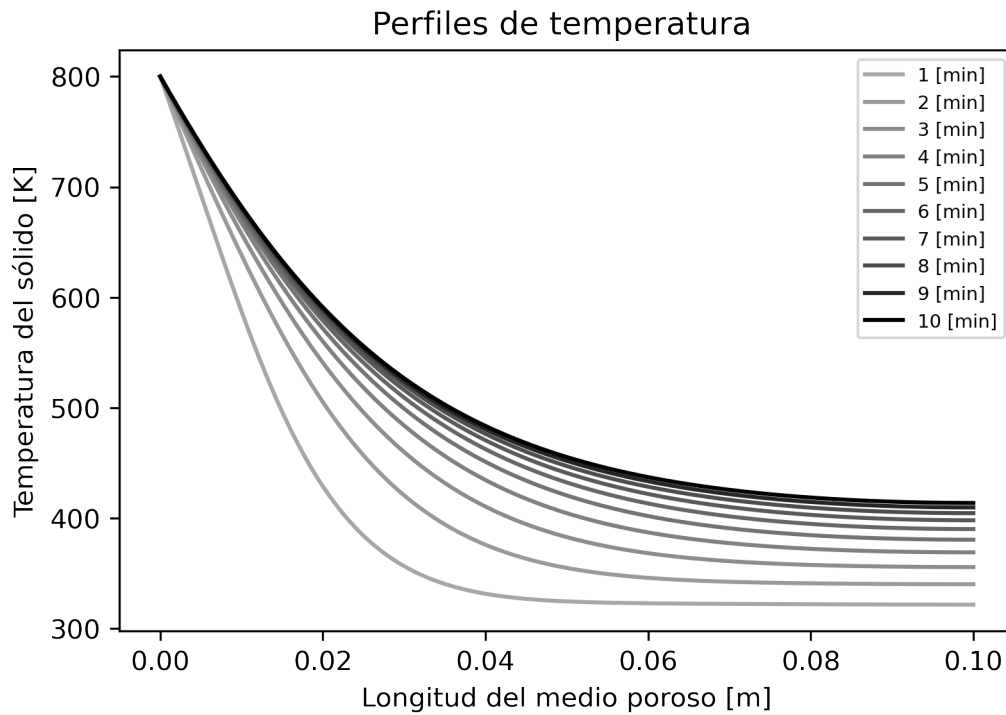


Figura 5.20: Perfil de temperatura del sólido para la oxidación considerando Caso 5 (menos favorable).

Para la oxidación, el perfil de temperatura de fase sólida muestra un comportamiento cualitativamente similar a los perfiles mostrados anteriormente. Se observa un aumento de temperatura mayor en los primeros 0.04[m]. Un aspecto interesante es que -al igual que en el Caso 1 (figura 5.15) la temperatura en el borde derecho es la mitad de la temperatura en el borde izquierdo. La temperatura en la zona reducida (entre 0 y 0.04[m]) tiene un valor promedio aproximado de 600[K], por lo que se espera una oxidación mucho más lenta que la observada en el Caso 1.

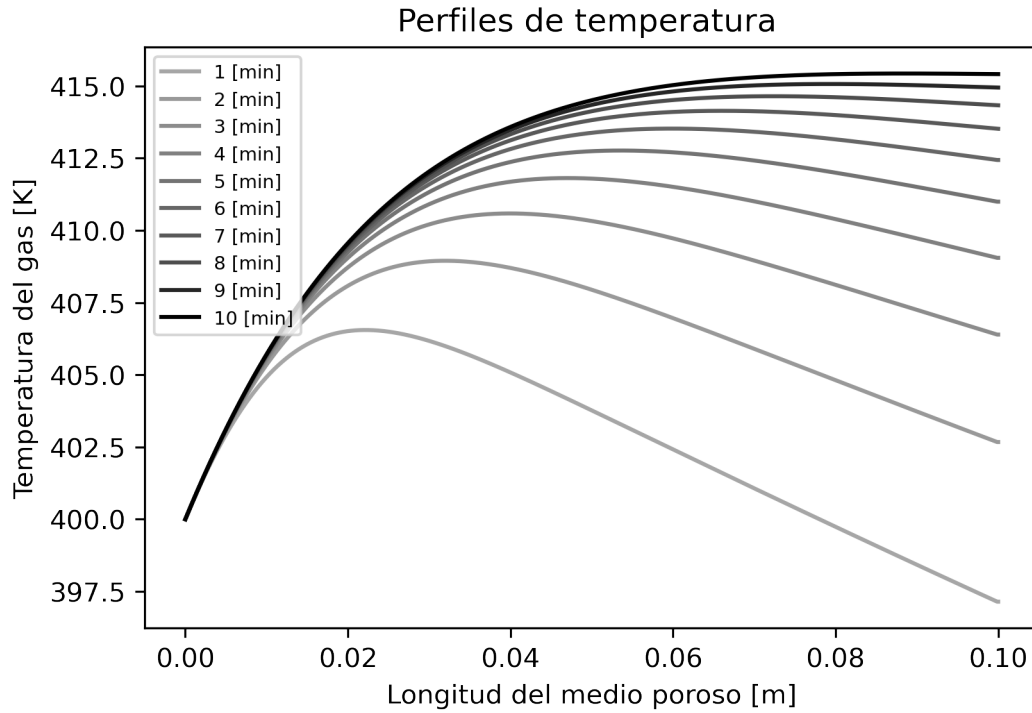


Figura 5.21: Perfil de temperatura del gas para la oxidación considerando Caso 5 (menos favorable).

La temperatura de fase gaseosa tiene un comportamiento bastante diferente a la observada en el caso anterior (figura 5.15). Inicialmente el vapor ingresa a 400[K], sin embargo, la inicial temperatura del sólido para este caso se fijó en 300[K], por lo el sentido la transferencia de calor en los primeros instantes de tiempo depende según la longitud del medio poroso. En los primeros tramos el vapor gana calor, aumentando su temperatura; a medida que el vapor avanza hacia sectores más fríos del medio poroso, el vapor se enfría transfiriendo calor hacia el sólido.

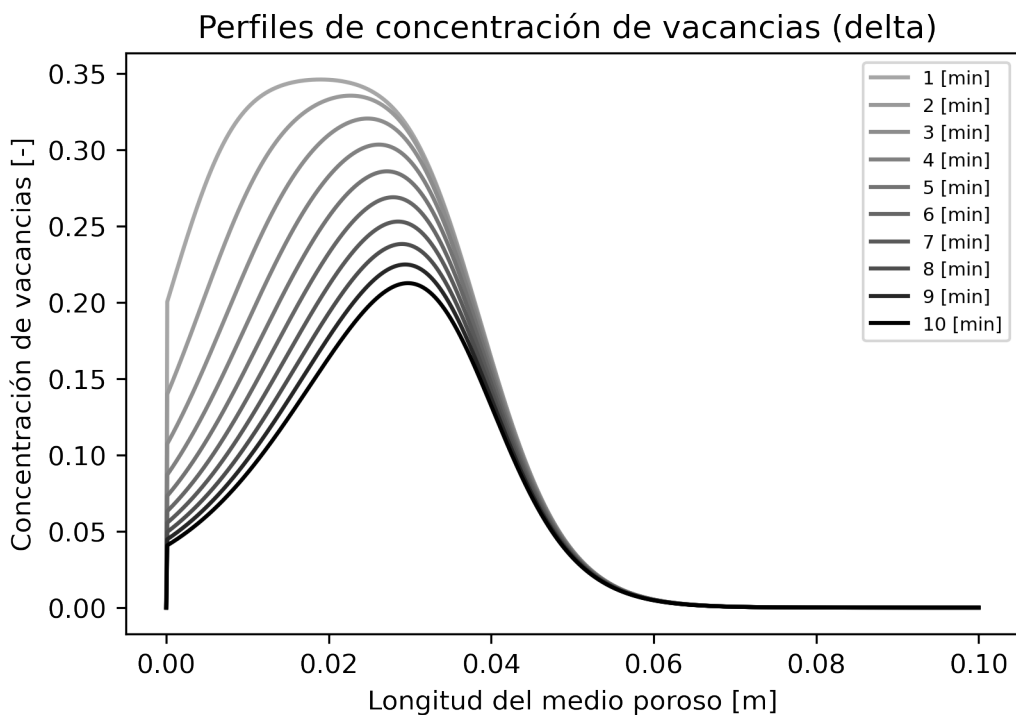


Figura 5.22: Distribución de vacancias del sólido para la oxidación considerando Caso 5 (menos favorable).

Tal como se señaló al observar la temperatura del sólido para la oxidación (figura 5.20), se puede observar una disminución bastante lenta en la concentración de vacancias, consecuencia de la baja temperatura; esto, sumado a gran cantidad de material sin reducir, explica la gran diferencia entre la producción de hidrógeno del caso más favorable respecto al presente caso.

Conclusiones

Señalar en primer lugar que se pudieron completar tanto el objetivo general como los objetivos específicos planteados. Se logró desarrollar e implementar un modelo numérico que permite la simulación de un reactor de Water-Splitting en una dimensión; para esto fue necesario completar ordenadamente cada objetivo específico, cuyo desarrollo fue explicado a lo largo de los 5 capítulos de este documento.

Mediante el modelo de comparación, análisis y contraste de resultados obtenidos con resultados experimentales de la literatura fue posible determinar la cercanía del modelo con el problema físico real, así como también identificar aspectos y parámetros que inciden en diferencias entre el modelo y experimento. Para una mejor comparación entre modelo, simulación y experimento es preciso conocer de manera detallada los parámetros operacionales así como el comportamiento físico de los objetos de estudio (por ejemplo, tener una buena caracterización de la temperatura del sólido).

En este sentido, los resultados obtenidos en el modelo unidimensional deben ser contrastados con futuras pruebas a realizar en el reactor; de esta manera será posible ajustar el modelo y obtener una simulación computacional más fidedigna del proceso. Una vez ajustado, el modelo puede ser acoplado a otros modelos o software que permitan el diseño de reactores más complejos o incluso a simulaciones de una planta productora de hidrógeno completa , en donde se pueda hacer un estudio completo para determinar rendimiento, aspectos de diseño y materiales, costos o cualquier otra variable de interés.

Propuesta de mejoras al modelo

A partir de las consideraciones tomadas para el modelo desarrollado en este trabajo se desprenden distintas mejoras que podrían significar un entendimiento más completo del problema a estudiar, así como también podrían injerir en tener resultados de modelación más cercanos a resultados de eventuales experimentos a realizar en el reactor.

- Iteración de $f(\alpha)$: El punto de equilibrio de la reacción puede no ser necesariamente la reducción/oxidación completa del material, tal como se planteó en el modelo de

comparación cinética presentado en el Capítulo 4 . El realizar el mismo procedimiento para el modelo del reactor de Water-Splitting significaría un incremento significativo del costo computacional de la modelación, sin embargo esto llevaría consigo resultados más fidedignos. El elevado costo computacional puede ser compensado al optimizar el código (ya sea manualmente o mediante bibliotecas como Numba) o al transcribirlo en un lenguaje de programación que presente mejores velocidades para cálculos numéricos (Fortran, por ejemplo).

- **Modelación en 2 (o más) dimensiones:** Un modelo en 1-D es muy útil en lo que respecta a un acercamiento inicial al entendimiento de un problema termofísico o termoquímico. Sin embargo, es una simplificación que coarta el entendimiento del problema en lo que respecta a la variación espacial de la temperatura a través del medio poroso y a la influencia de variables no consideradas en este modelo como la presión del flujo de gas. El paso lógico a seguir en la profundización de este trabajo es realizar un modelo en dos dimensiones del proceso. Si bien el costo computacional incrementaría de manera considerable, esto resultaría en un acercamiento más completo a la realidad, y permitiría realizar evaluaciones en lo que respecta al diseño del reactor y de los accesorios, como bombas o compresores a utilizar para el transporte de los fluidos.

Referencias

- [1] REVISTA Electricidad. El salto de las energías renovables en Chile bate todos los récords proyectados. [En línea] <https://www.revistaei.cl/2019/11/29/el-salto-de-las-energias-renovables-en-chile-bate-todos-los-records-proyectados/?mc_cid=21ade1b0a4&mc_eid=89e31b35a1>. [Consulta: 24 - 03 - 2021]
- [2] CENTRO Nacional del Hidrógeno. El Hidrógeno. [En línea] <<https://www.cnh2.es/el-hidrogeno/>>. [Consulta: 27 - 05 - 2021]
- [3] EL periódico de la energía. El hidrógeno y toda su gama de colores. [En línea] <<https://elperiodicodelaenergia.com/el-hidrogeno-y-toda-su-gama-de-colores/>>. [Consulta: 27 - 05 - 2021]
- [4] ARIEMA. Almacenamiento de Hidrógeno. [En línea] <<https://www.ariema.com/almacenamiento-de-h2>>. [Consulta: 27 - 05 - 2021]
- [5] CORDIS Resultados de investigaciones de la UE. Hidruros metálicos en el almacenamiento y la compresión de hidrógeno. [En línea] <<https://cordis.europa.eu/article/id/188584-metal-hydrides-for-hydrogen-storage-and-hydrogen-compression/es>>. [Consulta: 27 - 05 - 2021]
- [6] TECNOLOGÍA del hidrógeno y perspectivas para Chile. Rodrigo Vásquez, Felipe Salinas, Deutsche Gesellschaft für Internationale Zusammenarbeit (GIZ) GmbH. Santiago. 2018. 135p.
- [7] JUNTA de Castilla y León. El recorrido de la energía en Castilla y León: El hidrógeno y las pilas de combustible. [En línea] <<https://energia.jcyl.es/web/es/biblioteca/recorrido-energia-castilla-leon.html>>. [Consulta: 27 - 05 - 2021]
- [8] J. Lédé, F. Lapicque & J.Villermaux. Production of hydrogen by direct thermal decomposition of water. International Journal of Hydrogen Energy. 8(9):675-679. Jan. 1983.

- [9] E. Funk, Thermochemical hydrogen production: past and present. *International Journal of Hydrogen Energy*, 26(3), 185–190. Mar. 2001.
- [10] ANALYTICAL Model of CeO₂ Oxidation and Reduction. B. Bulfin [et al]. *The Journal of Physical Chemistry*, 117(46):24129-24137, Oct. 2013.
- [11] D. Arifin, A. W. Weimer. Kinetics and mechanism of solar-thermochemical H₂ and CO production by oxidation of reduced CeO₂. *Solar Energy*, 160:178-185, Jan. 2018.
- [12] LEVENSPIEL, Octave. *Chemical reaction engineering*. 3er ed. USA, John Wiley & sons, 1999. 668 p.
- [13] STATISTICAL thermodynamics of non-stoichiometric ceria and ceria zirconia solid solutions. B. Bulfin [et al]. *Physical Chemistry Chemical Physics*, 18(33):23147–23154, Aug. 2016.
- [14] NONG, G., LI, Y., & YIN, Y. Energy analysis on the water cycle consisting of photo catalyzing water splitting and hydrogen reacting with oxygen in a hydrogen fuel cell. *Chemical Physics Letters: X*, 4, 100033, Oct. 2019.
- [15] NUMERICAL modeling of solar thermo-chemical water-splitting reactor. Darryl L. James [et al]. *Proceedings of the ASME 2006 International Solar Energy Conference*: 221-227. Jul. 2006.
- [16] MODELING Chemical and Thermal States of Reactive Metal Oxides in a CR5 Solar Thermochemical Heat Engine. R.E. Hogan [et al]. *Proceedings of the ASME 2012 6th International Conference on Energy Sustainability*: 1137-1144. Jul. 2012.
- [17] MODELADO de la hidrólisis del agua para la producción de hidrógeno mediante la descomposición termoquímica del CeO₂ con energía solar concentrada. J. Valle-Hernández [et al]. *Revista de Aplicación Científica y Técnica*, Vol. 2, No. 6:9-15. Nov. 2016.
- [18] MODELADO y simulación de un proceso para la producción de hidrógeno a partir de la descomposición termoquímica del óxido de cerio con energía solar concentrada.

- J. Valle-Hernandez [et al]. Ingeniería, Tecnología y Ciencias Aplicadas 01:42-47. Mar. 2016.
- [19] TOLEDO Torres, Mario Gonzalo. Estudio teórico-computacional de la combustión superadiabática en medios porosos inertes. Tesis (Doctor en Ciencias de la Ingeniería mención Ingeniería de Procesos). Santiago. Universidad de Santiago de Chile, Facultad de Ingeniería. 2005.
- [20] HYDROGEN production in ultrarich combustion of hydrocarbon fuels in porous media. M. Toledo [et al]. International Journal of Hydrogen Energy (34):1818-1827. Jan 2009.
- [21] CFD online. Tridiagonal matrix algorithm - TDMA (Thomas Algorithm). [En línea] <[https://www.cfd-online.com/Wiki/Tridiagonal_matrix_algorithm_-_TDMA_\(Thomas_algorithm\)](https://www.cfd-online.com/Wiki/Tridiagonal_matrix_algorithm_-_TDMA_(Thomas_algorithm))>. [Consulta: 04 - 04 - 2021]
- [22] AN Evaluation of the Thermophysical Properties of Stoichiometric CeO₂ in Comparison to UO₂ and PuO₂. A. Nelson [et al]. Journal of American Ceramic Society, Vol 97 (11):3652-3659. Nov 2014.
- [23] NIST: National Institute of Standards and Technology, Libro web de Química del NIST. [En línea] <<https://webbook.nist.gov/cgi/cbook.cgi?ID=C7440371&Units=SI&Mask=1Thermo-Gas>>[Consulta 09 - 04 - 2021]
- [24] B. Younglove, H. Hanley. The Viscosity and Thermal Conductivity Coefficients of Gaseous and Liquid Argon. Journal of Physical and Chemical Reference Data 15, 1323. Aug 1986.
- [25] G.F.C. Rogers y Y.R. Mayhew. Thermodynamic and Transport Properties of Fluids . 5ta ed. USA, Wiley-Blackwell, 1995. 33p.

Anexos

Anexo A: Tablas de datos de sensibilización

Casos	Temperatura de borde					
	Ambos		Reducción		Oxidación	
	mol H2	%	mol H2	%	mol H2	%
Caso 1	1,80	143,49	0,98	31,64	1,15	54,66
Caso 2	1,50	102,10	0,92	24,03	1,08	45,91
Caso 3	1,23	66,02	0,87	17,46	0,99	33,95
Caso 4	0,98	32,59	0,82	10,15	0,88	18,60
Caso 5	0,74	0,00	0,74	0,00	0,74	0,00

Figura 7.1: Sensibilización temperatura de borde.

Casos	Velocidad del gas					
	Ambos		Reducción		Oxidación	
	mol H2	%	mol H2	%	mol H2	%
Caso 1	0,85	15,26	0,79	6,60	0,80	7,42
Caso 2	0,82	10,24	0,77	4,50	0,78	5,15
Caso 3	0,79	6,23	0,76	2,78	0,76	3,22
Caso 4	0,76	2,88	0,75	1,31	0,75	1,52
Caso 5	0,74	0,00	0,74	0,00	0,74	0,00

Figura 7.2: Sensibilización velocidad del gas.

	Temperatura inicial	
	Oxidación	
Casos	mol H2	%
Caso 1	0,94	26,95
Caso 2	0,88	19,17
Caso 3	0,83	11,54
Caso 4	0,78	5,05
Caso 5	0,74	0,00

Figura 7.3: Sensibilización temperatura inicial (oxidación).

Casos	Tiempo de reacción					
	Ambos		Reducción		Oxidación	
	mol H2	%	mol H2	%	mol H2	%
Caso 1	1,63	119,49	0,87	18,03	1,14	53,49
Caso 2	1,51	103,34	0,86	16,64	1,10	49,07
Caso 3	1,35	81,78	0,85	14,47	1,05	42,13
Caso 4	1,12	50,57	0,82	10,38	0,96	29,58
Caso 5	0,74	0,00	0,74	0,00	0,74	0,00

Figura 7.4: Sensibilización tiempo de reacción.

Aporte de calor volumétrico [W/m ³]	Aporte de calor volumétrico					
	Ambos		Reducción		Oxidación	
	mol H2	%	mol H2	%	mol H2	%
1*10 ⁹	1,43	92,43	0,84	13,00	1,04	40,07
5*10 ⁸	1,02	38,01	0,78	5,82	0,92	24,24
1*10 ⁸	0,79	6,70	0,75	1,09	0,78	5,35
5*10 ⁷	0,77	3,30	0,75	0,54	0,76	2,69
1*10 ⁷	0,75	0,65	0,74	0,11	0,75	0,54
5*10 ⁶	0,74	0,32	0,74	0,05	0,74	0,27
1*10 ⁶	0,74	0,06	0,74	0,01	0,74	0,05
5*10 ⁵	0,74	0,03	0,74	0,01	0,74	0,03
1*10 ⁵	0,74	0,01	0,74	0,00	0,74	0,01
1*10 ²	0,74	0,00	0,74	0,00	0,74	0,00

Figura 7.5: Sensibilización aporte de calor volumétrico.

Anexo B: Código en Python

Debido a la extensión de los códigos, estos se incluyen como archivos de pdf adjuntos a este informe, es posible que esto implique diferencias en la paginación señaladas en el índice.

Anexo B1

May 31, 2021

1 Modelo 1-D: determinación dt

1.1 Constantes

```
[1]: R=8.3145*10**(-3) # [kJ /mol K ] Constante universal de los gases.
Ared= 720000. #+-360000 [1/s] Factor pre-exponencial de reducción. Bulfin
Ered=232. # +-5 [kJ/mol] Energía de activación de reducción. Bulfin
lp=0.1 # [m] Longitud del medio poroso. ***DATO CÁMARA DE REACCIÓN ***
vol=7.3*10**(-4) # [m3] Volumen de la cámara de reacción. ***DATO CÁMARA DE
    ↳ REACCIÓN
e=0.65 # [-] Porosidad del medio poroso. ***DATO PELLETS BULFIN***
dp=0.01 # [m] diámetro de partícula ***DATO PELLETS BULFIN***
cCe=(1-e)*34.55 # [kmol/m3] concentración de cerio. Cálculo
molCeO2=(1-e)*30.6225 # =10.7179 moles totales de CeO2.
sigma=5.67*10**(-8) # [W/m2K4] Constante de Stefan-Boltzmann.

#Cálculo moles totales CeO2
#7.22 g/cm3 * 730 cm3 / 172.115(g/mol)= 30.6225 moles de CeO2 totales

yox=1 #Fracción molar de oxidante
Aox=1.0 # Factor pre-exponencial de oxidación. Arifin
Eox=29.0 # Energía de activación de oxidación. Arifin
gamma=0.89 # [-] Orden de reacción para la oxidación. Arifin
```

1.2 Función de Reducción

```
[2]: #Definición de función reducción
import numpy as np
def reduccion(dt,t_reaccion): #Los inputs se eligen desde el apartado
    ↳ 'variables' ubicado mas abajo. Al seleccionar se debe marcar con #

    #Funciones entalpía de reacción
    def entalpia_red(delta): #Valores EN kJ/mol

        if delta==0:
            entalpia=430
        elif 395-31.4*np.log10(delta) > 430:
            entalpia=430
```

```

else:
    entalpia= 395-31.4*np.log10(delta)
return entalpia

#.....Variables.....

u_in=1. # [m/s] Velocidad inicial del gas
Ts_border=2200 #[K] Temperatura del solido en x=0 en la reducción.
Qce=1000000 #[W/m3] Aporte de calor volumétrico externo.
nx=800 #[nodos] Cantidad de nodos temporales
#dt=0.02 #[segundos] Intervalo de tiempo
#t_reaccion=15 #[minutos] Tiempo de reacción.
#.....

#Constantes
Tg_border=300. #Temperatura del gas en x=0 en la reduccion.
d_border=0.35 #Concentracion de vacancias en el sólido en x=0 en la
→reducción.
d_max=0.35 #Concentración de vacancias máxima.
dx=lp/nx #[-] Espaciamiento
molxnodo=molCeO2/nx #cantidad de moles de CeO2 por nodo.

#####
#RESOLUCION DE ECUACIONES

#condiciones iniciales
d=np.zeros(nx)
d[0]=d_border
d_final=np.zeros(nx)+0.35 #condición final de reducción (consideración de
→reducción completa)

Ts=np.zeros(nx)+300
Ts[0]=Ts_border
Tg=np.zeros(nx)+300
Tg[0]=Tg_border

kred=np.zeros(nx) #mod eq vacancias
kox=np.zeros(nx) #mod eq vacancias

#se crean los arreglos de A, B, C, F, alfa y beta vacías, los cuales se usan
→para aplicar TDMA.
A_s=np.zeros(nx)
B_s=np.zeros(nx)
C_s=np.zeros(nx)
F_s=np.zeros(nx)

```

```

alfa_s=np.zeros(nx)
beta_s=np.zeros(nx)

A_g=np.zeros(nx)
B_g=np.zeros(nx)
C_g=np.zeros(nx)
F_g=np.zeros(nx)
alfa_g=np.zeros(nx)
beta_g=np.zeros(nx)

tiempo=0 #[min] tiempo de reacción
nt=0

while tiempo < t_reaccion:

    nt+=dt #contador para el tiempo de reacción

    #primero calculo todas las propiedades en función de la temperatura

    #sólido
    rho_s=(-0.0002*Ts+6.8561)*1000 #7220 #[kg/m3]densidad del sólido *ref. Nelson 2014
    cp_s= (0.0599*np.log(Ts)+0.0390)*1000 #[J/kgK]calor específico del sólido * ref. Nelson 2014
    lambda_s= (6.7760*10**(-2)+2.7930*10**(-4)*Ts)**(-1)#[W/mK] Coeficiente de transferencia de calor del sólido. ref. Nelson 2014
    lambda_ef= (1.-e)*lambda_s + (32.*sigma*dp*e*Ts**3.)/(9.*(1.-e)) #[W/mK] Coeficiente de transferencia de calor efectivo. ref. Toledo

    #gas (argón)
    rho_ar=1.603*300/Tg #[kg/m3]. Gases ideales
    cp_ar=(20.7860+Tg*2.8259*10**(-7)-Tg**2*1.4642*10**(-7)+Tg**3*1.0921*10**(-8)-Tg**(-1)*3.6614*10**(-8))*1000/39.948 #[J/kgK]. ref. webbook. nist.gob. Chase 1998
    lambda_ar=(0.0442*Tg + 4.7827)/1000 #[W/Km] Conductividad térmica. Ajuste Younglove 1986.
    mu_ar= (0.0566*Tg+6.0645)*10**(-6) ##[Pa*s] viscosidad dinámica del argón. Ajuste Younglove 1986.
    nu_ar=mu_ar/rho_ar # viscosidad cinematica del argon

```

```

u_ar=1.603*u_in/rho_ar # m/s velocidad del argon (depende de la
→ densidad)

#parámetros para a_ar. ref. Toledo
ags_ar=6./(1.-e) #superficie específica interna del medio poroso
Pr_ar=(mu_ar*cp_ar)/lambda_ar #Número de Prandtl
Re_ar=(e*u_ar*dp)/mu_ar #Número de Reynolds
hgs_ar=(lambda_ar/dp)*(2.0+1.1*Pr_ar**(1/3)*Re_ar**0.6) #Coeficiente
→ convectivo interfásico
a_ar = ags_ar*hgs_ar #Coeficiente intersticial de intercambio de calor

#Cálculo de vacancias
d_n=d.copy() #d_n sería d anterior (d^n), d sería el valor actual
→ d^(n+1)

kred=(1-d_n/d_final)**(1/3)*Ared*np.exp(-Ered/(R*Ts)) #La temperatura
→ todavía no se calcula para n+1, por tanto Ts corresponde a la temperatura en
→ n.
kox=y_ox**gamma*(1-d_n/d_final)*Aox*np.exp(-Eox/(R*Ts)) Para el caso
→ de la reducción será cero en la reducción

d=(d_n/dt+d_max*kred)/(kred+kox+1/dt) #como son estáticas no requiere
→ recorrer x

#Cálculo de temperaturas
for i in range(nx):
    if i==0:
        #solido
        beta_s[i+1]=Ts_border

        #gas
        beta_g[i+1]=Tg_border

    elif i<(nx-1):
        #solido
        A_s[i]=(lambda_ef[i-1]+lambda_ef[i])/(2*dx**2)
        B_s[i]=(lambda_ef[i]+lambda_ef[i+1])/(2*dx**2)
        C_s[i]=(1-e)*(cp_s[i]*rho_s[i])/dt + a_ar[i] + A_s[i]+B_s[i]
        F_s[i]=Ts[i]*(1-e)*(cp_s[i]*rho_s[i])/dt + a_ar[i]*Tg[i]+Qce/
→ nx-(d[i]-d_n[i])/dt*(cCe/nx)*(10**6)*entalpia_red(d[i])

        alfa_s[i+1]=B_s[i]/(C_s[i]-A_s[i]*alfa_s[i])
        beta_s[i+1]=(F_s[i]+beta_s[i]*A_s[i])/(C_s[i]-A_s[i]*alfa_s[i])

```

```

        #gas
        A_g[i]=(lambda_ar[i-1]+lambda_ar[i])/
        ↪ (2*dx**2)+u_ar[i]*cp_ar[i]*rho_ar[i]/dx
        B_g[i]=(lambda_ar[i]+lambda_ar[i+1])/(2*dx**2)
        C_g[i]=(cp_ar[i]*rho_ar[i])/dt + a_ar[i]/e + A_g[i]+B_g[i]
        F_g[i]=Tg[i]*(cp_ar[i]*rho_ar[i])/dt + a_ar[i]*Ts[i]/e

        alfa_g[i+1]=B_g[i]/(C_g[i]-A_g[i]*alfa_g[i])
        beta_g[i+1]=(F_g[i]+beta_g[i]*A_g[i])/(C_g[i]-A_g[i]*alfa_g[i])

    else:
        #solido
        Ts[i]=beta_s[i]/(1-alfa_s[i])

        #gas
        Tg[i]=beta_g[i]/(1-alfa_g[i])

    for j in range(nx-1,-1,-1):
        if j!=0:
            #solido
            Ts[j-1]=alfa_s[j]*Ts[j]+beta_s[j]
            #gas
            Tg[j-1]=alfa_g[j]*Tg[j]+beta_g[j]

        if j==0:
            break
    tiempo=nt/60 #[minutos]
    return Ts,Tg,d

```

```

[3]: from numba import njit #Numba es una biblioteca que optimiza códigos y ahorra
    ↪ mucho tiempo de iteración.
    #Optimización de función reducción
    reduccion_njit=njit(reduccion)

```

1.3 Gráficos y otros procedimientos

```

[4]: #Listas vacías de resultados a obtener
    dt=[2.0,0.2,0.02,0.002] #Intervalos de tiempo
    t_reaccion=[2.5, 5, 7.5, 10]
    Ts_dt=[[0,0,0,0],[0,0,0,0],[0,0,0,0],[0,0,0,0]]
    Tg_dt=[[0,0,0,0],[0,0,0,0],[0,0,0,0],[0,0,0,0]]
    d_dt=[[0,0,0,0],[0,0,0,0],[0,0,0,0],[0,0,0,0]]

```

```
[5]: for i in range(len(dt)):
      for j in range(len(t_reaccion)):
          resultados=reduccion_njit(dt[i],t_reaccion[j])
          print(i,j)
          Ts_dt[i][j]=resultados[0]
          Tg_dt[i][j]=resultados[1]
          d_dt[i][j]=resultados[2]
      print('Iteración número:', i)
```

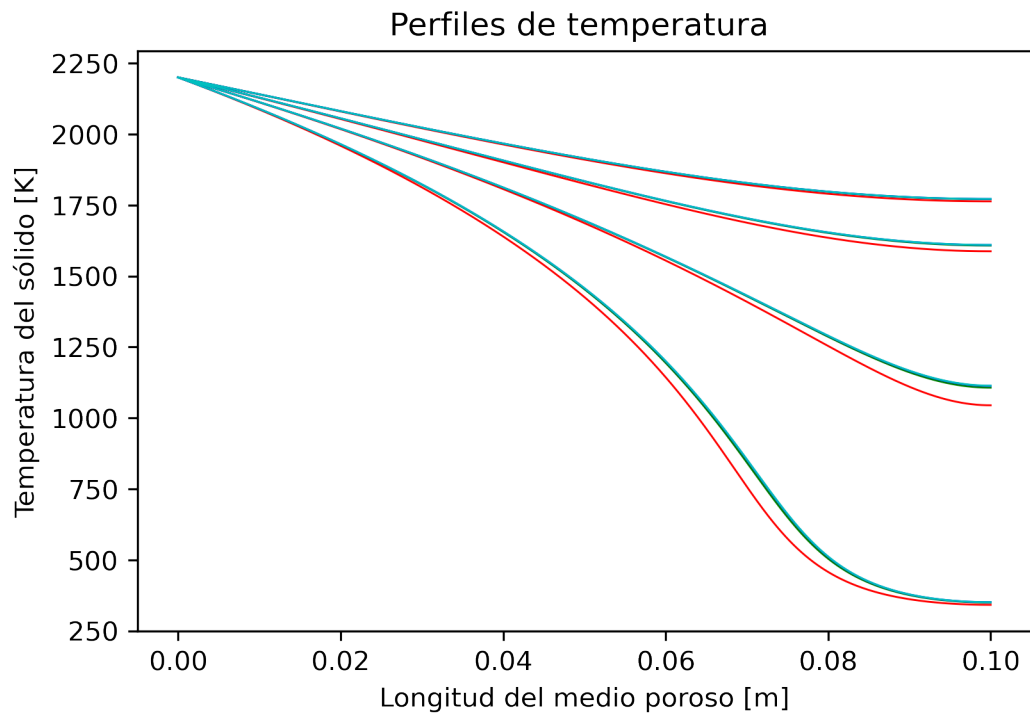
```
0 0
0 1
0 2
0 3
Iteración número: 0
1 0
1 1
1 2
1 3
Iteración número: 1
2 0
2 1
2 2
2 3
Iteración número: 2
3 0
3 1
3 2
3 3
Iteración número: 3
```

```
[10]: #Importes y gráfica de Ts
import matplotlib.pyplot as plt
plt.rcParams['figure.dpi'] = 200
lp=0.1 #Tengo que definirlo fuera de la función
nodos=800
colores=['r','g','b','c','m','k']
for i in range(len(t_reaccion)):
    for j in range(len(dt)):

        puntos=np.linspace(0,lp,nodos)
        plt.plot(puntos,Ts_dt[i][j],colores[i],linewidth=0.7)

#Datos extra
plt.title('Perfiles de temperatura ')
plt.xlabel('Longitud del medio poroso [m]')
plt.ylabel('Temperatura del sólido [K]')
#plt.legend(('tiempo=2.5 [min]', 'tiempo=5 [min]', 'tiempo=7.5 [min]', 'tiempo=10 [min]'), #Se agregó manualmente la leyenda correcta.
```

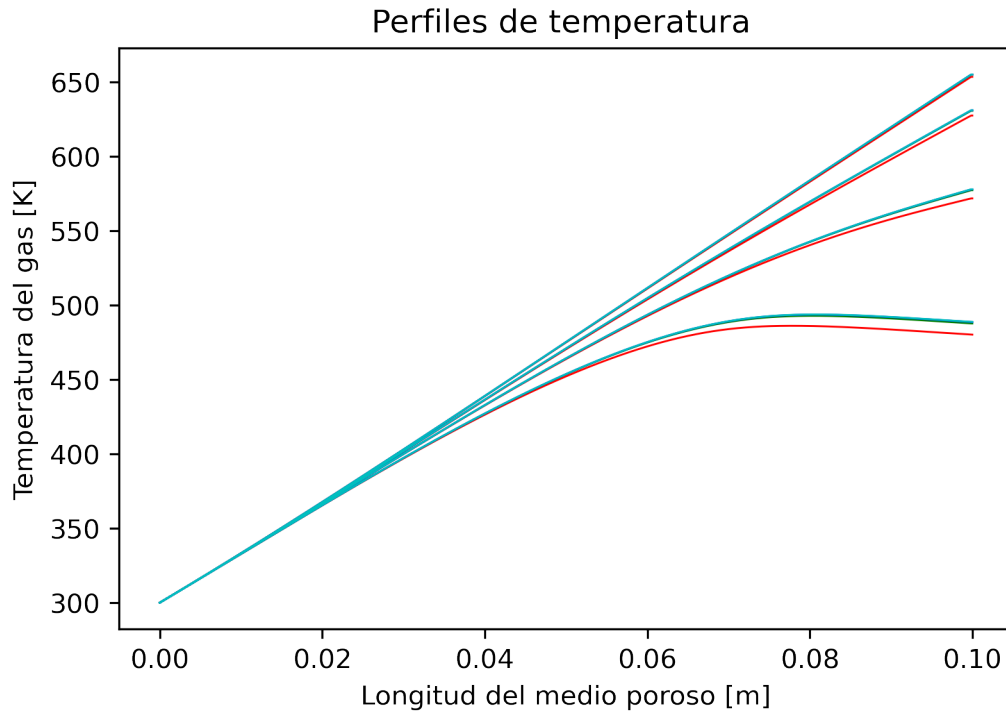
```
#prop = {'size': 7}, loc='lower left')
plt.show()
```



```
[13]: #Gráfica de Tg
for i in range(len(t_reaccion)):
    for j in range(len(dt)):

        puntos=np.linspace(0,lp,nodos)
        plt.plot(puntos,Tg_dt[i][j],colores[i],linewidth=0.7)

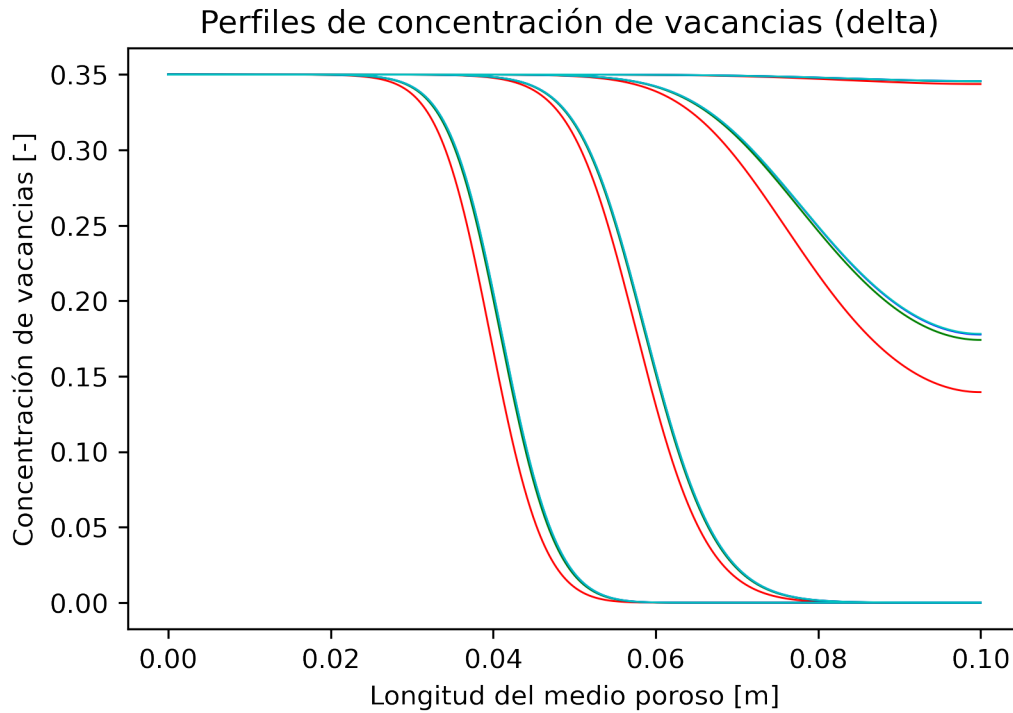
#Datos extra
plt.title('Perfiles de temperatura ')
plt.xlabel('Longitud del medio poroso [m]')
plt.ylabel('Temperatura del gas [K]')
#plt.legend(('dt=2 [s]','dt=0.2 [s]','dt=0.02 [s]','dt=0.002 [s]'),
#prop = {'size': 10}, loc='upper left')
plt.show()
```



```
[17]: #Gráfica de d
for i in range(len(t_reaccion)):
    for j in range(len(dt)):

        puntos=np.linspace(0,lp,nodos)
        plt.plot(puntos,d_dt[i][j],colores[i],linewidth=0.7)

#Datos extra
plt.title('Perfiles de concentración de vacancias (delta) ')
plt.xlabel('Longitud del medio poroso [m]')
plt.ylabel('Concentración de vacancias [-]')
#plt.legend(('dt=2 [s]', 'dt=0.2 [s]', 'dt=0.02 [s]', 'dt=0.002 [s]'),
            #prop = {'size': 10}, loc='lower left')
plt.show()
```



1.4 Función de Oxidación

```
[18]: #Definición de función oxidación
def oxidacion(dt,t_reaccion):
    def entalpia_ox(delta):
        if delta==0:
            entalpia=-430+285.85
        elif 395-31.4*np.log10(delta) > 430:
            entalpia=-430+285.85
        else:
            entalpia= -(395-31.4*np.log10(delta))+285.85
        return entalpia

    #.....Variables.....

    u_in=1 #[m/s] Velocidad inicial del gas.
    Ts_inicial=700 #[K] Temperatura inicial del sólido para la
    Ts_bordeo=1200 #[K] Temperatura del sólido en x=0 en la oxidación.
    Qce=1000000 #[W/m3] Aporte de calor volumétrico externo.
    nx=800 #[nodos] Cantidad de nodos espaciales.
    #dt=0.02 #[s] Intervalo de tiempo.
```

```

#t_reaccion=15 #[min] Tiempo de reacción.
#.....

#Constantes
dx=lp/nx #[-] Espaciamiento

#Condiciones de borde de temperatura y delta.

Ts_bordeo=float(Ts_bordeo) #Temperatura del solido en x=0 en la oxidación.
Tg_bordeo=400. #Temperatura del gas en x=0 en la oxidación.
d_bordeo=0 #Concentración de vacancias en el sólido en x=0 en la oxidación.

#####
#RESOLUCION DE ECUACIONES

#condiciones iniciales
d=reduccion_njit(dt,10)[2] #Las condiciones finales de la reducción se
↪ asignan aquí.
d_inicial=d.copy()
dox=d.copy()
dox[0]=0

Tsox=np.zeros(nx)+float(Ts_inicial)
Tsox[0]=Ts_bordeo
Tgox=np.zeros(nx)+300
Tgox[0]=Tg_bordeo

kox=np.zeros(nx) #mod eq vacancias

#se crean los arreglos de A, B, C, F, alfa y beta vacías, los cuales se
↪ usan para aplicar TDMA.
A_s=np.zeros(nx)
B_s=np.zeros(nx)
C_s=np.zeros(nx)
F_s=np.zeros(nx)
alfa_s=np.zeros(nx)
beta_s=np.zeros(nx)

A_g=np.zeros(nx)
B_g=np.zeros(nx)
C_g=np.zeros(nx)

```

```

F_g=np.zeros(nx)
alfa_g=np.zeros(nx)
beta_g=np.zeros(nx)

#condición final de reducción
dox_final=np.zeros(nx) #Asumimos que todo el material se oxidará

tiempo=0 #[min] tiempo de reacción
nt=0

while tiempo < t_reaccion:

    nt+=dt

    #sólido
    rho_s=(-0.0002*Tsox+6.8561)*1000 #7220 #[kg/m3]densidad del sólido *ref.
    ↪ Nelson 2014
    cp_s= (0.0599*np.log(Tsox)+0.039)*1000 #[J/kgK]calor específico del
    ↪sólido * ref. Nelson 2014
    lambda_s= (6.776*10**(-2)+2.793*10**(-4)*Tsox)**(-1)#[W/mK] Coeficiente
    ↪de transferencia de calor del sólido. ref. Nelson 2014
    lambda_ef= (1.-e)*lambda_s + (32.*sigma*dp*e*Tsox**3.)/(9.*(1.-e)) #[W/
    ↪mK] Coeficiente de transferencia de calor efectivo. ref. Toledo

    #gas
    #nota: La temperatura del vapor varía muy poco, por ende se
    ↪considerarán propiedades constantes.

    rho_h2o=np.zeros(nx)+1/1.673 #kg/m3 #1/v volumen específico
    cp_h2o=np.zeros(nx)+2.01*10**3 #j/kgK calor específico
    lambda_h2o=np.zeros(nx)+24.8*10**(-3) #W/mK conductividad térmica del
    ↪vapor
    mu_h2o= np.zeros(nx)+12.0*10**(-6)#Pa*s o kg/ms Viscosidad dinámica del
    ↪vapor
    nu_h2o=mu_h2o/rho_h2o #m**2/s viscosidad cinemática
    u_h2o=np.zeros(nx)+u_in

    #parámetros para a_ar. ref. Toledo
    ags_h2o=6./(1.-e) #superficie específica interna del medio poroso
    Pr_h2o=(mu_h2o*cp_h2o)/lambda_h2o #Número de Prandtl
    Re_h2o=(e*u_h2o*dp)/mu_h2o #Número de Reynolds
    hgs_h2o=(lambda_h2o/dp)*(2.0+1.1*Pr_h2o**(1/3)*Re_h2o**0.6)
    ↪#Coeficiente convectivo interfásico
    a_h2o = ags_h2o*hgs_h2o #Coeficiente intersticial de intercambio de
    ↪calor

```

```

#Cálculo de vacancias
dox_n=dox.copy() #d_n sería d anterior (d~n), d sería el valor actual
↪ d^(n+1)
kox=yox**gamma*(1-(d_inicial-dox_n)/(d_inicial))*Aox*np.exp(-Eox/
↪ (R*Tsox))
dox=(dox_n/dt)/(kox+1/dt) #como son estáticas no requiere recorrer x

#Cálculo de temperaturas
for i in range (nx):
    if i==0:
        #solido
        beta_s[i+1]=Ts_bordeo

        #gas
        beta_g[i+1]=Tg_bordeo

    elif i<(nx-1):
        #solido
        A_s[i]=(lambda_ef[i-1]+lambda_ef[i])/(2*dx**2)
        B_s[i]=(lambda_ef[i]+lambda_ef[i+1])/(2*dx**2)
        C_s[i]=(1-e)*(cp_s[i]*rho_s[i])/dt + a_h2o[i] + A_s[i]+B_s[i]
        F_s[i]=Tsox[i]*(1-e)*(cp_s[i]*rho_s[i])/dt +
↪ a_h2o[i]*Tgox[i]+Qce/nx+(dox[i]-dox_n[i])/dt*(cCe/
↪ nx)*(10**6)*entalpia_ox(d[i])

        alfa_s[i+1]=B_s[i]/(C_s[i]-A_s[i]*alfa_s[i])
        beta_s[i+1]=(F_s[i]+beta_s[i]*A_s[i])/(C_s[i]-A_s[i]*alfa_s[i])

        #gas
        A_g[i]=(lambda_h2o[i-1]+lambda_h2o[i])/
↪ (2*dx**2)+u_h2o[i]*cp_h2o[i]*rho_h2o[i]/dx
        B_g[i]=(lambda_h2o[i]+lambda_h2o[i+1])/(2*dx**2)
        C_g[i]=(cp_h2o[i]*rho_h2o[i])/dt + a_h2o[i]/e + A_g[i]+B_g[i]
        F_g[i]=Tgox[i]*(cp_h2o[i]*rho_h2o[i])/dt + a_h2o[i]*Tsox[i]/e

        alfa_g[i+1]=B_g[i]/(C_g[i]-A_g[i]*alfa_g[i])
        beta_g[i+1]=(F_g[i]+beta_g[i]*A_g[i])/(C_g[i]-A_g[i]*alfa_g[i])

    else:
        #solido
        Tsox[i]=beta_s[i]/(1-alfa_s[i])

        #gas
        Tgox[i]=beta_g[i]/(1-alfa_g[i])

```

```

    for j in range(nx-1,-1,-1):
        if j!=0:
            #solido
            Tsox[j-1]=alfa_s[j]*Tsox[j]+beta_s[j]
            #gas
            Tgox[j-1]=alfa_g[j]*Tgox[j]+beta_g[j]

        if j==0:

            break
    tiempo=nt/60
    return Tsox,Tgox,dox

```

```

[19]: #Optimización de función oxidación
oxidacion_njit=njit(oxidacion)

```

1.5 Gráficos y otros procedimientos

```

[20]: #Listas vacías de resultados a obtener

Tsox_dt=[[0,0,0,0],[0,0,0,0],[0,0,0,0],[0,0,0,0]]
Tgox_dt=[[0,0,0,0],[0,0,0,0],[0,0,0,0],[0,0,0,0]]
dox_dt=[[0,0,0,0],[0,0,0,0],[0,0,0,0],[0,0,0,0]]

```

```

[21]: #Calculo de oxidación
for i in range(len(dt)):
    for j in range(len(t_reaccion)):
        resultadosox=oxidacion_njit(dt[i],t_reaccion[j])
        print(i,j)
        Tsox_dt[i][j]=resultadosox[0]
        Tgox_dt[i][j]=resultadosox[1]
        dox_dt[i][j]=resultadosox[2]
    print('Iteración número:', i)

```

```

0 0
0 1
0 2
0 3
Iteración número: 0
1 0
1 1
1 2
1 3
Iteración número: 1
2 0
2 1

```

```

2 2
2 3
Iteración número: 2
3 0
3 1
3 2
3 3
Iteración número: 3

```

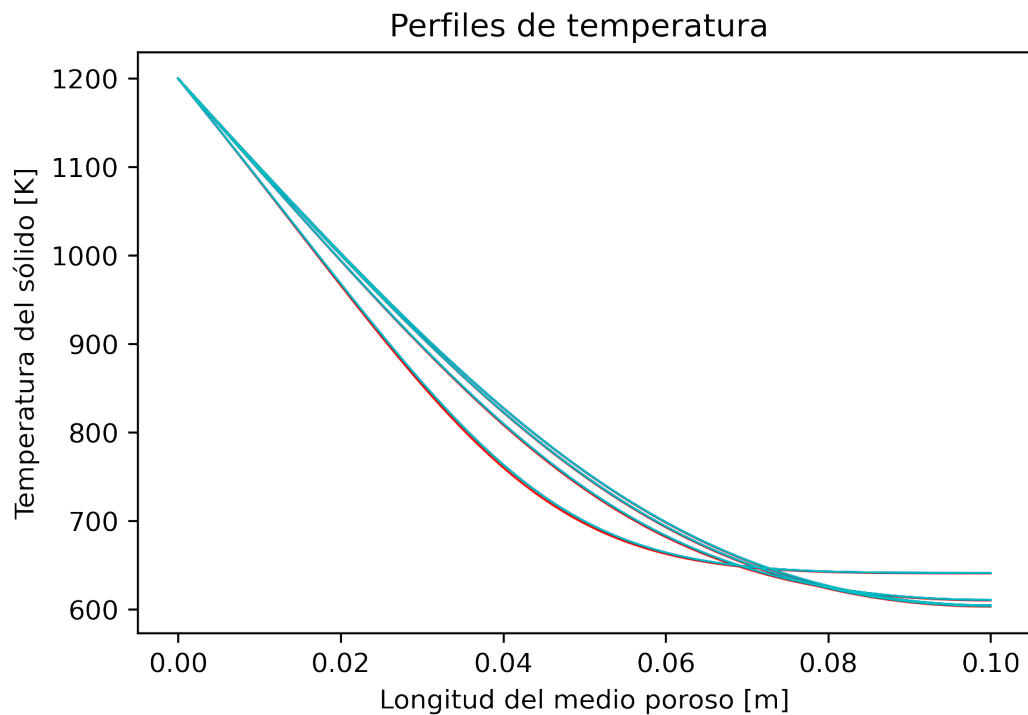
```

[22]: #Gráfica de Ts
for i in range(len(t_reaccion)):
    for j in range(len(dt)):

        puntos=np.linspace(0,lp,nodos)
        plt.plot(puntos,Tsox_dt[i][j],colores[i],linewidth=0.7)

#Datos extra
plt.title('Perfiles de temperatura ')
plt.xlabel('Longitud del medio poroso [m]')
plt.ylabel('Temperatura del sólido [K]')
#plt.legend(('dt=2 [s]','dt=0.2 [s]','dt=0.02 [s]','dt=0.002 [s]'), #Se agregó
→manualmente la leyenda correcta.
        #prop = {'size': 10}, loc='lower left')
plt.show()

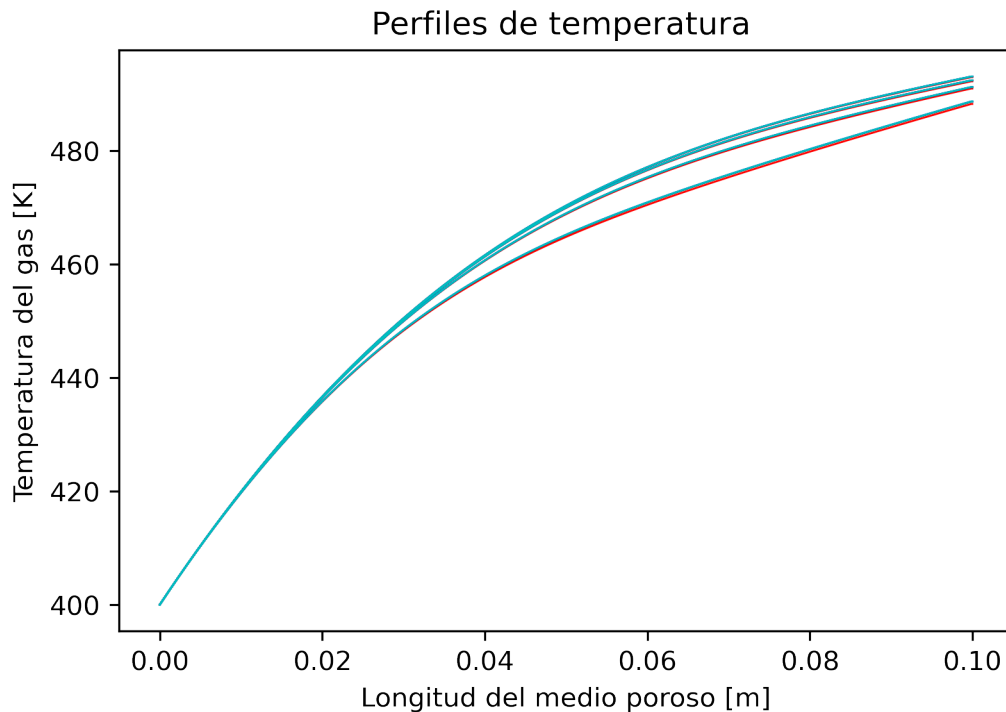
```



```
[23]: #Gráfica de Tg
for i in range(len(t_reaccion)):
    for j in range(len(dt)):

        puntos=np.linspace(0,lp,nodos)
        plt.plot(puntos,Tgox_dt[i][j],colores[i],linewidth=0.7)

#Datos extra
plt.title('Perfiles de temperatura ')
plt.xlabel('Longitud del medio poroso [m]')
plt.ylabel('Temperatura del gas [K]')
#plt.legend(('dt=2 [s]','dt=0.2 [s]','dt=0.02 [s]','dt=0.002 [s]'),
            #prop = {'size': 10}, loc='upper left')
plt.show()
```

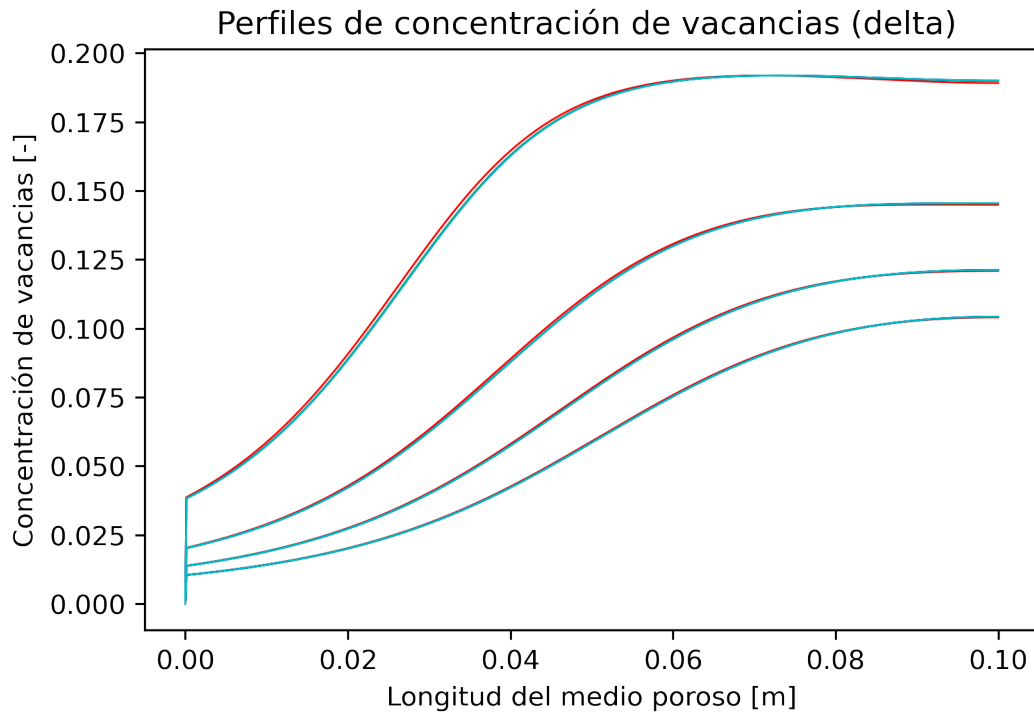


```
[24]: #Gráfica de d
for i in range(len(t_reaccion)):
    for j in range(len(dt)):

        puntos=np.linspace(0,lp,nodos)
        plt.plot(puntos,dox_dt[i][j],colores[i],linewidth=0.7)

#Datos extra
```

```
plt.title('Perfiles de concentración de vacancias (delta) ')
plt.xlabel('Longitud del medio poroso [m]')
plt.ylabel('Concentración de vacancias [-]')
#plt.legend(('dt=2 [s]', 'dt=0.2 [s]', 'dt=0.02 [s]', 'dt=0.002 [s]'),
#           #prop = {'size': 10}, loc='upper right')
plt.show()
```



Anexo B2

May 31, 2021

1 Modelo 1-D: determinación nx

1.1 Constantes

```
[2]: R=8.3145*10**(-3) # [kJ /mol K ] Constante universal de los gases.
Ared= 720000. #+-360000 [1/s] Factor pre-exponencial de reducción. Bulfin.
Ered=232. # +-5 [kJ/mol] Energía de activación de reducción. Bulfin.
lp=0.1 #[m] Longitud del medio poroso. ***DATO CÁMARA DE REACCIÓN ***
vol=7.3*10**(-4) #[m3] Volumen de la cámara de reacción. ***DATO CÁMARA DE
    ↪ REACCIÓN***
e=0.65 #[-] Porosidad del medio poroso.***DATO PELLETS BULFIN***
dp=0.01 #[m] Diámetro de partícula. ***DATO PELLETS BULFIN***
cCe=(1-e)*34.55 #[kmol/m3] Concentración de cerio. Cálculo.
molCeO2=(1-e)*30.6225 #=10.7179 Moles totales de CeO2.
sigma=5.67*10**(-8) #[W/m2K4] Constante de Stefan-Boltzmann.
#Cálculo moles totales CeO2
#7.22 g/cm3 * 730 cm3 / 172.115(g/mol)= 30.6225 moles de CeO2 totales.
yox=1 #Fracción molar de oxidante
Aox=1.0 # Factor pre-exponencial de oxidación. Arifin
Eox=29.0 # Energía de activación de oxidación. Arifin
gamma=0.89 #[-] Orden de reacción para la oxidación. Arifin
```

1.2 Función de Reducción

```
[ ]: #Definición de función reducción
import numpy as np
def reduccion(nx): #Los inputs se eligen desde el apartado 'variables' ubicado
    ↪ mas abajo. Al seleccionar se debe marcar #
    #Funciones entalpía de reacción
    def entalpia_red(delta): #Valores en kJ/mol
        if delta==0:
            entalpia=430
        elif 395-31.4*np.log10(delta) > 430:
            entalpia=430
        else:
            entalpia= 395-31.4*np.log10(delta)
        return entalpia
    #.....Variables.....
```

```

u_in=1. # [m/s] Velocidad inicial del gas.
Ts_border=2200 #[K] Temperatura del solido en x=0 en la reducci3n.
Qce=1000000 #[W/m3] Aporte de calor volum3trico externo.
#nx=800 #[nodos] Cantidad de nodos temporales.
dt=0.02 #[segundos] Intervalo de tiempo.
t_reaccion=5 #[minutos] Tiempo de reacci3n.
#.....
→

#Constantes
Tg_border=300. #Temperatura del gas en x=0 en la reducci3n.
d_border=0.35 #Concentraci3n de vacancias en el s3lido en x=0 en la
→reducci3n.
d_max=0.35 #Concentraci3n de vacancias m3xima.
dx=lp/nx #[-] Espaciamiento.
molxnodo=molCeO2/nx #Cantidad de moles de CeO2 por nodo.
#####
#RESOLUCI3N DE ECUACIONES
d=np.zeros(nx)
d[0]=d_border
d_final=np.zeros(nx)+0.35 #condici3n final de reducci3n (consideraci3n de
→reducci3n completa).
Ts=np.zeros(nx)+300
Ts[0]=Ts_border
Tg=np.zeros(nx)+300
Tg[0]=Tg_border
kred=np.zeros(nx)
kox=np.zeros(nx)
#se crean los arreglos de A, B, C, F, alfa y beta vac3as, los cuales se usan
→para aplicar TDMA.
A_s=np.zeros(nx)
B_s=np.zeros(nx)
C_s=np.zeros(nx)
F_s=np.zeros(nx)
alfa_s=np.zeros(nx)
beta_s=np.zeros(nx)
A_g=np.zeros(nx)
B_g=np.zeros(nx)
C_g=np.zeros(nx)
F_g=np.zeros(nx)
alfa_g=np.zeros(nx)
beta_g=np.zeros(nx)

tiempo=0 #[min] tiempo de reacci3n
nt=0
while tiempo < t_reaccion:
    nt+=dt #Contador para el tiempo de reacci3n

```

```

#Primero calculo todas las propiedades en función de la temperatura
→
#sólido
rho_s=(-0.0002*Ts+6.8561)*1000
cp_s= (0.0599*np.log(Ts)+0.0390)*1000
lambda_s= (6.7760*10**(-2)+2.7930*10**(-4)*Ts)**(-1)
lambda_ef= (1.-e)*lambda_s + (32.*sigma*dp*e*Ts**3.)/(9.*(1.-e))
#gas (argón)
rho_ar=1.603*300/Tg
cp_ar=(20.7860+Tg*2.8259*10**(-7)-Tg**2*1.4642*10**(-7)+Tg**3*1.
→0921*10**(-8)-Tg**(-1)*3.6614*10**(-8))*1000/39.948
lambda_ar=(0.0442*Tg + 4.7827)/1000
mu_ar= (0.0566*Tg+6.0645)*10**(-6)
nu_ar=mu_ar/rho_ar
u_ar=1.603*u_in/rho_ar
#parámetros para a_ar. ref. Toledo
ags_ar=6./(1.-e) #Superficie específica interna del medio poroso.
Pr_ar=(mu_ar*cp_ar)/lambda_ar #Número de Prandtl.
Re_ar=(e*u_ar*dp)/mu_ar #Número de Reynolds.
hgs_ar=(lambda_ar/dp)*(2.0+1.1*Pr_ar**(1/3)*Re_ar**0.6) #Coeficiente
→convectivo interfásico.
a_ar = ags_ar*hgs_ar #Coeficiente intersticial de intercambio de calor.
→
#Cálculo de vacancias
d_n=d.copy()
kred=(1-d_n/d_final)**(1/3)*Ared*np.exp(-Ered/(R*Ts)) #La temperatura
→todavía no se calcula para n+1, por tanto Ts corresponde a la temperatura en
→n.

d=(d_n/dt+d_max*kred)/(kred+kox+1/dt)

#Cálculo de temperaturas
for i in range (nx):
    if i==0:
        #sólido
        beta_s[i+1]=Ts_border

        #gas
        beta_g[i+1]=Tg_border

    elif i<(nx-1):
        #sólido
        A_s[i]=(lambda_ef[i-1]+lambda_ef[i])/(2*dx**2)
        B_s[i]=(lambda_ef[i]+lambda_ef[i+1])/(2*dx**2)
        C_s[i]=(1-e)*(cp_s[i]*rho_s[i])/dt + a_ar[i] + A_s[i]+B_s[i]

```

```

        F_s[i]=Ts[i]*(1-e)*(cp_s[i]*rho_s[i])/dt + a_ar[i]*Tg[i]+Qce/
        ↪ nx-(d[i]-d_n[i])/dt*(cCe/nx)*(10**6)*entalpia_red(d[i])

        alfa_s[i+1]=B_s[i]/(C_s[i]-A_s[i]*alfa_s[i])
        beta_s[i+1]=(F_s[i]+beta_s[i]*A_s[i])/(C_s[i]-A_s[i]*alfa_s[i])

        #gas
        A_g[i]=(lambda_ar[i-1]+lambda_ar[i])/
        ↪ (2*dx**2)+u_ar[i]*cp_ar[i]*rho_ar[i]/dx
        B_g[i]=(lambda_ar[i]+lambda_ar[i+1])/(2*dx**2)
        C_g[i]=(cp_ar[i]*rho_ar[i])/dt + a_ar[i]/e + A_g[i]+B_g[i]
        F_g[i]=Tg[i]*(cp_ar[i]*rho_ar[i])/dt + a_ar[i]*Ts[i]/e

        alfa_g[i+1]=B_g[i]/(C_g[i]-A_g[i]*alfa_g[i])
        beta_g[i+1]=(F_g[i]+beta_g[i]*A_g[i])/(C_g[i]-A_g[i]*alfa_g[i])

    else:
        #sólido
        Ts[i]=beta_s[i]/(1-alfa_s[i])

        #gas
        Tg[i]=beta_g[i]/(1-alfa_g[i])

    for j in range(nx-1,-1,-1):
        if j!=0:
            #sólido
            Ts[j-1]=alfa_s[j]*Ts[j]+beta_s[j]
            #gas
            Tg[j-1]=alfa_g[j]*Tg[j]+beta_g[j]

        if j==0:
            break
    tiempo=nt/60 #[minutos]
    return Ts,Tg,d

```

```

[4]: from numba import njit #Numba es una biblioteca que optimiza códigos y ahorra
    ↪ mucho tiempo de iteración.
    #Optimización de función reducción
    reduccion_njit=njit(reduccion)

```

1.3 Gráficos y otros procedimientos

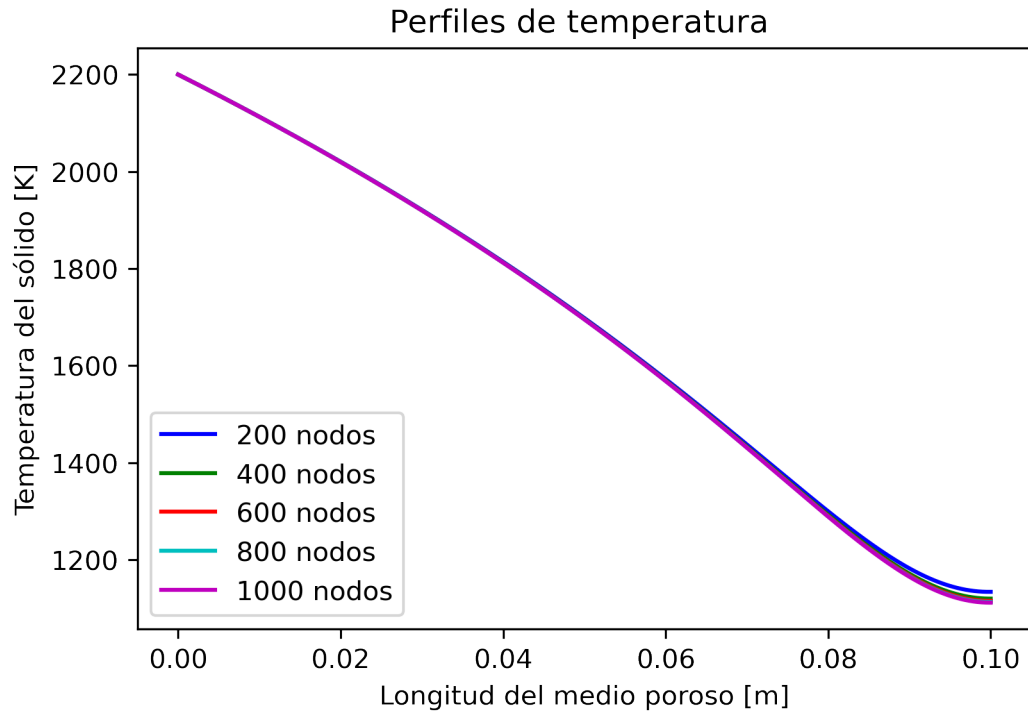
```
[5]: #Listas vacías de resultados a obtener
Ts_nx=[]
Tg_nx=[]
d_nx=[]
```

```
[6]: #Cálculo de reducción
nodos=[200, 400, 600, 800, 1000]
lp=0.1 #Tengo que definirlo fuera de la función
for i in range(len(nodos)):
    print(i)
    resultados=reduccion_njit(nodos[i])
    Ts_nx.append(resultados[0])
    Tg_nx.append(resultados[1])
    d_nx.append(resultados[2])
```

0
1
2
3
4

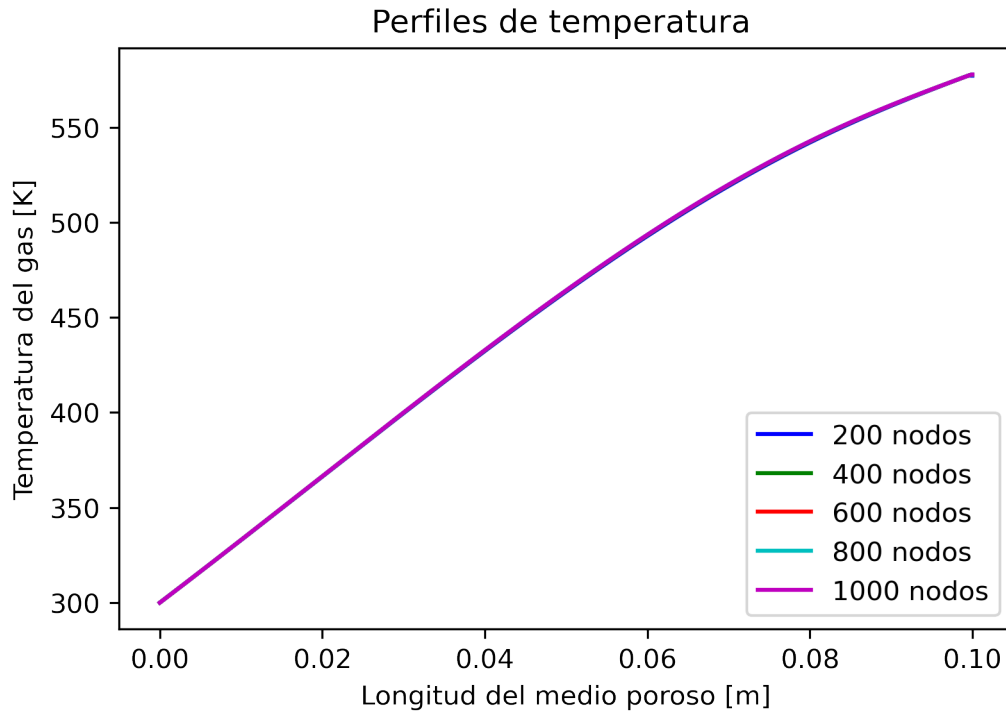
```
[7]: #Importes y gráfica de Ts
import matplotlib.pyplot as plt
plt.rcParams['figure.dpi'] = 300
puntos=[]
colores=['b','g','r','c','m','k']
for i in range(len(nodos)):
    puntos.append(np.linspace(0,lp,nodos[i]))
    plt.plot(puntos[i],Ts_nx[i],colores[i])

#Datos extra
plt.title('Perfiles de temperatura ')
plt.xlabel('Longitud del medio poroso [m]')
plt.ylabel('Temperatura del sólido [K]')
plt.legend(('200 nodos','400 nodos','600 nodos','800 nodos','1000 nodos'),
           prop = {'size': 10}, loc='lower left')
plt.show()
```



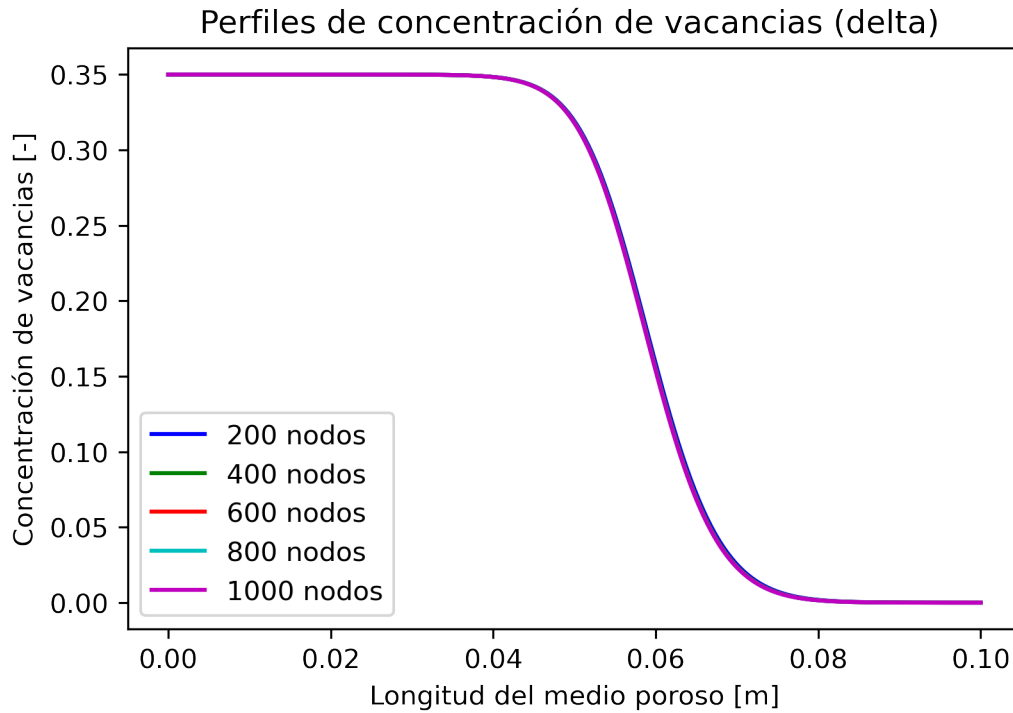
```
[8]: #Gráfica de Tg
colores=['b','g','r','c','m','k']
for i in range(len(nodos)):
    plt.plot(puntos[i],Tg_nx[i],colores[i])

#Datos extra
plt.title('Perfiles de temperatura ')
plt.xlabel('Longitud del medio poroso [m]')
plt.ylabel('Temperatura del gas [K]')
plt.legend(('200 nodos','400 nodos','600 nodos','800 nodos','1000 nodos'),
          prop = {'size': 10}, loc='lower right')
plt.show()
```



```
[9]: #Gráfica de d
colores=['b','g','r','c','m','k']
for i in range(len(nodos)):
    plt.plot(puntos[i],d_nx[i],colores[i])

#Datos extra
plt.title('Perfiles de concentración de vacancias (delta) ')
plt.xlabel('Longitud del medio poroso [m]')
plt.ylabel('Concentración de vacancias [-]')
plt.legend(('200 nodos','400 nodos','600 nodos','800 nodos','1000 nodos'),
           prop = {'size': 10}, loc='lower left')
plt.show()
```



1.4 Función de Oxidación

```
[10]: #Definición de función oxidación
def oxidacion(nx):
    def entalpia_ox(delta):
        if delta==0:
            entalpia=-430+285.85
        elif 395-31.4*np.log10(delta) > 430:
            entalpia=-430+285.85
        else:
            entalpia= -(395-31.4*np.log10(delta))+285.85
        return entalpia

    #.....Variables.....

    u_in=1 #[m/s] Velocidad inicial del gas.
    Ts_inicial=700 #[K] Temperatura inicial del sólido para la
    Ts_bordeo=1200 #[K] Temperatura del sólido en x=0 en la oxidación.
    Qce=1000000 #[W/m3] Aporte de calor volumétrico externo.
    #nx=800 #[nodos] Cantidad de nodos espaciales.
    dt=0.02 #[s] Intervalo de tiempo.
```

```

t_reaccion=5 #[min] Tiempo de reacción.
#.....

#Constantes
dx=lp/nx #[-] Espaciamiento

#Condiciones de borde de temperatura y delta.

Ts_bordeo=float(Ts_bordeo) #Temperatura del sólido en x=0 en la oxidación.
Tg_bordeo=400. #Temperatura del gas en x=0 en la oxidación.
d_bordeo=0 #Concentración de vacancias en el sólido en x=0 en la oxidación.

#####
#RESOLUCION DE ECUACIONES

#Condiciones iniciales
d=reduccion_njit(nx)[2] #Las condiciones finales de la reducción se asignan
→ aquí.
d_inicial=d.copy()
dox=d.copy()
dox[0]=0

Tsox=np.zeros(nx)+float(Ts_inicial)
Tsox[0]=Ts_bordeo
Tgox=np.zeros(nx)+300
Tgox[0]=Tg_bordeo

kox=np.zeros(nx)

A_s=np.zeros(nx)
B_s=np.zeros(nx)
C_s=np.zeros(nx)
F_s=np.zeros(nx)
alfa_s=np.zeros(nx)
beta_s=np.zeros(nx)

A_g=np.zeros(nx)
B_g=np.zeros(nx)
C_g=np.zeros(nx)
F_g=np.zeros(nx)
alfa_g=np.zeros(nx)

```

```

beta_g=np.zeros(nx)

dox_final=np.zeros(nx)

tiempo=0
nt=0

while tiempo < t_reaccion:

    nt+=dt

    #sólido
    rho_s=(-0.0002*Tsox+6.8561)*1000
    cp_s= (0.0599*np.log(Tsox)+0.039)*1000
    lambda_s= (6.776*10**(-2)+2.793*10**(-4)*Tsox)**(-1)
    lambda_ef= (1.-e)*lambda_s + (32.*sigma*dp*e*Tsox**3.)/(9.*(1.-e))

    #gas
    #nota: La temperatura del vapor varía muy poco, por ende se
    →considerarán propiedades constantes.

    rho_h2o=np.zeros(nx)+1/1.673 #kg/m3 #1/v Columna específico.
    cp_h2o=np.zeros(nx)+2.01*10**3 #j/kgK Calor específico.
    lambda_h2o=np.zeros(nx)+24.8*10**(-3) #W/mK Conductividad térmica del
    →vapor.
    mu_h2o= np.zeros(nx)+12.0*10**(-6) #Pa*s o kg/ms Viscosidad dinámica del
    →vapor.
    nu_h2o=mu_h2o/rho_h2o #m**2/s Viscosidad cinemática.
    u_h2o=np.zeros(nx)+u_in

    #parámetros para a_ar. ref. Toledo
    ags_h2o=6./(1.-e)
    Pr_h2o=(mu_h2o*cp_h2o)/lambda_h2o
    Re_h2o=(e*u_h2o*dp)/mu_h2o
    hgs_h2o=(lambda_h2o/dp)*(2.0+1.1*Pr_h2o**(1/3)*Re_h2o**0.6)
    a_h2o = ags_h2o*hgs_h2o

    #Cálculo de vacancias
    dox_n=dox.copy()
    kox=yox**gamma*(1-(d_inicial-dox_n)/(d_inicial))*Aox*np.exp(-Eox/
    →(R*Tsox))
    dox=(dox_n/dt)/(kox+1/dt)

    #Cálculo de temperaturas
    for i in range (nx):
        if i==0:

```

```

        #sólido
        beta_s[i+1]=Ts_bordeo

        #gas
        beta_g[i+1]=Tg_bordeo

    elif i<(nx-1):
        #sólido
        A_s[i]=(lambda_ef[i-1]+lambda_ef[i])/(2*dx**2)
        B_s[i]=(lambda_ef[i]+lambda_ef[i+1])/(2*dx**2)
        C_s[i]=(1-e)*(cp_s[i]*rho_s[i])/dt + a_h2o[i] + A_s[i]+B_s[i]
        F_s[i]=Tsox[i]*(1-e)*(cp_s[i]*rho_s[i])/dt +
        ↪a_h2o[i]*Tgox[i]+Qce/nx+(dox[i]-dox_n[i])/dt*(cCe/
        ↪nx)*(10**6)*entalpia_ox(d[i])

        alfa_s[i+1]=B_s[i]/(C_s[i]-A_s[i]*alfa_s[i])
        beta_s[i+1]=(F_s[i]+beta_s[i]*A_s[i])/(C_s[i]-A_s[i]*alfa_s[i])

        #gas
        A_g[i]=(lambda_h2o[i-1]+lambda_h2o[i])/
        ↪(2*dx**2)+u_h2o[i]*cp_h2o[i]*rho_h2o[i]/dx
        B_g[i]=(lambda_h2o[i]+lambda_h2o[i+1])/(2*dx**2)
        C_g[i]=(cp_h2o[i]*rho_h2o[i])/dt + a_h2o[i]/e + A_g[i]+B_g[i]
        F_g[i]=Tgox[i]*(cp_h2o[i]*rho_h2o[i])/dt + a_h2o[i]*Tsox[i]/e

        alfa_g[i+1]=B_g[i]/(C_g[i]-A_g[i]*alfa_g[i])
        beta_g[i+1]=(F_g[i]+beta_g[i]*A_g[i])/(C_g[i]-A_g[i]*alfa_g[i])

    else:
        #sólido
        Tsox[i]=beta_s[i]/(1-alfa_s[i])

        #gas
        Tgox[i]=beta_g[i]/(1-alfa_g[i])

for j in range(nx-1,-1,-1):
    if j!=0:
        #sólido
        Tsox[j-1]=alfa_s[j]*Tsox[j]+beta_s[j]
        #gas
        Tgox[j-1]=alfa_g[j]*Tgox[j]+beta_g[j]

    if j==0:

```

```

        break
    tiempo=nt/60
    return Tsox,Tgox,dox

```

```

[11]: #Optimización de función oxidación
oxidacion_njit=njit(oxidacion)

```

1.5 Gráficos y otros procedimientos

```

[12]: #Listas vacías de resultados a obtener
Tsox_nx=[]
Tgox_nx=[]
dox_nx=[]

```

```

[13]: #Calculo de oxidación
nodos=[200, 400, 600, 800, 1000]
lp=0.1 #Tengo que definirlo fuera de la función
for i in range(len(nodos)):
    print (i)
    #cálculo oxidación
    resultadosox=oxidacion_njit(nodos[i])
    Tsox_nx.append(resultadosox[0])
    Tgox_nx.append(resultadosox[1])
    dox_nx.append(resultadosox[2])

```

```

0
1
2
3
4

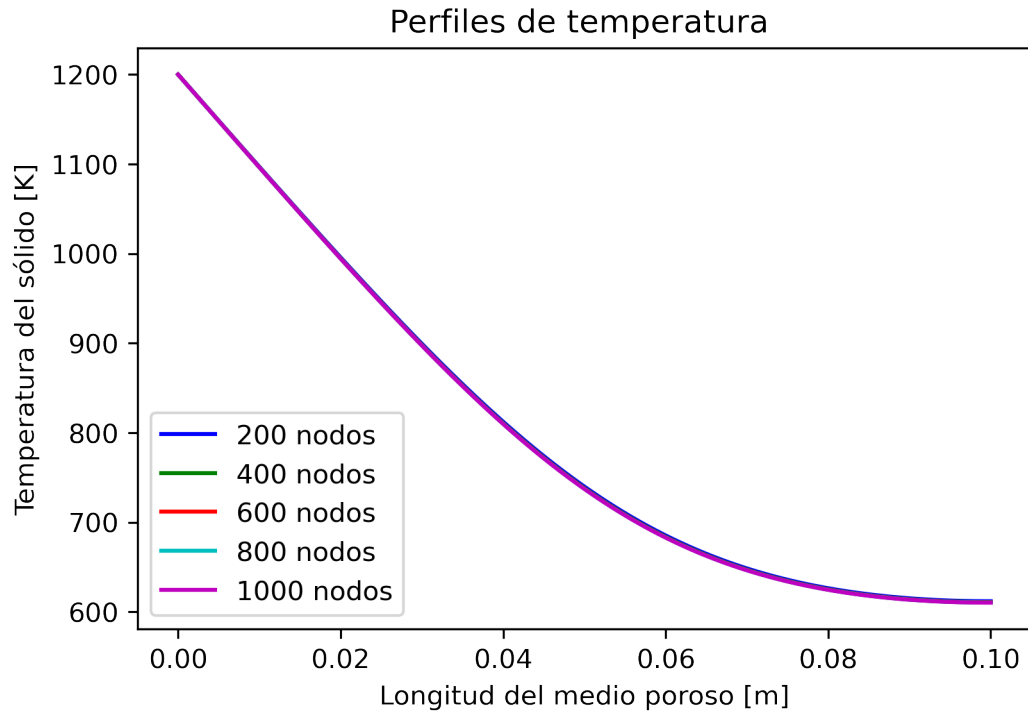
```

```

[14]: #Gráfica de Ts
for i in range(len(nodos)):
    puntos.append(np.linspace(0,lp,nodos[i]))
    plt.plot(puntos[i],Tsox_nx[i],colores[i])

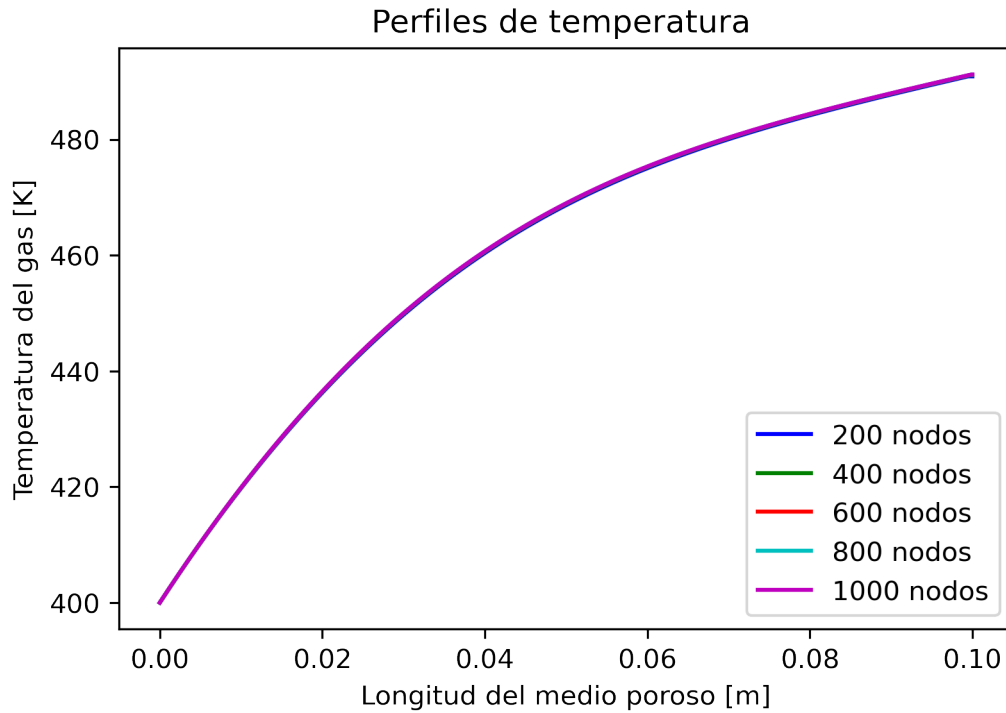
#Datos extra
plt.title('Perfiles de temperatura ')
plt.xlabel('Longitud del medio poroso [m]')
plt.ylabel('Temperatura del sólido [K]')
plt.legend(('200 nodos','400 nodos','600 nodos','800 nodos','1000 nodos'),
           prop = {'size': 10}, loc='lower left')
plt.show()

```



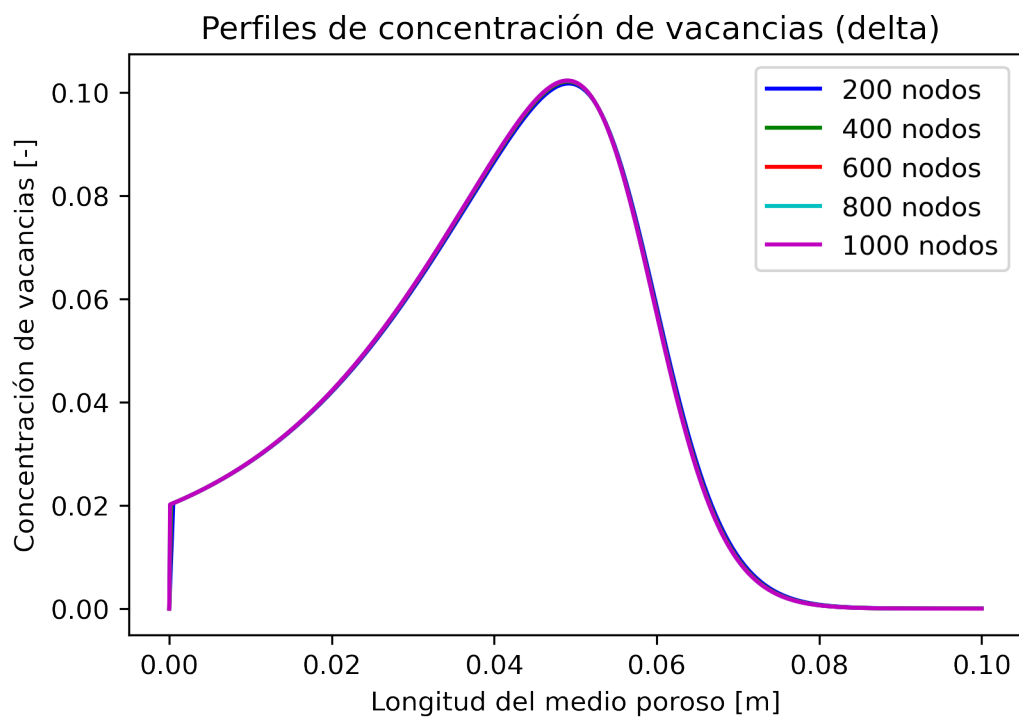
```
[15]: #Gráfica de Tg
colores=['b','g','r','c','m','k']
for i in range(len(nodos)):
    plt.plot(puntos[i],Tgox_nx[i],colores[i])

#Datos extra
plt.title('Perfiles de temperatura ')
plt.xlabel('Longitud del medio poroso [m]')
plt.ylabel('Temperatura del gas [K]')
plt.legend(('200 nodos','400 nodos','600 nodos','800 nodos','1000 nodos'),
           prop = {'size': 10}, loc='lower right')
plt.show()
```



```
[17]: #Gráfica de d
colores=['b','g','r','c','m','k']
for i in range(len(nodos)):
    plt.plot(puntos[i],dox_nx[i],colores[i])

#Datos extra
plt.title('Perfiles de concentración de vacancias (delta) ')
plt.xlabel('Longitud del medio poroso [m]')
plt.ylabel('Concentración de vacancias [-]')
plt.legend(('200 nodos','400 nodos','600 nodos','800 nodos','1000 nodos'),
           prop = {'size': 10}, loc='upper right')
plt.show()
```



Anexo B3

May 31, 2021

1 Modelo 1-D: Casos extremos

1.1 Constantes

```
[1]: R=8.3145*10**(-3) # [kJ /mol K ] Constante universal de los gases.
Ared= 720000. #+-360000 [1/s] Factor pre-exponencial de reducción. Bulfin
Ered=232. # +-5 [kJ/mol] Energía de activación de reducción. Bulfin
lp=0.1 # [m] Longitud del medio poroso. ***DATO CÁMARA DE REACCIÓN ***
vol=7.3*10**(-4) # [m3] Volumen de la cámara de reacción. ***DATO CÁMARA DE
    ↳ REACCIÓN
e=0.65 # [-] Porosidad del medio poroso. ***DATO PELLETS BULFIN***
dp=0.01 # [m] diámetro de partícula ***DATO PELLETS BULFIN***
cCe=(1-e)*34.55 # [kmol/m3] concentración de cerio. Cálculo
molCeO2=(1-e)*30.6225 # =10.7179 moles totales de CeO2 considerando pellets de
    ↳ CeO2.
sigma=5.67*10**(-8) # [W/m2K4] Constante de Stefan-Boltzmann.

#Cálculo moles totales CeO2
#7.22 g/cm3 * 730 cm3 / 172.115(g/mol)= 30.6225 moles de CeO2 totales (si todo
    ↳ el volumen fuera CeO2)

yox=1 #Fracción molar de oxidante
Aox=1.0 # Factor pre-exponencial de oxidación. Arifin
Eox=29.0 # Energía de activación de oxidación. Arifin
gamma=0.89 # [-] Orden de reacción para la oxidación. Arifin
```

1.2 Reducción

```
[2]: import numpy as np
def reduccion(Qce,u_in,Ts_border,t_reaccion): #Los inputs se eligen desde el
    ↳ apartado 'variables' ubicado mas abajo. Al seleccionar se debe marcar con #

    #Funciones entalpía de reacción
    def entalpia_red(delta): #Valores EN kJ/mol

        if delta==0:
            entalpia=430
        elif 395-31.4*np.log10(delta) > 430:
```

```

        entalpia=430
    else:
        entalpia= 395-31.4*np.log10(delta)
    return entalpia

#.....Variables.....
#Qce=1000000 #[W/m3] Aporte de calor volumétrico externo.
#u_in=1. # [m/s] Velocidad inicial del gas
#Ts_border=2100 #[K] Temperatura del solido en x=0 en la reducción.

nx=800 #[nodos] Cantidad de nodos temporales
dt=0.02 #[segundos] Intervalo de tiempo
#t_reaccion=15 #[minutos] Tiempo de reacción.
#.....

#Constantes
Tg_border=300.#Temperatura del gas en x=0 en la reducción.
d_border=0.35 #Concentracion de vacancias en el sólido en x=0 en la
↪reducción.
d_max=0.35 #Concentración de vacancias máxima.
dx=lp/nx #[-] Espaciamiento
molxnodo=molCeO2/nx #cantidad de moles de CeO2 por nodo.

#####
#RESOLUCION DE ECUACIONES

#condiciones iniciales
d=np.zeros(nx)
d[0]=d_border
d_final=np.zeros(nx)+0.35 #condición final de reducción (consideración de
↪reducción completa)

Ts=np.zeros(nx)+300
Ts[0]=Ts_border
Tg=np.zeros(nx)+300
Tg[0]=Tg_border

kred=np.zeros(nx) #mod eq vacancias
kox=np.zeros(nx) #mod eq vacancias

#se crean los arreglos de A, B, C, F, alfa y beta vacías, los cuales se usan
↪para aplicar TDMA.
A_s=np.zeros(nx)
B_s=np.zeros(nx)
C_s=np.zeros(nx)

```

```

F_s=np.zeros(nx)
alfa_s=np.zeros(nx)
beta_s=np.zeros(nx)

A_g=np.zeros(nx)
B_g=np.zeros(nx)
C_g=np.zeros(nx)
F_g=np.zeros(nx)
alfa_g=np.zeros(nx)
beta_g=np.zeros(nx)

tiempo=0 #[min] tiempo de reacción
nt=0

while tiempo < t_reaccion:

    nt+=dt #contador para el tiempo de reacción

    #primero calculo todas las propiedades en función de la temperatura

    #sólido
    rho_s=(-0.0002*Ts+6.8561)*1000 #7220 #[kg/m3]densidad del sólido *ref. Nelson 2014
    cp_s= (0.0599*np.log(Ts)+0.0390)*1000 #[J/kgK]calor específico del sólido * ref. Nelson 2014
    lambda_s= (6.7760*10**(-2)+2.7930*10**(-4)*Ts)**(-1)#[W/mK] Coeficiente de transferencia de calor del sólido. ref. Nelson 2014
    lambda_ef= (1.-e)*lambda_s + (32.*sigma*dp*e*Ts**3.)/(9.*(1.-e)) #[W/mK] Coeficiente de transferencia de calor efectivo. ref. Toledo

    #gas (argón)
    rho_ar=1.603*300/Tg #[kg/m3]. Gases ideales
    cp_ar=(20.7860+Tg*2.8259*10**(-7)-Tg**2*1.4642*10**(-7)+Tg**3*1.0921*10**(-8)-Tg**(-1)*3.6614*10**(-8))*1000/39.948 #[J/kgK]. ref. webbook. nist.gob. Chase 1998
    lambda_ar=(0.0442*Tg + 4.7827)/1000 #[W/Km] Conductividad térmica. Ajuste Younglove 1986.
    mu_ar= (0.0566*Tg+6.0645)*10**(-6) ##[Pa*s] viscosidad dinámica del argón. Ajuste Younglove 1986.

```

```

nu_ar=mu_ar/rho_ar # viscosidad cinematica del argon
u_ar=1.603*u_in/rho_ar # m/s velocidad del argon (depende de la
→ densidad)

#parámetros para a_ar. ref. Toledo
ags_ar=6./(1.-e) #superficie específica interna del medio poroso
Pr_ar=(mu_ar*cp_ar)/lambda_ar #Número de Prandtl
Re_ar=(e*u_ar*dp)/mu_ar #Número de Reynolds
hgs_ar=(lambda_ar/dp)*(2.0+1.1*Pr_ar**(1/3)*Re_ar**0.6) #Coeficiente
→ convectivo interfásico
a_ar = ags_ar*hgs_ar #Coeficiente intersticial de intercambio de calor

#Cálculo de vacancias
d_n=d.copy() #d_n sería d anterior (d^n), d sería el valor actual
→ d^(n+1)

kred=(1-d_n/d_final)**(1/3)*Ared*np.exp(-Ered/(R*Ts)) #La temperatura
→ todavía no se calcula para n+1, por tanto Ts corresponde a la temperatura en
→ n.

#kox=y_ox**gamma*(1-d_n/d_final)*Aox*np.exp(-Eox/(R*Ts)) Para el caso
→ de la reducción será cero en la reducción

d=(d_n/dt+d_max*kred)/(kred+kox+1/dt) #como son estáticas no requiere
→ recorrer x

#Cálculo de temperaturas
for i in range (nx):
    if i==0:
        #solido
        beta_s[i+1]=Ts_border

        #gas
        beta_g[i+1]=Tg_border

    elif i<(nx-1):
        #solido
        A_s[i]=(lambda_ef[i-1]+lambda_ef[i])/(2*dx**2)
        B_s[i]=(lambda_ef[i]+lambda_ef[i+1])/(2*dx**2)
        C_s[i]=(1-e)*(cp_s[i]*rho_s[i])/dt + a_ar[i] + A_s[i]+B_s[i]
        F_s[i]=Ts[i]*(1-e)*(cp_s[i]*rho_s[i])/dt + a_ar[i]*Tg[i]+Qce/
→ nx-(d[i]-d_n[i])/dt*(cCe/nx)*(10**6)*entalpia_red(d[i])

        alfa_s[i+1]=B_s[i]/(C_s[i]-A_s[i]*alfa_s[i])
        beta_s[i+1]=(F_s[i]+beta_s[i]*A_s[i])/(C_s[i]-A_s[i]*alfa_s[i])

```

```

        #gas
        A_g[i]=(lambda_ar[i-1]+lambda_ar[i])/
        ↪(2*dx**2)+u_ar[i]*cp_ar[i]*rho_ar[i]/dx
        B_g[i]=(lambda_ar[i]+lambda_ar[i+1])/(2*dx**2)
        C_g[i]=(cp_ar[i]*rho_ar[i])/dt + a_ar[i]/e + A_g[i]+B_g[i]
        F_g[i]=Tg[i]*(cp_ar[i]*rho_ar[i])/dt + a_ar[i]*Ts[i]/e

        alfa_g[i+1]=B_g[i]/(C_g[i]-A_g[i]*alfa_g[i])
        beta_g[i+1]=(F_g[i]+beta_g[i]*A_g[i])/(C_g[i]-A_g[i]*alfa_g[i])

    else:
        #solido
        Ts[i]=beta_s[i]/(1-alfa_s[i])

        #gas
        Tg[i]=beta_g[i]/(1-alfa_g[i])

    for j in range(nx-1,-1,-1):
        if j!=0:
            #solido
            Ts[j-1]=alfa_s[j]*Ts[j]+beta_s[j]
            #gas
            Tg[j-1]=alfa_g[j]*Tg[j]+beta_g[j]

        if j==0:

            break
    tiempo=nt/60 #[minutos]

    #Cálculo de producción de oxígeno

    prod_o2_nodo=molxnodo*d/2 #producción de oxígeno por nodo
    prod_o2=np.sum(prod_o2_nodo) #producción total de oxígeno
    ##ecuación de producción de o2
    return Ts,Tg,d,prod_o2

```

```

[3]: from numba import njit #Numba es una biblioteca que optimiza códigos y ahorra
    ↪mucho tiempo de iteración.
    reduccion_njit=njit(reduccion)

```

1.3 Oxidación

```
[4]: def oxidacion(Qce,u_in,Ts_bordeo,Ts_inicial,t_reaccion,Qred,ured,Tred,tred):  
    ↪#reduccion_tuple es una tupla con los parámetros de entrada de reducción  
    def entalpia_ox(delta):  
        if delta==0:  
            entalpia=-430+285.85  
        elif 395-31.4*np.log10(delta) > 430:  
            entalpia=-430+285.85  
        else:  
            entalpia= -(395-31.4*np.log10(delta))+285.85  
        return entalpia  
  
    #.....Variables.....  
    #=1000000 #[W/m3] Aporte de calor volumétrico externo.  
    #u_in=1 #[m/s] Velocidad inicial del gas.  
    #Ts_bordeo=1200 #[K] Temperatura del sólido en x=0 en la oxidación.  
    #Ts_inicial=700 #[K] Temperatura inicial del sólido para la  
  
    nx=800 #[nodos] Cantidad de nodos espaciales.  
    dt=0.02 #[s] Intervalo de tiempo.  
    #t_reaccion=15 #[min] Tiempo de reacción.  
    #.....  
  
    #Constantes  
    dx=lp/nx #[-] Espaciamiento  
    molxnodo=molCeO2/nx #cantidad de moles de CeO2 por nodo.  
  
    #Condiciones de borde de temperatura y delta.  
  
    Ts_bordeo=float(Ts_bordeo) #Temperatura del solido en x=0 en la oxidación.  
    Tg_bordeo=400. #Temperatura del gas en x=0 en la oxidación.  
    d_bordeo=0 #Concentración de vacancias en el sólido en x=0 en la oxidación.  
  
    #####  
    #RESOLUCION DE ECUACIONES  
  
    #condiciones iniciales  
    d=reduccion_njit(Qred,ured,Tred,tred)[2] #Las condiciones finales de la  
    ↪reducción se asignan aquí.  
    d_inicial=d.copy()  
    dox=d.copy()  
    dox[0]=0
```

```

Tsox=np.zeros(nx)+float(Ts_inicial)
Tsox[0]=Ts_bordeo
Tgox=np.zeros(nx)+300
Tgox[0]=Tg_bordeo

kox=np.zeros(nx) #mod eq vacancias

#se crean los arreglos de A, B, C, F, alfa y beta vacías, los cuales se
→usan para aplicar TDMA.
A_s=np.zeros(nx)
B_s=np.zeros(nx)
C_s=np.zeros(nx)
F_s=np.zeros(nx)
alfa_s=np.zeros(nx)
beta_s=np.zeros(nx)

A_g=np.zeros(nx)
B_g=np.zeros(nx)
C_g=np.zeros(nx)
F_g=np.zeros(nx)
alfa_g=np.zeros(nx)
beta_g=np.zeros(nx)

#condición final de reducción
dox_final=np.zeros(nx) #Asumimos que todo el material se oxidará

tiempo=0 #[min] tiempo de reacción
nt=0

while tiempo < t_reaccion:

    nt+=dt

    #sólido
    rho_s=(-0.0002*Tsox+6.8561)*1000 #7220 #[kg/m3]densidad del sólido *ref.
→Nelson 2014
    cp_s= (0.0599*np.log(Tsox)+0.039)*1000 #[J/kgK]calor específico del
→sólido * ref. Nelson 2014
    lambda_s= (6.776*10**(-2)+2.793*10**(-4)*Tsox)**(-1)#[W/mK] Coeficiente
→de transferencia de calor del sólido. ref. Nelson 2014
    lambda_ef= (1.-e)*lambda_s + (32.*sigma*dp*e*Tsox**3.)/(9.*(1.-e)) #[W/
→mK] Coeficiente de transferencia de calor efectivo. ref. Toledo

```

```

#gas
#nota: La temperatura del vapor varía muy poco, por ende se
→considerarán propiedades constantes.

rho_h2o=np.zeros(nx)+1/1.673 #kg/m3 #1/v volumen específico
cp_h2o=np.zeros(nx)+2.01*10**3 #j/kgK calor específico
lambda_h2o=np.zeros(nx)+24.8*10**(-3) #W/mK conductividad térmica del
→vapor
mu_h2o= np.zeros(nx)+12.0*10**(-6)#Pa*s o kg/ms Viscosidad dinámica del
→vapor
nu_h2o=mu_h2o/rho_h2o #m**2/s viscosidad cinemática
u_h2o=np.zeros(nx)+u_in

#parámetros para a_ar. ref. Toledo
ags_h2o=6./(1.-e) #superficie específica interna del medio poroso
Pr_h2o=(mu_h2o*cp_h2o)/lambda_h2o #Número de Prandtl
Re_h2o=(e*u_h2o*dp)/mu_h2o #Número de Reynolds
hgs_h2o=(lambda_h2o/dp)*(2.0+1.1*Pr_h2o**(1/3)*Re_h2o**0.6)
→#Coeficiente convectivo interfásico
a_h2o = ags_h2o*hgs_h2o #Coeficiente intersticial de intercambio de
→calor

#Cálculo de vacancias
dox_n=dox.copy() #d_n sería d anterior (d~n), d sería el valor actual
→d^(n+1)
kox=yox**gamma*(1-(d_inicial-dox_n)/(d_inicial))*Aox*np.exp(-Eox/
→(R*Tsox))
dox=(dox_n/dt)/(kox+1/dt) #como son estáticas no requiere recorrer x

#Cálculo de temperaturas
for i in range (nx):
    if i==0:
        #solido
        beta_s[i+1]=Ts_bordeo

        #gas
        beta_g[i+1]=Tg_bordeo

    elif i<(nx-1):
        #solido
        A_s[i]=(lambda_ef[i-1]+lambda_ef[i])/(2*dx**2)
        B_s[i]=(lambda_ef[i]+lambda_ef[i+1])/(2*dx**2)
        C_s[i]=(1-e)*(cp_s[i]*rho_s[i])/dt + a_h2o[i] + A_s[i]+B_s[i]

```

```

        F_s[i]=Tsox[i]*(1-e)*(cp_s[i]*rho_s[i])/dt + u
        a_h2o[i]*Tgox[i]+Qce/nx+(dox[i]-dox_n[i])/dt*(cCe/
        nx)*(10**6)*entalpia_ox(d[i])

        alfa_s[i+1]=B_s[i]/(C_s[i]-A_s[i]*alfa_s[i])
        beta_s[i+1]=(F_s[i]+beta_s[i]*A_s[i])/(C_s[i]-A_s[i]*alfa_s[i])

        #gas
        A_g[i]=(lambda_h2o[i-1]+lambda_h2o[i])/
        (2*dx**2)+u_h2o[i]*cp_h2o[i]*rho_h2o[i]/dx
        B_g[i]=(lambda_h2o[i]+lambda_h2o[i+1])/(2*dx**2)
        C_g[i]=(cp_h2o[i]*rho_h2o[i])/dt + a_h2o[i]/e + A_g[i]+B_g[i]
        F_g[i]=Tgox[i]*(cp_h2o[i]*rho_h2o[i])/dt + a_h2o[i]*Tsox[i]/e

        alfa_g[i+1]=B_g[i]/(C_g[i]-A_g[i]*alfa_g[i])
        beta_g[i+1]=(F_g[i]+beta_g[i]*A_g[i])/(C_g[i]-A_g[i]*alfa_g[i])

    else:
        #solido
        Tsox[i]=beta_s[i]/(1-alfa_s[i])

        #gas
        Tgox[i]=beta_g[i]/(1-alfa_g[i])

    for j in range(nx-1,-1,-1):
        if j!=0:
            #solido
            Tsox[j-1]=alfa_s[j]*Tsox[j]+beta_s[j]
            #gas
            Tgox[j-1]=alfa_g[j]*Tgox[j]+beta_g[j]

        if j==0:
            break
    tiempo=nt/60

    #Cálculo de producción de oxígeno
    prod_h2_nodo=molxnodo*(d-dox) #producción de hidrógeno por nodo
    prod_h2=np.sum(prod_h2_nodo) #producción total de hidrógeno
    return Tsox,Tgox,dox,prod_h2

```

```
[5]: oxidacion_njit=njit(oxidacion)
```

1.4 Perfiles de temperatura y vacancias para casos extremos

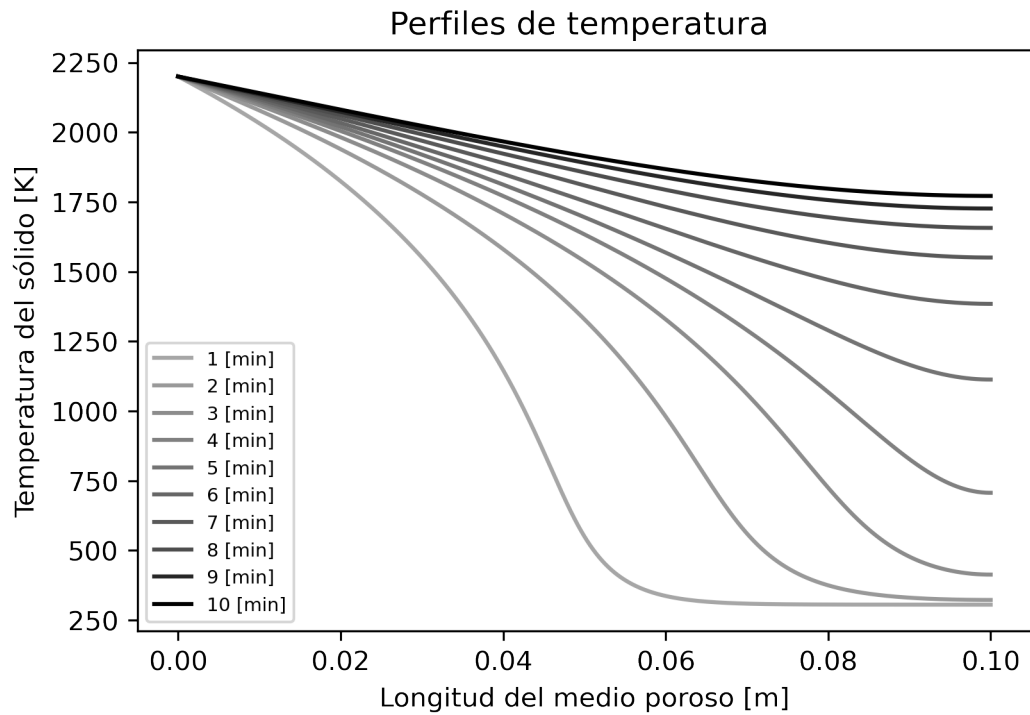
1.4.1 Caso más favorable:

Reduccion

```
[6]: t_reac=[1,2,3,4,5,6,7,8,9,10]
Ts_red=[]
Tg_red=[]
d_red=[]
for i in range(len(t_reac)):
    resultados=reduccion_njit(10**6,1,2200,t_reac[i])
    Ts_red.append(resultados[0])
    Tg_red.append(resultados[1])
    d_red.append(resultados[2])

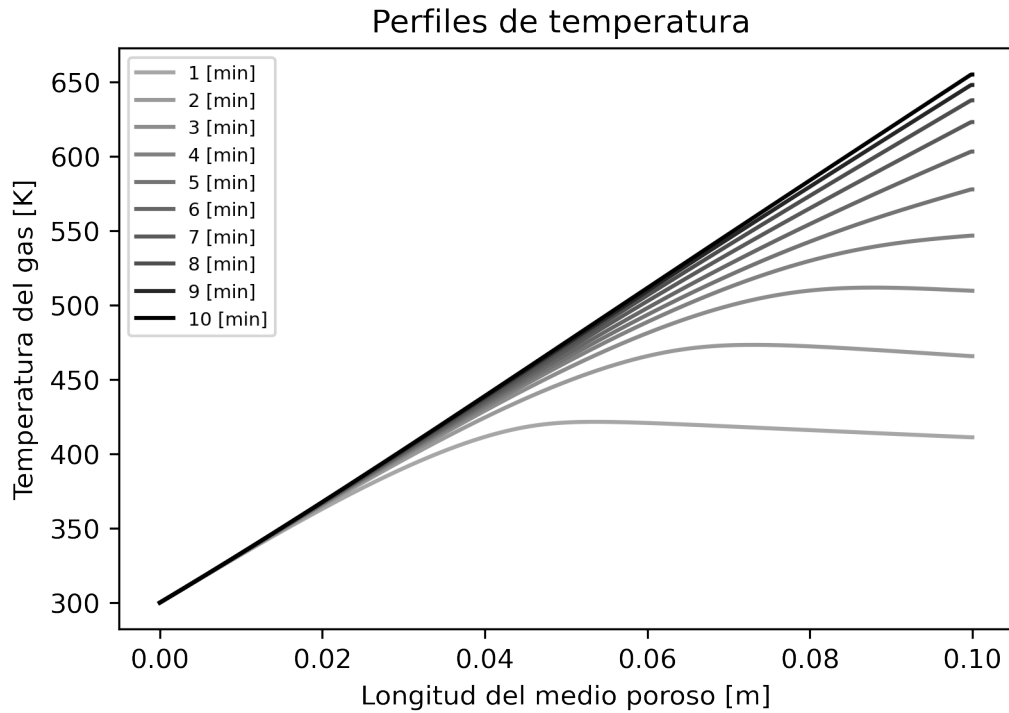
[7]: import matplotlib.pyplot as plt
plt.rcParams['figure.dpi'] = 200
#colores=['b','g','r','c','m','k','y']
colores=['0.65','0.60','0.55','0.50','0.45','0.40','0.35','0.30','0.15','0.0']
nodos=np.linspace(0,lp,800)
for i in range(len(t_reac)):
    plt.plot(nodos,Ts_red[i],colores[i])

#Datos extra
plt.title('Perfiles de temperatura ')
plt.xlabel('Longitud del medio poroso [m]')
plt.ylabel('Temperatura del sólido [K]')
plt.legend(('1 [min]','2 [min]','3 [min]','4 [min]','5 [min]','6 [min]','7 [min]','8 [min]','9 [min]','10 [min]'),
           prop = {'size': 7}, loc='lower left')
plt.show()
```



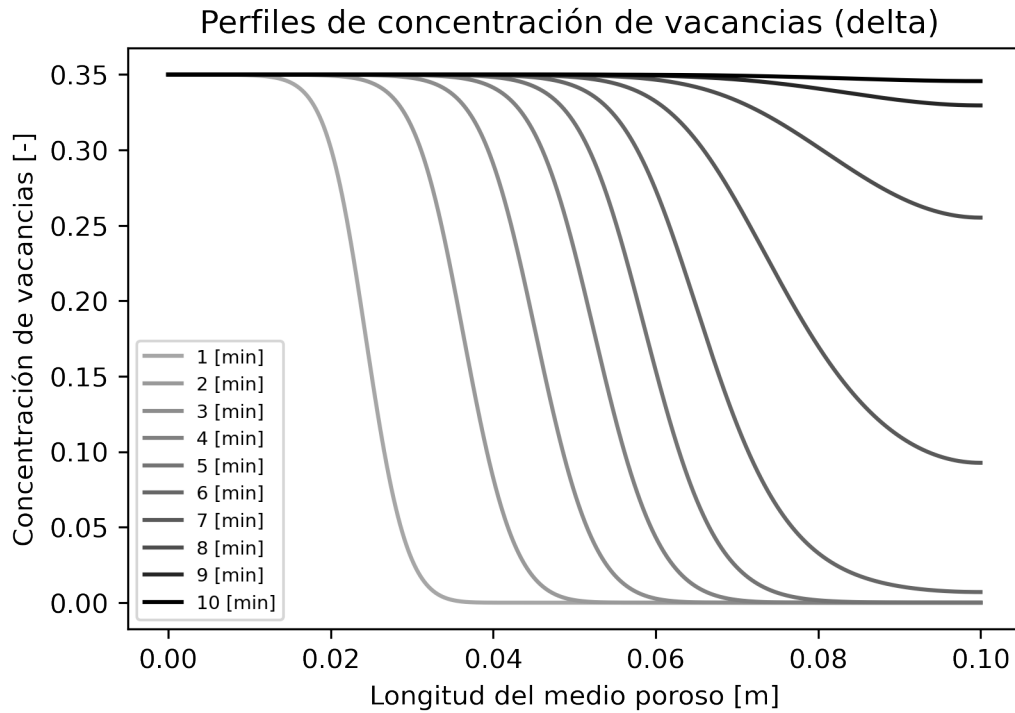
```
[8]: for i in range(len(t_reac)):
      plt.plot(nodos,Tg_red[i],colores[i])

#Datos extra
plt.title('Perfiles de temperatura ')
plt.xlabel('Longitud del medio poroso [m]')
plt.ylabel('Temperatura del gas [K]')
plt.legend(('1 [min]','2 [min]','3 [min]','4 [min]','5 [min]','6 [min]','7 [min]','8 [min]','9 [min]','10 [min]'),
           prop = {'size': 7}, loc='upper left')
plt.show()
```



```
[9]: for i in range(len(t_reac)):
      plt.plot(nodos,d_red[i],colores[i])

      #Datos extra
      plt.title('Perfiles de concentración de vacancias (delta) ')
      plt.xlabel('Longitud del medio poroso [m]')
      plt.ylabel('Concentración de vacancias [-]')
      plt.legend(('1 [min]','2 [min]','3 [min]','4 [min]','5 [min]','6 [min]','7 [min]','8 [min]','9 [min]','10 [min]'),
                prop = {'size': 7}, loc='lower left')
      plt.show()
```

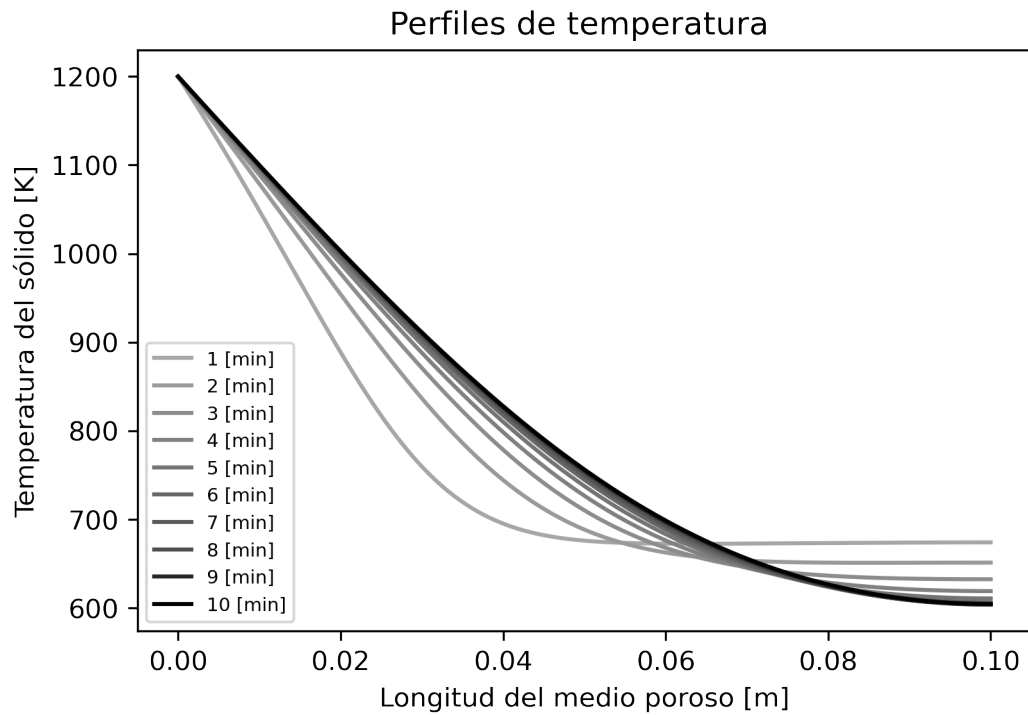


Oxidación

```
[10]: Ts_ox=[]
      Tg_ox=[]
      d_ox=[]
      for i in range(len(t_reac)):
          resultados=oxidacion_njit(10**6,1,1200,700,t_reac[i],10**6,1,2200,10)
          Ts_ox.append(resultados[0])
          Tg_ox.append(resultados[1])
          d_ox.append(resultados[2])

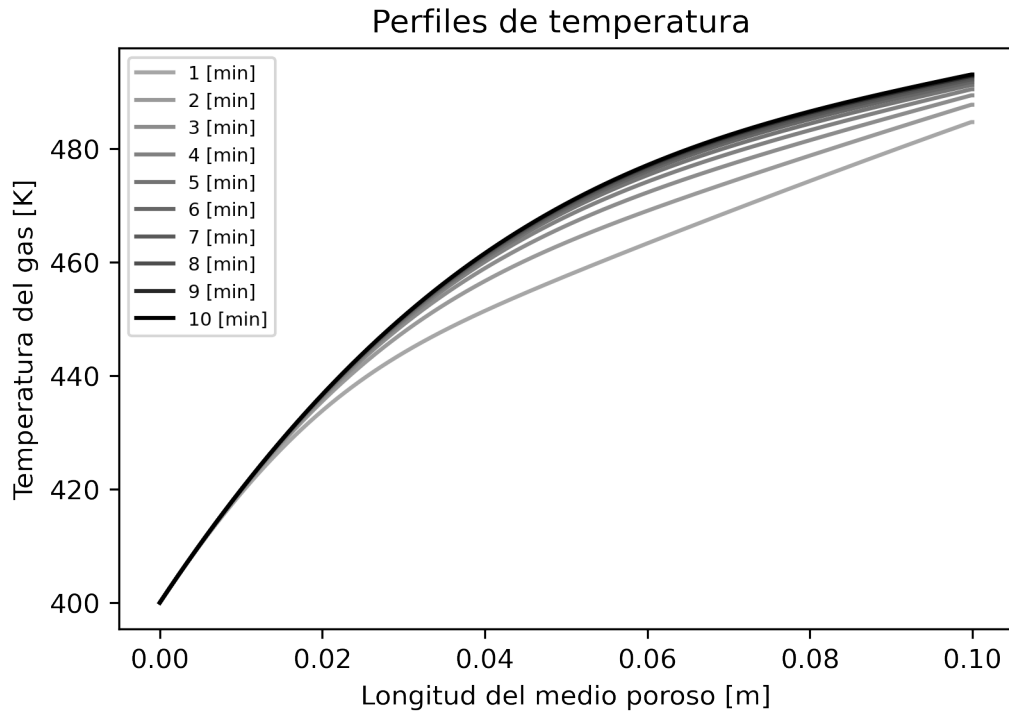
[11]: for i in range(len(t_reac)):
      plt.plot(nodos,Ts_ox[i],colores[i])

      #Datos extra
      plt.title('Perfiles de temperatura ')
      plt.xlabel('Longitud del medio poroso [m]')
      plt.ylabel('Temperatura del sólido [K]')
      plt.legend(('1 [min]','2 [min]','3 [min]','4 [min]','5 [min]','6 [min]','7 [min]','8 [min]','9 [min]','10 [min]'),
                  prop = {'size': 7}, loc='lower left')
      plt.show()
```



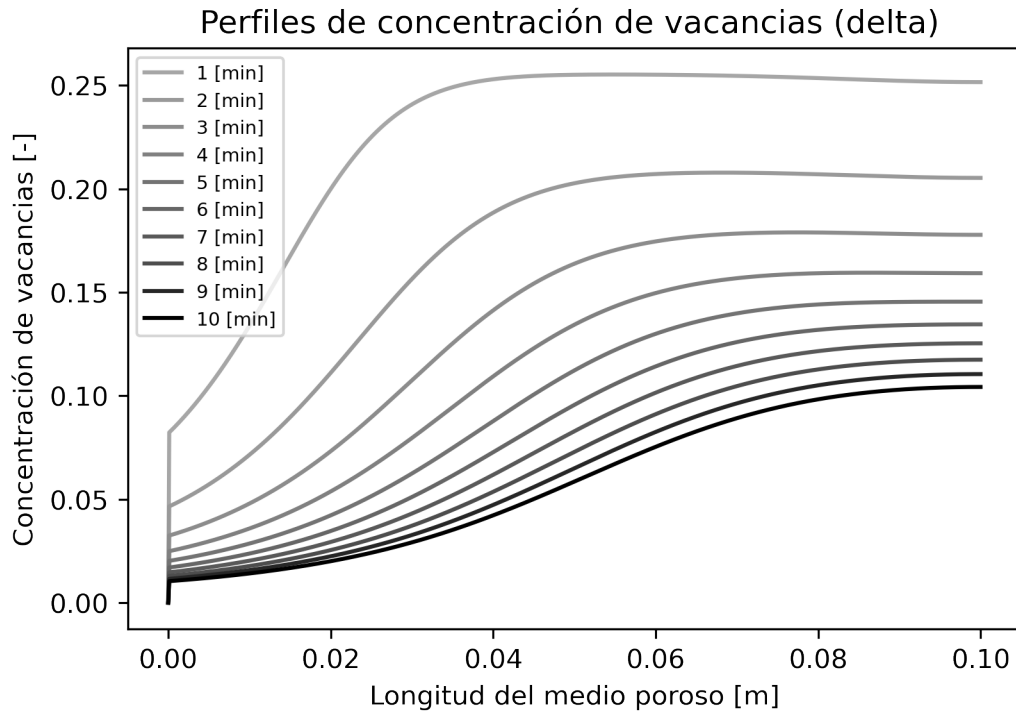
```
[12]: for i in range(len(t_reac)):
        plt.plot(nodos,Tg_ox[i],colores[i])

#Datos extra
plt.title('Perfiles de temperatura ')
plt.xlabel('Longitud del medio poroso [m]')
plt.ylabel('Temperatura del gas [K]')
plt.legend(('1 [min]','2 [min]','3 [min]','4 [min]','5 [min]','6 [min]','7 [min]','8 [min]','9 [min]','10 [min]'),
           prop = {'size': 7}, loc='upper left')
plt.show()
```



```
[13]: for i in range(len(t_reac)):
        plt.plot(nodos,d_ox[i],colores[i])

#Datos extra
plt.title('Perfiles de concentración de vacancias (delta) ')
plt.xlabel('Longitud del medio poroso [m]')
plt.ylabel('Concentración de vacancias [-]')
plt.legend(('1 [min]','2 [min]','3 [min]','4 [min]','5 [min]','6 [min]','7 [min]','8 [min]','9 [min]','10 [min]'),
           prop = {'size': 7}, loc='upper left')
plt.show()
```



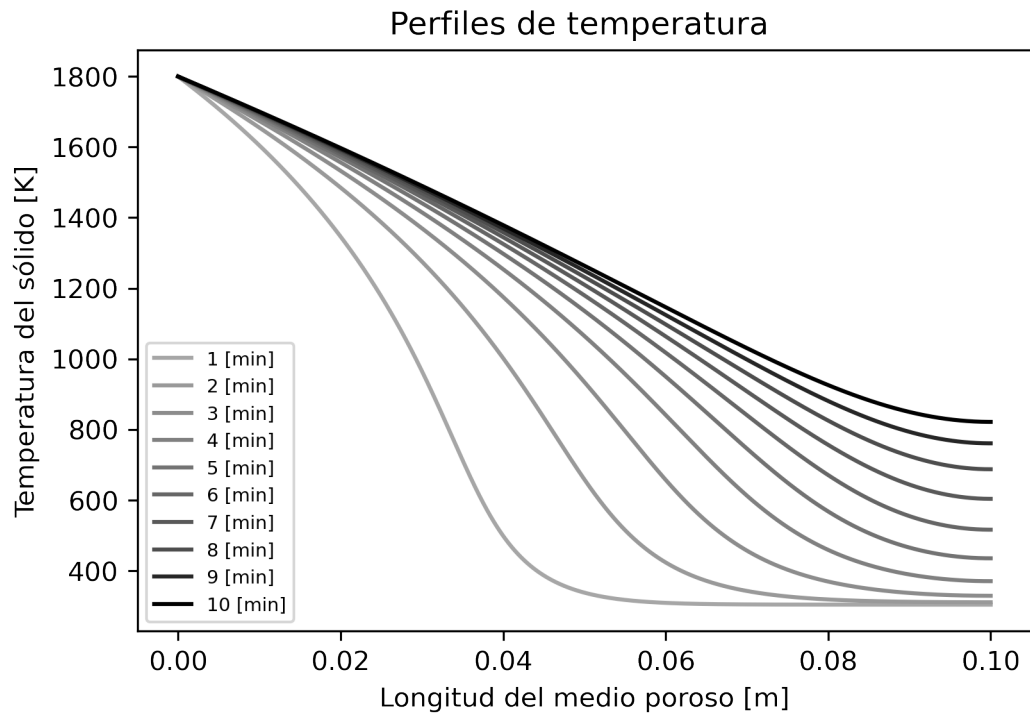
1.4.2 Caso mas desfavorable

Reducción

```
[32]: Ts_red=[]
      Tg_red=[]
      d_red=[]
      for i in range(len(t_reac)):
          resultados=reduccion_njit(10**2,3,1800,t_reac[i])
          Ts_red.append(resultados[0])
          Tg_red.append(resultados[1])
          d_red.append(resultados[2])

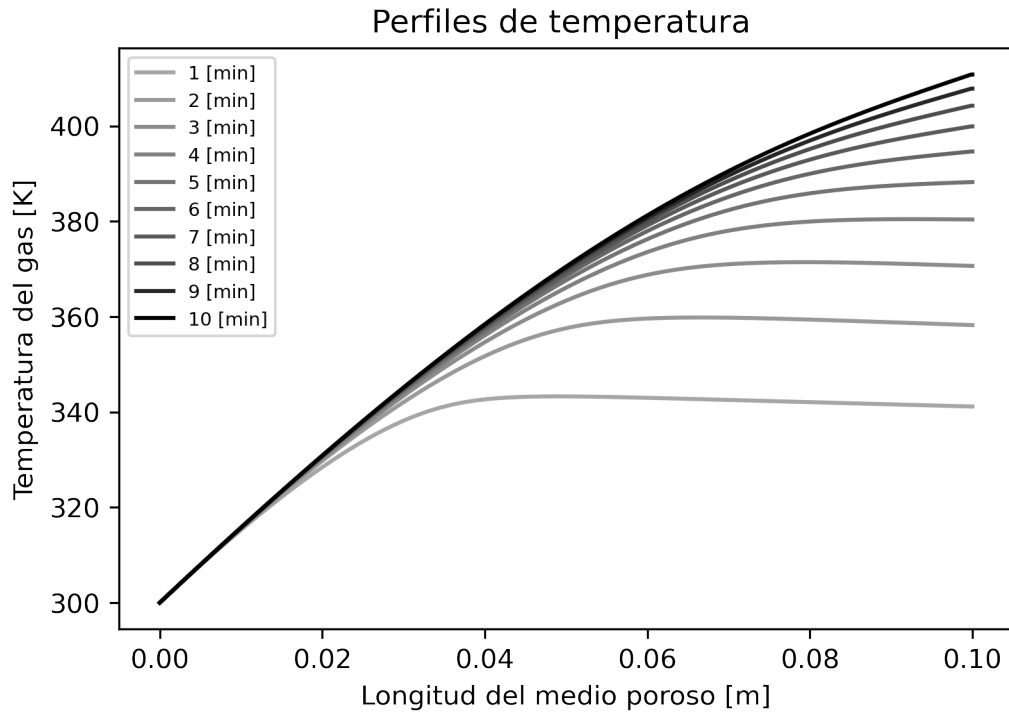
[34]: for i in range(len(t_reac)):
      plt.plot(nodos,Ts_red[i],colores[i])

      #Datos extra
      plt.title('Perfiles de temperatura ')
      plt.xlabel('Longitud del medio poroso [m]')
      plt.ylabel('Temperatura del sólido [K]')
      plt.legend(('1 [min]','2 [min]','3 [min]','4 [min]','5 [min]','6 [min]','7 [min]','8 [min]','9 [min]','10 [min]'),
          prop = {'size': 7}, loc='lower left')
      plt.show()
```



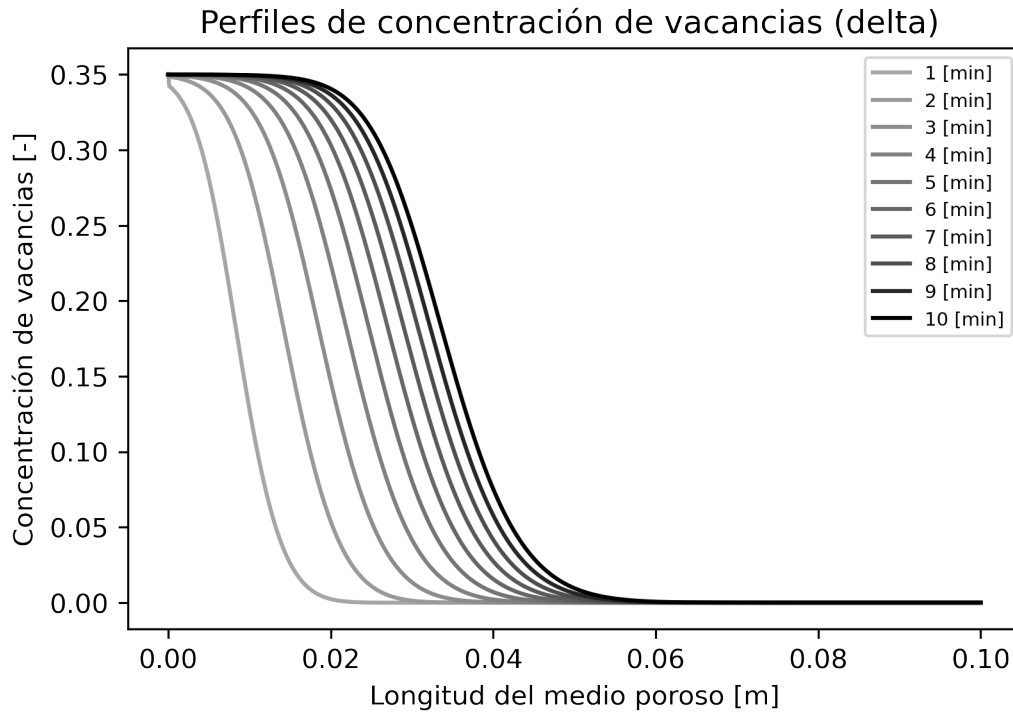
```
[35]: for i in range(len(t_reac)):
        plt.plot(nodos,Tg_red[i],colores[i])

#Datos extra
plt.title('Perfiles de temperatura ')
plt.xlabel('Longitud del medio poroso [m]')
plt.ylabel('Temperatura del gas [K]')
plt.legend(('1 [min]','2 [min]','3 [min]','4 [min]','5 [min]','6 [min]','7 [min]','8 [min]','9 [min]','10 [min]'),
           prop = {'size': 7}, loc='upper left')
plt.show()
```



```
[36]: for i in range(len(t_reac)):
        plt.plot(nodos,d_red[i],colores[i])

#Datos extra
plt.title('Perfiles de concentración de vacancias (delta) ')
plt.xlabel('Longitud del medio poroso [m]')
plt.ylabel('Concentración de vacancias [-]')
plt.legend(('1 [min]','2 [min]','3 [min]','4 [min]','5 [min]','6 [min]','7 [min]','8 [min]','9 [min]','10 [min]'),
           prop = {'size': 7}, loc='upper right')
plt.show()
```

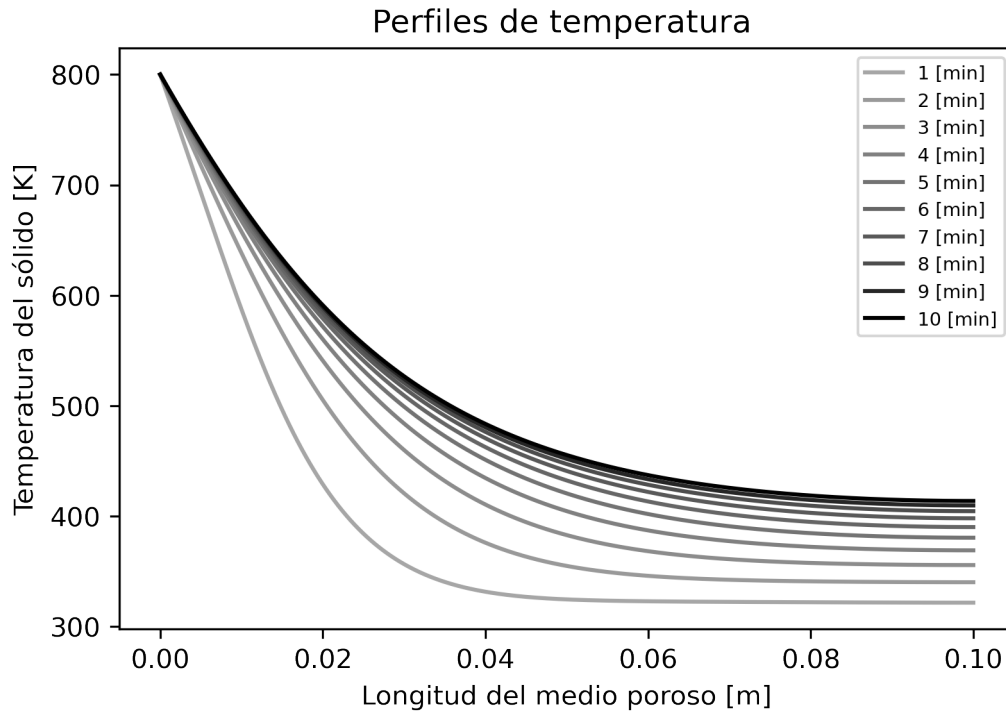


Oxidación

```
[37]: Ts_ox=[]
      Tg_ox=[]
      d_ox=[]
      for i in range(len(t_reac)):
          resultados=oxidacion_njit(10**2,3,800,300,t_reac[i],10**2,3,1800,10)
          Ts_ox.append(resultados[0])
          Tg_ox.append(resultados[1])
          d_ox.append(resultados[2])

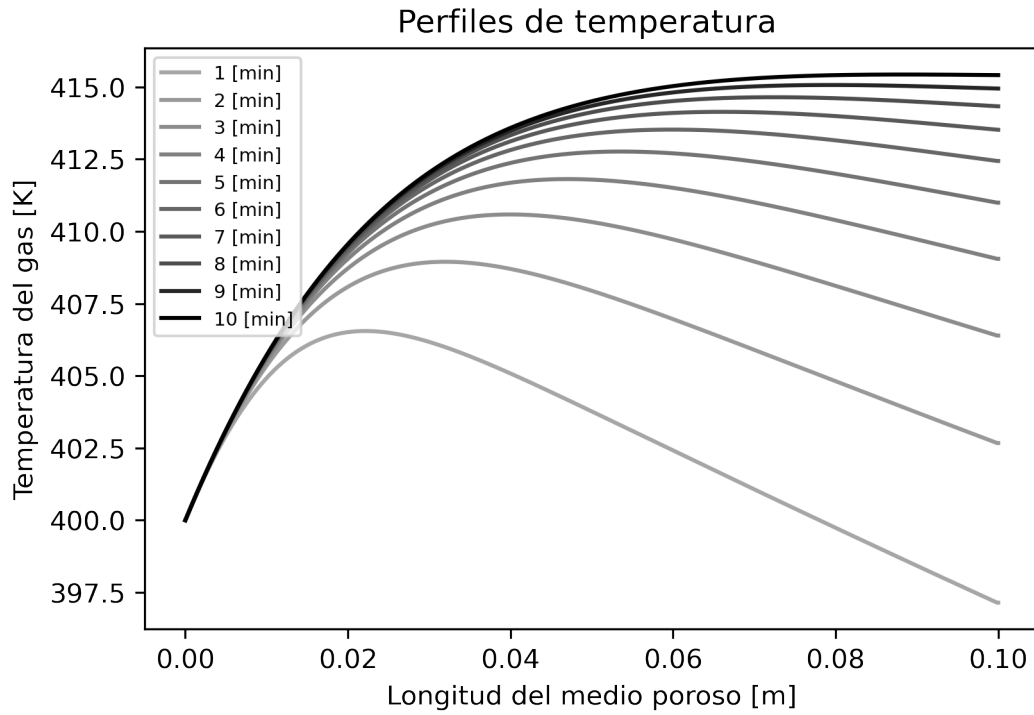
[39]: for i in range(len(t_reac)):
      plt.plot(nodos,Ts_ox[i],colores[i])

      #Datos extra
      plt.title('Perfiles de temperatura ')
      plt.xlabel('Longitud del medio poroso [m]')
      plt.ylabel('Temperatura del sólido [K]')
      plt.legend(('1 [min]','2 [min]','3 [min]','4 [min]','5 [min]','6 [min]','7 [min]','8 [min]','9 [min]','10 [min]'),
                  prop = {'size': 7}, loc='upper right')
      plt.show()
```



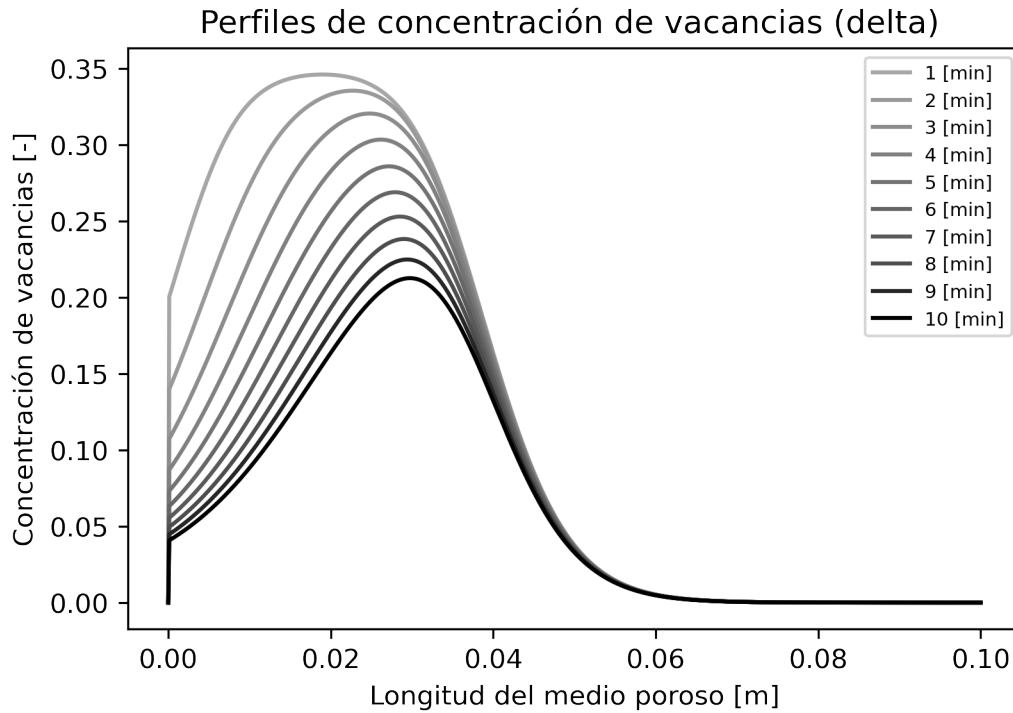
```
[40]: for i in range(len(t_reac)):
        plt.plot(nodos,Tg_ox[i],colores[i])

#Datos extra
plt.title('Perfiles de temperatura ')
plt.xlabel('Longitud del medio poroso [m]')
plt.ylabel('Temperatura del gas [K]')
plt.legend(('1 [min]','2 [min]','3 [min]','4 [min]','5 [min]','6 [min]','7 [min]','8 [min]','9 [min]','10 [min]'),
           prop = {'size': 7}, loc='upper left')
plt.show()
```



```
[41]: for i in range(len(t_reac)):
        plt.plot(nodos,d_ox[i],colores[i])

#Datos extra
plt.title('Perfiles de concentración de vacancias (delta) ')
plt.xlabel('Longitud del medio poroso [m]')
plt.ylabel('Concentración de vacancias [-]')
plt.legend(('1 [min]','2 [min]','3 [min]','4 [min]','5 [min]','6 [min]','7 [min]','8 [min]','9 [min]','10 [min]'),
           prop = {'size': 7}, loc='upper right')
plt.show()
```



1.5 Producción de O₂/H₂ para casos extremos y combinación

1.5.1 Caso +R +O

Reducción: $Q_{ce}=10^{**6}$ W/m³ $u_{gas}=1$ m/s $T_{border}=2200$ K tiempo=10 min

Oxidación: $Q_{ce}=10^{**6}$ W/m³ $u_{gas}=1$ m/s $T_{sborde}=1200$ K $T_{sinicial}=700$ K tiempo=10 min

```
[6]: red1=reduccion_njit(10**6,1,2200,10)
```

```
[7]: ox1=oxidacion_njit(10**6,1,1200,700,10,10**6,1,2200,10)
```

```
[8]: print('Produccion total de oxígeno:',red1[3], 'moles')
      print('Produccion total de hidrógeno:',ox1[3], 'moles')
```

Produccion total de oxígeno: 1.8708797308986194 moles

Produccion total de hidrógeno: 3.112462998949285 moles

1.5.2 Caso +R -O

Reducción: $Q_{ce}=10^{**6}$ W/m³ $u_{gas}=1$ m/s $T_{border}=2200$ K tiempo=10 min

Oxidación: $Q_{ce}=10^{**2}$ W/m³ $u_{gas}=3$ m/s $T_{sborde}=800$ K $T_{sinicial}=300$ K tiempo=10 min

```
[9]: red2=reduccion_njit(10**6,1,2200,10)
```

```
[10]: ox2=oxidacion_njit(10**2,3,800,300,10,10**6,1,2200,10)
```

```
[11]: print('Produccion total de oxígeno:',red2[3], 'moles')
      print('Produccion total de hidrógeno:',ox2[3], 'moles')
```

Produccion total de oxígeno: 1.8708797308986194 moles

Produccion total de hidrógeno: 1.00545185986426 moles

1.5.3 Caso -R +O

Reducción: $Q_{ce}=10^{**2}$ W/m³ $u_{gas}=3$ m/s $T_{border}=1800$ K tiempo=10 min

Oxidación: $Q_{ce}=10^{**6}$ W/m³ $u_{gas}=1$ m/s $T_{sborde}=1200$ K $T_{sinicial}=700$ K tiempo=10 min

```
[12]: red3=reduccion_njit(10**2,3,1800,10)
```

```
[13]: ox3=oxidacion_njit(10**6,1,1200,700,10,10**2,3,1800,10)
```

```
[14]: print('Produccion total de oxígeno:',red3[3], 'moles')
      print('Produccion total de hidrógeno:',ox3[3], 'moles')
```

Produccion total de oxígeno: 0.6435200959771489 moles

Produccion total de hidrógeno: 1.2114944745887772 moles

1.5.4 Caso -R -O

Reducción: $Q_{ce}=10^{**2}$ W/m³ $u_{gas}=3$ m/s $T_{border}=1800$ K tiempo=10 min

Oxidación: $Q_{ce}=10^{**2}$ W/m³ $u_{gas}=3$ m/s $T_{sborde}=800$ K $T_{sinicial}=300$ K tiempo=10 min

```
[15]: red4=reduccion_njit(10**2,3,1800,10)
```

```
[16]: ox4=oxidacion_njit(10**2,3,800,300,10,10**2,3,1800,10)
```

```
[17]: print('Produccion total de oxígeno:',red4[3], 'moles')
      print('Produccion total de hidrógeno:',ox4[3], 'moles')
```

Produccion total de oxígeno: 0.6435200959771489 moles

Produccion total de hidrógeno: 0.7411406343418204 moles

Anexo B4

May 31, 2021

1 Modelo 1-D: Sensibilización

1.1 Constantes

```
[2]: R=8.3145*10**(-3) # [kJ /mol K ] Constante universal de los gases.
Ared= 720000. #+-360000 [1/s] Factor pre-exponencial de reducción. Bulfin
Ered=232. # +-5 [kJ/mol] Energía de activación de reducción. Bulfin
lp=0.1 #[m]Longitud del medio poroso. ***DATO CÁMARA DE REACCIÓN ***
vol=7.3*10**(-4) #[m3]Volumen de la cámara de reacción. ***DATO CÁMARA DE
    ↳ REACCIÓN
e=0.65 #[-] Porosidad del medio poroso.***DATO PELLETS BULFIN***
dp=0.01 #[m]diámetro de partícula ***DATO PELLETS BULFIN***
cCe=(1-e)*34.55 #[kmol/m3] Concentración de cerio. Cálculo
molCeO2=(1-e)*30.6225 #=10.7179 Moles totales de CeO2.
sigma=5.67*10**(-8) #[W/m2K4] Constante de Stefan-Boltzmann.

#Calculo moles totales CeO2
#7.22 g/cm3 * 730 cm3 / 172.115(g/mol)= 30.6225 moles de CeO2 totales

yox=1 #Fracción molar de oxidante
Aox=1.0 # Factor pre-exponencial de oxidación. Arifin
Eox=29.0 # Energía de activación de oxidación. Arifin
gamma=0.89 #[-] Orden de reacción para la oxidación. Arifin
```

1.2 Función de la Reducción

```
[3]: import numpy as np
def reduccion(Qce,u_in,Ts_border,tred): #Los inputs se eligen desde el
    ↳ apartado 'variables' ubicado mas abajo. Al seleccionar se debe marcar con #

    #Funciones entalpía de reacción
    def entalpia_red(delta): #Valores EN kJ/mol

        if delta==0:
            entalpia=430
        elif 395-31.4*np.log10(delta) > 430:
            entalpia=430
        else:
```

```

        entalpia= 395-31.4*np.log10(delta)
    return entalpia

#.....Variables.....
#Qce=1000000 #[W/m3] Aporte de calor volumétrico externo.
#u_in=1. # [m/s] Velocidad inicial del gas
#Ts_border=2100 #[K] Temperatura del solido en x=0 en la reducción.

nx=800 #[nodos] Cantidad de nodos temporales
dt=0.02 #[segundos] Intervalo de tiempo
t_reaccion=tred #[minutos] Tiempo de reacción.
#.....

#Constantes
Tg_border=300. #Temperatura del gas en x=0 en la reducción.
d_border=0.35 #Concentración de vacancias en el sólido en x=0 en la
↪reducción.
d_max=0.35 #Concentración de vacancias máxima.
dx=lp/nx #[-] Espaciamiento
molxnodo=molCeO2/nx #cantidad de moles de CeO2 por nodo.

#####
#RESOLUCION DE ECUACIONES

#condiciones iniciales
d=np.zeros(nx)
d[0]=d_border
d_final=np.zeros(nx)+0.35 #condición final de reducción (consideración de
↪reducción completa)

Ts=np.zeros(nx)+300
Ts[0]=Ts_border
Tg=np.zeros(nx)+300
Tg[0]=Tg_border

kred=np.zeros(nx) #mod eq vacancias
kox=np.zeros(nx) #mod eq vacancias

#se crean los arreglos de A, B, C, F, alfa y beta vacías, los cuales se usan
↪para aplicar TDMA.
A_s=np.zeros(nx)
B_s=np.zeros(nx)
C_s=np.zeros(nx)
F_s=np.zeros(nx)
alfa_s=np.zeros(nx)

```

```

beta_s=np.zeros(nx)

A_g=np.zeros(nx)
B_g=np.zeros(nx)
C_g=np.zeros(nx)
F_g=np.zeros(nx)
alfa_g=np.zeros(nx)
beta_g=np.zeros(nx)

tiempo=0 #[min] tiempo de reacción
nt=0

while tiempo < t_reaccion:

    nt+=dt #contador para el tiempo de reacción

    #primero calculo todas las propiedades en función de la temperatura

    #sólido
    rho_s=(-0.0002*Ts+6.8561)*1000 #7220 #[kg/m3]densidad del sólido *ref. Nelson 2014
    cp_s= (0.0599*np.log(Ts)+0.0390)*1000 #[J/kgK]calor específico del sólido * ref. Nelson 2014
    lambda_s= (6.7760*10**(-2)+2.7930*10**(-4)*Ts)**(-1)#[W/mK] Coeficiente de transferencia de calor del sólido. ref. Nelson 2014
    lambda_ef= (1.-e)*lambda_s + (32.*sigma*dp*e*Ts**3.)/(9.*(1.-e)) #[W/mK] Coeficiente de transferencia de calor efectivo. ref. Toledo

    #gas (argón)
    rho_ar=1.603*300/Tg #[kg/m3]. Gases ideales
    cp_ar=(20.7860+Tg*2.8259*10**(-7)-Tg**2*1.4642*10**(-7)+Tg**3*1.0921*10**(-8)-Tg**(-1)*3.6614*10**(-8))*1000/39.948 #[J/kgK]. ref. webbook. nist.gov. Chase 1998
    lambda_ar=(0.0442*Tg + 4.7827)/1000 #[W/Km] Conductividad térmica. Ajuste Younglove 1986.
    mu_ar= (0.0566*Tg+6.0645)*10**(-6) ##[Pa*s] viscosidad dinámica del argón. Ajuste Younglove 1986.
    nu_ar=mu_ar/rho_ar # viscosidad cinematica del argon

```

```

u_ar=1.603*u_in/rho_ar # m/s velocidad del argon (depende de la
→ densidad)

#parámetros para a_ar. ref. Toledo
ags_ar=6./(1.-e) #superficie específica interna del medio poroso
Pr_ar=(mu_ar*cp_ar)/lambda_ar #Número de Prandtl
Re_ar=(e*u_ar*dp)/mu_ar #Número de Reynolds
hgs_ar=(lambda_ar/dp)*(2.0+1.1*Pr_ar**(1/3)*Re_ar**0.6) #Coeficiente
→ convectivo interfásico
a_ar = ags_ar*hgs_ar #Coeficiente intersticial de intercambio de calor

#Cálculo de vacancias
d_n=d.copy() #d_n sería d anterior (d^n), d sería el valor actual
→ d^(n+1)

kred=(1-d_n/d_final)**(1/3)*Ared*np.exp(-Ered/(R*Ts)) #La temperatura
→ todavía no se calcula para n+1, por tanto Ts corresponde a la temperatura en
→ n.
kox=y_ox**gamma*(1-d_n/d_final)*Aox*np.exp(-Eox/(R*Ts)) Para el caso
→ de la reducción será cero en la reducción

d=(d_n/dt+d_max*kred)/(kred+kox+1/dt) #como son estáticas no requiere
→ recorrer x

#Cálculo de temperaturas
for i in range(nx):
    if i==0:
        #solido
        beta_s[i+1]=Ts_border

        #gas
        beta_g[i+1]=Tg_border

    elif i<(nx-1):
        #solido
        A_s[i]=(lambda_ef[i-1]+lambda_ef[i])/(2*dx**2)
        B_s[i]=(lambda_ef[i]+lambda_ef[i+1])/(2*dx**2)
        C_s[i]=(1-e)*(cp_s[i]*rho_s[i])/dt + a_ar[i] + A_s[i]+B_s[i]
        F_s[i]=Ts[i]*(1-e)*(cp_s[i]*rho_s[i])/dt + a_ar[i]*Tg[i]+Qce/
→ nx-(d[i]-d_n[i])/dt*(cCe/nx)*(10**6)*entalpia_red(d[i])

        alfa_s[i+1]=B_s[i]/(C_s[i]-A_s[i]*alfa_s[i])
        beta_s[i+1]=(F_s[i]+beta_s[i]*A_s[i])/(C_s[i]-A_s[i]*alfa_s[i])

```

```

        #gas
        A_g[i]=(lambda_ar[i-1]+lambda_ar[i])/
        ↪ (2*dx**2)+u_ar[i]*cp_ar[i]*rho_ar[i]/dx
        B_g[i]=(lambda_ar[i]+lambda_ar[i+1])/(2*dx**2)
        C_g[i]=(cp_ar[i]*rho_ar[i])/dt + a_ar[i]/e + A_g[i]+B_g[i]
        F_g[i]=Tg[i]*(cp_ar[i]*rho_ar[i])/dt + a_ar[i]*Ts[i]/e

        alfa_g[i+1]=B_g[i]/(C_g[i]-A_g[i]*alfa_g[i])
        beta_g[i+1]=(F_g[i]+beta_g[i]*A_g[i])/(C_g[i]-A_g[i]*alfa_g[i])

    else:
        #solido
        Ts[i]=beta_s[i]/(1-alfa_s[i])

        #gas
        Tg[i]=beta_g[i]/(1-alfa_g[i])

    for j in range(nx-1,-1,-1):
        if j!=0:
            #solido
            Ts[j-1]=alfa_s[j]*Ts[j]+beta_s[j]
            #gas
            Tg[j-1]=alfa_g[j]*Tg[j]+beta_g[j]

        if j==0:

            break
    tiempo=nt/60 #[minutos]

    #Cálculo de producción de oxígeno

    prod_o2_nodo=molxnodo*d/2 #producción de oxígeno por nodo
    prod_o2=np.sum(prod_o2_nodo) #producción total de oxígeno
    ##ecuación de producción de o2
    return Ts,Tg,d,prod_o2

```

```

[4]: from numba import njit #Numba es una biblioteca que optimiza códigos y ahorra
    ↪ mucho tiempo de iteración.
    reduccion_njit=njit(reduccion)

```

1.3 Función de la Oxidación

```
[6]: def oxidacion(Qce,u_in,Ts_bordeo,Ts_inicial,tox,Qred,ured,Tred,tred):  
    ↪#reduccion_tuple es una tupla con los parámetros de entrada de reducción  
    def entalpia_ox(delta):  
        if delta==0:  
            entalpia=-430+285.85  
        elif 395-31.4*np.log10(delta) > 430:  
            entalpia=-430+285.85  
        else:  
            entalpia= -(395-31.4*np.log10(delta))+285.85  
        return entalpia  
  
    #.....Variables.....  
    #=1000000 #[W/m3] Aporte de calor volumétrico externo.  
    #u_in=1 #[m/s] Velocidad inicial del gas.  
    #Ts_bordeo=1200 #[K] Temperatura del sólido en x=0 en la oxidación.  
    #Ts_inicial=700 #[K] Temperatura inicial del sólido para la  
  
    nx=800 #[nodos] Cantidad de nodos espaciales.  
    dt=0.02 #[s] Intervalo de tiempo.  
    t_reaccion=tox #[min] Tiempo de reacción.  
    #.....  
  
    #Constantes  
    dx=lp/nx #[-] Espaciamiento  
    molxnodo=molCeO2/nx #cantidad de moles de CeO2 por nodo.  
  
    #Condiciones de borde de temperatura y delta.  
  
    Ts_bordeo=float(Ts_bordeo) #Temperatura del solido en x=0 en la oxidación.  
    Tg_bordeo=400. #Temperatura del gas en x=0 en la oxidación.  
    d_bordeo=0 #Concentración de vacancias en el sólido en x=0 en la oxidación.  
  
    #####  
    #RESOLUCION DE ECUACIONES  
  
    #condiciones iniciales  
    d=reduccion_njit(Qred,ured,Tred,tred)[2] #Las condiciones finales de la  
    ↪reducción se asignan aquí.  
    d_inicial=d.copy()  
    dox=d.copy()  
    dox[0]=0
```

```

Tsox=np.zeros(nx)+float(Ts_inicial)
Tsox[0]=Ts_bordeo
Tgox=np.zeros(nx)+300
Tgox[0]=Tg_bordeo

kox=np.zeros(nx) #mod eq vacancias

#se crean los arreglos de A, B, C, F, alfa y beta vacías, los cuales se
→usan para aplicar TDMA.
A_s=np.zeros(nx)
B_s=np.zeros(nx)
C_s=np.zeros(nx)
F_s=np.zeros(nx)
alfa_s=np.zeros(nx)
beta_s=np.zeros(nx)

A_g=np.zeros(nx)
B_g=np.zeros(nx)
C_g=np.zeros(nx)
F_g=np.zeros(nx)
alfa_g=np.zeros(nx)
beta_g=np.zeros(nx)

#condición final de reducción
dox_final=np.zeros(nx) #Asumimos que todo el material se oxidará

tiempo=0 #[min] tiempo de reacción
nt=0

while tiempo < t_reaccion:

    nt+=dt

    #sólido
    rho_s=(-0.0002*Tsox+6.8561)*1000 #7220 #[kg/m3]densidad del sólido *ref.
→Nelson 2014
    cp_s= (0.0599*np.log(Tsox)+0.039)*1000 #[J/kgK]calor específico del
→sólido * ref. Nelson 2014
    lambda_s= (6.776*10**(-2)+2.793*10**(-4)*Tsox)**(-1)#[W/mK] Coeficiente
→de transferencia de calor del sólido. ref. Nelson 2014
    lambda_ef= (1.-e)*lambda_s + (32.*sigma*dp*e*Tsox**3.)/(9.*(1.-e)) #[W/
→mK] Coeficiente de transferencia de calor efectivo. ref. Toledo

```

```

#gas
#nota: La temperatura del vapor varía muy poco, por ende se
→considerarán propiedades constantes.

rho_h2o=np.zeros(nx)+1/1.673 #kg/m3 #1/v volumen específico
cp_h2o=np.zeros(nx)+2.01*10**3 #j/kgK calor específico
lambda_h2o=np.zeros(nx)+24.8*10**(-3) #W/mK conductividad térmica del
→vapor
mu_h2o= np.zeros(nx)+12.0*10**(-6)#Pa*s o kg/ms Viscosidad dinámica del
→vapor
nu_h2o=mu_h2o/rho_h2o #m**2/s viscosidad cinemática
u_h2o=np.zeros(nx)+u_in

#parámetros para a_ar. ref. Toledo
ags_h2o=6./(1.-e) #superficie específica interna del medio poroso
Pr_h2o=(mu_h2o*cp_h2o)/lambda_h2o #Número de Prandtl
Re_h2o=(e*u_h2o*dp)/mu_h2o #Número de Reynolds
hgs_h2o=(lambda_h2o/dp)*(2.0+1.1*Pr_h2o**(1/3)*Re_h2o**0.6)
→#Coeficiente convectivo interfásico
a_h2o = ags_h2o*hgs_h2o #Coeficiente intersticial de intercambio de
→calor

#Cálculo de vacancias
dox_n=dox.copy() #d_n sería d anterior (d~n), d sería el valor actual
→d^(n+1)
kox=yox**gamma*(1-(d_inicial-dox_n)/(d_inicial))*Aox*np.exp(-Eox/
→(R*Tsox))
dox=(dox_n/dt)/(kox+1/dt) #como son estáticas no requiere recorrer x

#Cálculo de temperaturas
for i in range (nx):
    if i==0:
        #solido
        beta_s[i+1]=Ts_bordeo

        #gas
        beta_g[i+1]=Tg_bordeo

    elif i<(nx-1):
        #solido
        A_s[i]=(lambda_ef[i-1]+lambda_ef[i])/(2*dx**2)
        B_s[i]=(lambda_ef[i]+lambda_ef[i+1])/(2*dx**2)
        C_s[i]=(1-e)*(cp_s[i]*rho_s[i])/dt + a_h2o[i] + A_s[i]+B_s[i]

```

```

        F_s[i]=Tsox[i]*(1-e)*(cp_s[i]*rho_s[i])/dt + u
↪ a_h2o[i]*Tgox[i]+Qce/nx+(dox[i]-dox_n[i])/dt*(cCe/
↪ nx)*(10**6)*entalpia_ox(d[i])

        alfa_s[i+1]=B_s[i]/(C_s[i]-A_s[i]*alfa_s[i])
        beta_s[i+1]=(F_s[i]+beta_s[i]*A_s[i])/(C_s[i]-A_s[i]*alfa_s[i])

        #gas
        A_g[i]=(lambda_h2o[i-1]+lambda_h2o[i])/
↪ (2*dx**2)+u_h2o[i]*cp_h2o[i]*rho_h2o[i]/dx
        B_g[i]=(lambda_h2o[i]+lambda_h2o[i+1])/(2*dx**2)
        C_g[i]=(cp_h2o[i]*rho_h2o[i])/dt + a_h2o[i]/e + A_g[i]+B_g[i]
        F_g[i]=Tgox[i]*(cp_h2o[i]*rho_h2o[i])/dt + a_h2o[i]*Tsox[i]/e

        alfa_g[i+1]=B_g[i]/(C_g[i]-A_g[i]*alfa_g[i])
        beta_g[i+1]=(F_g[i]+beta_g[i]*A_g[i])/(C_g[i]-A_g[i]*alfa_g[i])

    else:
        #solido
        Tsox[i]=beta_s[i]/(1-alfa_s[i])

        #gas
        Tgox[i]=beta_g[i]/(1-alfa_g[i])

    for j in range(nx-1,-1,-1):
        if j!=0:
            #solido
            Tsox[j-1]=alfa_s[j]*Tsox[j]+beta_s[j]
            #gas
            Tgox[j-1]=alfa_g[j]*Tgox[j]+beta_g[j]

        if j==0:
            break
    tiempo=nt/60

    #Cálculo de producción de oxígeno
    prod_h2_nodo=molxnodo*(d-dox) #producción de hidrógeno por nodo
    prod_h2=np.sum(prod_h2_nodo) #producción total de hidrógeno
    return Tsox,Tgox,dox,prod_h2

```

```
[7]: oxidacion_njit=njit(oxidacion)
```

1.4 Sensibilización: Base caso desfavorable

Reducción: $Q_{ce}=10^{**6}$ W/m³ $u_{gas}=1$ m/s $T_{border}=2200$ K tiempo=10 min

Oxidación: $Q_{ce}=10^{**2}$ W/m³ $u_{gas}=3$ m/s $T_{borde}=800$ K $T_{sinicial}=300$ K tiempo=10 min

Solo se calcularán las cantidades de H₂ producido.

```
[8]: #Producción considerando el caso desfavorable:
base=oxidacion_njit(1*10**2,3,800,300,10,1*10**2,3,1800,10)[3]
```

```
[9]: def porcent(prod):
      r=(prod-base)/base*100
      return r
```

1.4.1 Aporte de calor volumétrico (Q_{ce})

Ambas etapas

```
[10]: #def oxidacion(Qce,u_in,Ts_bordeo,Ts_inicial,tox,Qred,ured,Tred,tred)
qce_prod1=oxidacion_njit(1*10**9,3,800,300,10,1*10**9,3,1800,10)
qce_prod2=oxidacion_njit(5*10**8,3,800,300,10,5*10**8,3,1800,10)
qce_prod3=oxidacion_njit(1*10**8,3,800,300,10,1*10**8,3,1800,10)
qce_prod4=oxidacion_njit(5*10**7,3,800,300,10,5*10**7,3,1800,10)
qce_prod5=oxidacion_njit(1*10**7,3,800,300,10,1*10**7,3,1800,10)
qce_prod6=oxidacion_njit(5*10**6,3,800,300,10,5*10**6,3,1800,10)
qce_prod7=oxidacion_njit(1*10**6,3,800,300,10,1*10**6,3,1800,10)
qce_prod8=oxidacion_njit(5*10**5,3,800,300,10,5*10**5,3,1800,10)
qce_prod9=oxidacion_njit(1*10**5,3,800,300,10,1*10**5,3,1800,10)
qce_prod10=oxidacion_njit(1*10**2,3,800,300,10,1*10**2,3,1800,10)
```

```
[11]: print('Caso 1: ' , qce_prod1[3], 'moles de hidrógeno.' , 'Se tiene un_
      ↪incremento de ',porcent(qce_prod1[3]),'%')
print('Caso 2: ' , qce_prod2[3], 'moles de hidrógeno.' , 'Se tiene un_
      ↪incremento de ',porcent(qce_prod2[3]),'%')
print('Caso 3: ' , qce_prod3[3], 'moles de hidrógeno.' , 'Se tiene un_
      ↪incremento de ',porcent(qce_prod3[3]),'%')
print('Caso 4: ' , qce_prod4[3], 'moles de hidrógeno.' , 'Se tiene un_
      ↪incremento de ',porcent(qce_prod4[3]),'%')
print('Caso 5: ' , qce_prod5[3], 'moles de hidrógeno.' , 'Se tiene un_
      ↪incremento de ',porcent(qce_prod5[3]),'%')
print('Caso 6: ' , qce_prod6[3], 'moles de hidrógeno.' , 'Se tiene un_
      ↪incremento de ',porcent(qce_prod6[3]),'%')
print('Caso 7: ' , qce_prod7[3], 'moles de hidrógeno.' , 'Se tiene un_
      ↪incremento de ',porcent(qce_prod7[3]),'%')
print('Caso 8: ' , qce_prod8[3], 'moles de hidrógeno.' , 'Se tiene un_
      ↪incremento de ',porcent(qce_prod8[3]),'%')
print('Caso 9: ' , qce_prod9[3], 'moles de hidrógeno.' , 'Se tiene un_
      ↪incremento de ',porcent(qce_prod9[3]),'%')
```

```
print('Caso 10: ' , qce_prod10[3], 'moles de hidrógeno.' , 'Se tiene un_
↪incremento de ',porcent(qce_prod10[3]),'%')
```

Caso 1: 1.426156188811162 moles de hidrógeno. Se tiene un incremento de 92.42720244015207 %
 Caso 2: 1.0228799005706388 moles de hidrógeno. Se tiene un incremento de 38.014278690713084 %
 Caso 3: 0.7908051047640906 moles de hidrógeno. Se tiene un incremento de 6.701085883163775 %
 Caso 4: 0.7655660576275878 moles de hidrógeno. Se tiene un incremento de 3.2956529643606265 %
 Caso 5: 0.7459605320352195 moles de hidrógeno. Se tiene un incremento de 0.650335101067485 %
 Caso 6: 0.7435464851844742 moles de hidrógeno. Se tiene un incremento de 0.32461461849144385 %
 Caso 7: 0.741621114175023 moles de hidrógeno. Se tiene un incremento de 0.0648297787138924 %
 Caso 8: 0.7413808094937868 moles de hidrógeno. Se tiene un incremento de 0.032406150848776995 %
 Caso 9: 0.7411886244235343 moles de hidrógeno. Se tiene un incremento de 0.006475165372152867 %
 Caso 10: 0.7411406343418204 moles de hidrógeno. Se tiene un incremento de 0.0 %

Reducción

```
[17]: #def oxidacion(Qce,u_in,Ts_bordeo,Ts_inicial,Qred,ured,Tred):
qcer_prod1=oxidacion_njit(1*10**2,3,800,300,10,1*10**9,3,1800,10)
qcer_prod2=oxidacion_njit(1*10**2,3,800,300,10,5*10**8,3,1800,10)
qcer_prod3=oxidacion_njit(1*10**2,3,800,300,10,1*10**8,3,1800,10)
qcer_prod4=oxidacion_njit(1*10**2,3,800,300,10,5*10**7,3,1800,10)
qcer_prod5=oxidacion_njit(1*10**2,3,800,300,10,1*10**7,3,1800,10)
qcer_prod6=oxidacion_njit(1*10**2,3,800,300,10,5*10**6,3,1800,10)
qcer_prod7=oxidacion_njit(1*10**2,3,800,300,10,1*10**6,3,1800,10)
qcer_prod8=oxidacion_njit(1*10**2,3,800,300,10,5*10**5,3,1800,10)
qcer_prod9=oxidacion_njit(1*10**2,3,800,300,10,1*10**5,3,1800,10)
qcer_prod10=oxidacion_njit(1*10**2,3,800,300,10,1*10**2,3,1800,10)
```

```
[18]: print('Caso 1: ' , qcer_prod1[3], 'moles de hidrógeno.' , 'Se tiene un_
↪incremento de ',porcent(qcer_prod1[3]),'%')
print('Caso 2: ' , qcer_prod2[3], 'moles de hidrógeno.' , 'Se tiene un_
↪incremento de ',porcent(qcer_prod2[3]),'%')
print('Caso 3: ' , qcer_prod3[3], 'moles de hidrógeno.' , 'Se tiene un_
↪incremento de ',porcent(qcer_prod3[3]),'%')
print('Caso 4: ' , qcer_prod4[3], 'moles de hidrógeno.' , 'Se tiene un_
↪incremento de ',porcent(qcer_prod4[3]),'%')
print('Caso 5: ' , qcer_prod5[3], 'moles de hidrógeno.' , 'Se tiene un_
↪incremento de ',porcent(qcer_prod5[3]),'%')
```

```

print('Caso 6: ' , qcer_prod6[3], 'moles de hidrógeno.' , 'Se tiene un_
↪incremento de ',porcent(qcer_prod6[3]),'%')
print('Caso 7: ' , qcer_prod7[3], 'moles de hidrógeno.' , 'Se tiene un_
↪incremento de ',porcent(qcer_prod7[3]),'%')
print('Caso 8: ' , qcer_prod8[3], 'moles de hidrógeno.' , 'Se tiene un_
↪incremento de ',porcent(qcer_prod8[3]),'%')
print('Caso 9: ' , qcer_prod9[3], 'moles de hidrógeno.' , 'Se tiene un_
↪incremento de ',porcent(qcer_prod9[3]),'%')
print('Caso 10: ' , qcer_prod10[3], 'moles de hidrógeno.' , 'Se tiene un_
↪incremento de ',porcent(qcer_prod10[3]),'%')

```

Caso 1: 0.8375125722325913 moles de hidrógeno. Se tiene un incremento de 13.003191759463476 %

Caso 2: 0.784281239192641 moles de hidrógeno. Se tiene un incremento de 5.820839237769251 %

Caso 3: 0.7492522051790446 moles de hidrógeno. Se tiene un incremento de 1.0944712057823844 %

Caso 4: 0.7451696473604646 moles de hidrógeno. Se tiene un incremento de 0.5436232790315485 %

Caso 5: 0.7419422882618432 moles de hidrógeno. Se tiene un incremento de 0.10816488570143737 %

Caso 6: 0.7415412026258085 moles de hidrógeno. Se tiene un incremento de 0.05404754042986393 %

Caso 7: 0.7412207009783977 moles de hidrógeno. Se tiene un incremento de 0.010803163781234224 %

Caso 8: 0.7411806611224269 moles de hidrógeno. Se tiene un incremento de 0.005400699779736873 %

Caso 9: 0.741148632887401 moles de hidrógeno. Se tiene un incremento de 0.0010792210290341815 %

Caso 10: 0.7411406343418204 moles de hidrógeno. Se tiene un incremento de 0.0 %

Oxidación

```

[19]: qceo_prod1=oxidacion_njit(1*10**9,3,800,300,10,1*10**2,3,1800,10)
qceo_prod2=oxidacion_njit(5*10**8,3,800,300,10,1*10**2,3,1800,10)
qceo_prod3=oxidacion_njit(1*10**8,3,800,300,10,1*10**2,3,1800,10)
qceo_prod4=oxidacion_njit(5*10**7,3,800,300,10,1*10**2,3,1800,10)
qceo_prod5=oxidacion_njit(1*10**7,3,800,300,10,1*10**2,3,1800,10)
qceo_prod6=oxidacion_njit(5*10**6,3,800,300,10,1*10**2,3,1800,10)
qceo_prod7=oxidacion_njit(1*10**6,3,800,300,10,1*10**2,3,1800,10)
qceo_prod8=oxidacion_njit(5*10**5,3,800,300,10,1*10**2,3,1800,10)
qceo_prod9=oxidacion_njit(1*10**5,3,800,300,10,1*10**2,3,1800,10)
qceo_prod10=oxidacion_njit(1*10**2,3,800,300,10,1*10**2,3,1800,10)

```

```

[20]: print('Caso 1: ' , qceo_prod1[3], 'moles de hidrógeno.' , 'Se tiene un_
↪incremento de ',porcent(qceo_prod1[3]),'%')

```

```

print('Caso 2: ' , qceo_prod2[3], 'moles de hidrógeno.' , 'Se tiene un_
↳incremento de ',porcent(qceo_prod2[3]),'%')
print('Caso 3: ' , qceo_prod3[3], 'moles de hidrógeno.' , 'Se tiene un_
↳incremento de ',porcent(qceo_prod3[3]),'%')
print('Caso 4: ' , qceo_prod4[3], 'moles de hidrógeno.' , 'Se tiene un_
↳incremento de ',porcent(qceo_prod4[3]),'%')
print('Caso 5: ' , qceo_prod5[3], 'moles de hidrógeno.' , 'Se tiene un_
↳incremento de ',porcent(qceo_prod5[3]),'%')
print('Caso 6: ' , qceo_prod6[3], 'moles de hidrógeno.' , 'Se tiene un_
↳incremento de ',porcent(qceo_prod6[3]),'%')
print('Caso 7: ' , qceo_prod7[3], 'moles de hidrógeno.' , 'Se tiene un_
↳incremento de ',porcent(qceo_prod7[3]),'%')
print('Caso 8: ' , qceo_prod8[3], 'moles de hidrógeno.' , 'Se tiene un_
↳incremento de ',porcent(qceo_prod8[3]),'%')
print('Caso 9: ' , qceo_prod9[3], 'moles de hidrógeno.' , 'Se tiene un_
↳incremento de ',porcent(qceo_prod9[3]),'%')
print('Caso 10: ' , qceo_prod10[3], 'moles de hidrógeno.' , 'Se tiene un_
↳incremento de ',porcent(qceo_prod10[3]),'%')

```

Caso 1: 1.0380963985023381 moles de hidrógeno. Se tiene un incremento de 40.06739752223047 %
 Caso 2: 0.9208281220830642 moles de hidrógeno. Se tiene un incremento de 24.244722177568576 %
 Caso 3: 0.7807721888017558 moles de hidrógeno. Se tiene un incremento de 5.347373038739226 %
 Caso 4: 0.7610725042313876 moles de hidrógeno. Se tiene un incremento de 2.689350572077045 %
 Caso 5: 0.7451408142456025 moles de hidrógeno. Se tiene un incremento de 0.5397329087662976 %
 Caso 6: 0.7431414173590593 moles de hidrógeno. Se tiene un incremento de 0.26995996772132846 %
 Caso 7: 0.7415408681092219 moles de hidrógeno. Se tiene un incremento de 0.054002405057298745 %
 Caso 8: 0.7413407378812252 moles de hidrógeno. Se tiene un incremento de 0.026999402020712326 %
 Caso 9: 0.7411806240880675 moles de hidrógeno. Se tiene un incremento de 0.005395702838854005 %
 Caso 10: 0.7411406343418204 moles de hidrógeno. Se tiene un incremento de 0.0 %

1.4.2 Temperatura de borde

Ambas etapas

```

[13]: tb_prod1=oxidacion_njit(10**2,3,1200,300,10,10**2,3,2200,10)
      tb_prod2=oxidacion_njit(10**2,3,1100,300,10,10**2,3,2100,10)
      tb_prod3=oxidacion_njit(10**2,3,1000,300,10,10**2,3,2000,10)
      tb_prod4=oxidacion_njit(10**2,3,900,300,10,10**2,3,1900,10)

```

```
tb_prod5=oxidacion_njit(10**2,3,800,300,10,10**2,3,1800,10)
```

```
[27]: print('Caso 1: ' , tb_prod1[3], 'moles de hidrógeno.' , 'Se tiene un incremento_
      ↪de ',porcent(tb_prod1[3]),'%')
      print('Caso 2: ' , tb_prod2[3], 'moles de hidrógeno.' , 'Se tiene un incremento_
      ↪de ',porcent(tb_prod2[3]),'%')
      print('Caso 3: ' , tb_prod3[3], 'moles de hidrógeno.' , 'Se tiene un incremento_
      ↪de ',porcent(tb_prod3[3]),'%')
      print('Caso 4: ' , tb_prod4[3], 'moles de hidrógeno.' , 'Se tiene un incremento_
      ↪de ',porcent(tb_prod4[3]),'%')
      print('Caso 5: ' , tb_prod5[3], 'moles de hidrógeno.' , 'Se tiene un incremento_
      ↪de ',porcent(tb_prod5[3]),'%')
```

Caso 1: 1.8045755718999006 moles de hidrógeno. Se tiene un incremento de 143.48625460301164 %

Caso 2: 1.4978329766535017 moles de hidrógeno. Se tiene un incremento de 102.09834776953927 %

Caso 3: 1.2304350568324232 moles de hidrógeno. Se tiene un incremento de 66.0191062017706 %

Caso 4: 0.9826796917791244 moles de hidrógeno. Se tiene un incremento de 32.5901787387229 %

Caso 5: 0.7411406343418204 moles de hidrógeno. Se tiene un incremento de 0.0 %

Reducción

```
[29]: tbr_prod1=oxidacion_njit(10**2,3,800,300,10,10**2,3,2200,10)
      tbr_prod2=oxidacion_njit(10**2,3,800,300,10,10**2,3,2100,10)
      tbr_prod3=oxidacion_njit(10**2,3,800,300,10,10**2,3,2000,10)
      tbr_prod4=oxidacion_njit(10**2,3,800,300,10,10**2,3,1900,10)
      tbr_prod5=oxidacion_njit(10**2,3,800,300,10,10**2,3,1800,10)
```

```
[50]: print('Caso 1: ' , tbr_prod1[3], 'moles de hidrógeno.' , 'Se tiene un_
      ↪incremento de ',porcent(tbr_prod1[3]),%', '3ER')
      print('Caso 2: ' , tbr_prod2[3], 'moles de hidrógeno.' , 'Se tiene un_
      ↪incremento de ',porcent(tbr_prod2[3]),'%')
      print('Caso 3: ' , tbr_prod3[3], 'moles de hidrógeno.' , 'Se tiene un_
      ↪incremento de ',porcent(tbr_prod3[3]),'%')
      print('Caso 4: ' , tbr_prod4[3], 'moles de hidrógeno.' , 'Se tiene un_
      ↪incremento de ',porcent(tbr_prod4[3]),'%')
      print('Caso 5: ' , tbr_prod5[3], 'moles de hidrógeno.' , 'Se tiene un_
      ↪incremento de ',porcent(tbr_prod5[3]),'%')
```

Caso 1: 0.9756210605253199 moles de hidrógeno. Se tiene un incremento de 31.637777679229924 % 3ER

Caso 2: 0.9192017870985753 moles de hidrógeno. Se tiene un incremento de 24.025285418992624 %

Caso 3: 0.8705109803084967 moles de hidrógeno. Se tiene un incremento de 17.455573203264628 %

Caso 4: 0.8163297963698938 moles de hidrógeno. Se tiene un incremento de

10.145059998612282 %

Caso 5: 0.7411406343418204 moles de hidrógeno. Se tiene un incremento de 0.0 %

Oxidación

```
[31]: tbo_prod1=oxidacion_njit(10**2,3,1200,300,10,10**2,3,1800,10)
tbo_prod2=oxidacion_njit(10**2,3,1100,300,10,10**2,3,1800,10)
tbo_prod3=oxidacion_njit(10**2,3,1000,300,10,10**2,3,1800,10)
tbo_prod4=oxidacion_njit(10**2,3,900,300,10,10**2,3,1800,10)
tbo_prod5=oxidacion_njit(10**2,3,800,300,10,10**2,3,1800,10)
```

```
[48]: print('Caso 1: ' , tbo_prod1[3], 'moles de hidrógeno.' , 'Se tiene un_
      ↪ incremento de ',porcent(tbo_prod1[3]),'%', '1ER')
print('Caso 2: ' , tbo_prod2[3], 'moles de hidrógeno.' , 'Se tiene un_
      ↪ incremento de ',porcent(tbo_prod2[3]),'%')
print('Caso 3: ' , tbo_prod3[3], 'moles de hidrógeno.' , 'Se tiene un_
      ↪ incremento de ',porcent(tbo_prod3[3]),'%')
print('Caso 4: ' , tbo_prod4[3], 'moles de hidrógeno.' , 'Se tiene un_
      ↪ incremento de ',porcent(tbo_prod4[3]),'%')
print('Caso 5: ' , tbo_prod5[3], 'moles de hidrógeno.' , 'Se tiene un_
      ↪ incremento de ',porcent(tbo_prod5[3]),'%')
```

Caso 1: 1.1462352581671034 moles de hidrógeno. Se tiene un incremento de 54.65826660348107 % 1ER

Caso 2: 1.0814263878262074 moles de hidrógeno. Se tiene un incremento de 45.91378986885291 %

Caso 3: 0.9927253835988035 moles de hidrógeno. Se tiene un incremento de 33.945615393279056 %

Caso 4: 0.8790209247507035 moles de hidrógeno. Se tiene un incremento de 18.603795827674386 %

Caso 5: 0.7411406343418204 moles de hidrógeno. Se tiene un incremento de 0.0 %

1.4.3 Velocidad del gas

Ambas etapas

```
[34]: u_prod1=oxidacion_njit(10**2,1,800,300,10,10**2,1,1800,10)
u_prod2=oxidacion_njit(10**2,1.5,800,300,10,10**2,1.5,1800,10)
u_prod3=oxidacion_njit(10**2,2,800,300,10,10**2,2,1800,10)
u_prod4=oxidacion_njit(10**2,2.5,800,300,10,10**2,2.5,1800,10)
u_prod5=oxidacion_njit(10**2,3,800,300,10,10**2,3,1800,10)
```

```
[35]: print('Caso 1: ' , u_prod1[3], 'moles de hidrógeno.' , 'Se tiene un incremento_
      ↪ de ',porcent(u_prod1[3]),'%')
print('Caso 2: ' , u_prod2[3], 'moles de hidrógeno.' , 'Se tiene un incremento_
      ↪ de ',porcent(u_prod2[3]),'%')
print('Caso 3: ' , u_prod3[3], 'moles de hidrógeno.' , 'Se tiene un incremento_
      ↪ de ',porcent(u_prod3[3]),'%')
print('Caso 4: ' , u_prod4[3], 'moles de hidrógeno.' , 'Se tiene un incremento_
      ↪ de ',porcent(u_prod4[3]),'%')
```

```
print('Caso 5: ' , u_prod5[3], 'moles de hidrógeno.' , 'Se tiene un incremento_
↳de ',porcent(u_prod5[3]),'%')
```

Caso 1: 0.8542648268215864 moles de hidrógeno. Se tiene un incremento de 15.26352587322748 %
 Caso 2: 0.8170302151328686 moles de hidrógeno. Se tiene un incremento de 10.239565512211168 %
 Caso 3: 0.7872972897510617 moles de hidrógeno. Se tiene un incremento de 6.227786370157847 %
 Caso 4: 0.76246642820671 moles de hidrógeno. Se tiene un incremento de 2.8774287735320545 %
 Caso 5: 0.7411406343418204 moles de hidrógeno. Se tiene un incremento de 0.0 %

Reducción

```
[12]: ur_prod1=oxidacion_njit(10**2,3,800,300,10,10**2,1,1800,10)
ur_prod2=oxidacion_njit(10**2,3,800,300,10,10**2,1.5,1800,10)
ur_prod3=oxidacion_njit(10**2,3,800,300,10,10**2,2,1800,10)
ur_prod4=oxidacion_njit(10**2,3,800,300,10,10**2,2.5,1800,10)
ur_prod5=oxidacion_njit(10**2,3,800,300,10,10**2,3,1800,10)
```

```
[11]: print('Caso 1: ' , ur_prod1[3], 'moles de hidrógeno.' , 'Se tiene un incremento_
↳de ',porcent(ur_prod1[3]),'%')
print('Caso 2: ' , ur_prod2[3], 'moles de hidrógeno.' , 'Se tiene un incremento_
↳de ',porcent(ur_prod2[3]),'%')
print('Caso 3: ' , ur_prod3[3], 'moles de hidrógeno.' , 'Se tiene un incremento_
↳de ',porcent(ur_prod3[3]),'%')
print('Caso 4: ' , ur_prod4[3], 'moles de hidrógeno.' , 'Se tiene un incremento_
↳de ',porcent(ur_prod4[3]),'%')
print('Caso 5: ' , ur_prod5[3], 'moles de hidrógeno.' , 'Se tiene un incremento_
↳de ',porcent(ur_prod5[3]),'%')
```

Caso 1: 0.8757731681172327 moles de hidrógeno. Se tiene un incremento de 18.16558525292227 %
 Caso 2: 0.7744802889539656 moles de hidrógeno. Se tiene un incremento de 4.498424869357335 %
 Caso 3: 0.7617621292223669 moles de hidrógeno. Se tiene un incremento de 2.7823997126887554 %
 Caso 4: 0.7508259997533435 moles de hidrógeno. Se tiene un incremento de 1.30681883609368 %
 Caso 5: 0.7411406343418204 moles de hidrógeno. Se tiene un incremento de 0.0 %

Oxidación

```
[38]: uo_prod1=oxidacion_njit(10**2,1,800,300,10,10**2,3,1800,10)
uo_prod2=oxidacion_njit(10**2,1.5,800,300,10,10**2,3,1800,10)
uo_prod3=oxidacion_njit(10**2,2,800,300,10,10**2,3,1800,10)
uo_prod4=oxidacion_njit(10**2,2.5,800,300,10,10**2,3,1800,10)
uo_prod5=oxidacion_njit(10**2,3,800,300,10,10**2,3,1800,10)
```

```
[39]: print('Caso 1: ' , uo_prod1[3], 'moles de hidrógeno.' , 'Se tiene un incremento_
↳de ',porcent(uo_prod1[3]),'%')
print('Caso 2: ' , uo_prod2[3], 'moles de hidrógeno.' , 'Se tiene un incremento_
↳de ',porcent(uo_prod2[3]),'%')
print('Caso 3: ' , uo_prod3[3], 'moles de hidrógeno.' , 'Se tiene un incremento_
↳de ',porcent(uo_prod3[3]),'%')
print('Caso 4: ' , uo_prod4[3], 'moles de hidrógeno.' , 'Se tiene un incremento_
↳de ',porcent(uo_prod4[3]),'%')
print('Caso 5: ' , uo_prod5[3], 'moles de hidrógeno.' , 'Se tiene un incremento_
↳de ',porcent(uo_prod5[3]),'%')
```

Caso 1: 0.7961198458307156 moles de hidrógeno. Se tiene un incremento de 7.418188794589596 %
 Caso 2: 0.7793134529350599 moles de hidrógeno. Se tiene un incremento de 5.150549952930239 %
 Caso 3: 0.7649952199690732 moles de hidrógeno. Se tiene un incremento de 3.2186314610097058 %
 Caso 4: 0.7524118376296595 moles de hidrógeno. Se tiene un incremento de 1.5207914349276743 %
 Caso 5: 0.7411406343418204 moles de hidrógeno. Se tiene un incremento de 0.0 %

1.4.4 Temperatura inicial de oxidación

```
[40]: ti_prod1=oxidacion_njit(10**2,3,800,700,10,10**2,3,1800,10)
ti_prod2=oxidacion_njit(10**2,3,800,600,10,10**2,3,1800,10)
ti_prod3=oxidacion_njit(10**2,3,800,500,10,10**2,3,1800,10)
ti_prod4=oxidacion_njit(10**2,3,800,400,10,10**2,3,1800,10)
ti_prod5=oxidacion_njit(10**2,3,800,300,10,10**2,3,1800,10)
```

```
[51]: print('Caso 1: ' , ti_prod1[3], 'moles de hidrógeno.' , 'Se tiene un incremento_
↳de ',porcent(ti_prod1[3]),'%','4T0')
print('Caso 2: ' , ti_prod2[3], 'moles de hidrógeno.' , 'Se tiene un incremento_
↳de ',porcent(ti_prod2[3]),'%')
print('Caso 3: ' , ti_prod3[3], 'moles de hidrógeno.' , 'Se tiene un incremento_
↳de ',porcent(ti_prod3[3]),'%')
print('Caso 4: ' , ti_prod4[3], 'moles de hidrógeno.' , 'Se tiene un incremento_
↳de ',porcent(ti_prod4[3]),'%')
print('Caso 5: ' , ti_prod5[3], 'moles de hidrógeno.' , 'Se tiene un incremento_
↳de ',porcent(ti_prod5[3]),'%')
```

Caso 1: 0.9408676937771343 moles de hidrógeno. Se tiene un incremento de 26.948604648115683 % 4T0
 Caso 2: 0.8831899464035025 moles de hidrógeno. Se tiene un incremento de 19.166310073908008 %
 Caso 3: 0.8266421171987948 moles de hidrógeno. Se tiene un incremento de 11.536472147813761 %
 Caso 4: 0.7785495358368107 moles de hidrógeno. Se tiene un incremento de 5.047476789369643 %

Caso 5: 0.7411406343418204 moles de hidrógeno. Se tiene un incremento de 0.0 %

1.4.5 Tiempo de reacción

Reducción

```
[42]: trr_prod1=oxidacion_njit(10**2,3,800,300,10,10**2,3,1800,50)
trr_prod2=oxidacion_njit(10**2,3,800,300,10,10**2,3,1800,40)
trr_prod3=oxidacion_njit(10**2,3,800,300,10,10**2,3,1800,30)
trr_prod4=oxidacion_njit(10**2,3,800,300,10,10**2,3,1800,20)
trr_prod5=oxidacion_njit(10**2,3,800,300,10,10**2,3,1800,10)
```

```
[52]: print('Caso 1: ' , trr_prod1[3], 'moles de hidrógeno.' , 'Se tiene un_
      ↪ incremento de ',porcent(trr_prod1[3]),'%','5T0')
print('Caso 2: ' , trr_prod2[3], 'moles de hidrógeno.' , 'Se tiene un_
      ↪ incremento de ',porcent(trr_prod2[3]),'%')
print('Caso 3: ' , trr_prod3[3], 'moles de hidrógeno.' , 'Se tiene un_
      ↪ incremento de ',porcent(trr_prod3[3]),'%')
print('Caso 4: ' , trr_prod4[3], 'moles de hidrógeno.' , 'Se tiene un_
      ↪ incremento de ',porcent(trr_prod4[3]),'%')
print('Caso 5: ' , trr_prod5[3], 'moles de hidrógeno.' , 'Se tiene un_
      ↪ incremento de ',porcent(trr_prod5[3]),'%')
```

Caso 1: 0.8747465620336865 moles de hidrógeno. Se tiene un incremento de 18.02706821094981 % 5T0

Caso 2: 0.8644690690665221 moles de hidrógeno. Se tiene un incremento de 16.64035528617657 %

Caso 3: 0.8484152574448596 moles de hidrógeno. Se tiene un incremento de 14.474260097519249 %

Caso 4: 0.8180560074750318 moles de hidrógeno. Se tiene un incremento de 10.377972758370898 %

Caso 5: 0.7411406343418204 moles de hidrógeno. Se tiene un incremento de 0.0 %

Oxidación

```
[44]: tro_prod1=oxidacion_njit(10**2,3,800,300,50,10**2,3,1800,10)
tro_prod2=oxidacion_njit(10**2,3,800,300,40,10**2,3,1800,10)
tro_prod3=oxidacion_njit(10**2,3,800,300,30,10**2,3,1800,10)
tro_prod4=oxidacion_njit(10**2,3,800,300,20,10**2,3,1800,10)
tro_prod5=oxidacion_njit(10**2,3,800,300,10,10**2,3,1800,10)
```

```
[49]: print('Caso 1: ' , tro_prod1[3], 'moles de hidrógeno.' , 'Se tiene un_
      ↪ incremento de ',porcent(tro_prod1[3]),'%','2D0')
print('Caso 2: ' , tro_prod2[3], 'moles de hidrógeno.' , 'Se tiene un_
      ↪ incremento de ',porcent(tro_prod2[3]),'%')
print('Caso 3: ' , tro_prod3[3], 'moles de hidrógeno.' , 'Se tiene un_
      ↪ incremento de ',porcent(tro_prod3[3]),'%')
print('Caso 4: ' , tro_prod4[3], 'moles de hidrógeno.' , 'Se tiene un_
      ↪ incremento de ',porcent(tro_prod4[3]),'%')
```

```
print('Caso 5: ' , tro_prod5[3], 'moles de hidrógeno.' , 'Se tiene un_
↳ incremento de ',porcent(tro_prod5[3]),'%')
```

Caso 1: 1.1375680270467687 moles de hidrógeno. Se tiene un incremento de 53.48882173448779 % 2D0

Caso 2: 1.104817093954971 moles de hidrógeno. Se tiene un incremento de 49.069831387145335 %

Caso 3: 1.0533718400920051 moles de hidrógeno. Se tiene un incremento de 42.1284694540417 %

Caso 4: 0.9603465856542029 moles de hidrógeno. Se tiene un incremento de 29.576836184005916 %

Caso 5: 0.7411406343418204 moles de hidrógeno. Se tiene un incremento de 0.0 %

Ambas etapas

```
[46]: tr_prod1=oxidacion_njit(10**2,3,800,300,50,10**2,3,1800,50)
tr_prod2=oxidacion_njit(10**2,3,800,300,40,10**2,3,1800,40)
tr_prod3=oxidacion_njit(10**2,3,800,300,30,10**2,3,1800,30)
tr_prod4=oxidacion_njit(10**2,3,800,300,20,10**2,3,1800,20)
tr_prod5=oxidacion_njit(10**2,3,800,300,10,10**2,3,1800,10)
```

```
[47]: print('Caso 1: ' , tr_prod1[3], 'moles de hidrógeno.' , 'Se tiene un incremento_
↳ de ',porcent(tr_prod1[3]),'%')
print('Caso 2: ' , tr_prod2[3], 'moles de hidrógeno.' , 'Se tiene un incremento_
↳ de ',porcent(tr_prod2[3]),'%')
print('Caso 3: ' , tr_prod3[3], 'moles de hidrógeno.' , 'Se tiene un incremento_
↳ de ',porcent(tr_prod3[3]),'%')
print('Caso 4: ' , tr_prod4[3], 'moles de hidrógeno.' , 'Se tiene un incremento_
↳ de ',porcent(tr_prod4[3]),'%')
print('Caso 5: ' , tr_prod5[3], 'moles de hidrógeno.' , 'Se tiene un incremento_
↳ de ',porcent(tr_prod5[3]),'%')
```

Caso 1: 1.626764666529521 moles de hidrógeno. Se tiene un incremento de 119.4947343528385 %

Caso 2: 1.5070087126404712 moles de hidrógeno. Se tiene un incremento de 103.33640375538036 %

Caso 3: 1.3472684585762689 moles de hidrógeno. Se tiene un incremento de 81.78310514207983 %

Caso 4: 1.1159125244195431 moles de hidrógeno. Se tiene un incremento de 50.56690629445028 %

Caso 5: 0.7411406343418204 moles de hidrógeno. Se tiene un incremento de 0.0 %

Anexo B5

May 31, 2021

1 Modelo de Comparación: Cinética Bulfin+Arifin

1.1 Etapa de Reducción

```
[1]: import numpy as np
dT=17.
Tred=1500.+273.15 #temperatura a la que se fijan los reactantes
Ts=np.array([750., 800., 850., 900., 950., 1000., 1100.])+273.15 #Este arreglo
    ↳ contiene las distintas Tox que ocupa Arifin
time=240.0 #Segundos que se mantiene a 240°C una vez que alcanza la temperatura
    ↳ de reducción
timer=np.zeros(len(Ts))
dt=0.02
d=np.zeros(len(Ts))
d_max=0.35
R=8.3145*10**(-3)# [kJ /mol K ] Constante universal de los gases
Ared= 720000. #+-360000 [1/s] Factor pre-exponencial de reducción
Ered=232. # +-5 [kJ/mol] Energía de activación de reducción
tiempo_total=np.zeros(len(Ts))
y_ox=0 #Para la reducción este valor es cero. En el trabajo de Arifin se señala
    ↳ que la presion parcial de O2 es inferior a 10ppm.
Aox=1.0
Eox=29.0
gamma=0.89
kred=np.zeros(len(Ts))
kox=np.zeros(len(Ts))
print_counter=0 #

#d_final=np.array([0.35, 0.35, 0.35, 0.35, 0.35, 0.35, 0.35]) #para la
    ↳ iteración de f(a) se consideró como
#condición inicial de iteración la reducción completa.

#Luego de 10 iteraciones se obtuvo una distribución de vacancias de
#D [0.34933276 0.34933308 0.3493334 0.34933371 0.34933402 0.3493343 0.34933472]
```

```

d_final=np.array([0.34933276, 0.34933308, 0.3493334, 0.34933371, 0.34933402, 0.
↪3493343, 0.34933472])
#este d_final es obtenido mediante proceso iterativo.

#----Al ejecutar este código se obtendrá el resultado de la iteración 11 de
↪f(a) considerando la reducción completa como condicion inicial

for i in range(len(Ts)):
    tt=0
    while timer[i] < time:
        d_n=d.copy()
        tt+=dt

        if Ts[i] < Tred:
            Ts[i]+=dT*dt

        if Ts[i] >= Tred:

            timer[i]+=dt

        kred[i]=(1-d_n[i]/d_final[i])** (1/3)*Ared*np.exp(-Ered/(R*Ts[i]))
        kox[i]=0 #y_ox=0 implica que kox=0

        d[i]=(d_n[i]/dt+d_max*kred[i])/(kred[i]+kox[i]+1/dt)

        tiempo_total[i]=tt

```

```

[2]: print('d_anterior: ',d_final)
      print('d_actual: ',d)
      print('diferencia: ',d_final-d)
      #iteracion 1

```

```

d_anterior: [0.34933276 0.34933308 0.3493334 0.34933371 0.34933402 0.3493343
0.34933472]
d_actual: [0.34933016 0.34933048 0.3493308 0.34933111 0.34933143 0.34933171
0.34933213]
diferencia: [2.59906523e-06 2.59814024e-06 2.59743308e-06 2.59543545e-06
2.59487111e-06 2.59213760e-06 2.59178144e-06]

```

```

[3]: umol_o2=(d/2)/172.115*10**6
      print('MODELO NUMÉRICO - micro-moles de O2 por g de CeO2:',umol_o2)

```

MODELO NUMÉRICO - micro-moles de O2 por g de CeO2: [1014.81614309 1014.81707538
1014.81800705 1014.81891341 1014.81981561
1014.82063696 1014.82185811]

1.1.1 Cálculos según estequiometría de la reducción

```
[4]: print('Considerando una reacción estequiométrica:')
print('-----')

umol_o2=(d/2)/172.115*10**6
print('MODELO NUMÉRICO - micro-moles de O2 por g de CeO2:',umol_o2)
umol_o2_arifin=0.056*np.array([750., 800., 850., 900., 950., 1000., 1100. ])+28.
↪99
print('RESULTADOS DE ARIFIN - micro-moles de O2 por g de CeO2:' ,umol_o2_arifin_
↪)
#print('Producción total de O2 en micro moles con 0,2g de CeO2', (d/2)/172.
↪115*10**6*0.2)
d_arifin=(0.056*np.array([750., 800., 850., 900., 950., 1000., 1100. ])+28.
↪99)*172.115*2*10**-6
```

Considerando una reacción estequiométrica:

MODELO NUMÉRICO - micro-moles de O2 por g de CeO2: [1014.81614309 1014.81707538
1014.81800705 1014.81891341 1014.81981561
1014.82063696 1014.82185811]

RESULTADOS DE ARIFIN - micro-moles de O2 por g de CeO2: [70.99 73.79 76.59 79.39
82.19 84.99 90.59]

1.2 Etapa de Oxidación

```
[5]: d_final=np.array([0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0]) #Asumimos inicialmente_
↪condición final como oxidación completa de la ceria.
#d_final se obtiene mediante iteración

#Luego de 1100 iteraciones se obtiene un d_final de [0.19972006 0.18600724 0.
↪17338739 0.16185165 0.1513547 0.14182989 0.12538496]
d_final=np.array([0.19972006, 0.18600724, 0.17338739, 0.16185165, 0.1513547, 0.
↪14182989, 0.12538496])
#El dox obtenido será la iteración 1101 de f(a).

dox=d.copy() #'d' corresponde a la distribución de vacancias final de la etapa_
↪de reducción
#Luego de correr el programa, dox será la distribución de vacancias final de la_
↪etapa de oxidación.
```

```

#La oxidación es isotérmica en el experimento de arifin:
Tox=np.array([750., 800., 850., 900., 950., 1000., 1100.])+273.15

#d_inicial es fijo, son los valores que se obtuvieron al final de la reducción
d_inicial=d.copy()

for i in range(len(Tox)):
    tt=0
    while tt < 600:
        d_n=dox.copy()
        kox[i]=0.3**gamma*(1-(d_inicial[i]-d_n[i])/
→(d_inicial[i]-d_final[i]))*Aox*np.exp(-Eox/(R*Tox[i]))

        if dox[i]>=0.35:
            break
        dox[i]=(d_n[i]/dt)/(1/dt+kox[i])

        tt+=dt
print('dox:',dox)
print('d_final',d_final)
print('diferencia entre dox y d_final;',dox-d_final)

```

```

dox: [0.19972986 0.1860172  0.17339739 0.1618616  0.15136452 0.14183954
      0.12539417]
d_final [0.19972006 0.18600724 0.17338739 0.16185165 0.1513547  0.14182989
      0.12538496]
diferencia entre dox y d_final; [9.79597246e-06 9.95965358e-06 1.00013409e-05
      9.94809173e-06
      9.82381484e-06 9.64852690e-06 9.20585035e-06]

```

1.3 Resultados segun estequiometría

```

[6]: print('Considerando una reacción estequiométrica:')
print('-----')

print('Moles de hidrógeno producidos:')
umol_h2=(d-dox)/172.115*10**6
print('MODELO NUMÉRICO - micro-moles de H2 por g de CeO2:',umol_h2)

```

```

umol_h2_arifin=0.059*np.array([750., 800., 850., 900., 950., 1000., 1100. ])+56.
↪45
print('RESULTADOS DE ARIFIN - micro-moles de H2 por g de CeO2:', umol_h2_arifin)

print('Diferencia de resultados:', umol_h2-umol_h2_arifin)

```

Considerando una reacción estequiométrica:

```

-----
Moles de hidrógeno producidos:
MODELO NUMÉRICO - micro-moles de H2 por g de CeO2: [ 869.18807171  948.86141363
1022.18523212 1089.21079786 1150.20132652
1205.54378953 1301.09497933]
RESULTADOS DE ARIFIN - micro-moles de H2 por g de CeO2: [100.7  103.65 106.6
109.55 112.5  115.45 121.35]
Diferencia de resultados: [ 768.48807171  845.21141363  915.58523212
979.66079786 1037.70132652
1090.09378953 1179.74497933]

```

```

[7]: #Razon de moles de H2 sobre moles de O2 producidos
print('Razon de moles de H2 sobre moles de O2 producidos:')
print('-----')
#Arifin:
razon_arifin=(umol_h2_arifin)/(umol_o2_arifin)
print('moles de H2/moles de O2 segun arifin:', razon_arifin)

razon_modelo=umol_h2/umol_o2
print('moles de H2/moles de O2 segun el modelo numérico:', razon_modelo)

```

Razon de moles de H2 sobre moles de O2 producidos:

```

-----
moles de H2/moles de O2 segun arifin: [1.41850965 1.40466188 1.39182661
1.37989671 1.36877966 1.35839511
1.33955183]
moles de H2/moles de O2 segun el modelo numérico: [0.85649807 0.93500734
1.00725965 1.07330558 1.13340448 1.18793779
1.28209199]

```

1.4 Función para ajuste lineal

```

[8]: import matplotlib.pyplot as plt
plt.rcParams['figure.dpi'] = 200
def b1(x,y):
    t1=x-np.average(x)
    t2=y-np.average(y)
    sxy=sum(t1*t2)
    sxx=sum(t1*t1)
    return sxy/sxx
def b0(x,y):

```

```

    return np.average(y)-b1(x,y)*np.average(x)
def reg_lin(x,y):

    return b0(x,y)+b1(x,y)*x

```

1.5 Modelo vs Experimento de Arifin: oxígeno producido

```

[9]: T_ox=np.array([750., 800., 850., 900., 950., 1000., 1100.])
puntosx=np.linspace(T_ox[0],T_ox[-1],100)

puntosy1=b0(T_ox,umol_o2)+b1(T_ox,umol_o2)*puntosx
puntosy2=b0(T_ox,umol_o2_arifin)+b1(T_ox,umol_o2_arifin)*puntosx

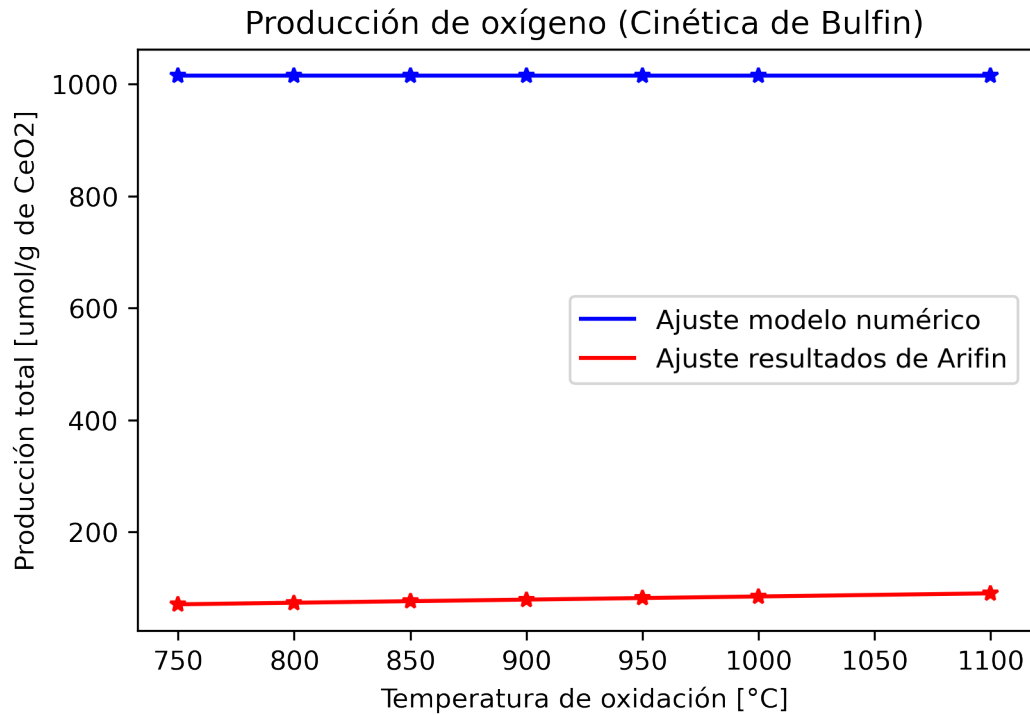
plt.plot(puntosx,puntosy1,'b')#rectas
plt.plot(puntosx,puntosy2,'r') #rectas

plt.plot(T_ox,umol_o2,'b*') #puntos
plt.plot(T_ox,umol_o2_arifin,'r*')#puntos

#Datos extra
plt.title('Producción de oxígeno (Cinética de Bulfin)')
plt.xlabel('Temperatura de oxidación [°C]')
plt.ylabel('Producción total [umol/g de CeO2]')
plt.legend(('Ajuste modelo numérico','Ajuste resultados de Arifin'),
          prop = {'size': 10}, loc='center right')

plt.show()

```



1.6 Modelo vs Experimento de Arifin: hidrógeno producido

```
[10]: T_ox=np.array([750., 800., 850., 900., 950., 1000., 1100.])
puntosx=np.linspace(T_ox[0],T_ox[-1],100)

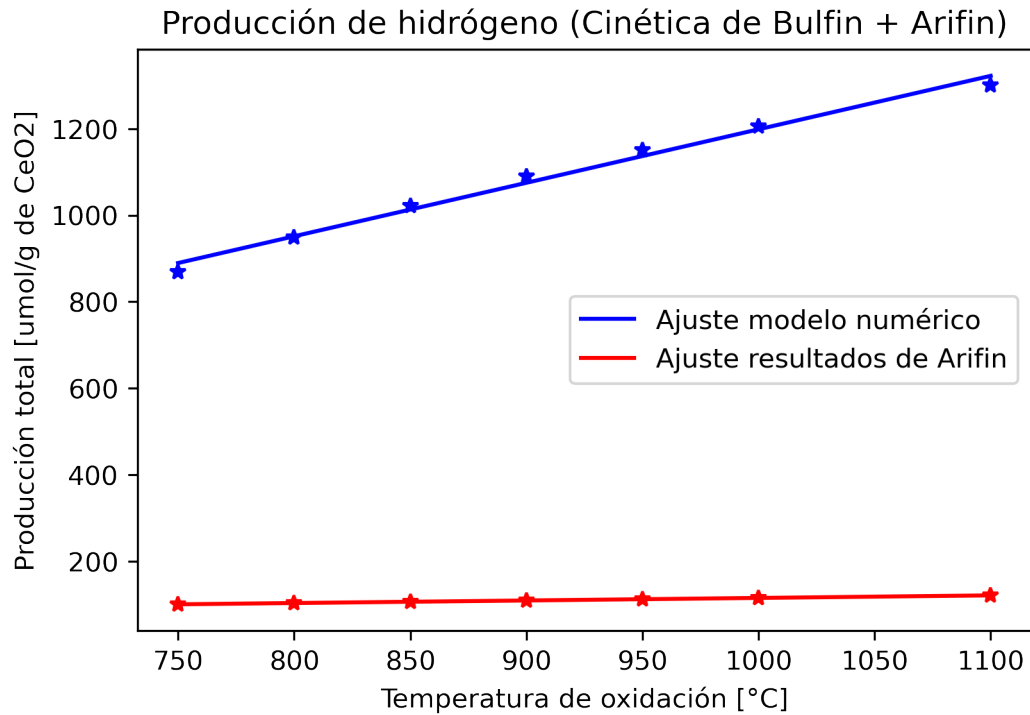
puntosy1=b0(T_ox,umol_h2)+b1(T_ox,umol_h2)*puntosx
puntosy2=b0(T_ox,umol_h2_arifin)+b1(T_ox,umol_h2_arifin)*puntosx

plt.plot(puntosx,puntosy1,'b')#rectas
plt.plot(puntosx,puntosy2,'r') #rectas

plt.plot(T_ox,umol_h2,'b*') #puntos
plt.plot(T_ox,umol_h2_arifin,'r*')#puntos

#Datos extra
plt.title('Producción de hidrógeno (Cinética de Bulfin + Arifin)')
plt.xlabel('Temperatura de oxidación [°C]')
plt.ylabel('Producción total [umol/g de CeO2]')
plt.legend(('Ajuste modelo numérico','Ajuste resultados de Arifin'),
          prop = {'size': 10}, loc='center right')

plt.show()
```



1.7 Producción de O2 y H2

```
[22]: T_ox=np.array([750., 800., 850., 900., 950., 1000., 1100.])
puntosx=np.linspace(T_ox[0],T_ox[-1],100)
#Oxígeno
puntosy102=b0(T_ox,umol_o2)+b1(T_ox,umol_o2)*puntosx #Ajuste lineal resultados
puntosy202=b0(T_ox,umol_o2_arifin)+b1(T_ox,umol_o2_arifin)*puntosx #Ajuste
↪lineal arifin
#Hidrógeno
puntosy1h2=b0(T_ox,umol_h2)+b1(T_ox,umol_h2)*puntosx
puntosy2h2=b0(T_ox,umol_h2_arifin)+b1(T_ox,umol_h2_arifin)*puntosx

plt.plot(puntosx,puntosy102,'r')#Modelo O2
plt.plot(puntosx,puntosy202,'r--') #Arifin O2

plt.plot(puntosx,puntosy1h2,'b')#Modelo H2
plt.plot(puntosx,puntosy2h2,'b--') #Arifin H2

plt.plot(T_ox,umol_o2,'r.')
plt.plot(T_ox,umol_o2_arifin,'r.')
```

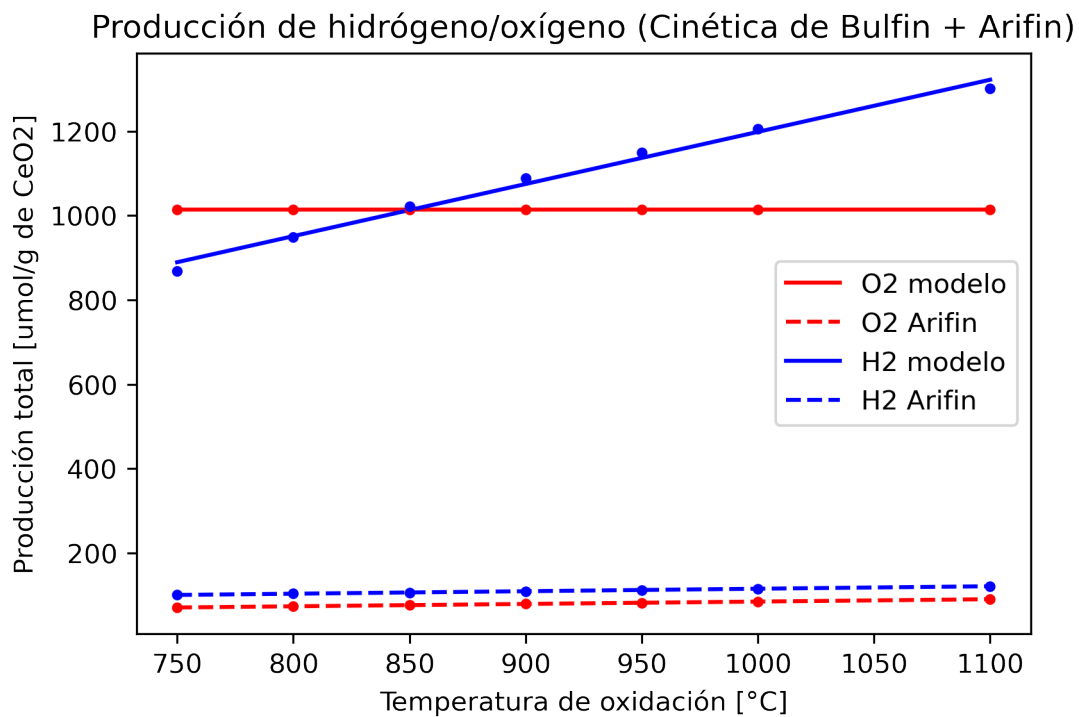
```

plt.plot(T_ox,umol_h2,'b.') #puntos
plt.plot(T_ox,umol_h2_arifin,'b.')#puntos

#Datos extra
plt.title('Producción de hidrógeno/oxígeno (Cinética de Bulfin + Arifin)')
plt.xlabel('Temperatura de oxidación [°C]')
plt.ylabel('Producción total [umol/g de CeO2]')
plt.legend(('O2 modelo','O2 Arifin','H2 modelo','H2 Arifin'),
           prop = {'size': 10}, loc='center right')

plt.show()

```



Anexo B6

May 31, 2021

1 Modelo de Comparación: Cinética Arifin (sólo oxidación)

1.1 Concentración de vacancia (delta) inicial de oxidación

```
[1]: import numpy as np
d_arifin=(0.056*np.array([750., 800., 850., 900., 950., 1000., 1100. ])+28.
    ↳99)*172.115*2*10**-6
d=d_arifin
print(d_arifin)
```

```
[0.02443689 0.02540073 0.02636458 0.02732842 0.02829226 0.02925611
 0.0311838 ]
```

1.2 Reacción de Oxidación

```
[2]: dT=17.
Tred=1500.+273.15 #temperatura a la que se fijan los reactantes
Ts=np.array([750., 800., 850., 900., 950., 1000., 1100.])+273.15 #Este arreglo
    ↳contiene las distintas Tox que ocupa Arifin
time=240.0 #estos son los segundos que se mantiene a 240°C una vez que alcanza
    ↳la temperatura de reducción
timer=np.zeros(len(Ts))
dt=0.02 #paso de tiempo
d_max=0.35 #Concentración de vacancias máxima.
R=8.3145*10**(-3)# [kJ /mol K ] Constante universal de los gases
Ared= 720000. #+-360000 [1/s] Factor pre-exponencial de reducción
Ered=232. # +-5 [kJ/mol] Energía de activación de reducción
tiempo_total=np.zeros(len(Ts))
y_ox=0*10**-7 #Fracción molar promedio de oxígeno CORREGIR CON CADA ITERACIÓN
Aox=1.0
Eox=29.0
gamma=0.89
kred=np.zeros(len(Ts))
kox=np.zeros(len(Ts))
print_counter=0 #
```

```

#d_final=np.array([0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0]) #Asumimos condición
    ↪ final como oxidación completa de la ceria.
#d_final se obtiene mediante iteración

#Luego de 120 iteraciones se obtiene un d_final de
d_final=np.array([0.01192768, 0.01139412, 0.01089149, 0.01042392, 0.00999278, 0.
    ↪ 00959759, 0.00890827])
#El dox obtenido será la iteración 121 de f(a). Este f(a) solo corresponde a la
    ↪ componente de la oxidación, (esto debido a que se desprecia la reducción por
    ↪ las temperaturas de trabajo.)

dox=d.copy() #'d' corresponde a la distribución de vacancias final de la etapa
    ↪ de reducción
#Luego de correr el programa, dox será la distribución de vacancias final de la
    ↪ etapa de oxidación.

#La oxidación es isotérmica en el experimento de arifin:
Tox=np.array([750., 800., 850., 900., 950., 1000., 1100.])+273.15

#d_inicial es fijo, son los valores que se obtuvieron al final de la reducción
d_inicial=d.copy()

for i in range(len(Tox)):
    tt=0
    while tt < 600:
        d_n=dox.copy()
        kox[i]=0.3**gamma*(1-(d_inicial[i]-d_n[i])/
    ↪ (d_inicial[i]-d_final[i]))*Aox*np.exp(-Eox/(R*Tox[i]))
            #(d_n[i]**2/d_max)
        if dox[i]>=0.35:
            break
        dox[i]=(d_n[i]/dt)/(1/dt+kox[i]) #como son estáticas no requiere
    ↪ recorrer

    tt+=dt

```

1.3 Resultados segun estequiometría

```
[3]: print('Considerando una reacción estequiométrica:')
print('-----')

print('Moles de hidrógeno producidos:')
umol_h2=(d-dox)/172.115*10**6
print('MODELO NUMÉRICO - micro-moles de H2 por g de CeO2:',umol_h2)
umol_h2_arifin=0.059*np.array([750., 800., 850., 900., 950., 1000., 1100. ])+56.
    ↪45
print('RESULTADOS DE ARIFIN - micro-moles de H2 por g de CeO2:', umol_h2_arifin)

print('Diferencia de resultados:', umol_h2-umol_h2_arifin)
```

Considerando una reacción estequiométrica:

```
-----
Moles de hidrógeno producidos:
MODELO NUMÉRICO - micro-moles de H2 por g de CeO2: [ 72.62499041  81.32342947
89.84276989  98.15888352 106.26369696
114.15990295 129.36565844]
RESULTADOS DE ARIFIN - micro-moles de H2 por g de CeO2: [100.7  103.65 106.6
109.55 112.5  115.45 121.35]
Diferencia de resultados: [-28.07500959 -22.32657053 -16.75723011 -11.39111648
-6.23630304
-1.29009705  8.01565844]
```

1.4 Función para ajuste lineal

```
[4]: import matplotlib.pyplot as plt
plt.rcParams['figure.dpi'] = 300
def b1(x,y):
    t1=x-np.average(x)
    t2=y-np.average(y)
    sxy=sum(t1*t2)
    sxx=sum(t1*t1)
    return sxy/sxx
def b0(x,y):
    return np.average(y)-b1(x,y)*np.average(x)
def reg_lin(x,y):
    return b0(x,y)+b1(x,y)*x
```

1.5 Modelo vs Experimento de Arifin: hidrógeno producido

```
[5]: T_ox=np.array([750., 800., 850., 900., 950., 1000., 1100.])
puntosx=np.linspace(T_ox[0],T_ox[-1],100)

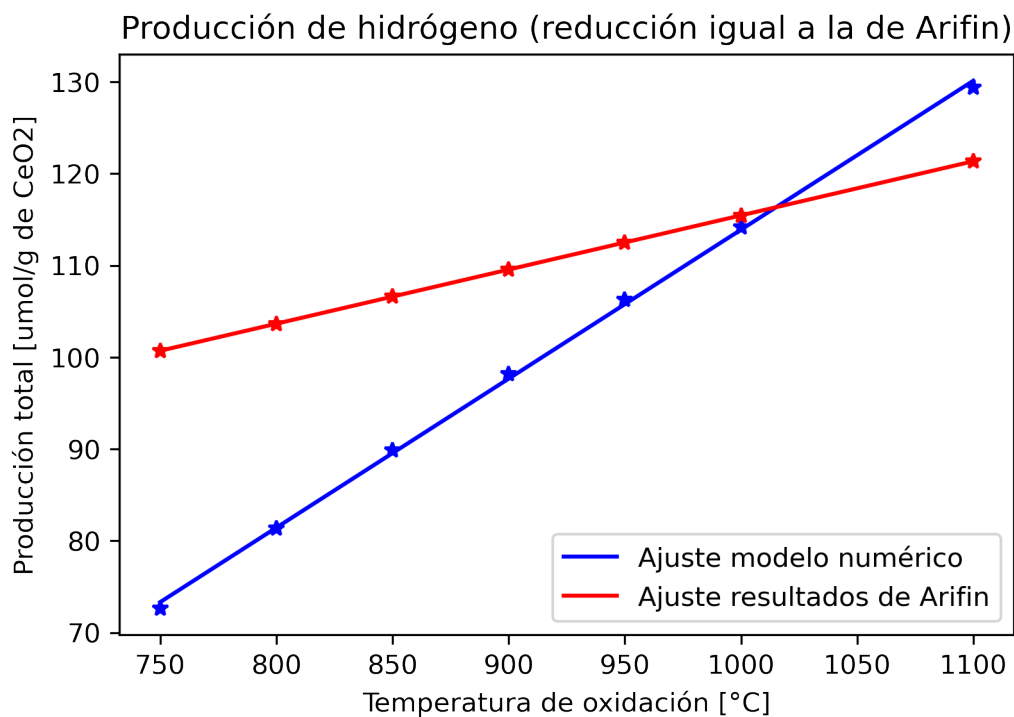
puntosy1=b0(T_ox,umol_h2)+b1(T_ox,umol_h2)*puntosx
puntosy2=b0(T_ox,umol_h2_arifin)+b1(T_ox,umol_h2_arifin)*puntosx

plt.plot(puntosx,puntosy1,'b')#rectas
plt.plot(puntosx,puntosy2,'r') #rectas

plt.plot(T_ox,umol_h2,'b*') #puntos
plt.plot(T_ox,umol_h2_arifin,'r*')#puntos

#Datos extra
plt.title('Producción de hidrógeno (reducción igual a la de Arifin)')
plt.xlabel('Temperatura de oxidación [°C]')
plt.ylabel('Producción total [umol/g de CeO2]')
plt.legend(('Ajuste modelo numérico','Ajuste resultados de Arifin'),
           prop = {'size': 10}, loc='lower right')

plt.show()
```



Anexo B7

May 31, 2021

1 Modelo de Comparación: Variación nred=1.65

1.1 Etapa de Reducción

```
[1]: import numpy as np
dT=17.
Tred=1500.+273.15 #temperatura a la que se fijan los reactantes
Ts=np.array([750., 800., 850., 900., 950., 1000., 1100.])+273.15 #Este arreglo
    ↪ contiene las distintas Tox que ocupa Arifin
time=240.0 #estos son los segundos que se mantiene a 240°C una vez que alcanza
    ↪ la temperatura de reducción
timer=np.zeros(len(Ts))
dt=0.02 #paso de tiempo
d=np.zeros(len(Ts))
d_max=0.35 #Concentración de vacancias máxima.
R=8.3145*10**(-3) # [kJ /mol K ] Constante universal de los gases
Ared= 720000. #+-360000 [1/s] Factor pre-exponencial de reducción
Ered=232. # +-5 [kJ/mol] Energía de activación de reducción
tiempo_total=np.zeros(len(Ts))
y_ox=0 #Fracción molar promedio de oxígeno CORREGIR CON CADA ITERACIÓN
Aox=1.0
Eox=29.0
gamma=0.89
kred=np.zeros(len(Ts))
kox=np.zeros(len(Ts))
print_counter=0 #

#d_final=np.array([0.35, 0.35, 0.35, 0.35, 0.35, 0.35, 0.35]) #para la
    ↪ iteración de f(a) se consideró como
#condición inicial de iteración la reducción completa.

d_final=np.array([0.0298415, 0.02984592, 0.02985032, 0.02985469, 0.02985896, 0.
    ↪ 02986298, 0.02986875])
```

```

#----Al ejecutar este código se obtendrá el resultado de la iteración 1651 de
→f(a) considerando la reducción completa como condicion inicial

#este d_final es obtenido mediante proceso iterativo.

for i in range(len(Ts)):
    tt=0
    while timer[i] < time:
        d_n=d.copy()
        tt+=dt

        if Ts[i] < Tred:
            Ts[i]+=dT*dt

        if Ts[i] >= Tred:

            timer[i]+=dt

        kred[i]=(1-d_n[i]/d_final[i])**1.65*Ared*np.exp(-Ered/(R*Ts[i]))

        kox[i]=y_ox**gamma*(1-d_n[i]/d_final[i])*Aox*np.exp(-Eox/(R*Ts[i]))
            #el término f(a) es en este caso (reducción) d(t)/d_final, puesto
→que d_inicial=0

        d[i]=(d_n[i]/dt+d_max*kred[i])/(kred[i]+kox[i]+1/dt) #como son
→estáticas no requiere recorrer x

        tiempo_total[i]=tt

```

```

[2]: print('d_anterior: ',d_final)
      print('d_actual: ',d)
      print('diferencia: ',d_final-d)
      #iteracion 1

```

```

d_anterior: [0.0298415  0.02984592 0.02985032 0.02985469 0.02985896 0.02986298
0.02986875]
d_actual: [0.02983156 0.02983598 0.02984038 0.02984475 0.02984902 0.02985304
0.02985881]
diferencia: [9.93681352e-06 9.93804238e-06 9.93926149e-06 9.94047754e-06
9.94166677e-06 9.94278363e-06 9.94438025e-06]

```

```

[3]: umol_o2=(d/2)/172.115*10**6
      print('MODELO NUMÉRICO - micro-moles de O2 por g de CeO2:',umol_o2)

```

```

MODELO NUMÉRICO - micro-moles de O2 por g de CeO2: [86.661718  86.67455468

```

```
86.68733329 86.70002476 86.7124258 86.72410079
86.7408582 ]
```

1.2 Cálculos según estequiometría de la reducción

```
[4]: print('Considerando una reacción estequiométrica:')
print('-----')

umol_o2=(d/2)/172.115*10**6
print('MODELO NUMÉRICO - micro-moles de O2 por g de CeO2:',umol_o2)
umol_o2_arifin=0.056*np.array([750., 800., 850., 900., 950., 1000., 1100. ])+28.
    ↪99
print('RESULTADOS DE ARIFIN - micro-moles de O2 por g de CeO2:' ,umol_o2_arifin_
    ↪)
#print('Producción total de O2 en micro moles con 0,2g de CeO2', (d/2)/172.
    ↪115*10**6*0.2)
d_arifin=(0.056*np.array([750., 800., 850., 900., 950., 1000., 1100. ])+28.
    ↪99)*172.115*2*10**-6

#print('-La cantidad de oxígeno obtenida por Arifin corresponderia a una_
    ↪concentración de vacancias de:')
#print(d_arifin)
#print('La concentración de vacancias obtenidas mediante el modelo numérico es:_
    ↪')
#print(d)

#RESULTADOS
```

Considerando una reacción estequiométrica:

```
-----
MODELO NUMÉRICO - micro-moles de O2 por g de CeO2: [86.661718 86.67455468
86.68733329 86.70002476 86.7124258 86.72410079
86.7408582 ]
RESULTADOS DE ARIFIN - micro-moles de O2 por g de CeO2: [70.99 73.79 76.59 79.39
82.19 84.99 90.59]
```

```
[5]: print('Verificación de fracción molar de oxidante:')
print('-----')
moles_producto_o2=(d/2)/172.115*0.2
moles_he=500*tiempo_total/60*0.18/4.0026 #500 standar cm3/min sccm , 0.18 g/cm3_
    ↪4.0026 g/mol
#print('moles de O2 totales:', moles_producto_o2)
#print('moles de helio totales', moles_he)
y_ox_r=moles_producto_o2/(moles_producto_o2+moles_he)
#print('fracción molar de oxidante: ',y_ox_r)
y_ox_prom=np.average(y_ox_r)
print('fracción molar de oxidante promedio:',y_ox_prom )
```

Verificación de fracción molar de oxidante:

fracción molar de oxidante promedio: 1.684385738291143e-07

1.3 Etapa de Oxidación

```
[4]: #d_final=np.array([0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0]) #Asumimos condición
    ↪ final como oxidación completa de la ceria.
    #d_final se obtiene mediante iteración

    #Luego de 140 iteraciones se obtiene un d_final de
    d_final=np.array([0.01476997, 0.01359096, 0.0125308, 0.0115819, 0.01073468, 0.
    ↪ 00997889, 0.0087017 ])
    #El dox obtenido será la iteración 141 de f(a). Este f(a) solo corresponde a la
    ↪ componente de la oxidación, (esto debido a que se desprecia la reducción por
    ↪ las temperaturas de trabajo.)

    dox=d.copy() #'d' corresponde a la distribución de vacancias final de la etapa
    ↪ de reducción
    #Luego de correr el programa, dox será la distribución de vacancias final de la
    ↪ etapa de oxidación.

    #La oxidación es isotérmica en el experimento de arifin:
    Tox=np.array([750., 800., 850., 900., 950., 1000., 1100.])+273.15

    #d_inicial es fijo, son los valores que se obtuvieron al final de la reducción
    d_inicial=d.copy()

    for i in range(len(Tox)):
        tt=0
        while tt < 600:
            d_n=dox.copy()
            kox[i]=0.3**gamma*(1-(d_inicial[i]-d_n[i])/
            ↪ (d_inicial[i]-d_final[i]))*Aox*np.exp(-Eox/(R*Tox[i]))
                #(d_n[i]**2/d_max)
            if dox[i]>=0.35:
                break
            dox[i]=(d_n[i]/dt)/(1/dt+kox[i]) #como son estáticas no requiere
            ↪ recorrer

            tt+=dt
```

```

tiempo: 4.000000000000003
Temperatura instantánea: [1023.15 1073.15 1123.15 1173.15 1223.15 1273.15
1373.15]
Distribución de vacancias: [0.02856543 0.02983598 0.02984038 0.02984475
0.02984902 0.02985304
0.02985881]
Iteración de caso número 1

```

1.4 Resultados segun estequiometría

```

[7]: print('Considerando una reacción estequiométrica:')
print('-----')

print('Moles de hidrógeno producidos:')
umol_h2=(d-dox)/172.115*10**6
print('MODELO NUMÉRICO - micro-moles de H2 por g de CeO2:',umol_h2)
umol_h2_arifin=0.059*np.array([750., 800., 850., 900., 950., 1000., 1100. ])+56.
↪45
print('RESULTADOS DE ARIFIN - micro-moles de H2 por g de CeO2:', umol_h2_arifin)
print('Diferencia de resultados:', umol_h2-umol_h2_arifin)

```

```

Considerando una reacción estequiométrica:
-----
Moles de hidrógeno producidos:
MODELO NUMÉRICO - micro-moles de H2 por g de CeO2: [ 87.45368255  94.32990601
100.51603463 106.05593358 111.00471887
115.42096859 122.87857882]
RESULTADOS DE ARIFIN - micro-moles de H2 por g de CeO2: [100.7  103.65 106.6
109.55 112.5  115.45 121.35]
Diferencia de resultados: [-13.24631745 -9.32009399 -6.08396537 -3.49406642
-1.49528113
-0.02903141  1.52857882]

```

```

[8]: #Razon de moles de H2 sobre moles de O2 producidos
print('Razon de moles de H2 sobre moles de O2 producidos:')
print('-----')
#Arifin:
razon_arifin=(umol_h2_arifin)/(umol_o2_arifin)
print('moles de H2/moles de O2 segun arifin:', razon_arifin)

razon_modelo=umol_h2/umol_o2
print('moles de H2/moles de O2 segun el modelo numérico:', razon_modelo)

```

```

Razon de moles de H2 sobre moles de O2 producidos:
-----
moles de H2/moles de O2 segun arifin: [1.41850965 1.40466188 1.39182661
1.37989671 1.36877966 1.35839511

```

```

1.33955183]
moles de H2/moles de O2 segun el modelo numérico: [1.00913857 1.08832294
1.1595239 1.22325148 1.28014777 1.33089842
1.41661705]

```

1.5 Función para ajuste lineal

```

[9]: import matplotlib.pyplot as plt
plt.rcParams['figure.dpi'] = 300
def b1(x,y):
    t1=x-np.average(x)
    t2=y-np.average(y)
    sxy=sum(t1*t2)
    sxx=sum(t1*t1)
    return sxy/sxx
def b0(x,y):
    return np.average(y)-b1(x,y)*np.average(x)
def reg_lin(x,y):
    return b0(x,y)+b1(x,y)*x

```

1.6 Modelo vs Experimento de Arifin: oxígeno producido

```

[10]: T_ox=np.array([750., 800., 850., 900., 950., 1000., 1100.])
puntosx=np.linspace(T_ox[0],T_ox[-1],100)

puntosy1=b0(T_ox,umol_o2)+b1(T_ox,umol_o2)*puntosx
puntosy2=b0(T_ox,umol_o2_arifin)+b1(T_ox,umol_o2_arifin)*puntosx

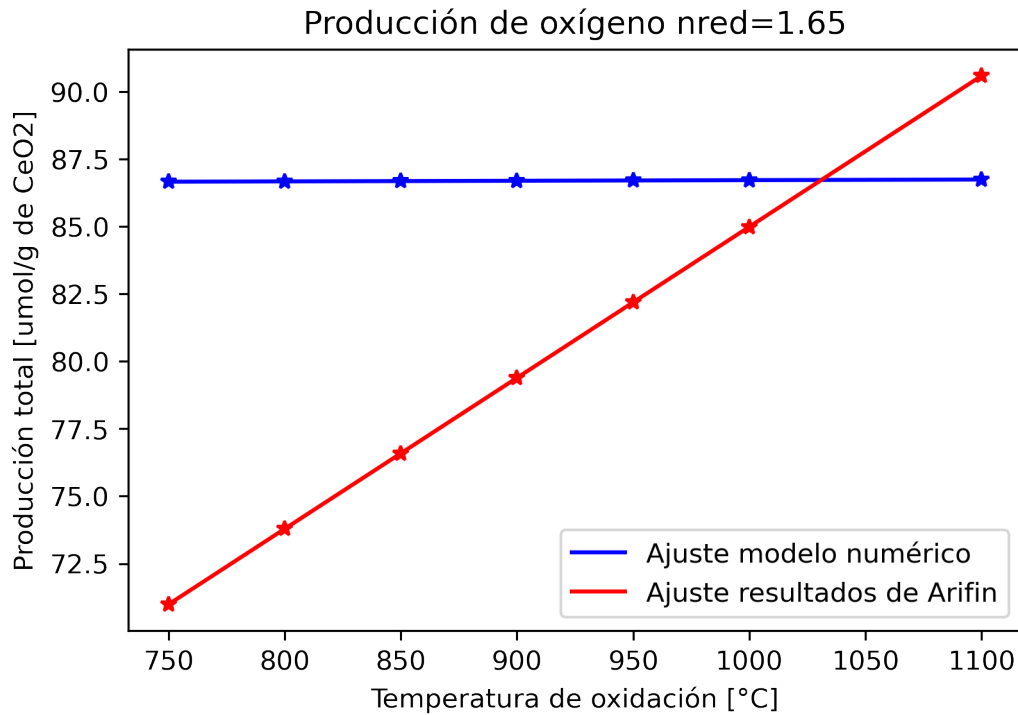
plt.plot(puntosx,puntosy1,'b')#rectas
plt.plot(puntosx,puntosy2,'r') #rectas

plt.plot(T_ox,umol_o2,'b*') #puntos
plt.plot(T_ox,umol_o2_arifin,'r*')#puntos

#Datos extra
plt.title('Producción de oxígeno nred=1.65')
plt.xlabel('Temperatura de oxidación [°C]')
plt.ylabel('Producción total [umol/g de CeO2]')
plt.legend(('Ajuste modelo numérico','Ajuste resultados de Arifin'),
          prop = {'size': 10}, loc='lower right')

plt.show()

```



1.7 Modelo vs Experimento de Arifin: hidrógeno producido

```
[11]: T_ox=np.array([750., 800., 850., 900., 950., 1000., 1100.])
puntosx=np.linspace(T_ox[0],T_ox[-1],100)

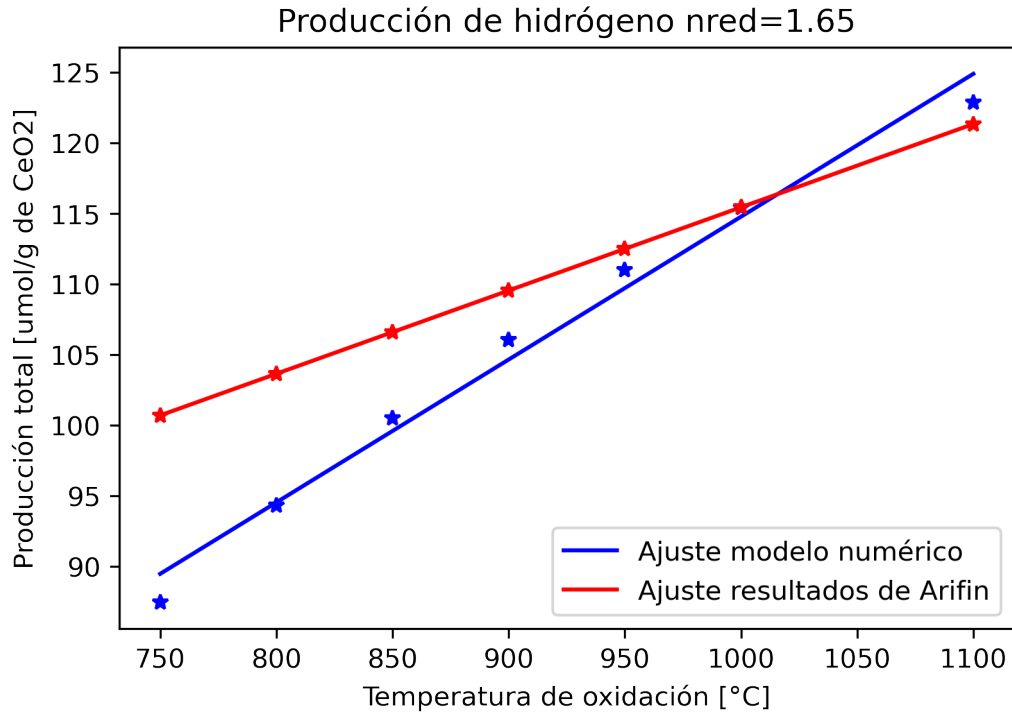
puntosy1=b0(T_ox,umol_h2)+b1(T_ox,umol_h2)*puntosx
puntosy2=b0(T_ox,umol_h2_arifin)+b1(T_ox,umol_h2_arifin)*puntosx

plt.plot(puntosx,puntosy1,'b')#rectas
plt.plot(puntosx,puntosy2,'r') #rectas

plt.plot(T_ox,umol_h2,'b*') #puntos
plt.plot(T_ox,umol_h2_arifin,'r*')#puntos

#Datos extra
plt.title('Producción de hidrógeno nred=1.65')
plt.xlabel('Temperatura de oxidación [°C]')
plt.ylabel('Producción total [umol/g de CeO2]')
plt.legend(('Ajuste modelo numérico','Ajuste resultados de Arifin'),
          prop = {'size': 10}, loc='lower right')

plt.show()
```



1.8 Producción de O2 y H2

```
[16]: T_ox=np.array([750., 800., 850., 900., 950., 1000., 1100.])
puntosx=np.linspace(T_ox[0],T_ox[-1],100)
#Oxígeno
puntosy102=b0(T_ox,umol_o2)+b1(T_ox,umol_o2)*puntosx #Ajuste lineal resultados
puntosy202=b0(T_ox,umol_o2_arifin)+b1(T_ox,umol_o2_arifin)*puntosx #Ajuste
↪ lineal arifin
#Hidrógeno
puntosy1h2=b0(T_ox,umol_h2)+b1(T_ox,umol_h2)*puntosx
puntosy2h2=b0(T_ox,umol_h2_arifin)+b1(T_ox,umol_h2_arifin)*puntosx

plt.plot(puntosx,puntosy102,'r') #Modelo O2
plt.plot(puntosx,puntosy202,'r--') #Arifin O2

plt.plot(puntosx,puntosy1h2,'b') #Modelo H2
plt.plot(puntosx,puntosy2h2,'b--') #Arifin H2

plt.plot(T_ox,umol_o2,'r.')
plt.plot(T_ox,umol_o2_arifin,'r.')
```

```

plt.plot(T_ox,umol_h2,'b.') #puntos
plt.plot(T_ox,umol_h2_arifin,'b.')#puntos

#Datos extra
plt.title('Producción de hidrógeno/oxígeno (Cinética de Bulfin + Arifin)')
plt.xlabel('Temperatura de oxidación [°C]')
plt.ylabel('Producción total [umol/g de CeO2]')
plt.legend(('O2 modelo','O2 Arifin','H2 modelo','H2 Arifin'),
           prop = {'size': 10}, loc='upper left')

plt.show()

```

