

UNIVERSIDADE FEDERAL DE SÃO CARLOS
CENTRO DE CIÊNCIAS EXATAS E DE TECNOLOGIA
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA E
ENGENHARIA DE MATERIAIS

MODELAGEM TERMODINÂMICA COMPUTACIONAL DE LIGAS
MULTICOMPONENTES CÚBICAS DE CORPO CENTRADO PARA
ARMAZENAGEM DE HIDROGÊNIO

Otávio Abreu Pedroso

São Carlos

2021

UNIVERSIDADE FEDERAL DE SÃO CARLOS
CENTRO DE CIÊNCIAS EXATAS E DE TECNOLOGIA
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA E
ENGENHARIA DE MATERIAIS

MODELAGEM TERMODINÂMICA COMPUTACIONAL DE LIGAS
MULTICOMPONENTES CÚBICAS DE CORPO CENTRADO PARA
ARMAZENAGEM DE HIDROGÊNIO

Otávio Abreu Pedroso

Dissertação apresentada
ao Programa de Pós-Graduação em Ciência
e Engenharia de Materiais como requisito
parcial à obtenção do título de MESTRE EM
CIÊNCIA E ENGENHARIA DE MATERIAIS

Orientador: Dr. Guilherme Zepon

Agência Financiadora: CNPq - Processo: 132353/2020-2

São Carlos

2021

À todos que perderam alguém querido nestes tempos sombrios.

Tempos melhores virão.

VITAE DO CANDIDATO

Bacharel em Engenharia Física pela UFSCar (2019).



UNIVERSIDADE FEDERAL DE SÃO CARLOS

Centro de Ciências Exatas e de Tecnologia
Programa de Pós-Graduação em Ciência e Engenharia de Materiais

Folha de Aprovação

Defesa de Dissertação de Mestrado do candidato Otávio Abreu Pedroso, realizada em 20/09/2021.

Comissão Julgadora:

Prof. Dr. Guilherme Zepon (UFSCar)

Prof. Dr. Walter José Botta Filho (UFSCar)

Prof. Dr. Ricardo Floriano (UNICAMP)

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001.

O Relatório de Defesa assinado pelos membros da Comissão Julgadora encontra-se arquivado junto ao Programa de Pós-Graduação em Ciência e Engenharia de Materiais.

AGRADECIMENTOS

Primeiramente, gostaria de agradecer ao meu orientador Prof. Guilherme Zepon, pela orientação, ajuda, conselhos, por me incentivar e principalmente por acreditar no meu potencial. Fico grato por todo aprendizado até o momento e na expectativa para os próximos anos!

Agradeço também o Prof. Francisco Gil Coury, pelas discussões, questionamentos e apontamentos sobre meu trabalho e as aulas que tive o prazer de frequentar.

Ao Prof. Walter José Botta Filho, muito obrigado pelos questionamentos e sugestões na qualificação que ajudaram a aprimorar essa dissertação.

Gostaria de agradecer aos meus colegas do LH₂M, em especial ao Renato Belli Strozi, pelas discussões construtivas e pelas ótimas indicações no clube do livro.

Aos meus amigos, em especial para Leonardo Lemes, Lucas Francisco, Conrado Silva, Paulo Quirino e Eduardo Neres, que em muitos momentos por mais difícil que seja na pandemia mantiveram contato (à distância), com certeza foram fundamentais as conversas que tivemos!

À Camila Corrêa Ianagui, minha companheira! Por ser minha amiga em todos os momentos, por me acalmar, pelas risadas e por toda cumplicidade do nosso relacionamento. Sem você não conseguiria ter concluído esse desafio nesse momento tão conturbado.

À minha família, em especial aos meus pais, por todo suporte e incentivo para chegar até aqui.

Este trabalho é financiado em parte pelo Instituto Serrapilheira (bolsa número Serra-1709-17362) e em parte pelo Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) pela bolsa de estudos, processo: 132353/2020-2.

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001.

RESUMO

Recentemente, ligas metálicas multicomponentes para armazenamento de hidrogênio tem chamado a atenção pela possibilidade de se explorar um grande campo composicional, que pode permitir o controle das propriedades de armazenagem pelo controle da composição da liga, podendo satisfazer as necessidades para cada aplicação. O laboratório de hidrogênio em metais do DEMa/UFSCar já possui vasta experiência na síntese, caracterização e avaliação das propriedades destas ligas. Um próximo passo é desenvolver métodos e modelos para prever as propriedades termodinâmicas que determinam suas condições de operação (pressão e temperatura para absorção/dessorção) em função unicamente da composição química, visando aprimorar as ferramentas disponíveis para realização do design de ligas para armazenagem de hidrogênio com propriedades adequadas para diferentes aplicações. Recentemente, Zepon et al. propuseram um modelo termodinâmico para calcular diagramas pressão-composição-temperatura (PCT) de ligas multicomponentes cúbicas de corpo centrado (CCC). Neste modelo, a entropia configuracional das fases de interesse é modelada através do modelo de solução sólida intersticial com efeito de boqueio de sítios, e a entalpia das fases é modelada com base em dados experimentais e cálculos de *ab initio* através de teoria do funcional densidade (DFT). Este trabalho de mestrado teve como objetivo adicionar novos elementos químicos ao modelo recentemente proposto através de cálculos de DFT e desenvolver um programa em código aberto para o cálculo de diagramas PCT com base no modelo e nos dados obtidos por DFT. Foi realizada a incorporação dos elementos Mg, Al, Sc, Mn, Fe, Co, Cu, Zn, Mo e Pd ao modelo e a sua implementação em um software de código aberto na linguagem de programação Python. Utilizando esse software, as ligas reportadas na literatura foram avaliadas de forma que foi possível observar que as ligas do sistema TiVNbCr são melhor descritas pelo modelo, além de que algumas ligas contendo Mg apresentam discrepância entre a estrutura cristalina observada experimentalmente e as obtidas pelo modelo.

Palavras-chave: Sustentabilidade; Energias renováveis; Armazenamento de hidrogênio; DFT; ligas multicomponentes;

ABSTRACT

COMPUTATIONAL THERMODYNAMIC MODELING OF BODY CENTERED CUBIC MULTICOMPONENT ALLOYS FOR HYDROGEN STORAGE.

Recently, multicomponent metal alloys for hydrogen storage have drawn attention due to the possibility of exploring a large compositional field, which can allow the control of storage properties by controlling the alloy composition, thus satisfying the needs for each application. The laboratory of hydrogen in metals at DEMa/UFSCar has vast experience in the synthesis, characterization, and evaluation of the properties of these alloys. The next step is to develop methods and models to predict the thermodynamic properties that determine their operating conditions (pressure and temperature for absorption/desorption) based solely on the chemical composition, aiming to improve the tools available to carry out the design of alloys for storage of hydrogen with suitable properties for different applications. Recently, Zepon et al. proposed a thermodynamic model to calculate pressure-composition-temperature (PCT) diagrams of multicomponent cubic body-centered alloys (CCC). of sites, and the enthalpy of the phases is modeled based on experimental data and calculated from ab initio through density functional theory (DFT). This master's work aimed to add new chemical elements to the recently proposed model through DFT calculations and to develop an open code program for the calculation of PCT diagrams based on the model and the data obtained by DFT. The elements Mg, Al, Sc, Mn, Fe, Co, Cu, Zn, Mo, and Pd were incorporated into the model and their implementation in open-source software in the Python programming language. Using this software, the alloys reported in the literature were evaluated so that it was possible to observe that the alloys of the TiVNbCr system are better described by the model, in addition to the fact that some alloys containing Mg show a discrepancy between the crystal structure observed experimentally and those obtained by the model.

Keywords: Sustainability; Renewable energy; Hydrogen storage; DFT; Multicomponent alloys;

SUMÁRIO

FOLHA DE APROVAÇÃO	i
AGRADECIMENTOS	iii
RESUMO	v
ABSTRACT	vii
SUMÁRIO	ix
LISTA DE TABELAS	xi
LISTA DE FIGURAS	xiii
LISTA DE SIGLAS E ABREVIATURAS	xvii
LISTA DE SÍMBOLOS	xix
1 INTRODUÇÃO	1
2 REVISÃO BIBLIOGRÁFICA	5
2.1 Ligas multicomponentes para armazenagem de hidrogênio	5
2.1.1 Diagrama de Fases	7
2.1.2 Fases do sistema MH	8
2.2 Modelagem termodinâmica para cálculo de diagramas PCT de li- gas multicomponentes	12
2.2.1 Considerações e parâmetros do modelo	12
2.3 Cálculo de Energia Total através de DFT	18
2.3.1 Teoria do Funcional de Densidade (DFT)	19
3 OBJETIVOS	25

4	MATERIAIS E MÉTODOS	27
4.1	Cálculos de Energia Total por DFT	27
4.2	Adição de elementos no modelo termodinâmico	28
4.3	Implementação do cálculo dos diagramas PCT em Software de código aberto	30
5	RESULTADOS	33
5.1	Cálculos das entalpias parciais molares de Hidrogênio e energias de ligação utilizadas no modelo	33
5.2	Desenvolvimento do programa em código aberto	39
5.3	Análise sistemática das ligas reportadas na literatura	48
5.3.1	Sistema TiVZrNb	48
5.3.2	Sistema TiVZrNbTa	51
5.3.3	Sistema TiVZrNbHf	57
5.3.4	Sistema TiVCrNbMo	62
5.3.5	Sistema TiZrMoNbHf	63
5.3.6	Sistema TiVNbCr	65
5.3.7	Análise global dos sistemas	69
5.4	Estudo sistemático do teor de Cr na liga (TiVNb)	71
5.5	Análise das ligas multicomponentes contendo Mg	74
5.6	Considerações gerais sobre o modelo	80
6	CONCLUSÕES	83
7	SUGESTÕES PARA FUTUROS TRABALHOS	85
8	REFERÊNCIAS BIBLIOGRÁFICAS	87
	APÊNDICE A: Dados obtidos do DFT	93
	APÊNDICE B: Código desenvolvido para os cálculos do modelo	131

LISTA DE TABELAS

2.1	Ligas multicomponentes CCC para armazenamento de Hidrogênio reportadas.	6
2.2	Ligas multicomponentes CCC para armazenamento de Hidrogênio contendo Mg reportadas.	6
2.3	Parâmetros da entropia configuracional possíveis para cada fase. .	14
5.1	Entalpia parcial molar de hidrogênio para cada elemento do modelo. Valores em kJ/mol de H	35
5.2	Coeficientes da interpolação polinomial para determinar Energia de ligação [kJ/mol] em função da distância da ligação dos átomos para as estruturas CFC e L1 ₀	37
5.3	Energias de ligação dos pares atômicos considerados para o modelo. Energias em kJ/mol	38

LISTA DE FIGURAS

1.1	Exemplo de uma liga multicomponente de estrutura cristalina CCC que sofre uma transição de fase para um hidreto CFC ao ocorrer a hidrogenação da liga, onde cada cor nos átomos da estrutura representa a probabilidade de um elemento ocupar aquela posição.	3
2.1	a) Diagrama PCT de um sistema M-H onde podemos ver os limites do platô de equilíbrio entre as fases. b) Representação esquemática das etapas de interesse do sistema M-H de estrutura CCC no decorrer da hidrogenação da liga em temperatura constante, desde a liga CCC sem hidrogênio, até o hidreto CFC.	8
2.2	a) Gráfico das energias livres de Gibbs das fases β e δ em função da concentração e sua relação com o potencial químico. b) Gráfico dos potenciais químicos do metal e do hidrogênio em função da concentração nos intervalos das fases β e δ e no equilíbrio. Ambos os gráficos em temperatura constante.	12
2.3	a) Célula unitária da estrutura B2. b) Célula unitária da estrutura L1 ₀ . Retirado de [5].	17
2.4	Diagrama esquemático do ciclo de auto-consistência.	22
2.5	Energia total em função do volume da célula unitária do Sc com estrutura cristalina CFC	23
4.1	Fluxograma do software do Modelo	31
5.1	Energia total em função do volume da célula unitária para as estruturas CCC, CFC e MH ₂ para o elemento Paládio. As energias devem ser somadas ao valor de referência no canto superior esquerdo.	34

5.2	Energia total em função do volume da célula unitária para as estruturas B2 e L10 para determinar as energias de ligação entre os elementos Co - V. As energias no eixo y devem ser somadas ao valor de referência no canto superior esquerdo.	36
5.3	Gráficos obtidos através do código para a liga $TiVZr_{0.5}NbTa_{0.5}$ na temperatura de 25 °C onde houve dois platôs de equilíbrio.	42
5.4	Gráficos obtidos através do código para a liga $TiZrMo_2Hf$ na temperatura de 25 °C onde houve apenas um platô de equilíbrio. . . .	45
5.5	Gráficos obtidos através do código para a liga $TiVNbTa$, liga com $H_M > 0$, na temperatura de 25 °C.	46
5.6	Gráficos obtidos através do código para a liga $TiZrNbHf$, na temperatura de 25 °C, $H_M < 0$	47
5.7	Diagramas PCT de ligas reportadas do sistema $TiVZrNb$	50
5.8	Comparação entre os diagramas PCI reportado e calculado para a liga $TiVZrNb$	51
5.9	Comparação entre os PCIs calculados e reportados para a liga $TiZrNbHfTa$	52
5.10	Diagramas PCT calculados das ligas reportadas do sistema $TiVZrNbTa$ utilizando os parâmetros reportados em [5].	55
5.11	Exemplos de diagramas PCT de duas ligas reportadas do sistema $TiVZrNbTa$ obtidas com parâmetros diferentes.	56
5.12	Diagramas PCT de ligas reportadas do sistema $TiVZrNbHf$ obtidos através do código desenvolvido. Ligas com variação da composição dos elementos.	59
5.13	PCTs de ligas reportadas do sistema $TiVZrNbHf$ obtidos através do código desenvolvido. Ligas com algum dos elementos retirados.	60
5.14	Comparação entre os diagramas PCI experimentais e os calculados para a liga $TiVZrNbHf$. Dados experimentais tirados de [13]. . .	61
5.15	Comparação entre os diagramas PCI experimentais e os calculados para a liga $TiV_{0.5}ZrNbHf$. Dados experimentais tirados de [13]. . .	62

5.16 PCTs de ligas reportadas do sistema TiVCrNbMo obtidos através do código desenvolvido.	63
5.17 Diagramas PCT de ligas reportadas do sistema TiZrMoNbHf obtidos através do código desenvolvido.	65
5.18 Diagramas PCT de ligas reportadas do sistema TiVNbCr obtidos através do código desenvolvido.	66
5.19 Comparação entre PCIs calculados com os experimentais reportados. Dados experimentais retirados de [5, 11, 12]	69
5.20 Mapa da Pressão do platô em função da capacidade reversível em %massa H, destacando os sistemas e as transições de fase.	70
5.21 Diagramas PCT calculados para teores diferentes de cromo no sistema $(\text{TiVNb})_{1-x}\text{Cr}_x$ não reportados.	72
5.22 Comparação da pressão do platô em função do teor de Cr nas ligas. 73	
5.23 Comparação da capacidade reversível em função do teor de Cr nas ligas.	74
5.24 Gráfico de variação da energia livre de Gibbs da liga $\text{Mg}_{0.28}\text{Al}_{0.19}\text{V}_{0.28}\text{Cr}_{0.19}\text{Ni}_{0.06}$, sendo exemplo para as ligas com Mg que apresentam $H_M < 0$	75
5.25 Diagramas PCT calculados para as ligas com Mg que não tiveram PCI reportado.	77
5.26 Diagramas PCT calculados para as ligas com Mg que tiveram PCI reportado.	78
5.27 Gráfico de variação da energia livre de Gibbs da liga $\text{Mg}_{0.1}\text{Ti}_{0.3}\text{V}_{0.25}\text{Zr}_{0.1}\text{Nb}_{0.25}$, sendo exemplo para as ligas com Mg que apresentam $H_M > 0$	78
5.28 Comparação entre os gráficos PCI experimental e calculado para a liga $\text{Mg}_{0.1}\text{Ti}_{0.3}\text{V}_{0.25}\text{Zr}_{0.1}\text{Nb}_{0.25}$. Dados experimentais tirados de [24]	79
5.29 Comparação entre os gráficos PCI experimental e calculado para a liga $\text{Mg}_{0.12}\text{Al}_{0.11}\text{Ti}_{0.33}\text{Mn}_{0.11}\text{Nb}_{0.33}$. Dados experimentais tirados de [26]	80

LISTA DE SIGLAS E ABREVIATURAS

c-T Composição-Temperatura

CCC Cubica de corpo centrado

CFC Cubica de face centrada

DFT Teoria do funcional de densidade

MH Metal-hidrogênio

PBE-GGA Perdew-Burke-Ernzerhof *generalized gradient approximation*

PCT Pressão-composição-Temperatura

PPGCEM Programa de Pós-Graduação em Ciência e Engenharia de Materiais

TCC Tetragonal de corpo centrado

UFSCar Universidade Federal de São Carlos

LISTA DE SÍMBOLOS

c_H^i - concentração de hidrogênio na fase i, razão entre o número de átomos de hidrogênio e do metal

E - energia da parte eletrônica do sistema [kJ]

E_T - energia total do sistema [kJ]

E_{total}^i - energia total da fase i [kJ]

E_H - energia de Hartree [kJ]

\hat{H} - operador hamiltoniano do sistema de muitos corpos

$H(i)$ - entalpia da fase i [kJ]

H^0 - entalpia padrão do gás hidrogênio [kJ]

H_2 - gás hidrogênio

H_M - parcela da entalpia devido a transição de fase da liga

h_m - parcela da entalpia de mistura ligada a adição de hidrogênio na liga

h_i^j - entalpia parcial molar de hidrogênio do elemento i e fase j [kJ/mol]

n_H - número de átomos de hidrogênio

n_M - número de átomos de metal

P_{H_2} - Pressão de hidrogênio [atm]

p_{ij} - probabilidade de termos a ligação i-j

\mathbf{r}_i - coordenadas do elétron i

\mathbf{R}_i - coordenadas dos núcleo I

$S(i)$ - entropia da fase i [kJ/K]

S^0 - entropia padrão do gás hidrogênio [kJ/K]

T - temperatura [K]

\hat{T}_e - operador energia cinética dos elétrons [J]

\hat{T}_N - operador energia cinética dos núcleos [J]

$v_n(\mathbf{r})$ - potencial externo

$v_H(\mathbf{r})$ - potencial de Hartree

v_{xc} - potencial de correlação e troca

\hat{V}_{ee} - operador energia de repulsão entre os elétrons [J]

\hat{V}_{Ne} - operador energia de atração entre os núcleos e elétrons [J]

\hat{V}_{NN} - operador energia de repulsão entre os núcleos [J]

$n(\mathbf{r})$ - densidade eletrônica

Z_i - número atômico do átomo I

α - fase CCC de uma liga multicomponente

β - fase hidreto intermediário com rede CCC distorcida de uma liga multicomponente

δ - fase hidreto CFC de uma liga multicomponente

ΔG_m - variação da energia livre de Gibbs de mistura [kJ]

ΔH_m - variação da entalpia de mistura [kJ]

ΔS_m - variação da entropia de mistura [kJ/K]

ε_{M-M} - energia de ligação média entre os átomos da estrutura

ε_{i-k} - energia de ligação entre o elemento i e o elemento j

δ_{ij} - degenerescência da ligação i-j

μ_H^i - potencial químico do hidrogênio na fase i [kJ/mol]

μ_M^i - potencial químico do metal na fase i [kJ/mol]

$\Psi(\mathbf{r}, \mathbf{R})$ - função de onda do sistema em função da posição dos elétrons e núcleos

$\psi(\mathbf{r})$ - função de onda de muitos corpos dos elétrons

$\phi_i(\mathbf{r})$ - função de onda do elétron i

1 INTRODUÇÃO

Com o aumento da demanda mundial pela energia elétrica, além do surgimento de políticas ambientais para a preservação do meio ambiente, é possível notar que cada vez mais a ciência caminha em busca do desenvolvimento de tecnologias para obter energia elétrica de maneiras mais sustentáveis, as chamadas fontes de energia renovável, que devem causar menos danos ao meio ambiente do que combustíveis fósseis como petróleo e carvão mineral. No âmbito nacional, temos um grande potencial hídrico para a geração de energia, além da possibilidade de aproveitar muitas fontes de energia renovável, como a energia solar, eólica e das marés.

Apesar do Brasil possuir o Sistema Interligado Nacional de produção e transmissão de energia elétrica (SIN) que possui malhas por quase todo o país, com exceção de uma parte da região Norte [1], ainda há a necessidade de pensar em maneiras de armazenar a energia excedente, para que seja facilitado o seu transporte para regiões aonde o SIN não chega, ou quando há a falha do sistema em alguma região. Ou até mesmo aplicações alternativas como, por exemplo, baterias para carros movidos a energia elétrica e baterias para aplicações em *Nobreak* ou geradores de energia elétrica, necessários em hospitais e na indústria em geral para suprir momentos de falta de energia elétrica vinda da rede de distribuição.

Um grande candidato para o armazenamento dessa energia é o hidrogênio, que pode ser gerado sem danos ao meio ambiente a partir da eletrólise da água utilizando a energia excedente gerada pelas fontes renováveis. O hidrogênio pode ser utilizado em células à combustível para a produção de energia elétrica. Porém, a utilização do hidrogênio no estado gasoso deve ser feita de forma cuidadosa já que é altamente inflamável, além disso, na forma gasosa o hidrogênio tem uma baixa densidade volumétrica de energia, cerca de 0,7 kJ/L na pressão atmosférica [2], fazendo necessário que o armazenamento na forma gasosa seja feito em altas pressões para ser viável. Apesar do desenvolvimento de tecnologias para aumentar a segurança desse armazenamento em altas pressões,

sempre há um risco associado à sua utilização.

Uma alternativa para o armazenamento de hidrogênio é na forma de hidretos metálicos, ou seja, armazenamento em estado sólido, onde o hidrogênio é incorporado na liga metálica, logo, sendo mais seguro para o armazenamento e transporte.

Nos últimos anos, várias ligas e compósitos foram estudados para o armazenamento de hidrogênio em estado sólido. Sendo aquelas que possuem elementos leves as que possuem maior potencial para aplicação tecnológica. Essas ligas podem ser classificadas em alguns sistemas: hidretos binários como MgH_2 , TiH_2 e ZrH_2 , hidretos complexos (geralmente com Mg, Al ou Li), soluções sólidas de estrutura cúbica de corpo centrado, como exemplo ligas Ti-V-Cr, e compostos intermetálicos das famílias AB, A_2B , AB_2 e AB_5 ($TiFe$, Mg_2Ni , ZrV_2 e $LaNi_5$) [3, 4].

Recentemente, surgiu um grande interesse na investigação de ligas multicomponentes ou de alta entropia, ligas com cinco ou mais elementos onde os elementos que a compõe são aproximadamente equimolares e não tem uma posição preferencial para ocupar na estrutura cristalina. Dentre as ligas multicomponentes para armazenamento de hidrogênio, uma alternativa são as ligas monofásicas de estrutura cristalina cúbica de corpo centrado (CCC), que dependendo da composição são capazes de absorver grandes quantidades de hidrogênio. Em algumas delas ocorre a formação de hidretos multicomponentes que podem possuir diferentes estruturas cristalinas, como podemos ver no exemplo da figura 1.1. O interesse em se estudar ligas multicomponente com estrutura do tipo CCC é a capacidade de se controlar a composição química da liga monofásica CCC, controlando assim, suas propriedades de armazenagem de hidrogênio.

O armazenamento de hidrogênio em ligas metálicas tem uma relação direta com as entalpias de dissolução de H na liga e entalpia de formação dos hidretos metálicos. Essas entalpias podem ser determinadas experimentalmente ou estimadas através de cálculos *ab initio* onde são calculadas as energias totais das fases de interesse com diferentes teores de hidrogênio. O método da Teoria do funcional de densidade (DFT) é um dos mais adequados e que vem sendo

amplamente aplicado para determinar essas energias totais.

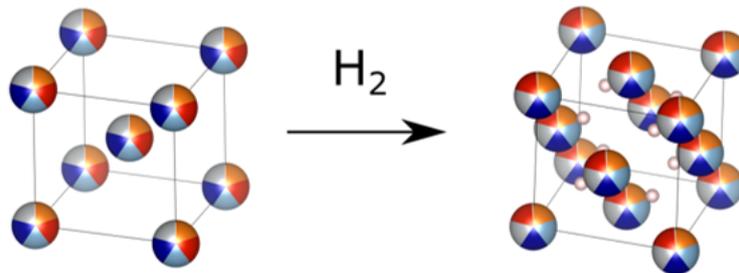


Figura 1.1: Exemplo de uma liga multicomponente de estrutura cristalina CCC que sofre uma transição de fase para um hidreto CFC ao ocorrer a hidrogenação da liga, onde cada cor nos átomos da estrutura representa a probabilidade de um elemento ocupar aquela posição.

Os diagramas pressão-composição-temperatura (PCT) são importantes para realizar o design de ligas, visto que trazem informações sobre a estabilidade das fases em função da temperatura e pressão de H_2 , ou seja, informações termodinâmicas essenciais para a utilização de uma liga em aplicações que envolvem a armazenagem de hidrogênio em estado sólido. Zepon et al. [5] propuseram um modelo termodinâmico para a determinação de diagramas PCT para ligas CCC, no qual, foi combinado um modelo para a entalpia das ligas à um modelo para a entropia configuracional proposto por Garcés [6]. No artigo em que foi proposto, os diagramas PCT calculados com o modelo foram comparados com 6 ligas com diagramas PCT experimentais reportados, apresentando uma boa concordância. Porém, o modelo foi limitado à oito elementos, a saber: Ti, V, Cr, Ni, Zr, Nb, Hf e Ta. Para ampliar o modelo e difundir sua utilização, o presente trabalho visa ser uma continuação do trabalho de Zepon et. al., expandindo a quantidade de elementos abrangidos pelo modelo, adicionando os elementos Mg, Al, Sc, Mn, Fe, Co, Cu, Zn, Mo e Pd, e desenvolvendo um código aberto para implementar o modelo para cálculos de diagramas PCT. Os elementos podem ser divididos em formadores ou não formadores de hidretos com base na entalpia de formação do hidreto [7]. Portanto, a escolha dos elementos para a adição ao modelo foi levado em conta se são elementos formadores, como Mg e Sc, ou não formadores, como Mn, Fe, Co, Cu, Zn, Mo e Pd, para que seja permitido realizar o design de ligas com uma ampla gama de propriedades.

2 REVISÃO BIBLIOGRÁFICA

2.1 Ligas multicomponentes para armazenagem de hidrogênio

Ligas multicomponentes vêm sendo estudadas para armazenamento de hidrogênio desde 2016 [8]. Devido as inúmeras possibilidades de combinação de elementos, vislumbra-se que ligas multicomponentes podem ser projetadas de acordo com as propriedades necessárias para cada aplicação específica.

Desde então, algumas ligas multicomponentes foram estudadas e, em sua maioria, ligas que possuem estrutura CCC monofásica. Algumas ligas CCC quando hidrogenadas sofrem transformação de fase formando um hidreto com estrutura cúbica de face centrada (CFC), ou CFC distorcida, com razão entre átomo de hidrogênio e átomos de metal (H/M) próximos à 2 como, por exemplo, as ligas TiZrNbHfTa [9], TiVNbZrHf [10], TiVNbCr [11]. Além disso, algumas ligas CCC formam um hidreto intermediário (H/M~1) tipicamente com estrutura tetragonal de corpo centrado como a liga $Ti_{0.2}Zr_{0.2}Nb_{0.2}Hf_{0.2}Ta_{0.2}$ [9]. Porém, algumas ligas com monohidretos com estruturas CCC também foram reportados, como a liga $(TiVNb)_{85}Cr_{15}$ [12].

Existem poucos dados a respeito dos diagramas PCT para essas ligas multicomponentes. Essa escassez de medições se dá por ser um procedimento experimentalmente trabalhoso, de forma que o desenvolvimento de um modelo para a previsão de diagramas PCT se torna interessante para facilitar a obtenção de informações a respeito de propriedades de armazenagem de hidrogênio das ligas.

A tabela 2.1 apresenta as ligas multicomponentes CCC reportadas na literatura, indicando aquelas que possuem diagramas PCT ou curvas pressão-composição-isoterma (PCI) experimentais reportadas.

Tabela 2.1: Ligas multicomponentes CCC para armazenamento de Hidrogênio reportadas.

Ligas com PCI experimental	Ligas sem PCI experimental	
$Ti_{0.2}V_{0.2}Zr_{0.2}Nb_{0.2}Hf_{0.2}$ [8, 10, 13]	$Ti_{0.2}V_{0.2}Zr_{0.2}Nb_{0.2}Mo_{0.2}$ [14]	$Ti_{0.2}V_{0.2}Cr_{0.2}Nb_{0.2}Mo_{0.2}$ [15]
$Ti_{28.3}V_{1.3}Nb_{55.4}Cr_{15}$ [5]	$Ti_{0.25}V_{0.25}Nb_{0.25}Ta_{0.25}$ [16]	$Ti_{0.25}V_{0.25}Cr_{0.25}Nb_{0.25}$ [15]
$Ti_{0.2}Zr_{0.2}Nb_{0.2}Hf_{0.2}Ta_{0.2}$ [9]	$Ti_{0.25}V_{0.25}Zr_{0.19}Nb_{0.25}Ta_{0.06}$ [16]	$Ti_{0.2}Zr_{0.2}Nb_{0.2}Mo_{0.2}Hf_{0.2}$ [17]
$Ti_{0.3}V_{0.25}Zr_{0.1}Nb_{0.25}Ta_{0.1}$ [18]	$Ti_{0.25}V_{0.25}Zr_{0.125}Nb_{0.25}Ta_{0.125}$ [16]	$Ti_{0.2}Zr_{0.2}Mo_{0.4}Hf_{0.2}$ [19]
$Ti_{0.25}Zr_{0.25}Nb_{0.25}Ta_{0.25}$ [20]	$Ti_{0.25}V_{0.25}Zr_{0.04}Nb_{0.25}Ta_{0.21}$ [16]	$Ti_{0.2}Zr_{0.2}Nb_{0.1}Mo_{0.3}Hf_{0.2}$ [19]
$Ti_{0.25}V_{0.25}Cr_{0.25}Nb_{0.25}$ [21]	$Ti_{0.2}V_{0.2}Zr_{0.4}Nb_{0.2}$ [16]	$Ti_{0.2}Zr_{0.2}Nb_{0.2}Mo_{0.2}Hf_{0.2}$ [19]
$Ti_{0.25}V_{0.25}Zr_{0.25}Nb_{0.25}$ [13]	$Ti_{0.21}V_{0.21}Zr_{0.37}Nb_{0.21}$ [16]	$Ti_{0.2}Zr_{0.2}Nb_{0.3}Mo_{0.1}Hf_{0.2}$ [19]
$Ti_{32.06}V_{32.06}Nb_{32.06}Ni_{3.8}$ [12]	$Ti_{0.22}V_{0.22}Zr_{0.33}Nb_{0.22}$ [16]	$Ti_{0.2}Zr_{0.2}Nb_{0.4}Hf_{0.2}$ [19]
$Ti_{0.22}V_{0.11}Zr_{0.22}Nb_{0.22}Hf_{0.22}$ [13]	$Ti_{0.24}V_{0.24}Zr_{0.28}Nb_{0.24}$ [16]	$Ti_{0.2}V_{0.2}Zr_{0.2}Nb_{0.2}Hf_{0.2}$ [22]
$Ti_{0.216}V_{0.216}Nb_{0.216}Cr_{0.35}$ [11]	$Ti_{0.25}V_{0.25}Zr_{0.25}Nb_{0.25}$ [16]	$Ti_{0.2}V_{0.2}Zr_{0.2}Nb_{0.2}Hf_{0.2}$ [13]
$Ti_{0.25}V_{0.25}Nb_{0.25}Cr_{0.25}$ [11]	$Ti_{0.325}V_{0.275}Zr_{0.125}Nb_{0.275}$ [23]	$Ti_{0.25}Zr_{0.25}Nb_{0.25}Hf_{0.25}$ [13]
$Ti_{0.283}V_{0.283}Nb_{0.283}Cr_{0.15}$ [11, 12]	$Ti_{0.25}Zr_{0.25}Nb_{0.25}Hf_{0.25}$ [15]	$Ti_{0.25}V_{0.25}Nb_{0.25}Hf_{0.25}$ [13]
$Ti_{31.76}V_{31.76}Nb_{31.76}Co_{4.7}$ [12]	$Ti_{0.25}V_{0.25}Nb_{0.25}Ta_{0.25}$ [15]	$Ti_{0.25}V_{0.25}Zr_{0.25}Hf_{0.25}$ [13]
	$Ti_{0.2}V_{0.2}Zr_{0.2}Nb_{0.2}Hf_{0.2}$ [15]	$Ti_{0.11}V_{0.22}Zr_{0.22}Nb_{0.22}Hf_{0.22}$ [13]
	$Ti_{0.25}V_{0.25}Zr_{0.25}Nb_{0.25}$ [15]	$Ti_{0.22}V_{0.22}Zr_{0.11}Nb_{0.22}Hf_{0.22}$ [13]
	$Ti_{0.2}V_{0.2}Cr_{0.2}Nb_{0.2}Ta_{0.2}$ [15]	$Ti_{0.22}V_{0.22}Zr_{0.22}Nb_{0.11}Hf_{0.22}$ [13]
	$Ti_{0.25}V_{0.25}Nb_{0.25}Mo_{0.25}$ [15]	$Ti_{0.22}V_{0.22}Zr_{0.22}Nb_{0.22}Hf_{0.11}$ [13]
	$Ti_{0.25}V_{0.25}Cr_{0.25}Mo_{0.25}$ [15]	

O elemento Mg possui uma boa visibilidade nos estudos para armazenamento de hidrogênio já que é um elemento barato, abundante e com baixo peso atômico. Desta forma, a adição de magnésio na composição de ligas multicomponentes leva a uma redução na sua massa molar, aumentando a capacidade gravimétrica de armazenagem de hidrogênio. Porém, não há muitos estudos de ligas multicomponentes com Mg para armazenamento de hidrogênio. De sete ligas de estrutura CCC reportadas, apenas duas possuem PCI experimental. A tabela 2.2 apresenta as ligas reportadas.

Tabela 2.2: Ligas multicomponentes CCC para armazenamento de Hidrogênio contendo Mg reportadas.

Liga com PCI experimental	Liga sem PCI experimental
$Mg_{0.1}Ti_{0.30}V_{0.25}Zr_{0.1}Nb_{0.25}$ [24]	$Mg_{0.2}Al_{0.2}Ti_{0.2}Fe_{0.2}Ni$ [25]
$Mg_{0.12}Al_{0.11}Ti_{0.33}Mn_{0.11}Nb_{0.33}$ [26]	$Mg_{0.2}V_{0.2}Al_{0.2}Cr_{0.2}Ni_{0.2}$ [27]
	$Mg_{0.28}Al_{0.19}V_{0.28}Cr_{0.19}Ni_{0.06}$ [27]
	$Mg_{0.22}Ti_{0.22}Cr_{0.11}Mn_{0.11}Ni_{0.11}Nb_{0.22}$ [28]
	$Mg_{0.22}Zr_{0.22}Ti_{0.22}Fe_{0.11}Co_{0.11}Ni_{0.11}$ [29]

Para que seja possível uma total compreensão sobre a capacidade de uma liga armazenar hidrogênio através da formação de um hidreto, é necessário com-

preender a termodinâmica de sistemas Metal-Hidrogênio (M-H), que será discutida nas seções seguintes.

2.1.1 Diagrama de Fases

Uma parte fundamental do estudo de materiais são os diagramas de fase, gráficos que contêm a informação da composição, temperatura e pressão em que cada fase é formada em equilíbrio. Para sistemas M-H, devido a ocupação intersticial do hidrogênio, é conveniente representar o eixo das abcissas como a razão entre o número de átomos de hidrogênio e do metal:

$$c_H = \frac{n_H}{n_M} \quad (2.1)$$

No estudo de sistemas M-H, são realizadas medidas isotérmicas da composição em função da pressão do gás, dessa forma, obtemos uma série de curvas que compõem os diagramas chamados de pressão-composição-isoterma (PCI). Nas curvas construídas podemos observar a formação de platôs nas regiões onde as fases coexistem, assim, podemos construir uma linha de contorno de fases ao ligarmos os pontos de inflexão das isotermas. Podemos ver um exemplo de diagrama PCI na figura 2.1 (a) e as possíveis fases formadas na figura 2.1 (b).

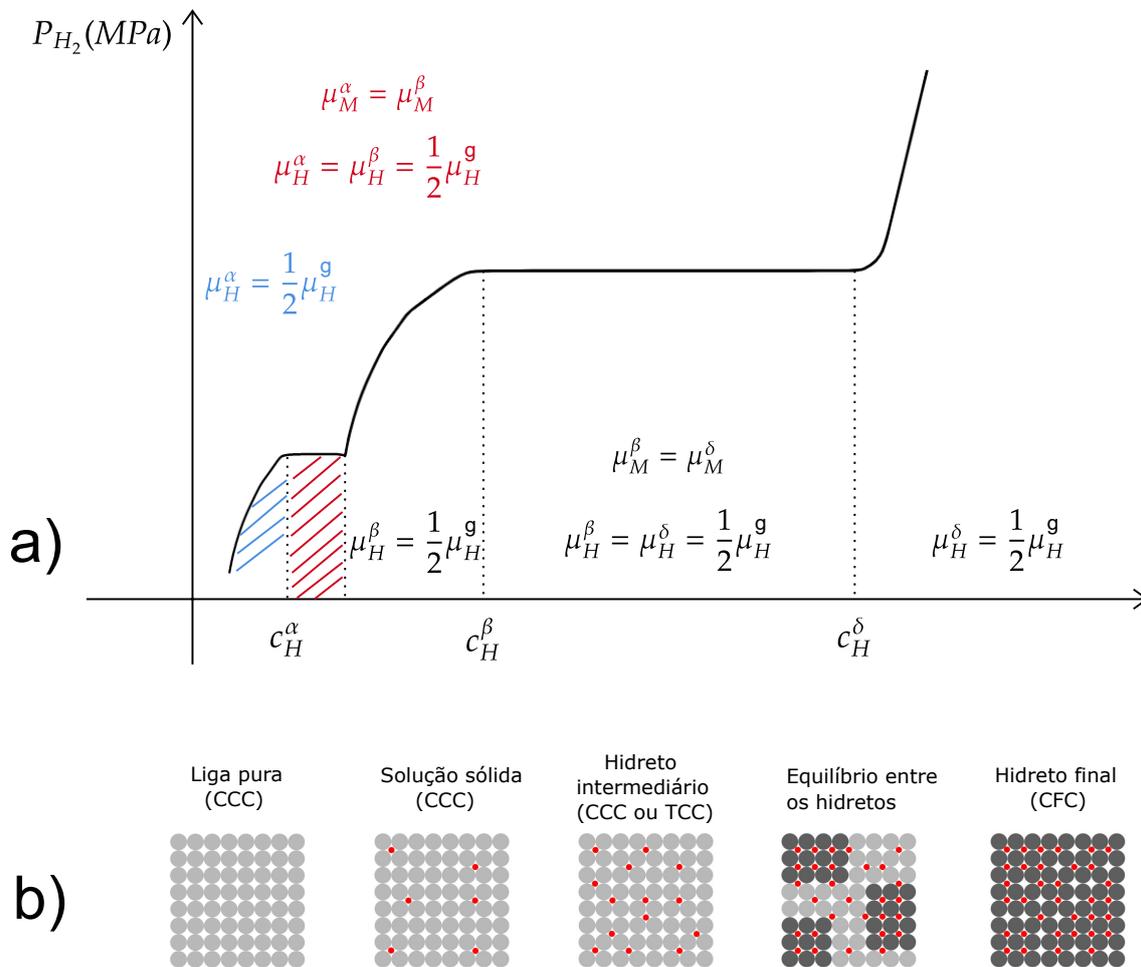


Figura 2.1: a) Diagrama PCT de um sistema M-H onde podemos ver os limites do platô de equilíbrio entre as fases. b) Representação esquemática das etapas de interesse do sistema M-H de estrutura CCC no decorrer da hidrogenação da liga em temperatura constante, desde a liga CCC sem hidrogênio, até o hidreto CFC.

2.1.2 Fases do sistema MH

Quando ocorre a mistura de mais de um elemento com a formação de uma solução sólida, há uma energia livre de Gibbs de mistura. Essa energia é devido a interação entre os átomos de elementos diferentes que se misturaram na solução sólida e pela configuração do arranjo dos átomos na rede cristalina.

Ao submeter uma liga a uma atmosfera rica em H_2 , os átomos de H começam a difundir na rede cristalina e a ocupar os sítios intersticiais da rede metálica formando uma solução sólida diluída, fase α , onde há poucos átomos de H na rede cristalina. Vamos considerar um sistema no qual dois hidretos, um intermediário (H/M~ 1) e um dihidreto (H/M~2), são formados. Quando a solução sólida

alcança uma concentração de hidrogênio c_H^α pode ocorrer a formação de um hidreto intermediário, fase β , possuindo uma estrutura diferente da solução sólida. Chegando a uma composição c_H^β ocorre uma transição de fase para um hidreto rico em hidrogênio, fase δ , que irá possuir uma estrutura diferente da solução sólida e do hidreto intermediário.

Uma maneira de estudar a formação desses hidretos e as mudanças de fase das ligas é através da energia livre de Gibbs e do potencial químico dos elementos nas diferentes fases.

A estabilidade relativa entre todas as fases possíveis de uma liga é ditada pela variação da energia livre de Gibbs das possíveis fases no sistema em função da temperatura e composição. A variação da energia livre de uma fase (a partir de um estado de referência) em função da concentração de hidrogênio e da temperatura é dada por:

$$\Delta G_m(c_H) = \Delta H_m(c_H) - T\Delta S_m(c_H) \quad (2.2)$$

$$\Delta H_m(c_H) = H(MH_{c_H}) - H(M^{CCC}) - \frac{c_H}{2}H^0(H_2) \quad (2.3)$$

$$\Delta S_m(c_H) = S(MH_{c_H}) - S(M^{CCC}) - \frac{c_H}{2}S^0(H_2) \quad (2.4)$$

$\Delta G_m(c_H)$ é a variação da energia livre de Gibbs de uma fase com composição c_H em relação as referências adotadas, que são o metal em sua forma estável, no caso desse trabalho, o metal de estrutura CCC, e o hidrogênio gasoso à pressão de H_2 de 1 atm. A variação da entalpia, $\Delta H_m(c_H)$, é composta pela entalpia da fase contendo hidrogênio $H(MH_{c_H})$ subtraída da entalpia da referência $H(M^{CCC})$ e da entalpia padrão do gás hidrogênio, $H^0(H_2)$. Já a variação da entropia $\Delta S_m(c_H)$, é composta pela entropia da fase contendo hidrogênio $S(MH_{c_H})$ subtraída da entropia da referência $S(M^{CCC})$ e da entropia padrão do gás H_2 , $S^0(H_2)$, conforme equações equação 2.4 e 2.5.

$$S^0(H_2) = \frac{7}{2}R \left(1 + \ln \left(\frac{T}{9.2} \right) \right) \quad (2.5)$$

onde R é a constante universal dos gases e T é a temperatura dada em Kelvin.

O potencial químico é a grandeza termodinâmica que serve de critério de equilíbrio entre as fases. Esse equilíbrio geralmente ocorre de forma completa, ou seja, quando o potencial químico de cada elemento é igual em cada fase que o contém. Porém, em temperaturas relativamente baixas, i.e., muito abaixo da temperatura de fusão das ligas, uma condição específica de equilíbrio químico pode ser observada, o para-equilíbrio. Na condição de para-equilíbrio, os átomos dos metais não possuem mobilidade para se reorganizarem de forma a alcançar o equilíbrio completo. Neste caso, apenas o hidrogênio possui mobilidade suficiente para mudar a composição química das fases [30]. Assim, tanto a variação da energia livre de Gibbs quanto o potencial químico dependem apenas do teor de hidrogênio (c_H) na fase.

Na grande maioria dos estudos, as ligas multicomponentes CCC são investigadas em temperaturas relativamente baixas (entre temperatura ambiente e 500 °C), nas quais as condições de para-equilíbrio são experimentalmente observadas, isto é, a composição da solução pobre em hidrogênio (em termos de elementos metálicos) é a mesma dos hidretos ricos em hidrogênio. O potencial químico do hidrogênio nas fases metálicas é dado pelas equações 2.6, 2.7 e 2.8. O equilíbrio entre as fases ocorre nos platôs que surgem no diagrama PCT, exemplificado na figura 2.1.

$$\mu_H^\alpha = \left(\frac{\partial \Delta G_m^\alpha(c_H)}{\partial c_H} \right)_{p,T} \quad (2.6)$$

$$\mu_H^\beta = \left(\frac{\partial \Delta G_m^\beta(c_H)}{\partial c_H} \right)_{p,T} \quad (2.7)$$

$$\mu_H^\delta = \left(\frac{\partial \Delta G_m^\delta(c_H)}{\partial c_H} \right)_{p,T} \quad (2.8)$$

Enquanto o potencial químico do hidrogênio gasoso em uma atmosfera de hidrogênio a uma determinada pressão P_{H_2} , em relação a referência $P_0 = 1$ atm,

é dado pela equação 2.9.

$$\mu_H^g = \frac{1}{2}RT \ln\left(\frac{P_{H_2}}{P_0}\right) \quad (2.9)$$

Na condição de para-equilíbrio, podemos expressar a energia livre de Gibbs por mol de átomos de metal pela equação 2.10.

$$\Delta G_m(c_H) = \mu_M + c_H \mu_H \quad (2.10)$$

Onde μ_M é o potencial químico dos átomos de metais na fase de interesse. Nas regiões de platô dos diagramas PCT, i.e., regiões onde duas fases sólidas coexistem em equilíbrio com o gás a uma determinada pressão, o potencial químico do hidrogênio nas fases sólidas e no gás são iguais e o potencial químico do metal nas fases sólidas também são iguais. Representando na forma de equação temos que as relações 2.11 e 2.12 determinam as condições de equilíbrio no platô.

$$\mu_M^\beta(c_H^\beta, p, T) = \mu_M^\delta(c_H^\delta, p, T) \quad (2.11)$$

$$\mu_H^\beta(c_H^\beta, p, T) = \mu_H^\delta(c_H^\delta, p, T) = \mu_H^g(p, T) \quad (2.12)$$

Graficamente, esta condição de equilíbrio é representada através de uma tangente comum às curvas de energia livre de Gibbs das fases em equilíbrio no platô. Podemos ver a tangente comum entre as energias livres de Gibbs em equilíbrio na figura 2.2a. Com a equação 2.10 é possível verificar o comportamento dos potenciais químicos no sistema em função da concentração de hidrogênio que é representado na figura 2.2b.

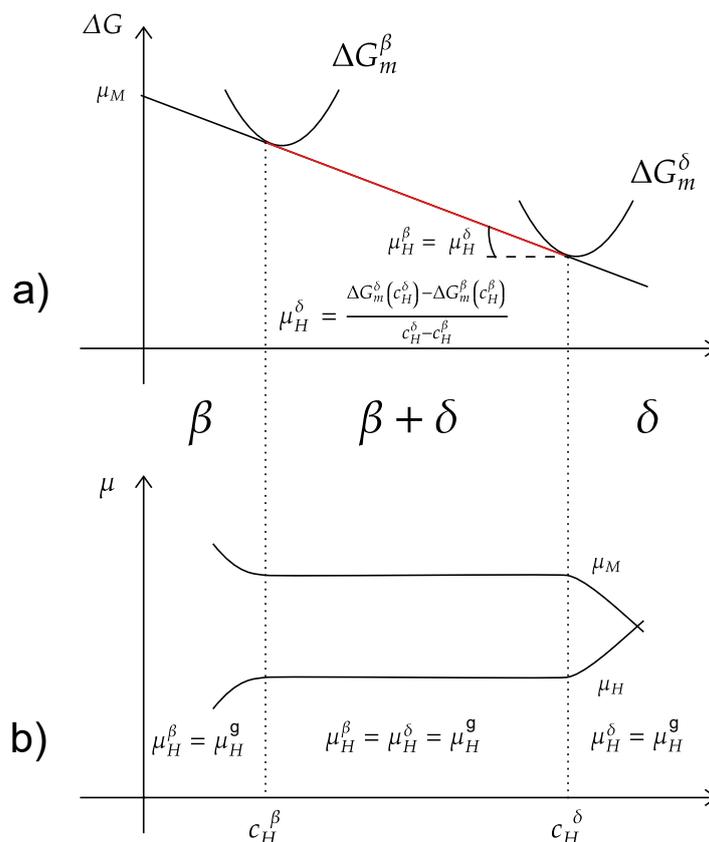


Figura 2.2: a) Gráfico das energias livres de Gibbs das fases β e δ em função da concentração e sua relação com o potencial químico. b) Gráfico dos potenciais químicos do metal e do hidrogênio em função da concentração nos intervalos das fases β e δ e no equilíbrio. Ambos os gráficos em temperatura constante.

2.2 Modelagem termodinâmica para cálculo de diagramas PCT de ligas multicomponentes

O modelo termodinâmico de ligas multicomponentes cúbica de corpo centrado para armazenamento de hidrogênio proposto por Zepon et al. [5] utiliza uma série de aproximações para se determinar $\Delta H_m(c_H)$ e $\Delta S_m(c_H)$ das fases de interesse para ligas multicomponentes CCC. A seguir será feita uma breve revisão do modelo proposto.

2.2.1 Considerações e parâmetros do modelo

Inicialmente, será tratado as aproximações a respeito da entropia. A princípio, é factível expressar a entropia como a soma de duas parcelas: uma ligada a configuração dos átomos na liga, chamada de entropia configuracional, S_c , e uma que independe da configuração atômica e que carrega a contribuição vibracional

e energética que os átomos contribuem para a liga, chamada de entropia não-configuracional, S_{nc} :

$$S = S_c + S_{nc} \quad (2.13)$$

Porém, como a variação da entropia entre uma fase MH_{c_H} e o estado de referência (M(CCC) e H_2 a 1 atm) é o que interessa, foi considerado que os valores da entropia não-configuracional entre a fase MH_{c_H} e M(CCC) são aproximadamente iguais, de forma que a principal contribuição seja obtida da entropia configuracional. Portanto, a equação 2.4 se torna:

$$\Delta S_m(c_H) = S_c(MH_{c_H}) - S_c(M^{CCC}) - \frac{c_H}{2} S^0(H_2) \quad (2.14)$$

Para ligas multicomponentes formadas por soluções sólidas desordenadas podemos aproximar a entropia configuracional por:

$$S_c(M^{CCC}) = -R \sum_i c_i \ln c_i \quad (2.15)$$

Onde $R = 8,314$ [J/K mol] é a constante universal dos gases e c_i é a fração do elemento i na fase.

Já para o hidreto, é importante levar em conta que os átomos de hidrogênio que ocupam os interstícios do metal interagem entre si de forma que pode ocorrer um efeito de bloqueio de interstícios vizinhos aos sítios intersticiais previamente ocupados. Uma equação para a entropia levando em conta o efeito de bloqueio de sítios devido a ocupação prévia de um sítio intersticial é proposta por J. Garcés [6] e foi adaptada para descrever a entropia configuracional das fases MH_{c_H} .

$$S_c(MH_{c_H}) = -R \left[\sum_i c_i \ln c_i + c_H \ln \left(\frac{c_H}{\theta - [(r-1)c_H]} \right) + (\theta - rc_H) \ln \left(\frac{\theta - rc_H}{\theta - [(r-1)c_H]} \right) \right] \quad (2.16)$$

onde θ é o número de interstícios por átomo e $r - 1$ representa a quantidade de sítios intersticiais bloqueados n^b . Por exemplo, no caso de cada átomo de H bloquear um vizinho ($n^b = 1$) é obtido o valor de $r = 2$, caso não houvesse bloqueio $r = 1$. Portanto, é possível reescrever a equação 2.14 como:

$$\Delta S_m(c_H) = -R \left[c_H \ln \left(\frac{c_H}{\theta - [(r-1)c_H]} \right) + (\theta - rc_H) \ln \left(\frac{\theta - rc_H}{\theta - [(r-1)c_H]} \right) \right] - \frac{c_H}{2} S^0(H_2) \quad (2.17)$$

Os valores de θ e r são diferentes para cada fase e cada liga, os parâmetros da entropia configuracional possíveis estão ilustrados na tabela 2.3. No modelo proposto, foram consideradas três fases que são tipicamente observadas durante a hidrogenação de ligas multicomponentes CCC. A fase α corresponde a solução sólida diluída com estrutura CCC. Foi considerado que para a fase α os átomos de hidrogênio podem ocupar apenas os interstícios tetraedrais da estrutura CCC ($\theta = 6$) e que o efeito de bloqueio de sítio pode estar presente e r pode assumir os valores de 5, 7 e 15 dependendo dos sítios vizinhos bloqueados. A fase β corresponde ao hidreto intermediário, que pode se apresentar ou com estrutura CCC ou TCC. Quando a estrutura é CCC assume-se que os átomos de hidrogênio ocupam apenas alguns sítios tetraedrais específicos representando um ordenamento e, neste caso, $\theta = 4$. No caso da estrutura TCC, é assumido que os átomos de H ocupam os sítios octaédricos e por isso $\theta=3$. Em ambos os casos, o fenômeno de bloqueio de sítios pode ocorrer e r pode assumir 2 ou 3. Já a fase δ é considerada como uma fase CFC na qual os átomos de hidrogênio ocupam os sítios tetraédricos ($\theta = 2$) e não ocorre o bloqueio de sítios ($r = 1$).

Tabela 2.3: Parâmetros da entropia configuracional possíveis para cada fase.

Fase	Sítio ocupado (θ)		Sítios Bloqueados (r)		
	Tetraedral	Octaedral	Primeiros Vizinhos	Segundos Vizinhos	Terceiros Vizinhos
α	6	-	5	7	15
β	4	3	2	3	-
δ	2	-	-	-	-

Para determinar $\Delta H_m(c_H)$ também foram feitas algumas aproximações. A

primeira é que a entalpia de uma fase varia linearmente com a composição de hidrogênio, como dado pela equação 2.18.

$$\Delta H_m(c_H) = H_M + h_m \cdot c_H \quad (2.18)$$

onde H_M é a parcela da entalpia ligada a transição da fase de referência (CCC) para uma determinada fase de interesse, sem levar em consideração a adição de hidrogênio a estrutura, enquanto h_m é a parcela de energia ligada a adição de átomos de H na fase resultante.

Nesse trabalho, a fase de referência é a liga CCC, logo, já que a solução sólida α possui uma estrutura CCC, então temos $H_M^\alpha = 0$. O mesmo pensamento pode ser levado em conta para os hidretos intermediários (β) tipicamente observados em ligas CCC, uma vez que estes hidretos usualmente apresentam estruturas que são apenas distorções da estrutura CCC devido a ocupação dos interstícios por hidrogênio. Neste caso, aproximou-se que para os hidretos intermediários, quando $c_H \rightarrow 0$ ele é idêntico a estrutura da liga CCC, o que nos permite assumir $H_M^\beta = 0$.

Já no caso dos hidretos CFC (δ), existe uma parcela de energia ligada a transição de fase da referência CCC para a fase CFC, logo, H_M^δ representa a entalpia referente a essa transição de fase. Para determinar o valor dessa entalpia foi considerado que a energia total de uma estrutura multicomponente é dada pela média ponderada das energias de ligação dos componentes da liga naquela estrutura

Para determinar a energia de ligação média de um átomo com os primeiros vizinhos foi utilizada a equação 2.19.

$$\varepsilon_{M-M} = \sum_{i \leq j} \delta_{ij} \cdot p_{ij} \cdot \varepsilon_{i-j} \quad (2.19)$$

onde δ_{ij} é a degenerescência da ligação i-j, onde $\delta_{ij} = 1$ se $i = j$ e $\delta_{ij} = 2$ se $i \neq j$, p_{ij} é a probabilidade de termos a ligação i-j na solução sólida aleatória que compõe a liga multicomponente, pode ser calculada simplesmente por $p_{ij} = c_i \cdot c_j$, e ε_{i-j} é a energia de ligação entre o elemento i e o elemento j.

Com a energia média das ligações é possível determinar as energias totais das estruturas antes e depois da transição de fase através das equações 2.20 e 2.21.

$$E_{total}^{\alpha} = 4\varepsilon_{M-M}^{\alpha} \quad (2.20)$$

$$E_{total}^{\delta} = 6\varepsilon_{M-M}^{\delta} \quad (2.21)$$

Os valores 4 e 6 nas equações 2.20 e 2.21 advém do numero de coordenação dos átomos de metal na estrutura CCC e CFC; 8 e 12, dividindo por 2: temos 4 e 6; esses valores representam o número de ligações por átomos nas estruturas CCC e CFC, respectivamente. Assim, é possível estimar a entalpia referente a mudança na estrutura cristalina através da equação 2.22.

$$H_M^{\delta} = E_{total}^{\delta} - E_{total}^{\alpha} \quad (2.22)$$

Para determinar inicialmente as energias de ligação ε_{i-j} , pode ser utilizado simulações de DFT das ligas CCC sem hidrogênio para determinar as energias i-i através da energia total e utilizando as equações 2.20, 2.21 e 2.19. Para determinar as energias de ligação i-j pode ser feito simulações DFT de ligas binárias com os elementos i-j, porém, utilizando as estruturas B2 e L1₀, como pode ser visto na figura 2.3, onde a relação das energias de ligação dos átomos com a energia total são dadas pelas equações 2.20 e 2.23, respectivamente.

$$E_{total}^{L1_0} = 4\varepsilon_{i-j}^{\delta} + \varepsilon_{i-i}^{\delta} + \varepsilon_{j-j}^{\delta} \quad (2.23)$$

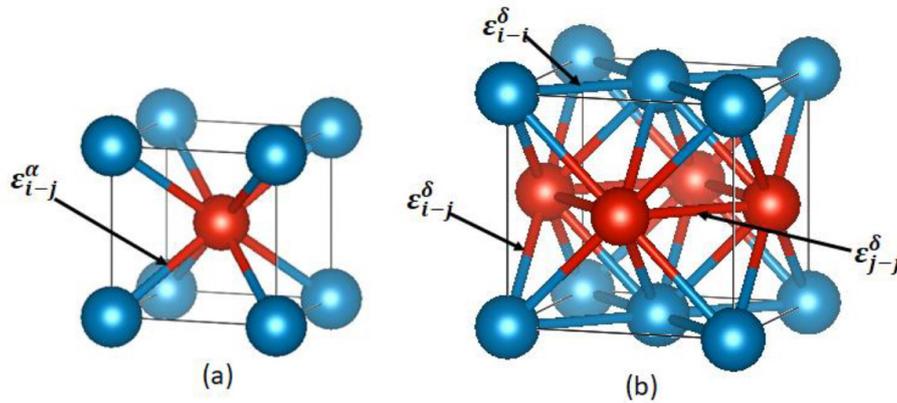


Figura 2.3: a) Célula unitária da estrutura B2. b) Célula unitária da estrutura L1₀. Retirado de [5].

Com os valores de ε_{i-i} e ε_{i-j} determinados é possível calcular H_M^δ para ligas multicomponentes.

Enquanto H_M está ligada a energia envolvida na transição de fase, h_m representa a entalpia parcial molar do hidrogênio na fase de interesse. Assumiu-se que h_m de uma liga multicomponente é aproximadamente o valor médio da entalpia parcial molar do hidrogênio na mesma fase de cada elemento puro presente na liga como representado na equação 2.24.

$$h_m = \sum c_i \cdot h_i \quad (2.24)$$

onde c_i é a fração atômica do elemento i e h_i é a entalpia parcial molar de hidrogênio do elemento i possuindo a mesma estrutura cristalina da fase de interesse. Para as fases α e β , Zepon et al. aproximou h_i aos valores tabelados de entalpia de dissolução de hidrogênio dos elementos puros em diluição infinita. Já, para o hidreto concentrado CFC (δ), h_i foi determinado através de cálculos de energia total de DFT como apresentado na equação 2.25.

$$h_i^\delta = \frac{E_{total}^{\delta-MH_2} - E_{total}^{\delta-M} - E_{total}^{H_2}}{2} \quad (2.25)$$

Onde $E_{total}^{\delta-MH_2}$ é a energia total do hidreto MH_2 com estrutura CFC, $E_{total}^{\delta-M}$ é a energia total do metal i com estrutura CFC e $E_{total}^{H_2}$ é a energia total do H_2 no estado padrão. O valor de $E_{total}^{H_2}$ foi determinado experimentalmente utilizando

dados experimentais das entalpias padrão de formação do NbH_2 , TiH_2 e VH_2 , que possuem estruturas CFC.

Com isso, é possível determinar a energia livre de Gibbs das fases utilizando as equações 2.17, 2.18 e 2.4, e tomando a sua derivada em relação a c_H obtemos o potencial químico do hidrogênio em uma determinada fase, que é dada pela equação 2.26.

$$\mu_H(c_H) = h_m - T \left[-R \ln \left(\frac{c_H [\theta - (r-1)c_H]^{(r-1)}}{[\theta - rc_H]^r} \right) - \frac{1}{2} S^0(H_2) \right] \quad (2.26)$$

Utilizando a relação 2.10 conseguimos determinar o potencial químico do metal através da equação 2.27.

$$\mu_M = \Delta G_m(c_H) - c_H \mu_H(c_H) \quad (2.27)$$

Com os potenciais químicos, é possível determinar os diagramas PCT através das condições de equilíbrio.

Os resultados obtidos pelo modelo no artigo em que foi proposto [5] foram promissores. Os diagramas PCT e as entalpias e entropias do platô calculados foram comparados com os experimentais para 6 ligas, apresentando um bom ajuste e demonstrando o potencial do modelo. Porém, o modelo abrange apenas 8 elementos e não foi disponibilizado um código aberto para realizar os cálculos. Portanto, é necessário expandir o modelo e desenvolver um código aberto para que o modelo seja amplamente difundido.

2.3 Cálculo de Energia Total através de DFT

As energias totais das estruturas cristalinas utilizadas pelo modelo podem ser calculadas através da teoria do funcional da densidade (DFT), portanto, é necessário entender um pouco melhor como são feitos os cálculos que serão utilizados nesse trabalho para a adição de novos elementos ao modelo.

Quando estamos estudando uma rede cristalina através da mecânica quântica para determinar sua energia total temos um problema com diversos átomos, e cada átomo com vários elétrons. Logo, há uma situação em que é necessário

levar em consideração não só a parcela de energia que cada entidade separada contribui para a total, como a interação entre cada uma delas. Portanto, quanto mais elétrons e núcleos uma estrutura possui, mais termos devem ser calculados e mais difícil se torna o cálculo da energia total, por isso, a Teoria do Funcional de Densidade é uma grande ferramenta para calcular grandezas físicas de sistemas através da mecânica quântica.

2.3.1 Teoria do Funcional de Densidade (DFT)

Quando é necessário abordar um problema de mecânica quântica, pode-se dizer que o principal objetivo é resolver a equação de Schroedinger para entender fenômenos ligados ao sistema estudado e conseguir realizar previsões de grandezas físicas do sistema. Por isso, a principal equação da mecânica quântica é a equação 2.28.

$$\hat{H}\Psi(r_1, \dots, r_N, \mathbf{R}_1, \dots, \mathbf{R}_M) = E_T\Psi(r_1, \dots, r_N, \mathbf{R}_1, \dots, \mathbf{R}_M) \quad (2.28)$$

onde \hat{H} é o Hamiltoniano, um operador matemático que contém os termos ligados as energias presentes no sistema, Ψ , que é a função de onda do sistema que leva em conta as posições dos elétrons r_n e dos núcleos R_M , que traz as informações do estado do sistema. Através da função de onda é possível determinar a densidade de probabilidade eletrônica, equação 2.29, que nos fornece a probabilidade de encontrar um elétron na posição \mathbf{r} e, por fim, E é a energia total do sistema.

$$n(\mathbf{r}) = N \int \Psi^*(\mathbf{r}, \mathbf{R})\Psi(\mathbf{r}, \mathbf{R})d\mathbf{r}d\mathbf{R} \quad (2.29)$$

Para um cristal, as principais energias que estão presentes na hamiltoniana são descritas na equação 2.30, que utiliza o sistema de unidades atômicas, ou seja, que apresenta $m_e = \hbar = e = 1$:

$$\hat{H} = \hat{T}_e + \hat{V}_{ee} + \hat{V}_{Ne} + \hat{T}_N + \hat{V}_{NN} \quad (2.30)$$

$$\hat{H} = - \sum_i \frac{1}{2} \nabla_i^2 + \frac{1}{2} \sum_{i \neq j} \frac{1}{|\mathbf{r}_i - \mathbf{r}_j|} - \sum_{i,I} \frac{Z_I}{|\mathbf{r}_i - \mathbf{R}_I|} - \sum_I \frac{1}{2m_I} \nabla_I^2 + \frac{1}{2} \sum_{I,J} \frac{Z_I Z_J}{|\mathbf{R}_I - \mathbf{R}_J|} \quad (2.31)$$

onde \hat{T}_N é a energia cinética dos núcleos atômicos, \hat{T}_e é a energia cinética dos elétrons, \hat{V}_{ee} é o potencial ligado a repulsão coulombiana entre os elétrons, \hat{V}_{Ne} é a energia potencial ligada a atração coulombiana entre os núcleos e elétrons e \hat{V}_{NN} é o potencial ligado a repulsão entre os núcleos.

Porém, escrever as funções de onda de muitos corpos se torna um grande empecilho já que não há um método de escrever funções de onda exatas para esses problemas, de forma que não é possível resolver a equação 2.28. Isso levou ao desenvolvimento de várias abordagens e aproximações para que se pudesse resolver esses tipos de sistemas. Dentre as abordagens, uma das mais utilizadas para estudar materiais com estruturas cristalinas é a Teoria do Funcional da Densidade.

A Teoria do Funcional de Densidade é fundamentada através de dois teoremas propostos por Hohenberg e Kohn[31].

1. A densidade eletrônica do estado fundamental determina unicamente o potencial sentido pelos elétrons.
2. A energia do estado fundamental pode ser obtida variacionalmente. A densidade que minimiza a energia total é a densidade exata do estado fundamental.

As verificações desses teoremas podem ser vistas em [32], a partir dos teoremas é possível chegar a algumas conclusões. A primeira de que a energia total é um funcional da densidade eletrônica já que temos uma relação indireta entre os dois, $n(\mathbf{r}) \rightarrow v_n(\mathbf{r}) \rightarrow \psi(\mathbf{r}) \rightarrow E$. A segunda é que a energia do estado fundamental obtida é a menor possível e única para a configuração dada no problema estudado, o que garante a validade da teoria. A equação 2.32 apresenta

as parcelas de energias que constituem a energia total de um sólido cristalino obtida através de DFT.

$$E[n] = \int d\mathbf{r} n(\mathbf{r}) v_n(\mathbf{r}) - \sum_i \int \phi_i^*(\mathbf{r}) \frac{\nabla^2}{2} \phi_i(\mathbf{r}) + \frac{1}{2} \iint d\mathbf{r} d\mathbf{r}' \frac{n(\mathbf{r})n(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|} + E_{xc}[n] \quad (2.32)$$

Na equação 2.32, a primeira integral é referente ao potencial externo que os elétrons sentem devido a presença dos núcleos atômicos, a segunda integral está relacionada a energia cinética de cada elétron, a última integral é relativa à energia coulombiana clássica que nasce da interação de cada elétron com todos os outros elétrons da nuvem eletrônica. O último termo é a energia de correlação e troca, que é a energia que surge da natureza quântica do problema e do princípio de exclusão de Pauli. Além disso, este último termo engloba outras parcelas de energia que não sabemos descrever, como por exemplo, a parcela da energia cinética dos elétrons que é influenciada pela interação com os outros elétrons da nuvem eletrônica. Uma forma geral de escrever o potencial que dá origem a essa energia é dado pela equação 2.33, porém, na prática o potencial de correlação e troca usualmente é descrito por aproximações.

$$v_{xc} = \frac{\delta E_{xc}}{\delta n(\mathbf{r})} \quad (2.33)$$

Assim, a equação de Schroedinger que é calculada pelo método DFT é dada pela equação 2.34 e 2.35.

$$\left[-\frac{1}{2} \nabla^2 + v_n(\mathbf{r}) + V_H(\mathbf{r}) + v_{xc}(\mathbf{r}) \right] \phi_i(\mathbf{r}) = \varepsilon_i \phi_i(\mathbf{r}) \quad (2.34)$$

$$\left[-\frac{1}{2} \nabla^2 + v^{KS} \right] \phi_i(\mathbf{r}) = \varepsilon_i \phi_i(\mathbf{r}) \quad (2.35)$$

Sendo v^{KS} o potencial efetivo que os elétrons sentem e ε_i a função de onda de cada elétron, como a energia de correlação e troca é composta por termos que não sabemos descrever de forma exata, são tomadas aproximações para que seja possível aplicar o método DFT. Para metais a aproximação comumente

utilizada é a *Perdew–Burke–Ernzerhof generalized gradient approximation* (PBE-GGA) [33].

Podemos ver na figura 2.4 o ciclo de auto-consistência onde essas equações são implementadas em softwares.

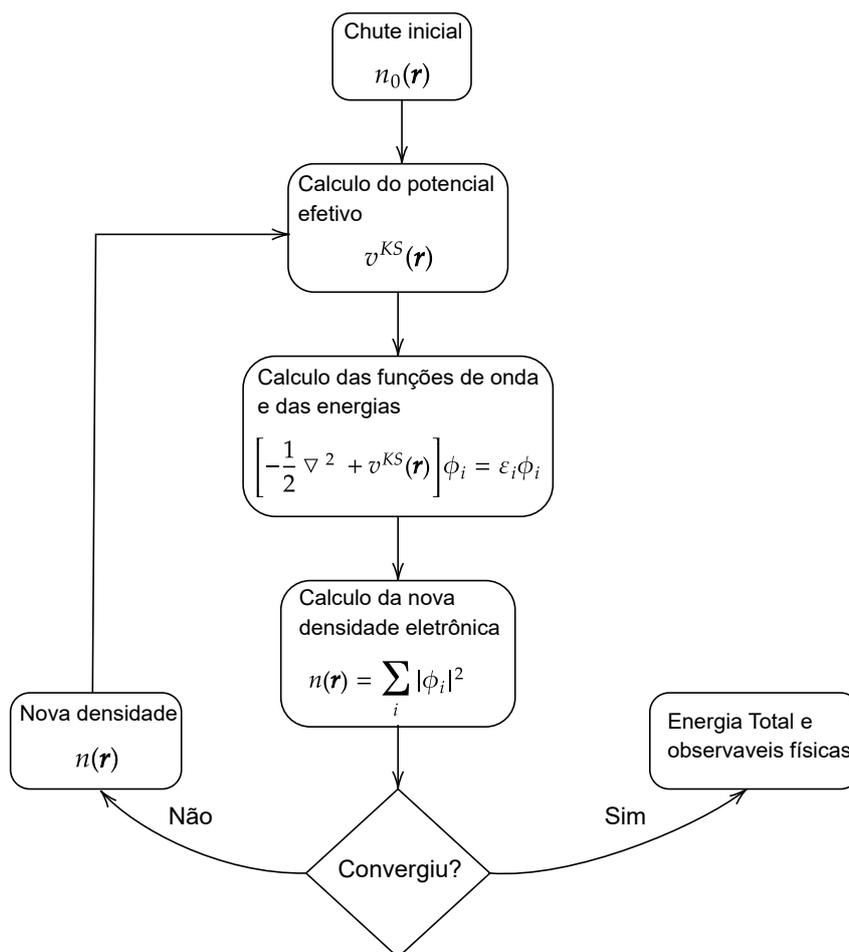


Figura 2.4: Diagrama esquemático do ciclo de auto-consistência.

Esses cálculos não levam em consideração contribuições de natureza térmica, ou seja, todos os cálculos são realizados em uma temperatura de 0 Kelvin. O que pode ser uma fonte de incertezas nas energias totais obtidas quando estas forem usadas em outras temperaturas no modelo proposto. Além disso, dependendo do potencial de correlação e troca utilizado, diferentes parâmetros obtidos podem ser subestimados ou superestimados, sendo necessário estudar qual se ajusta melhor a aplicação desejada. Por exemplo, o potencial PBE-GGA diminui os erros nos valores das energias de ligação e coesão de sólidos em compa-

ração ao potencial *local density approximation* (LDA), que é o potencial mais simples que pode ser utilizado, e superestima as ligações atômicas fazendo com que fiquem mais negativas do que realmente são [34].

Uma utilização do DFT, que foi utilizada no trabalho, é calcular a energia de uma estrutura cristalina variando seu volume para minimizar a energia total do sistema. Desta forma, é possível encontrar a energia total daquela estrutura que, como será visto mais a frente, permitirá fazer aproximações da entalpia de formação de diferentes fases. Podemos ver um exemplo do resultado desse tipo de cálculo na figura 2.5, que foi realizado para determinar a energia total do Sc com estrutura cúbica de face centrada (CFC).

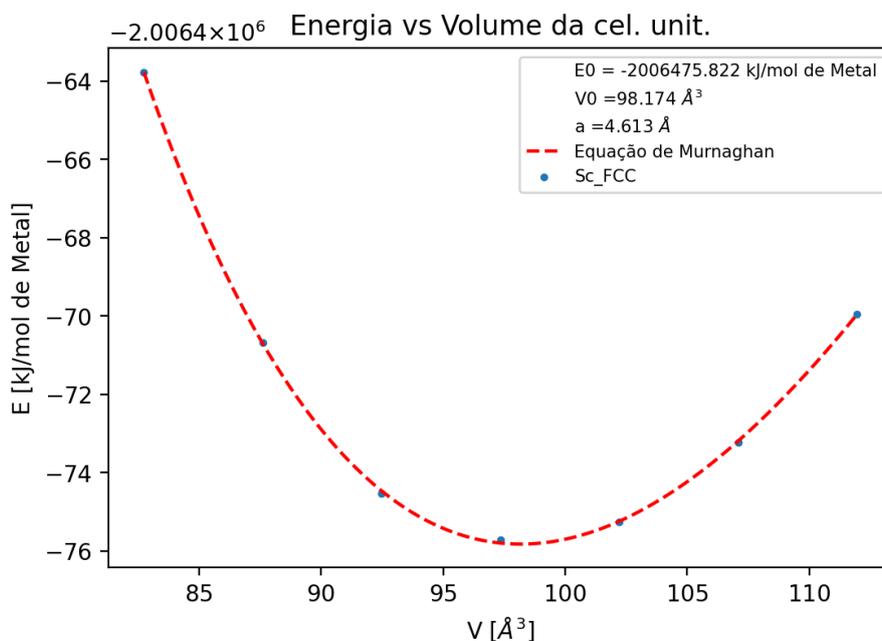


Figura 2.5: Energia total em função do volume da célula unitária do Sc com estrutura cristalina CFC

A equação de estado de Murnaghan é obtida através da expansão do módulo volumétrico em potências da pressão, tomando a pressão ambiente como referência. Então, em primeira ordem, é obtida a pressão em função do volume, como podemos ver na equação 2.36.

$$P(V) = \frac{B_0}{B'_0} \left[\left(\frac{V}{V_0} \right)^{-B'_0} - 1 \right] \quad (2.36)$$

onde P é a pressão em função do volume, B_0 é o módulo volumétrico a pressão ambiente, B'_0 é o módulo de compressibilidade a pressão ambiente, V_0 é o volume a pressão ambiente e V é o volume final.

É uma aproximação da pressão em função do volume que frequentemente é utilizada em cálculos *ab-initio* [35]. Essa utilização ocorre devido aos cálculos serem feitos em $T = 0$ K, onde é possível relacionar a pressão e volume com a energia. Essa relação é dada pela equação 2.37.

$$P = -\frac{dE}{dV} \quad (2.37)$$

Calculando a energia a partir das equações 2.36 e 2.37, é obtido a expressão da energia em função do volume:

$$E(V) = E_0 + \frac{B_0 V}{B'_0} \left[\left(\frac{V_0}{V} \right)^{B'_0} \frac{1}{B'_0 - 1} + 1 \right] - \frac{B_0 V_0}{B'_0 - 1} \quad (2.38)$$

Nos cálculos *ab-initio* de novas estruturas é necessário determinar o volume que minimiza a energia. Para isso, é utilizada a expressão 2.38 para ajustar os dados obtidos por DFT com o intuito de determinar V_0 que é o volume que fornece a menor energia E_0 .

Essas energias mínimas são utilizadas para obter as energias de ligação entre dois elementos em cada fase que será estudada no modelo.

3 OBJETIVOS

Este trabalho de mestrado tem como objetivo principal contribuir na implementação de ferramentas computacionais para o cálculo de diagramas PCT de ligas multicomponentes com estrutura do tipo CCC. Para tanto, será feita a incorporação de novos elementos, a saber: Mg, Al, Sc, Mn, Fe, Co, Cu, Zn, Mo e Pd, ao modelo inicialmente proposto por Zepon et al restrito a oito elementos: Ti, V, Cr, Ni, Zr, Nb, Hf e Ta. Além disso, busca-se desenvolver um programa de código aberto para o cálculo de diagramas PCT com base no modelo proposto por meio de dados obtidos por DFT. Por fim, pretende-se avaliar as ligas reportadas na literatura com o modelo proposto e verificar a faixa de propriedades obtidas, as limitações do modelo e as potencialidades para projetar ligas com propriedades otimizadas.

4 MATERIAIS E MÉTODOS

4.1 Cálculos de Energia Total por DFT

Para o modelo termodinâmico utilizado no trabalho é necessário a realização de diversos cálculos de DFT para a obtenção da energia total mínima das estruturas estudadas. Os cálculos de DFT foram efetuados utilizando o software Wien2K, que é um programa que realiza os cálculos de primeiros princípios utilizando como base para as funções de onda uma combinação de orbitais locais que se ajusta em torno das posições atômicas e de ondas planas aumentadas nas regiões entre os átomos (APW+lo) [36, 37]. Essa combinação faz com que o software seja de potencial total, já que considera todos os elétrons (de caroço e valência) dos átomos.

Alguns parâmetros de entrada são necessários para que os cálculos sejam feitos pelo software. Os principais são: a aproximação da energia de correlação e troca, que neste trabalho será utilizada a aproximação PBE-GGA [33]; a energia de corte que é dada por $RMT \cdot k_{max}$, onde k_{max} é o valor do maior vetor de onda na expansão de onda plana e RMT é o menor raio da esfera atômica na célula unitária; e o número de divisões na rede recíproca (*k-points*).

O cálculo das estruturas contendo apenas átomos metálicos foram realizados usando $RMT \cdot k_{max} = 7,5$. Enquanto as estruturas contendo átomos H foram calculadas usando $RMT \cdot k_{max} = 4,0$. Nos cálculos, foi utilizada uma divisão de 10.000 pontos (*k-points*) na zona de Brillouin irreduzível. O critério de convergência de energia para o cálculo do ciclo auto-consistente é de 0,0001 Ry.

Os valores de energia total foram calculados variando o volume da estrutura. A determinação do valor mínimo foi feita pelo ajuste dos resultados com a equação de estado de Murnaghan, equação 2.38. A estimativa inicial dos parâmetros de rede das estruturas é dada pela média ponderada dos parâmetros de rede de cada elemento que compõe a estrutura. Também é importante lembrar que todos os resultados obtidos são para a temperatura de 0 K, portanto, é importante destacar, que há um erro associado a temperatura.

4.2 Adição de elementos no modelo termodinâmico

Para adicionar um novo elemento A no modelo termodinâmico proposto por Zepon et al.[5], deve-se realizar uma série de cálculos por DFT como descrito na seção 4.1 para a obtenção das energias médias de ligação do novo elemento com os elementos já presentes no modelo.

Primeiramente, é preciso determinar a entalpia parcial molar de hidrogênio no novo elemento (h_A) para as estruturas CCC e CFC. Para as estruturas CCC, os valores foram considerados como sendo os valores experimentais da entalpia de dissolução infinita do hidrogênio nos elementos puros, que podem ser obtidas na referência [38]. Estes valores foram considerados mesmo que os elementos puros não possuíssem estrutura CCC. Esta aproximação é justificada por duas razões. Primeiramente, pois é reportado que a entalpia de dissolução de hidrogênio de vários elementos que possuem transformação alotrópica é praticamente idêntica para as diferentes formas alotrópicas, como no caso do Ti e Zr, por exemplo. Em segundo lugar, determinação de entalpia de dissolução de fases com baixos teores de hidrogênio por DFT é muito mais custosa computacionalmente, pois é necessário a construção de supercélulas com vários átomos para que baixos teores de H seja considerado. Por esta razão, as mesmas considerações feitas por Zepon et al. em [5] foram feitas aqui.

Para a estrutura CFC é necessário calcular as energias totais mínimas das estruturas A^δ e AH_2^δ . Com essas energias, pode-se utilizar a equação 2.25 para determinar a entalpia parcial molar da estrutura CFC. A energia total do gás hidrogênio ($E_{total}^{H_2}$) foi determinada em [5] como sendo -3073 kJ/mol. Este valor foi determinado utilizando dados experimentais da entalpia de formação padrão dos hidretos TiH_2 , VH_2 e NbH_2 , e cálculos de energias totais por DFT utilizando os mesmos parâmetros que os utilizados neste trabalho. Desta forma, o mesmo valor para $E_{total}^{H_2}$ será utilizada neste trabalho. A estrutura cristalina dos hidretos AH_2^δ é descrita com o grupo espacial 225 (Fm-3m) com os átomos de metal nas posições de Wyckoff 4a e os átomos de hidrogênio nas posições de Wyckoff 8c $(\frac{1}{4}, \frac{1}{4}, \frac{1}{4})$.

Outro parâmetro importante para a adição do novo elemento no modelo é a

energia de ligação de um átomo do elemento com ele mesmo e com os demais elementos inclusos no modelo. Para determinar as energias médias de ligação também são utilizados cálculos de DFT das estruturas das fases α (CCC) e δ (CFC) dos elementos puros. No caso do cálculo para a energia média de ligação entre dois átomos do mesmo elemento as estruturas utilizadas possuem apenas os átomos do elemento a ser adicionado, para a determinação das energias totais mínimas que serão usadas nas equações 2.20 e 2.21, para obtermos a energia médias de ligação em cada estrutura:

$$\varepsilon_{A-A}^{\alpha} = \frac{E_{total A}^{\alpha}}{4} \quad (4.1)$$

$$\varepsilon_{A-A}^{\delta} = \frac{E_{total A}^{\delta}}{6} \quad (4.2)$$

No caso da energia de ligação com outros elementos é necessário que os cálculos de DFT sejam feitos nas estruturas B2 e L1₀, que representam as estruturas das fases α e δ , respectivamente. Além disso, essas estruturas possuem uma composição $A_{0,5}j_{0,5}$, onde j é um elemento já incluso no modelo termodinâmico. Para a fase α , a energia mínima obtida pelo DFT é utilizada na equação 2.20:

$$\varepsilon_{A-j}^{\alpha} = \frac{E_{A_{0,5}j_{0,5}}^{B2}}{4} \quad (4.3)$$

Já para a fase δ , a energia total mínima obtida por DFT da estrutura L1₀ é utilizada na equação 2.23:

$$\varepsilon_{A-j}^{\delta} = \frac{E_{A_{0,5}j_{0,5}}^{L1_0} - \varepsilon_{A-A}^{\delta} - \varepsilon_{j-j}^{\delta}}{4} \quad (4.4)$$

para determinar $\varepsilon_{A-A}^{\delta}$ e $\varepsilon_{j-j}^{\delta}$ é necessário que seja construído uma relação entre energia de ligação e distância entre os átomos ligantes, já que as ligações entre dois átomos iguais numa rede CFC e na L1₀ possuem diferentes distâncias. Portanto, é necessário um ajuste de um polinômio de terceiro grau da energia em função da distância entre os átomos ligantes, utilizando os coeficientes desse

polinômio é possível estimar a energia de ligação do elemento (ε_{A-A}^δ e ε_{j-j}^δ) na distância em que ocorre essa ligação na estrutura $L1_0$. Essa interpolação polinomial é realizada na forma da equação 4.5, onde a , b , c e d são os coeficientes do polinômio e d_a é a distância entre os dois átomos adjacentes do elemento.

$$E(d_a) = ad_a^3 + bd_a^2 + cd_a + d \quad [\text{kJ/mol}] \quad (4.5)$$

Essa interpolação é aplicada para as mesmas distâncias onde foi calculado o DFT para a estrutura $L1_0$ do par de elementos, a menor energia encontrada de ε_{j-j}^δ é a utilizada.

Os valores obtidos das entalpias parciais molares de hidrogênio e das energias de ligação são incluídos no banco de dados utilizado pelo software que realiza os cálculos do modelo termodinâmico para obter os diagramas PCT. Para a adição dos elementos Mg, Al, Sc, Mn, Fe, Co, Cu, Zn, Mo e Pd foram necessárias a determinação da energia total de 280 estruturas.

4.3 Implementação do cálculo dos diagramas PCT em Software de código aberto

Com intuito de facilitar a utilização do modelo proposto para o design de ligas para o armazenamento de hidrogênio, foi realizada a implementação dos cálculos computacionalmente. Para isso, foi escolhido uma linguagem de programação de código aberto e de fácil aprendizado para que qualquer pessoa interessada possa conferir e aprimorar o programa que será desenvolvido. A linguagem Python foi escolhida, visto que esta linguagem vem ganhando cada vez mais adeptos no meio acadêmico, além de possuir diversas bibliotecas e ferramentas acessíveis para facilitar a implementação do cálculo dos diagramas PCT. O fluxograma do código desenvolvido pode ser visto na figura 4.1.

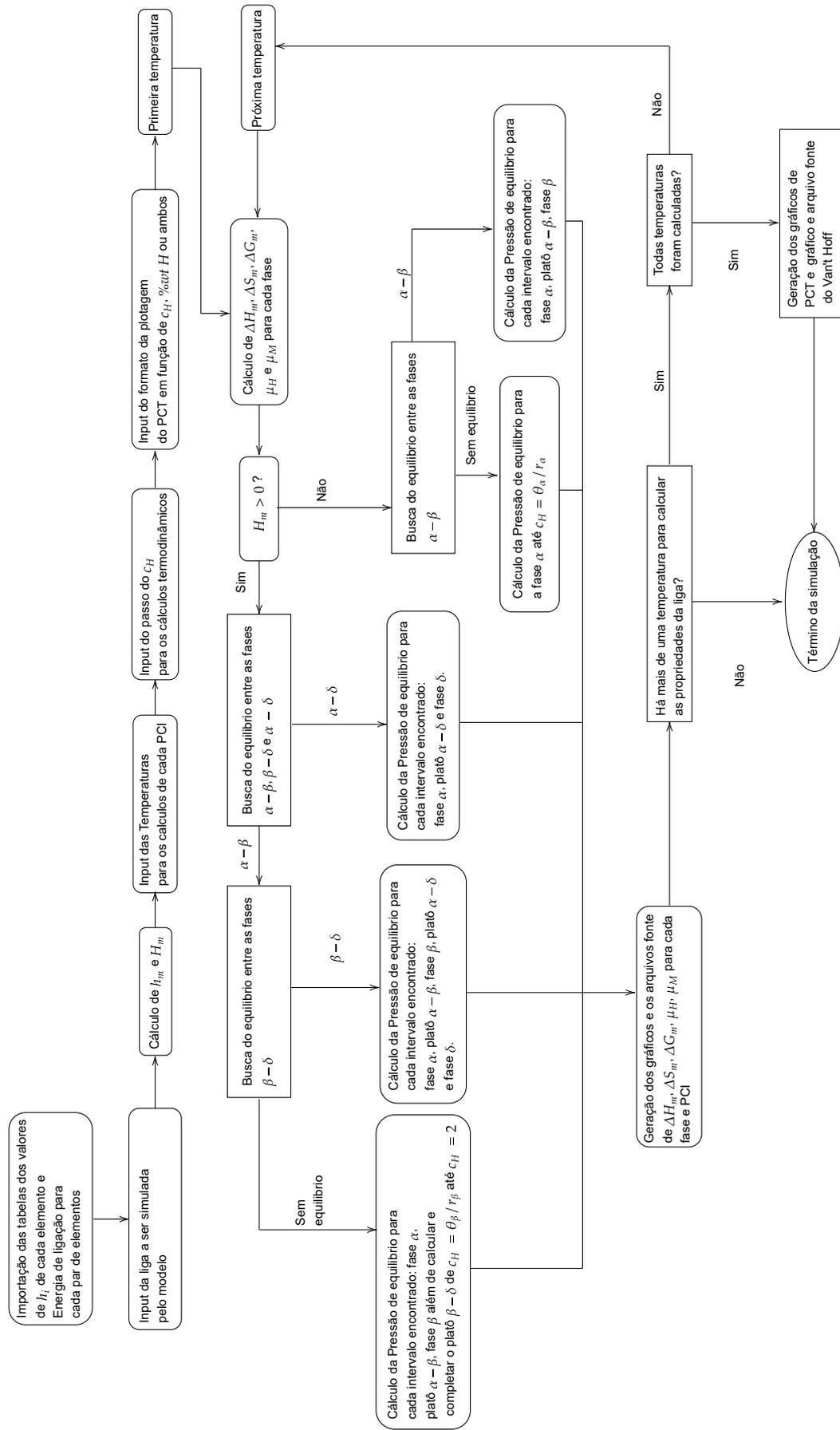
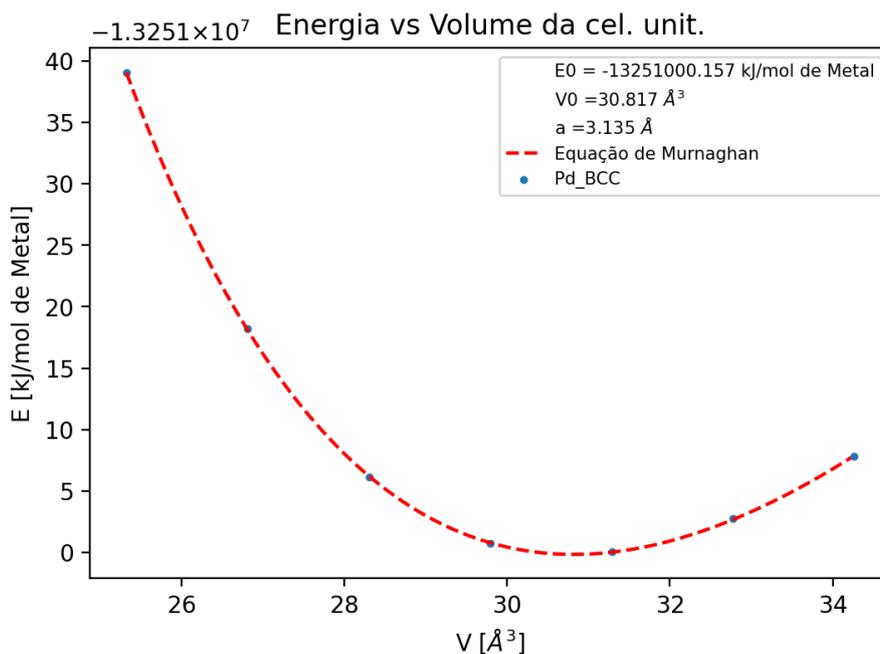


Figura 4.1: Fluxograma do software do Modelo

5 RESULTADOS

5.1 Cálculos das entalpias parciais molares de Hidrogênio e energias de ligação utilizadas no modelo

Foram realizadas 27 simulações de DFT para a determinação das entalpias parciais molares de hidrogênio. Essas simulações consistiram na utilização do DFT para as estruturas CFC e MH_2 (CFC) para cada um dos elementos a serem incorporados - Mg, Al, Sc, Mn, Fe, Co, Cu, Zn, Mo e Pd - onde foram obtidos valores da energia total em função do volume da célula unitária. Com esses valores foi ajustada a curva da equação de Murnaghan dada pela equação 2.38, o que permitiu obter os valores dos volumes que minimizam a energia para cada elemento em cada estrutura e a energia mínima da estrutura. Um exemplo desses ajustes pode ser visto nas figuras 5.1a, 5.1b e 5.1c para o Paládio, os ajustes para os outros elementos estão reportados no apêndice A.



(a) Pd - CCC

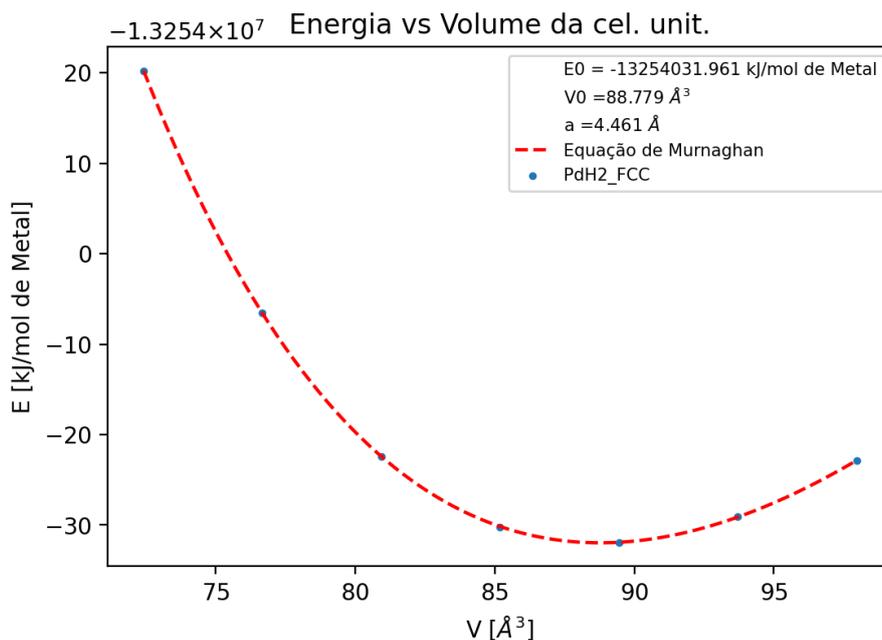
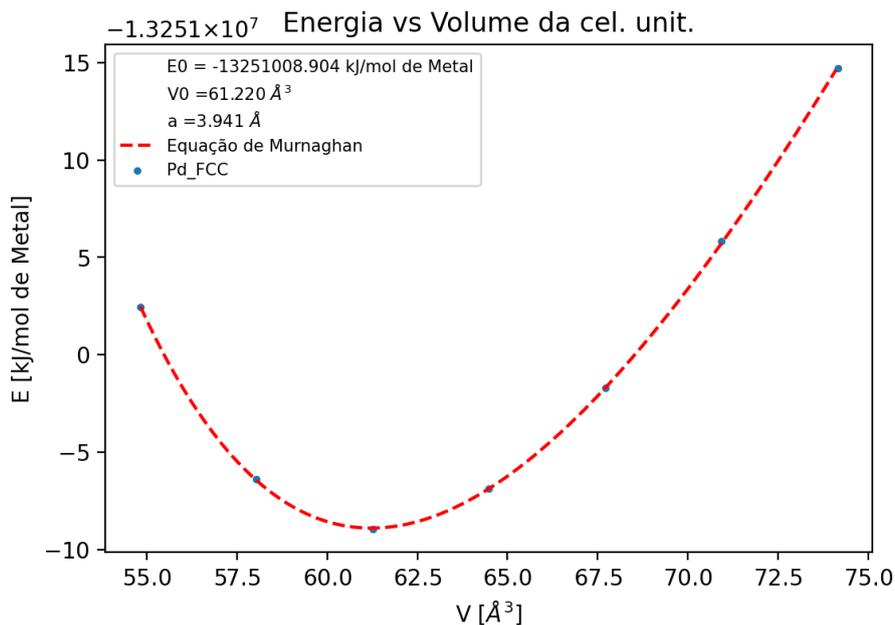


Figura 5.1: Energia total em função do volume da célula unitária para as estruturas CCC, CFC e MH₂ para o elemento Paládio. As energias devem ser somadas ao valor de referência no canto superior esquerdo.

Com os valores das energias mínimas obtidas através do ajuste foi possível calcular as entalpias parciais molares de hidrogênio para a fase δ com a equação 2.25. Os resultados dos cálculos são apresentados na tabela 5.1. Enquanto os

valores da entalpia parcial molar de hidrogênio para as fases α e β apresentados na tabela 5.1 foram aproximados como sendo a entalpia de dissolução de hidrogênio em diluição infinita dos elementos puros tirados da referência [38].

Tabela 5.1: Entalpia parcial molar de hidrogênio para cada elemento do modelo. Valores em kJ/mol de H

Sistema	Fase α	Fase β	Fase δ
Mg	21,0	21,0	-2,9
Al	63,0	63,0	55,4
Sc	-90,0	-90,0	-91,1
Ti	-58,0	-58,0	-64,2
V	-29,5	-29,5	-31,8
Cr	52,0	52,0	2,2
Mn	-2,0	-2,0	27,2
Fe	29,0	29,0	32,6
Co	21,0	21,0	18,9
Ni	16,0	16,0	18,5
Cu	55,0	55,0	72,1
Zn	15,0	15,0	84,8
Zr	-64,0	-64,0	-74,4
Nb	-35,3	-35,3	-40,8
Mo	15,0	15,0	0,8
Pd	-10,0	-10,0	25,6
Hf	-38,0	-38,0	-52,0
Ta	-36,0	-36,0	-6,4

Para a obtenção das energias de ligação que são utilizadas na determinação da entalpia referente a transição de fase das estruturas das ligas foram realizadas 253 otimizações por DFT. Sendo duas simulações para determinar a energia mínima, logo, a energia de ligação, para cada par de elementos nas estruturas B2 e L1₀. Como exemplo, são apresentados os gráficos resultantes do DFT e do ajuste da equação de Murnaghan para os pares de elementos Cobalto e Vanádio nas figuras 5.2a e 5.2b, as curvas para os demais elementos estão reportadas no Apêndice A.

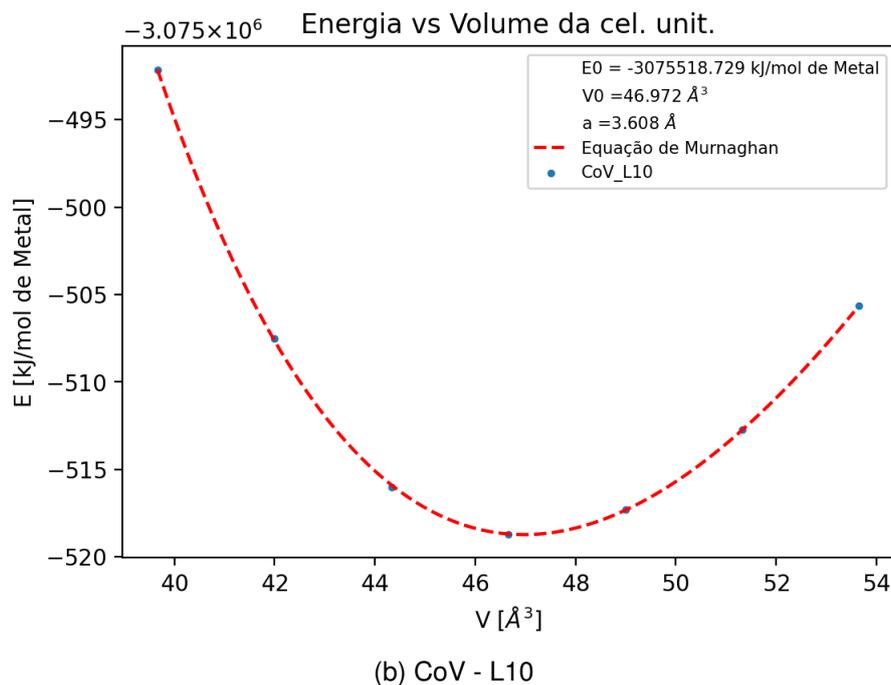
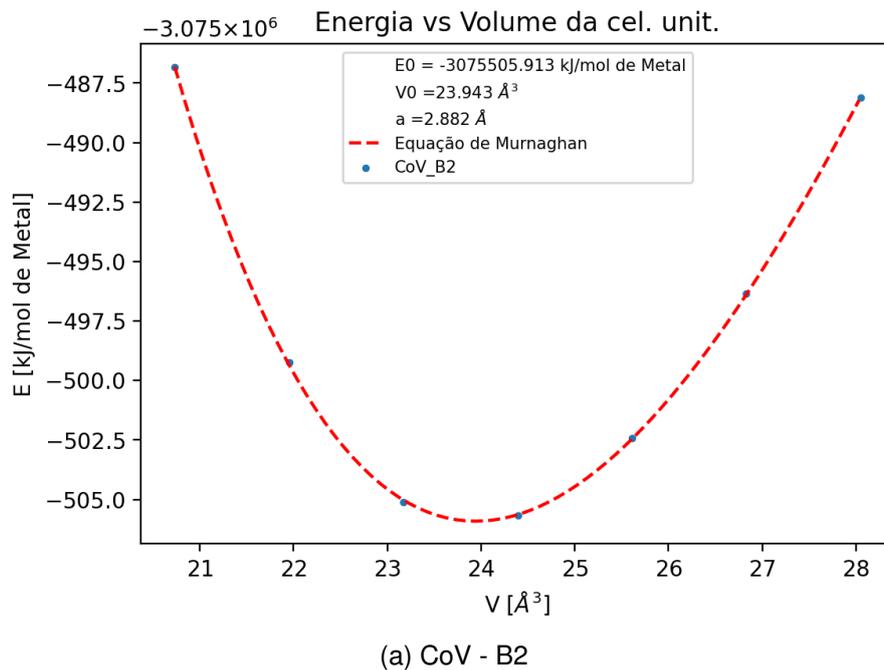


Figura 5.2: Energia total em função do volume da célula unitária para as estruturas B2 e L10 para determinar as energias de ligação entre os elementos Co - V. As energias no eixo y devem ser somadas ao valor de referência no canto superior esquerdo.

Com a energia mínima obtida nesses ajustes foi calculado as energias de ligação entre os pares de elementos para cada estrutura utilizando a equação 4.4. Porém, como as estruturas CFC e L1₀ são diferentes, foi necessário o uso de

uma interpolação polinomial, na forma da equação 4.5, para estimar a energia de ligação dos átomos simulados através da estrutura CFC para a distância utilizada na estrutura $L1_0$, os coeficientes dessa equação estão descritos na tabela 5.2.

Tabela 5.2: Coeficientes da interpolação polinomial para determinar Energia de ligação [kJ/mol] em função da distância da ligação dos átomos para as estruturas CFC e $L1_0$

Elemento	a	b	c	d	Elemento	a	b	c	d
Ti	-95,16	932,49	-3009,06	-370410,08	Mg	-14,79	176,5	-675,3	-86825
V	-150,64	1383,87	-4179,90	-411241,09	Al	-100,94	937,73	-2886,7	-103308
Cr	-192,97	1688,50	-4854,54	-455245,49	Mn	-318,39	2602,8	-7035,7	-500710
Zr	-67,69	743,94	-2681,70	-1571773,51	Fe	-336,37	2696,4	-7151,4	-550672
Nb	-121,25	1252,13	-4234,82	-1667059,85	Zn	-87,61	794,97	-2389,1	-783572
Hf	-73,99	808,10	-2892,52	-6603149,53	Sc	-49,375	536,04	-1920,9	-332137
Ta	-183,26	1824,25	-5991,88	-6831214,89	Cu	-233,89	1919,9	-5231,8	-719477
Ni	-168,60	1418,68	-3928,09	-661900,61	Mo	26,59	9,64	-700,9	-1770720
Co	-291,51	2346,6	-6247	-604251	Pd	14,71	7,11	-385,52	-2207800

Assim, com a interpolação foi possível determinar as energias de ligação dos pares de elementos na fase δ com a equação 4.4. Enquanto para a fase α não foi preciso um tratamento para determinar a energia de ligação ε_{j-j} , sendo apenas necessário a energia mínima através do ajuste com a equação de Murnaghan e a equação 4.3. Os valores das energias de ligação calculadas para as fases α e δ estão descritas na tabela 5.3.

5.2 Desenvolvimento do programa em código aberto

O modelo discutido em 2.2.1 foi implementado na linguagem de programação Python e o código do programa foi desenvolvido seguindo o fluxograma da figura 4.1. Para a sua utilização é necessário abrir o código em um interpretador de Python, como o Spyder, por exemplo, que é um interpretador voltado para programação científica. O código está disponibilizado no apêndice B e no link do *GitHub*: https://github.com/oapedroso/Model_MH.

Em conjunto com o código é necessário a utilização de quatro arquivos auxiliares de formato .csv, que são as tabelas que contém os valores das energias de ligação para os pares dos elementos nas fases α e δ , as entalpias parciais molares de hidrogênio dos elementos inclusos no modelo e a massa atômica de cada elemento para a conversão de $[H]/[M]$ para %massa H. Essas tabelas devem estar na mesma pasta do código do modelo e estão disponibilizadas no link do *GitHub*.

Ao executar o código, é necessário fornecer alguns parâmetros para realizar o cálculo dos diagramas PCT das ligas. As entradas necessárias são: i) os elementos da liga e sua composição; ii) as temperaturas em que as curvas PCI da liga devem ser determinadas; iii) o passo na composição de hidrogênio na liga, i.e., a variação de c_H à cada ponto em que as grandezas serão calculadas; iv) os parâmetros r e θ para o modelo de entropia para cada estrutura e, por fim; v) se os gráficos devem ser gerados em função de c_H , %massa H ou ambos.

Após receber todas as informações necessárias para a simulação dos diagramas PCT, o código calcula a entalpia parcial molar de hidrogênio (h_m) para cada possível fase da liga (α , β e δ) através da equação 2.24. Também é calculada o valor de H_M^δ usado na descrição da entalpia da fase delta, a partir das energias de ligação da tabela 5.3 e as equações 2.19, 2.20, 2.21 e 2.22. Para as fases α e β o valor de H_M é nulo já que o modelo considera que não há uma transição de fase entre α e β , e a fase α com $c_H = 0$ é a referência.

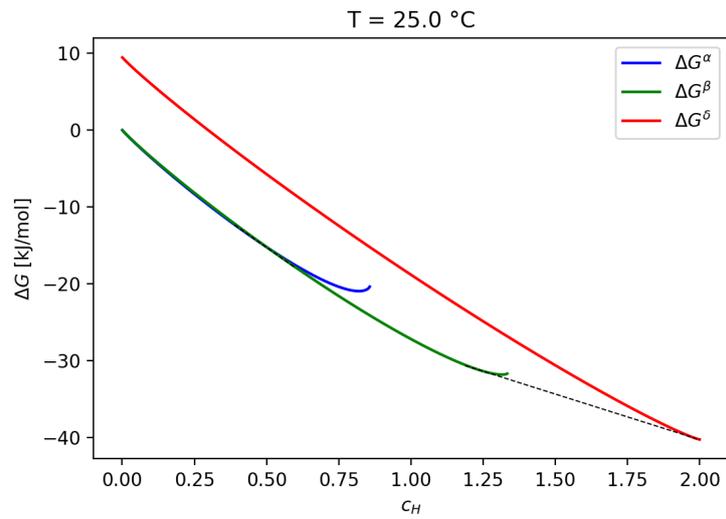
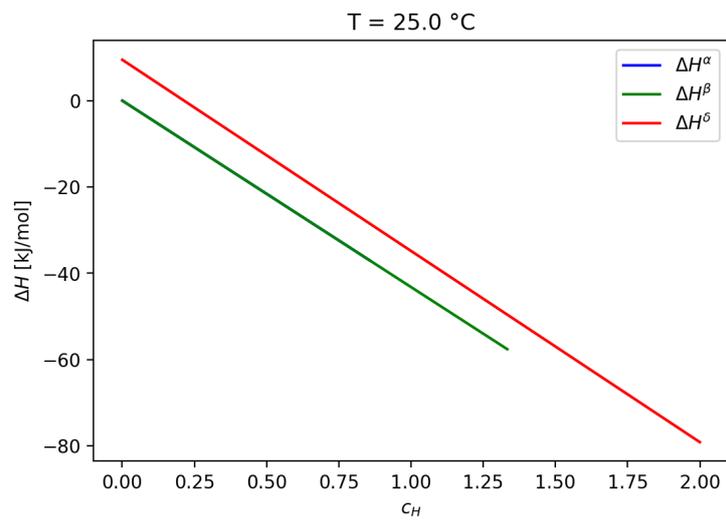
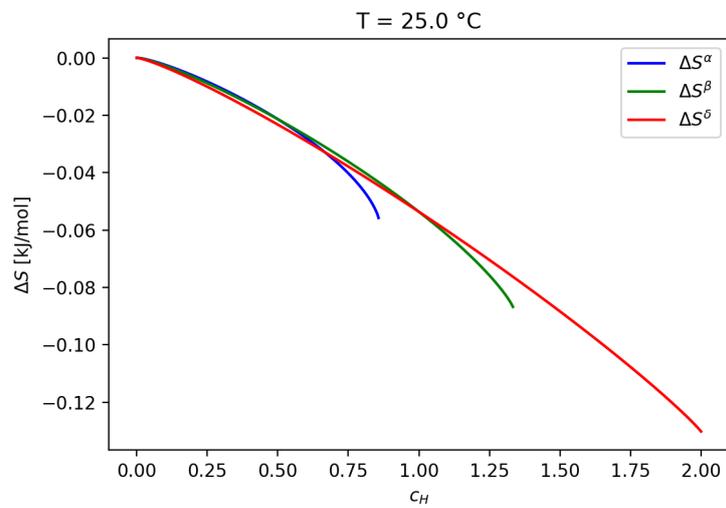
Com isso é possível calcular as grandezas termodinâmicas - ΔH_m , ΔS_m , ΔG_m , μ_H e μ_M - para cada valor de c_H , utilizando o passo escolhido (por padrão 0.0005), essas grandezas são calculadas e armazenadas para cada fase

no intervalo $0 < c_H < \theta/r$.

Com essas grandezas é possível buscar por equilíbrios químicos entre as fases, buscando as tangentes comuns entre as fases, ou seja, a condição de igualdade entre os potenciais químicos. Logo há a possibilidade de ocorrer quatro casos nessa busca: 1) encontrar um equilíbrio das fases $\alpha - \delta$; 2) encontrar um equilíbrio das fases $\alpha - \beta$ e um equilíbrio das fases $\beta - \delta$; 3) encontrar apenas o equilíbrio das fases $\alpha - \delta$; 4) não encontrar nenhum equilíbrio.

Encontrado as condições de equilíbrio, são definidos os limites dos intervalos de c_H para cada fase, onde são utilizados os valores calculados previamente dentro desses limites. Com isso são gerados os gráficos de cada grandeza em função da concentração de hidrogênio e é calculada a pressão do gás hidrogênio através do equilíbrio do gás com cada fase para cada c_H , assim, obtendo a curva PCI da liga. No caso do cálculo ser realizado para mais de uma temperatura, também são gerados o diagrama PCT e gráfico de Van't Hoff, isto é $\ln P_{Plat}$ vs $1/T$.

A liga $\text{TiVZr}_{0.5}\text{NbTa}_{0.5}$ é um exemplo da situação na qual foi encontrado um equilíbrio das fases $\alpha - \beta$ e um equilíbrio das fases $\beta - \delta$. Os gráficos da figura 5.3 foram obtidos com o código desenvolvido nesse trabalho. Para a simulação do PCI dessa liga foi utilizado os parâmetros: $\theta_\alpha = 6$, $r_\alpha = 7$, $\theta_\beta = 4$, $r_\beta = 2$, $\theta_\delta = 2$ e $r_\delta = 1$. A figura 5.3a apresenta o gráfico de ΔG_m e nele as tangentes comuns que determinam os potenciais químicos do metal e do hidrogênio no equilíbrio. As figuras 5.3b e 5.3c trazem a variação da entalpia e da entropia das fases, grandezas que compõem ΔG_m . Já nas figuras 5.3d e 5.3e exibem as curvas dos potenciais químicos em função da composição de hidrogênio na liga. Por fim, a figura 5.3f expõe o objetivo do código, o PCI da liga $\text{TiVZr}_{0.5}\text{NbTa}_{0.5}$. Nele é possível ver os intervalos de cada fase em cores diferentes e os valores da pressão de cada platô destacadas na legenda.

(a) $\text{TiVZr}_{0.5}\text{NbTa}_{0.5} - \Delta G_m(c_H)$ (b) $\text{TiVZr}_{0.5}\text{NbTa}_{0.5} - \Delta H_m(c_H)$ (c) $\text{TiVZr}_{0.5}\text{NbTa}_{0.5} - \Delta S_m(c_H)$

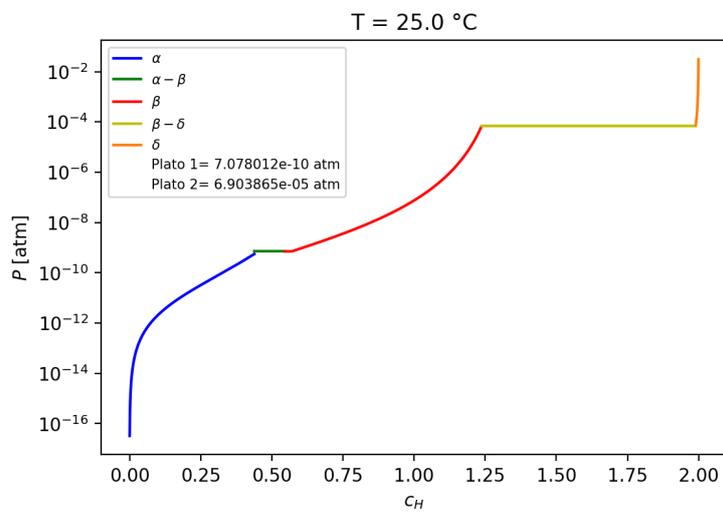
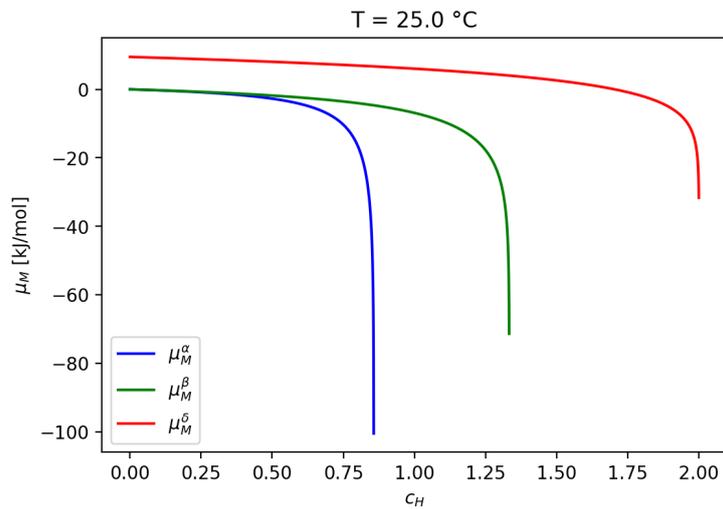
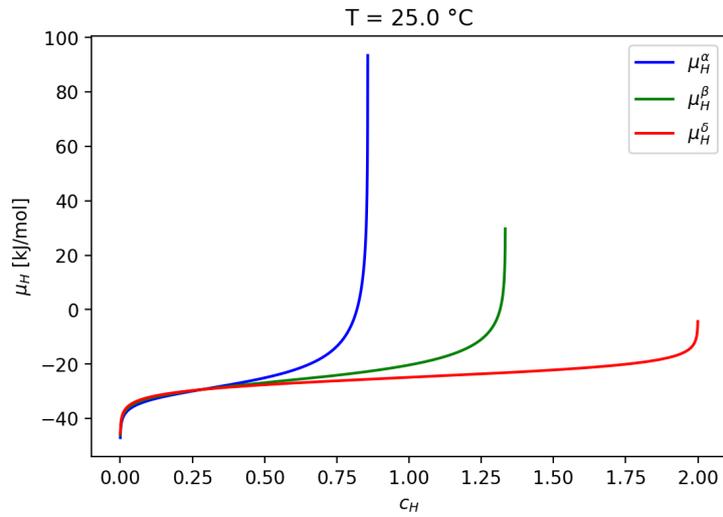
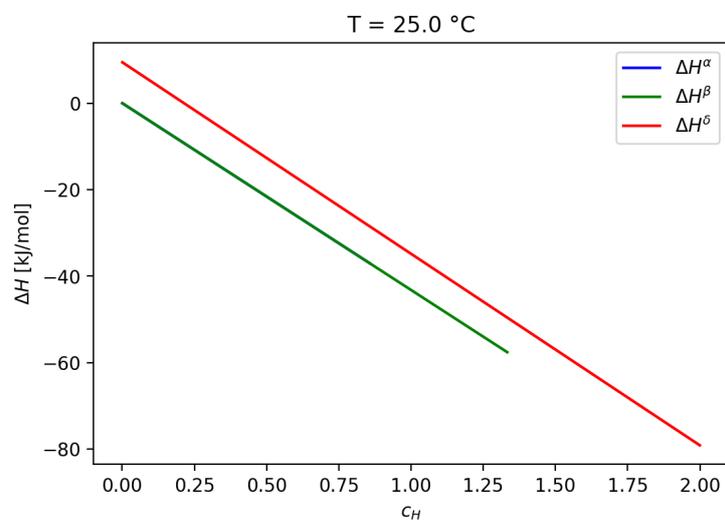
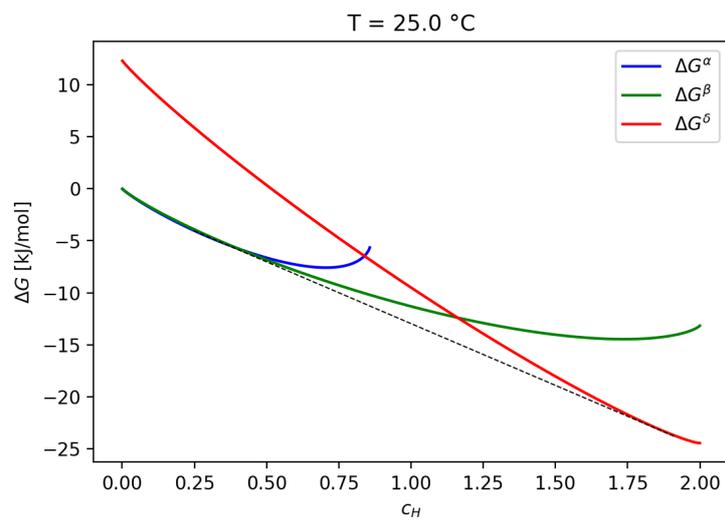
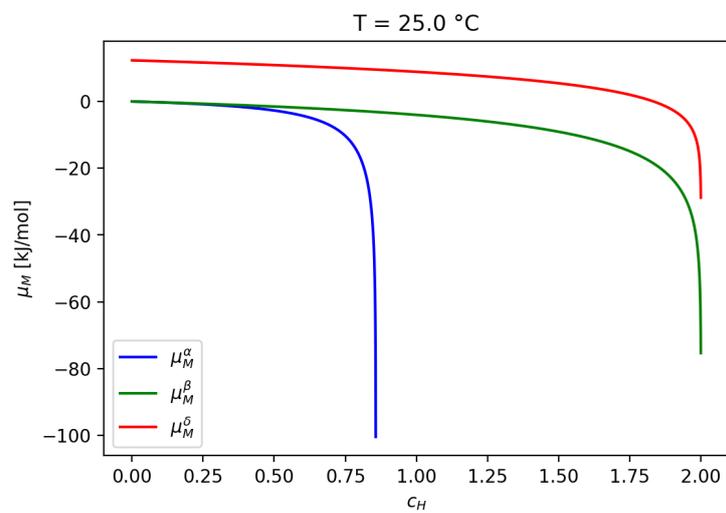
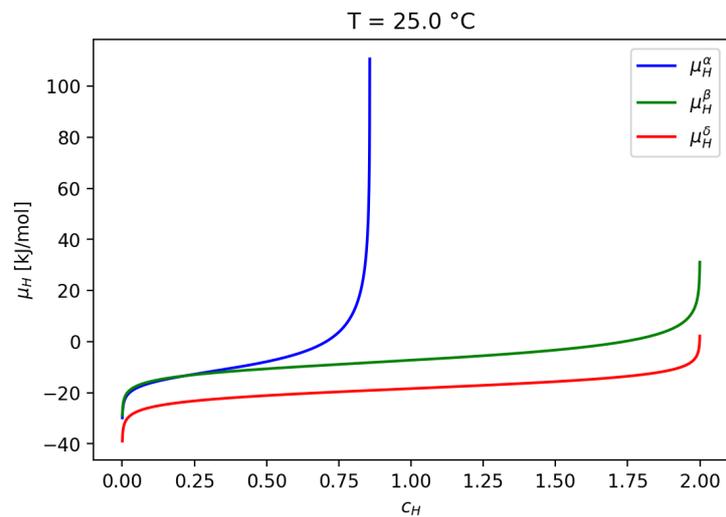
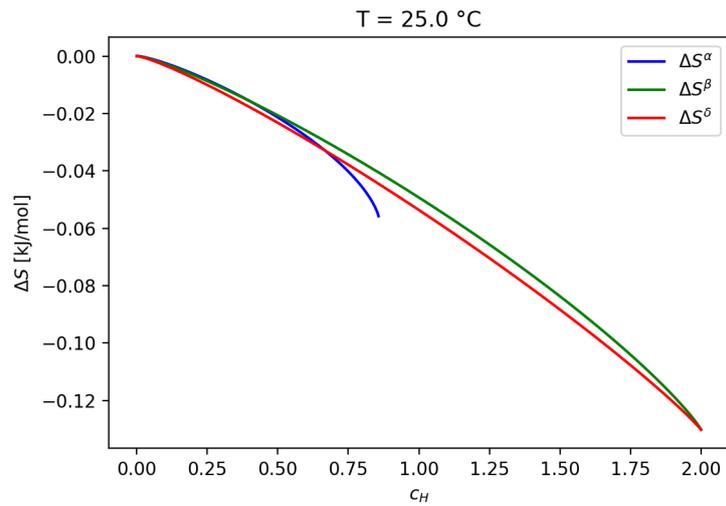


Figura 5.3: Gráficos obtidos através do código para a liga TiVZr_{0.5}NbTa_{0.5} na temperatura de 25 °C onde houve dois platôs de equilíbrio.

A liga TiZrMo_2Hf é um exemplo onde foi encontrado o equilíbrio entre as fases $\alpha - \delta$. Os gráficos obtidos para essa liga estão apresentados na figura 5.4. Para a simulação do PCI dessa liga foi utilizado os parâmetros: $\theta_\alpha = 6$, $r_\alpha = 7$, $\theta_\beta = 4$, $r_\beta = 2$, $\theta_\delta = 2$ e $r_\delta = 1$. Podemos destacar a figura 5.4a onde pode ser visto a curva de ΔG_m para cada fase e a tangente comum entre as fases α e δ , e a figura 5.4f onde é apresentado o PCI dessa liga a uma temperatura de 25 °C.





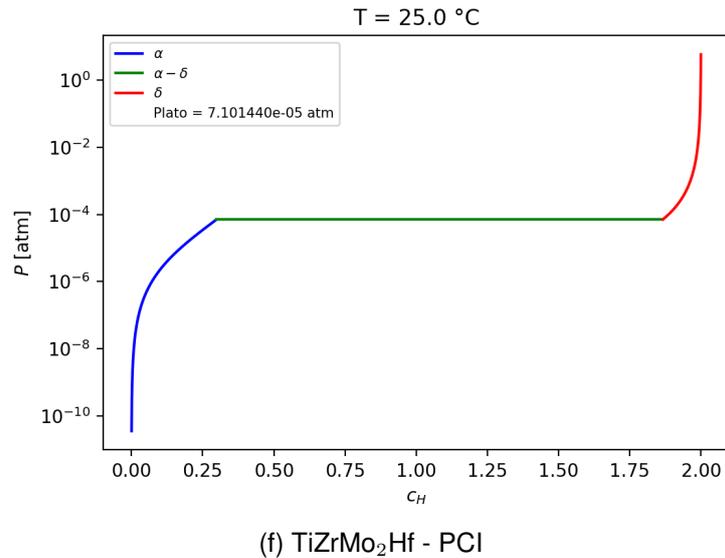
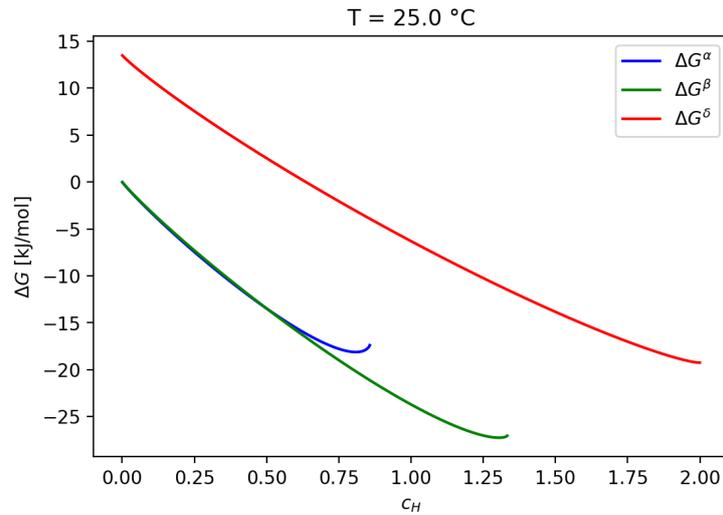
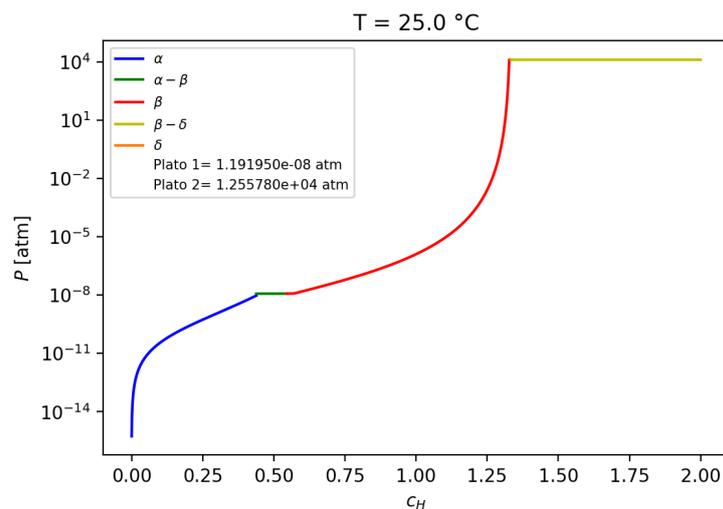


Figura 5.4: Gráficos obtidos através do código para a liga TiZrMo₂Hf na temperatura de 25 °C onde houve apenas um platô de equilíbrio.

As ligas onde se encontra apenas o equilíbrio dos platôs $\alpha - \beta$ podem ser divididas em dois casos, as que possuem $H_M > 0$ e as que possuem $H_M < 0$. No primeiro caso, apesar de encontrar o primeiro equilíbrio, o modelo não encontra a segunda transição, então é considerado que em $c_H = \theta_\beta/r_\beta$ a estrutura da fase β é saturada de hidrogênio e entra em equilíbrio com o hidreto CFC em $c_H = 2$, portanto, nesse caso o código calcula a pressão de platô considerando o equilíbrio entre $c_H = \theta_\beta/r_\beta$ e $c_H = 2$, para as fases β e δ , respectivamente. Um exemplo desse caso é a liga TiVNbTa onde pode ser visto na figura 5.5a as curvas de ΔG_m e o PCI resultante que é exposto na figura 5.5b. Estas curvas foram obtidas com os parâmetros de entropia: $\theta_\alpha = 6$, $r_\alpha = 7$, $\theta_\beta = 4$, $r_\beta = 3$, $\theta_\delta = 2$ e $r_\delta = 1$.

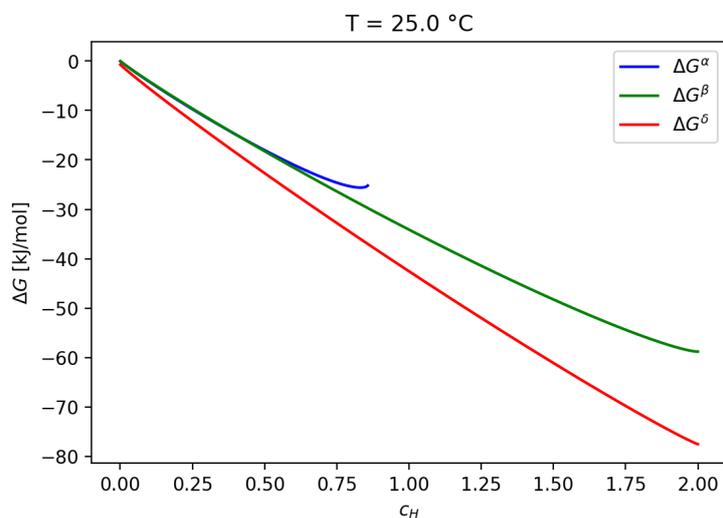
(a) TiVNbTa - $\Delta G_m(c_H)$ 

(b) TiVNbTa - PCI

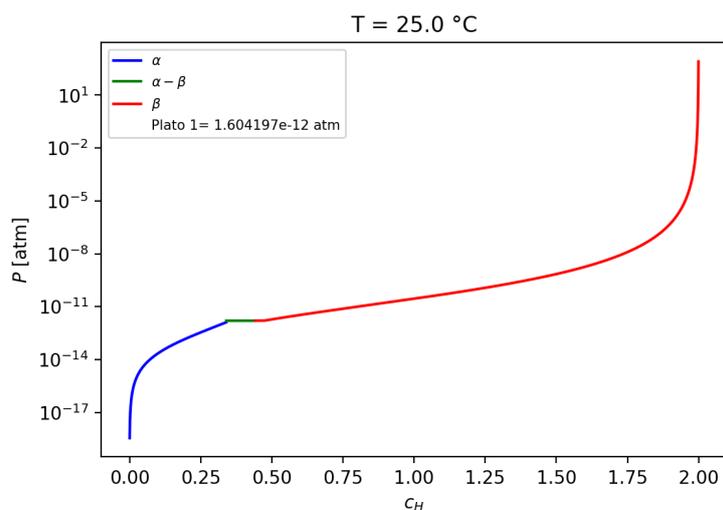
Figura 5.5: Gráficos obtidos através do código para a liga TiVNbTa, liga com $H_M > 0$, na temperatura de 25 °C.

Já no caso de $H_M < 0$, a estrutura CFC é vista como a mais estável, logo, não haveria uma transição, por isso não é possível encontrar um equilíbrio entre as fases β e δ . Porém, considerando que a liga é CCC inicialmente, logo há a procura por um equilíbrio entre as fases α e β . O valor negativo de H_M para algumas ligas pode vir das incertezas na obtenção das energias de ligação, mostrando a importância de refinar esses valores. Os gráficos de ΔG_m e do PCI para a liga TiZrNbHf, que é um exemplo desse caso, podem ser vistos nas figuras 5.6a e

5.6b. Para essa liga o valor de $H_M = -0,69$ kJ/mol. As curvas foram obtidas com os parâmetros de entropia: $\theta_\alpha = 6$, $r_\alpha = 7$, $\theta_\beta = 4$, $r_\beta = 2$, $\theta_\delta = 2$ e $r_\delta = 1$.



(a) TiZrNbHf - $\Delta G_m(c_H)$



(b) TiZrNbHf - PCI

Figura 5.6: Gráficos obtidos através do código para a liga TiZrNbHf, na temperatura de 25 °C, $H_M < 0$.

Por fim, na situação em que $H_M < 0$ e não é encontrado equilíbrio entre as fases α e β é gerado apenas o PCI da fase α até a composição $c_H = \theta_\alpha/r_\alpha$.

O código possui algumas limitações. A primeira delas é a necessidade de garantir previamente que a liga multicomponente a ser simulada será monofásica e de estrutura CCC, portanto, essa condição é essencial para obter resultados satisfatórios já que o modelo parte do pressuposto que a fase inicial é CCC.

A segunda limitação vem do fato de não conhecermos a maneira que os átomos de H ocupam os interstícios, principalmente para a fase β , o que influencia na escolha dos parâmetros de entrada θ e r para cada uma das fases, podendo levar a uma escolha incorreta que pode resultar em divisões por zero e gerando um aviso de erro nos cálculos. Portanto, é necessário ser investigado em projetos futuros se é possível determinar esses parâmetros a partir da composição química da liga.

A terceira limitação vem dos dados de DFT que alimentam o modelo, onde as suas incertezas podem levar a valores imprecisos de H_M . Logo, é necessário determinar de forma mais precisa essas energias para verificar se realmente há uma imprecisão nos valores calculados de H_M , ou se há um efeito do processamento na síntese de ligas que foram sintetizadas com estrutura CCC, mas possuem H_M negativos.

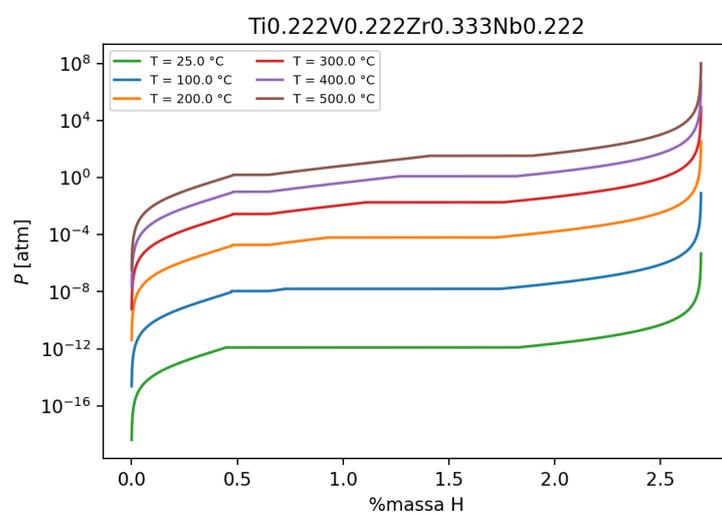
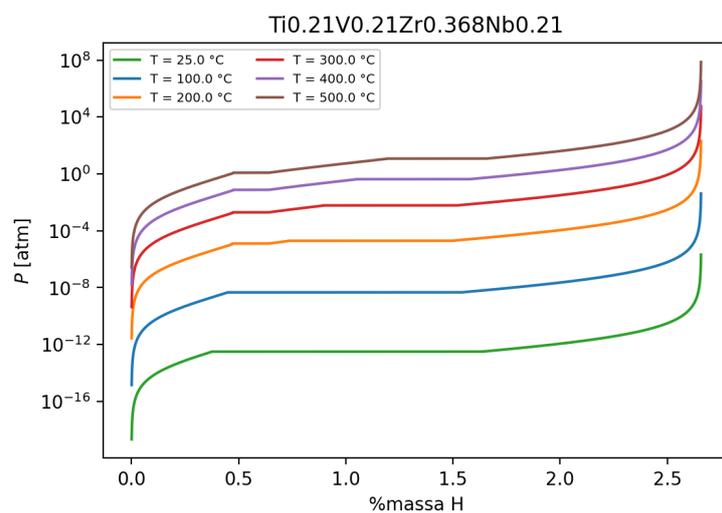
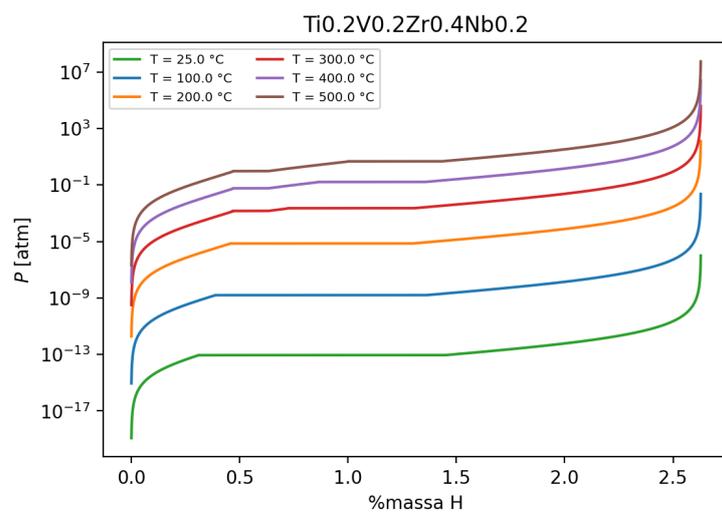
5.3 Análise sistemática das ligas reportadas na literatura

Com o código desenvolvido foi realizado uma análise sistemática das ligas reportadas na literatura. Para facilitar a análise, as ligas foram separadas nos sistemas: TiVZrNb, TiVZrNbTa, TiVZrNbHf, TiVCrNbMo, TiZrMoNbHf e TiVNbCr.

Para a análise individual de cada sistema, os diagramas PCT foram obtidos através do código desenvolvido utilizando por padrão os parâmetros $\theta_\alpha = 6$, $r_\alpha = 7$, $\theta_\beta = 4$, $r_\beta = 2$, $\theta_\delta = 2$ e $r_\delta = 1$ para a determinação da entropia configuracional. Algumas ligas tiveram os parâmetros para a fase β diferente e serão apontadas quando forem discutidas.

5.3.1 Sistema TiVZrNb

As ligas do sistema TiVZrNb que foram reportadas estão listadas na tabela 2.1. Os gráficos PCT obtidos através do modelo para essas ligas estão presentes na figura 5.7.



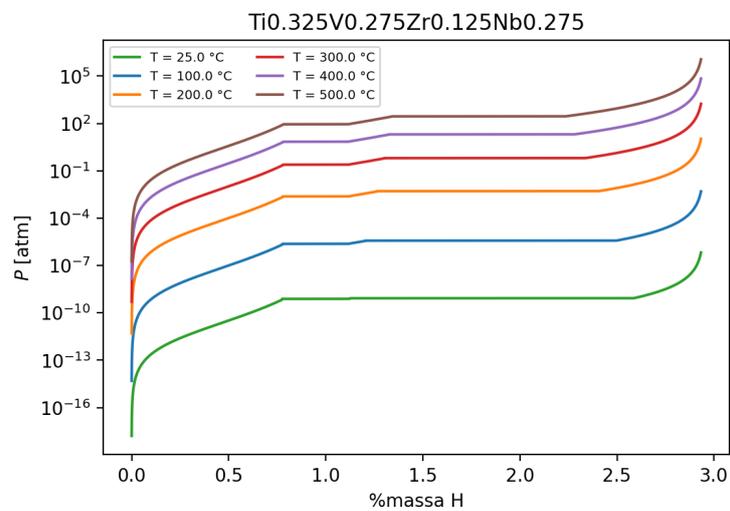
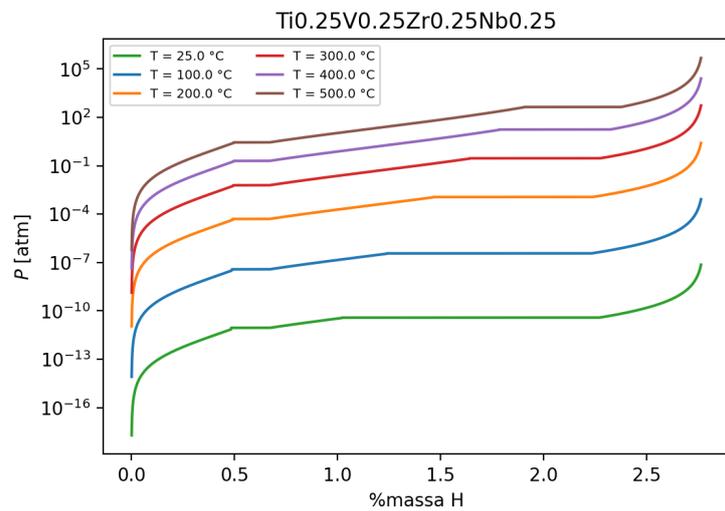
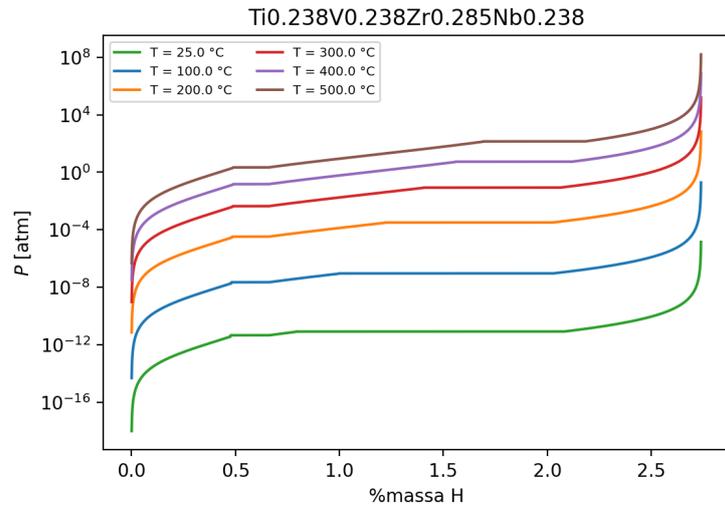


Figura 5.7: Diagramas PCT de ligas reportadas do sistema TiVZrNb.

Podemos observar que há uma tendência de mudança no comportamento da transição de fase com a razão $[Zr]/[M]$ na liga. Conforme essa razão vai diminuindo, as ligas tendem a possuir dois platôs de equilíbrio. Além disso, é possível observar que quanto maior a temperatura maior a diferença da pressão entre os platôs da liga, destacando que as ligas $TiVZr_2Nb$, $TiVZr_{1.75}Nb$ e $TiVZr_{1.5}Nb$ ainda apresentam apenas um platô em baixas temperaturas e passam a possuir dois platôs em temperaturas maiores.

Apenas a liga $TiVZrNb$ teve o PCI reportado na temperatura 294 °C [13]. Comparando com o PCT modelado, figura 5.7e, há uma discrepância na quantidade de platôs, enquanto experimentalmente foi observado uma transição direta para o hidreto CFC, o modelo encontrou dois platôs de equilíbrio. As pressões de platô experimental e simulada possuem uma ordem de grandeza de diferença, em torno de 0,03 atm para a medida experimental e 0,28 atm para o calculado. A comparação entre os diagramas PCI experimental e calculado pode ser vista na figura 5.8.

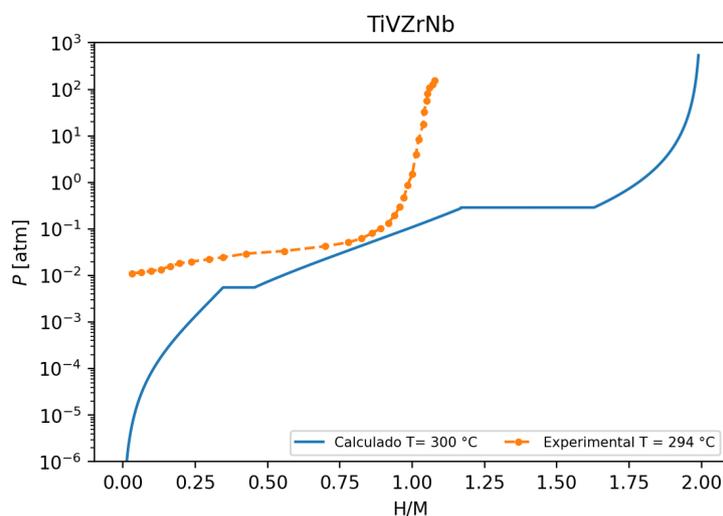


Figura 5.8: Comparação entre os diagramas PCI reportado e calculado para a liga $TiVZrNb$.

5.3.2 Sistema $TiVZrNbTa$

Dentre as ligas reportadas, apenas três tiveram seu PCI obtido experimentalmente. A primeira é a liga $TiZrNbHfTa$, que foi reportada na temperatura de 300 °C e com uma transição intermediária, sendo a pressão do platô do equilíbrio

entre as fases $\beta - \delta$ em torno de 22 atm [9]. Como foi reportada essa estrutura intermediária, devemos utilizar os parâmetros: $\theta_\beta = 3$ e $r_\beta = 2$, para a fase β , como relatado em [5].

A curva PCI obtida para essa liga está apresentada na figura 5.9, onde é observado que há o comportamento de um equilíbrio intermediário antes de alcançar o hidreto final e com uma pressão de platô $\beta - \delta$ por volta de 2 atm. Ou seja, as pressões de platô do modelo e a reportada possuem o valor dentro da mesma ordem de grandeza. Como não foi reportado a PCI em dessorção e sabendo que as ligas CCC podem apresentar histerese na pressão de platô que pode chegar a uma ordem de grandeza, é possível que o valor previsto pelo modelo esteja no intervalo de pressões que a histerese engloba como discutido em [5].

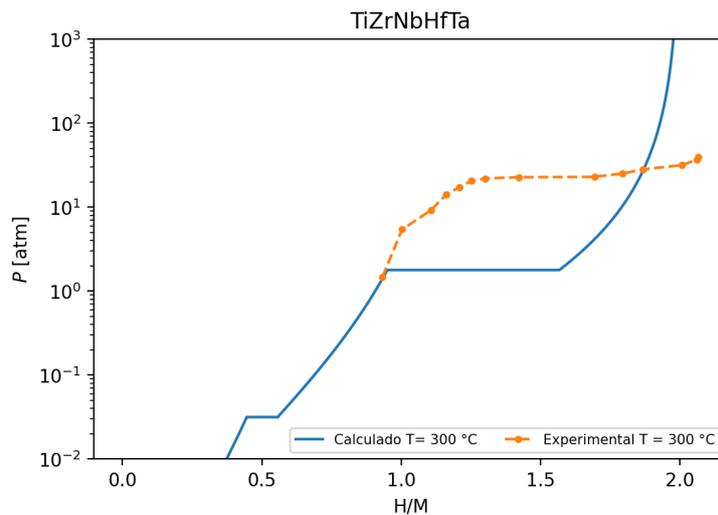
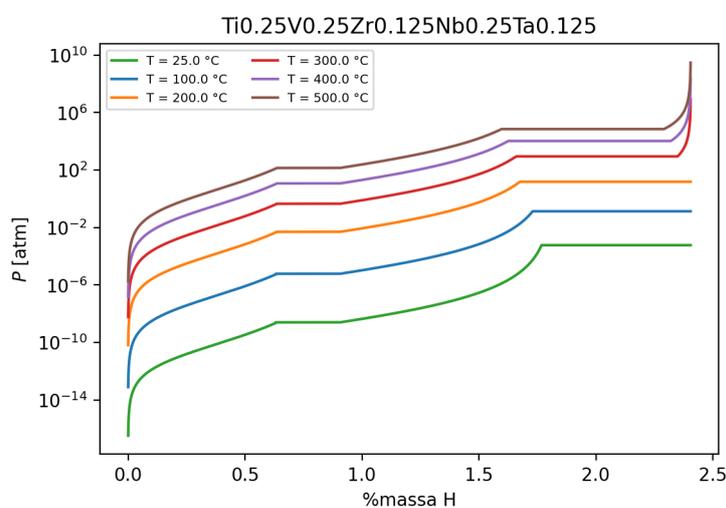
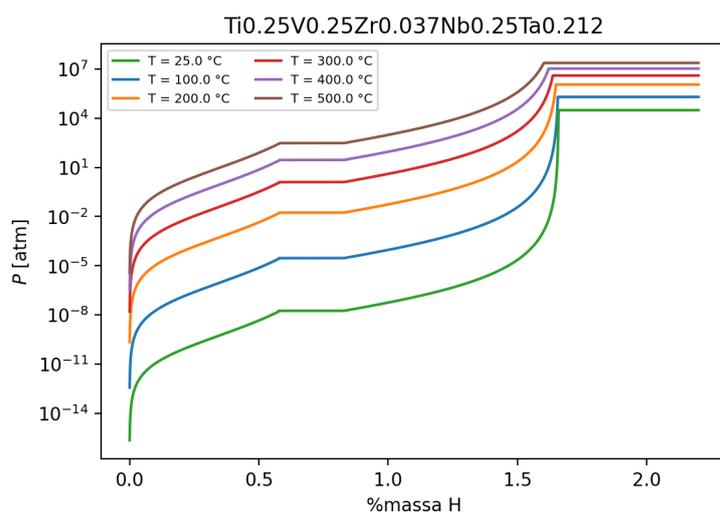


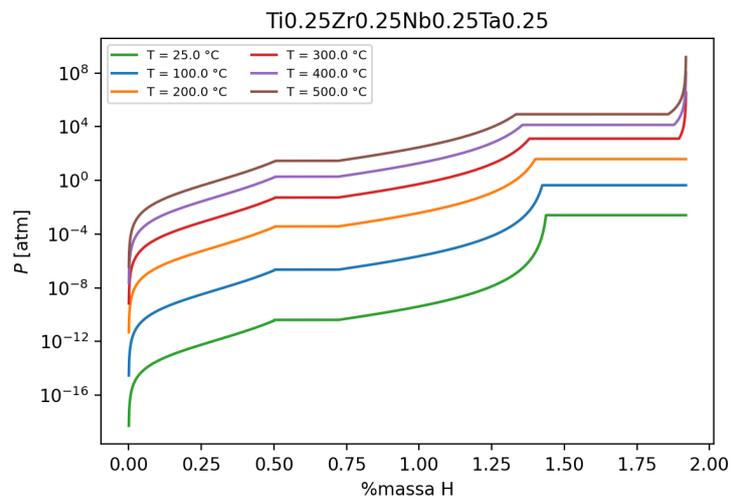
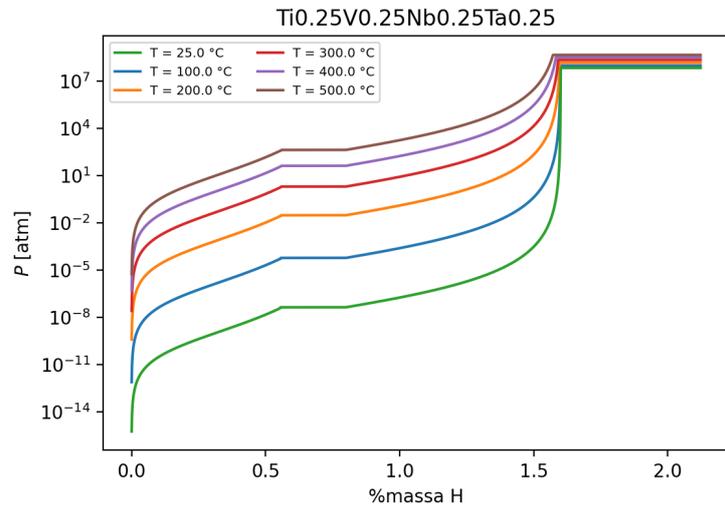
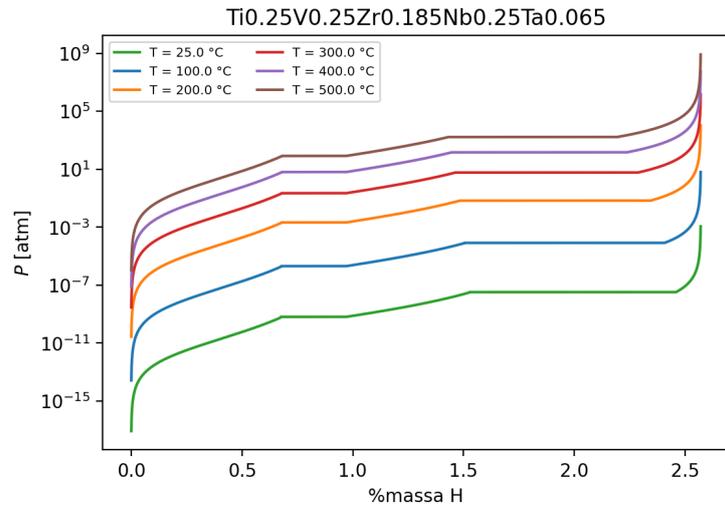
Figura 5.9: Comparação entre os PCIs calculados e reportados para a liga TiZrNbHfTa.

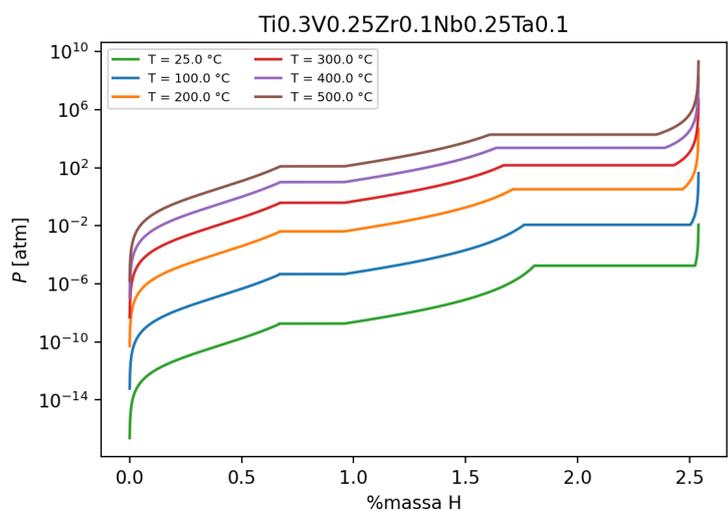
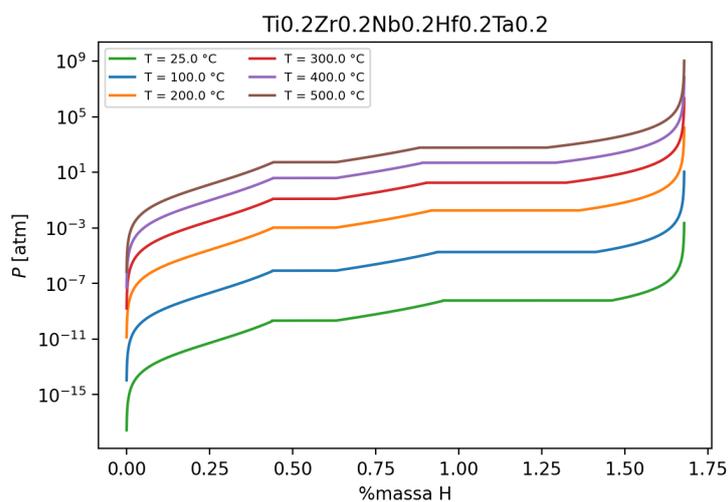
Outra liga que teve seu PCI reportado foi a liga $\text{Ti}_{0.3}\text{V}_{0.25}\text{Zr}_{0.1}\text{Nb}_{0.25}\text{Ta}_{0.1}$ [18], porém, a pressão do platô de equilíbrio não foi possível ser medida por ser muito baixa. Apesar de não ser possível realizar uma comparação direta na temperatura de 25 °C, pode ser visto na figura 5.10f dois platôs de equilíbrio, enquanto o artigo relata que houve uma transição direta, porém, ambos os platôs determinados estão em pressões baixas, abaixo de 0,001 atm, podendo então indicar um motivo para o equipamento não ser capaz de medir experimentalmente a pressão de platô da liga.

Por fim, na liga TiZrNbTa foi observado uma separação de fase durante a hidrogenação à 20 °C, 100 °C e 200 °C, dando origem a três hidretos [20] em cada temperatura. Visto que a separação de fase mostra que a condição de para-equilíbrio não foi respeitada, não sendo possível fazer uma comparação entre os diagramas PCT reportados e os obtidos através do modelo.

Os gráficos da figura 5.10 foram obtidos assumindo ser razoável utilizar os parâmetros da entropia configuracional $\theta_\beta = 3$ e $r_\beta = 2$ para as demais ligas do sistema TiVZrNbTa.





(f) $\text{Ti}_{0.3}\text{V}_{0.25}\text{Zr}_{0.1}\text{Nb}_{0.25}\text{Ta}_{0.1}$ 

(g) TiZrNbHfTa

Figura 5.10: Diagramas PCT calculados das ligas reportadas do sistema TiVZrNbTa utilizando os parâmetros reportados em [5].

No entanto, na figura 5.11 foram comparados os diagramas PCT para as ligas $\text{TiVZr}_{0.15}\text{NbTa}_{0.85}$ e TiZrNbHfTa com valores de θ_β e r_β diferentes, visando comparar os resultados obtidos com diferentes parâmetros da entropia configuracional. Foram utilizados os parâmetros $\theta_\beta = 4$ e $r_\beta = 3$ para comparação. A obtenção dos diagramas PCT com estes parâmetros diferentes tem a motivação de apresentar a importância de uma boa definição dos valores de θ e r para cada fase na previsão do modelo.

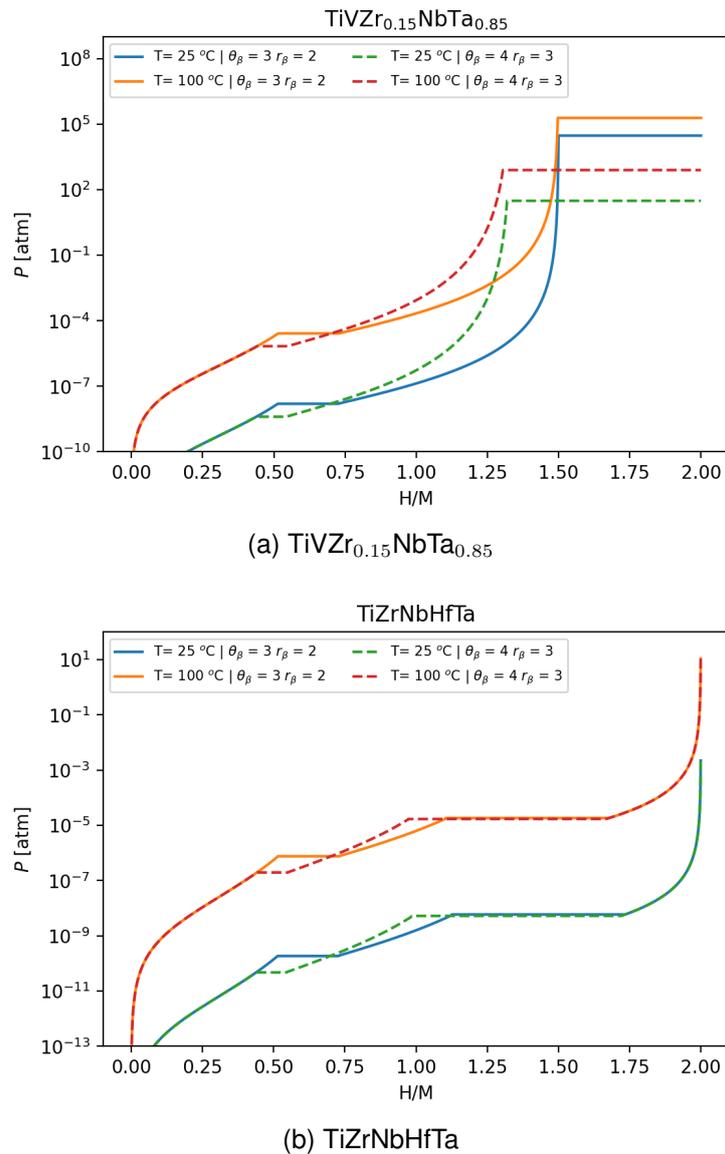


Figura 5.11: Exemplos de diagramas PCT de duas ligas reportadas do sistema TiVZrNbTa obtidas com parâmetros diferentes.

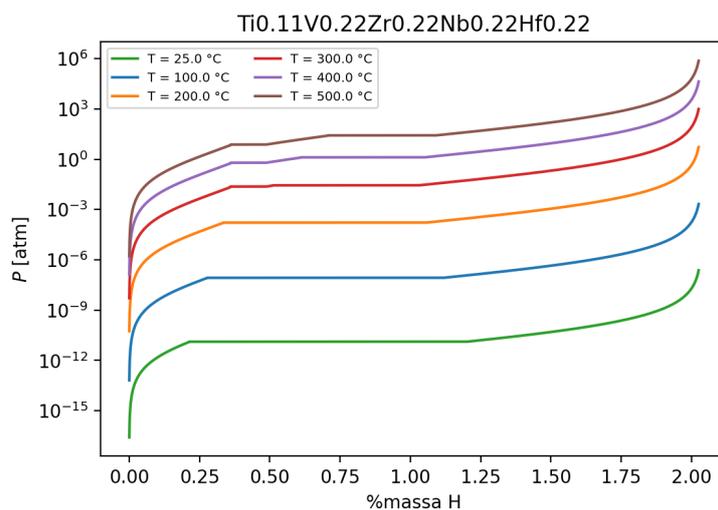
Comparando os gráficos dessas ligas é notável que uma mudança nos parâmetros pode resultar em grandes diferenças nos PCTs. Podemos observar que a diferença entre a pressão do platô do equilíbrio entre as fases $\beta - \delta$ da liga $\text{TiZr}_{0.15}\text{NbTa}_{0.85}$, figuras 5.11 e 5.10, é de 4 ordens de grandeza para a temperatura de 25 °C. Enquanto para a liga TiZrNbHfTa , que apesar de não possuir uma grande diferença na pressão de platô entre as fases $\beta - \delta$, há uma grande diferença no tamanho dos platôs.

Nos gráficos da figura 5.10 é observado que independentemente da tempe-

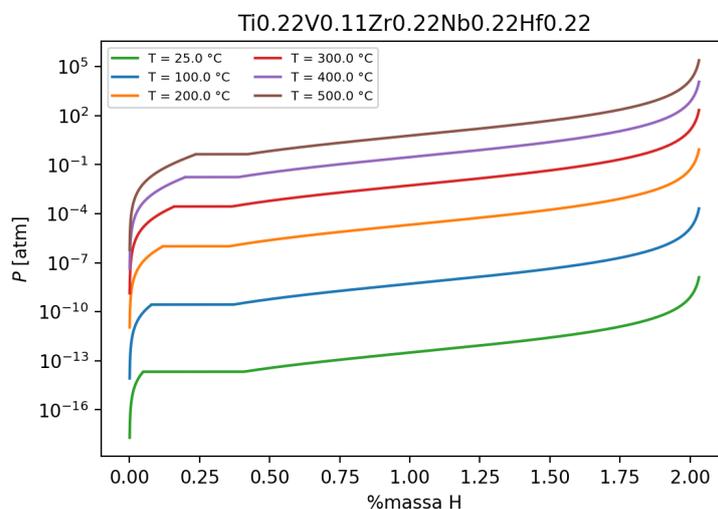
ratura ou composição, as ligas com Ta tendem a possuir uma transição intermediária, apresentando dois platôs de equilíbrio nos diagramas PCT de cada liga.

5.3.3 Sistema TiVZrNbHf

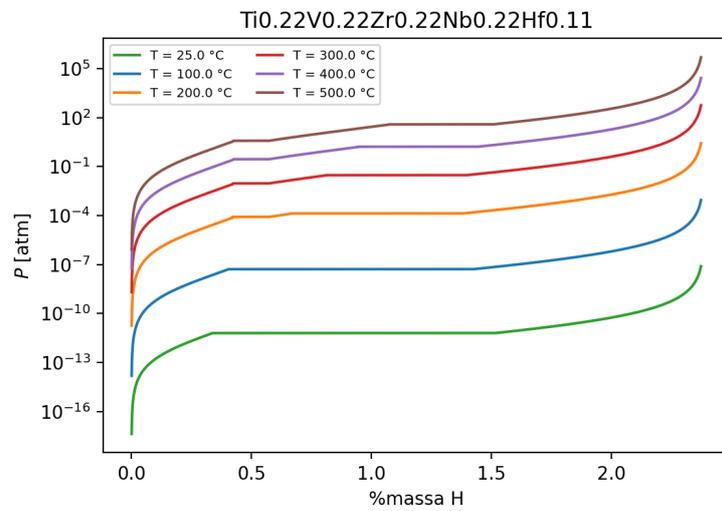
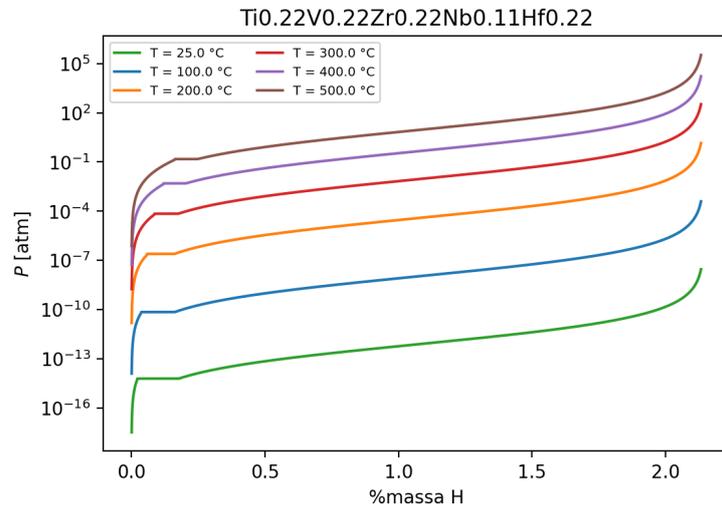
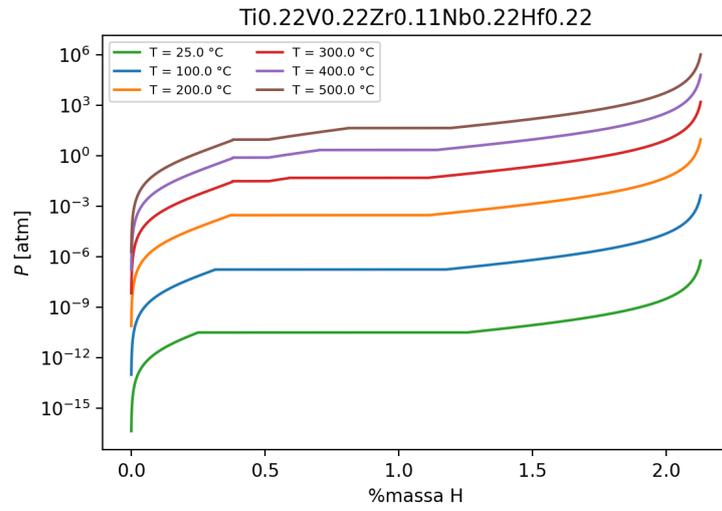
Algumas ligas do sistema TiVZrNbHf foram reportadas em [8], [10], [15], [13] e [22]. Há apenas algumas curvas PCI reportados para esse sistema, conforme indicado na tabela 2.1. Os gráficos PCT presentes na figura 5.12 foram obtidos através do modelo.



(a) $Ti_{0.5}VZrNbHf$



(b) $TiV_{0.5}ZrNbHf$



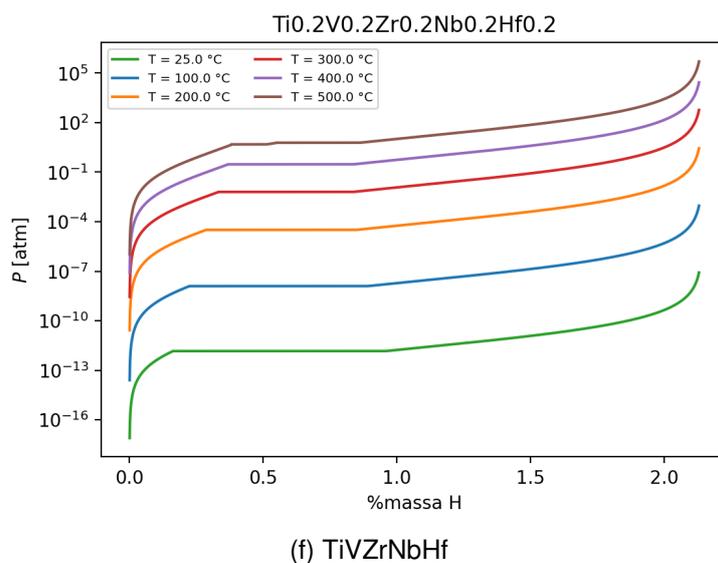
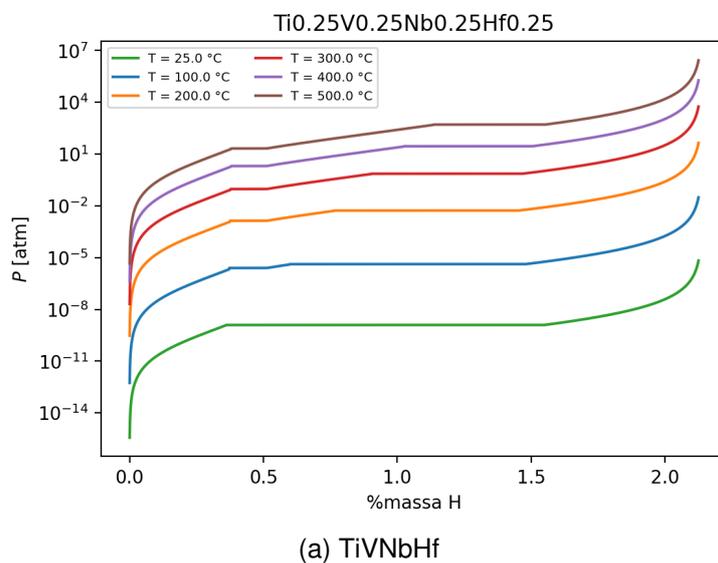


Figura 5.12: Diagramas PCT de ligas reportadas do sistema TiVZrNbHf obtidos através do código desenvolvido. Ligas com variação da composição dos elementos.

Nas ligas do sistema TiVZrNbHf, nos gráficos das figuras 5.12 e 5.13, há o predomínio do equilíbrio entre as fases α e δ , explicitado pelo único platô presente na maioria das temperaturas de cada gráfico. Dentre as ligas do sistema, é possível notar que as ligas que possuem uma menor quantidade de V ou Nb apresentam um platô com menor extensão, como pode ser visto nas ligas $\text{TiV}_{0.5}\text{ZrNbHf}$, $\text{TiVZrNb}_{0.5}\text{Hf}$, TiVZrHf e TiZrNbHf , figuras 5.12b, 5.12d, 5.13b e 5.13c, respectivamente.



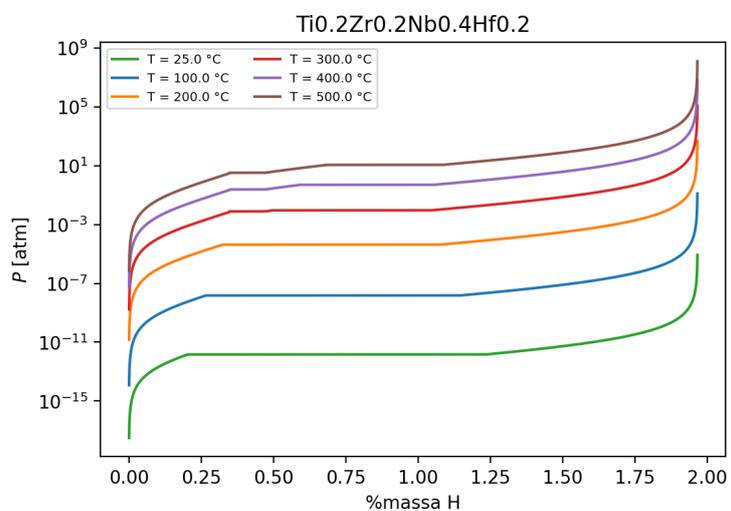
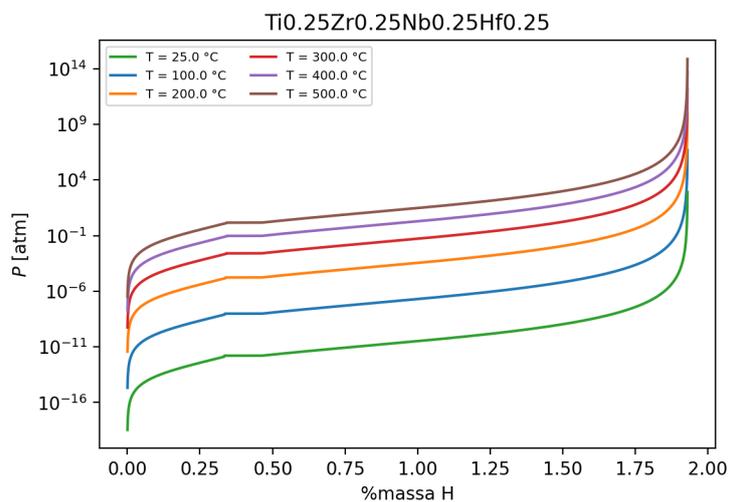
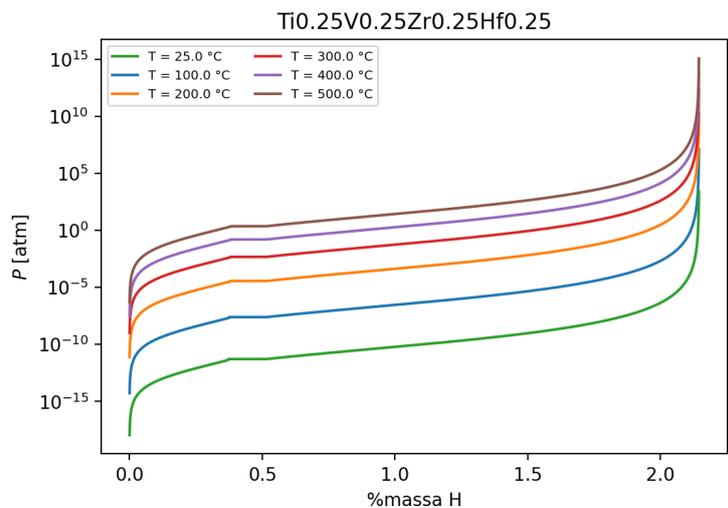


Figura 5.13: PCTs de ligas reportadas do sistema TiVZrNbHf obtidos através do código desenvolvido. Ligas com algum dos elementos retirados.

Já a liga TiVZrNbHf teve seu PCI reportado na literatura em mais de uma oportunidade em temperaturas em torno de 300 °C e com a pressão de platô em torno de 0,1 atm. Também foi relatado que houve apenas a transição direta para o dihidreto, com estrutura CFC ou TCC. Comparando esses resultados relatados com o diagrama PCT calculado, figura 5.14, é possível observar que o modelo previu que há um platô entre as fases $\alpha - \delta$, porém, a pressão do platô possui uma diferença de duas ordens de grandeza, o que pode estar ligado a histerese, já que não há medida de dessorção, porém, é um ponto que é preciso ser futuramente investigado para confirmar a capacidade de previsão do modelo.

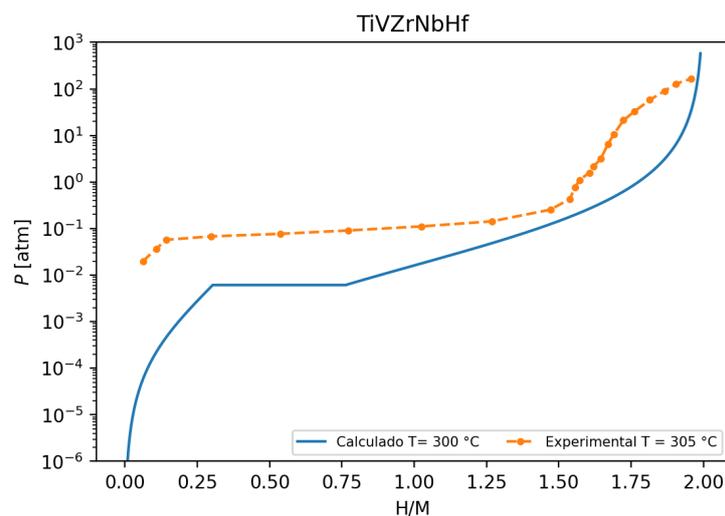


Figura 5.14: Comparação entre os diagramas PCI experimentais e os calculados para a liga TiVZrNbHf. Dados experimentais tirados de [13].

Outra liga desse sistema que teve seu PCI reportado foi TiV_{0.5}ZrNbHf na temperatura de 308 °C, exibindo apenas o platô $\alpha - \delta$ com pressão de platô por volta de 0,1 atm [13]. Enquanto isso no gráfico obtido pelo modelo, como podemos ver na figura 5.15, foi obtido apenas um platô entre as fases $\alpha - \delta$, mas a pressão do platô difere de 3 ordens de grandeza.

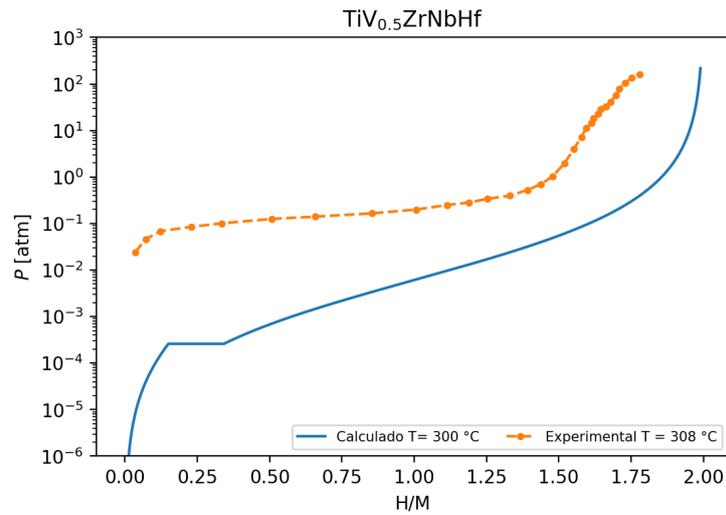
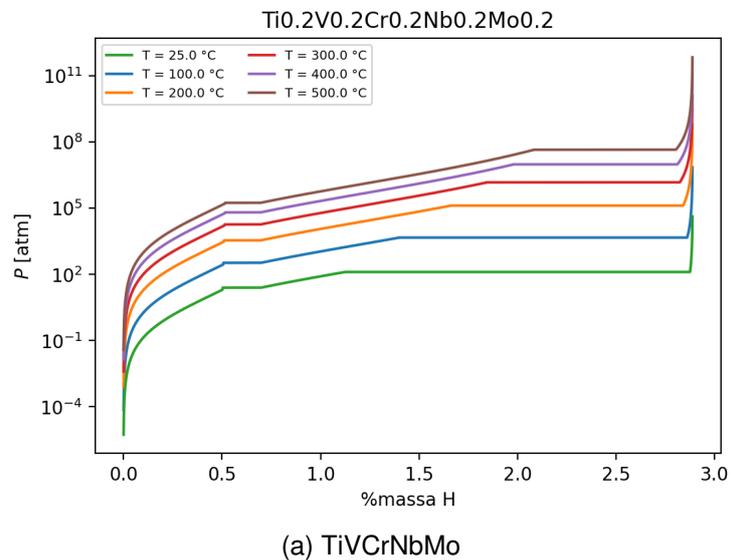
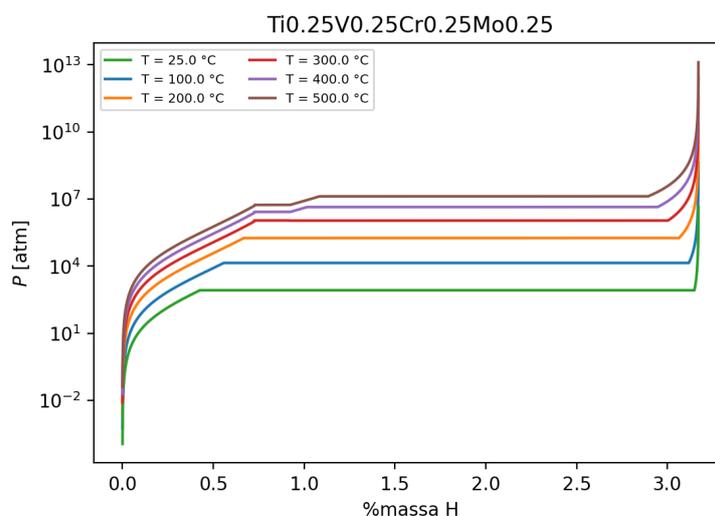


Figura 5.15: Comparação entre os diagramas PCI experimentais e os calculados para a liga $\text{TiV}_{0.5}\text{ZrNbHf}$. Dados experimentais tirados de [13].

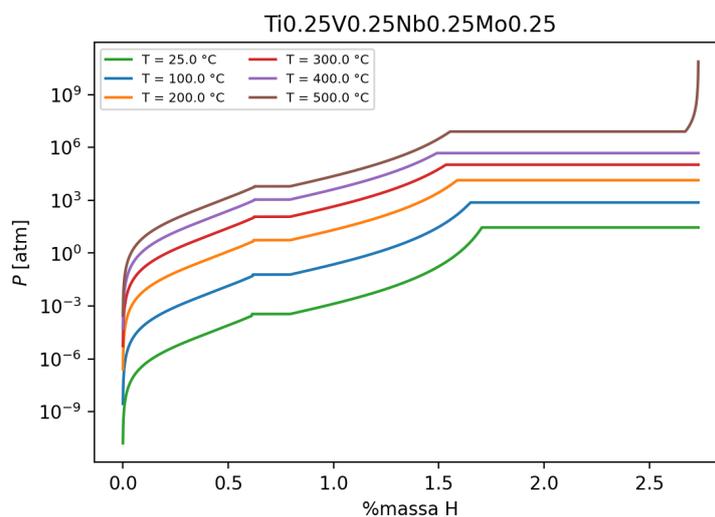
5.3.4 Sistema TiVCrNbMo

O sistema TiVCrNbMo possui apenas três ligas relacionadas e nenhuma delas possui um gráfico PCI experimental. Os diagramas PCT da figura 5.16 apresentam o resultado dos cálculos para essas ligas.





(b) TiVCrMo



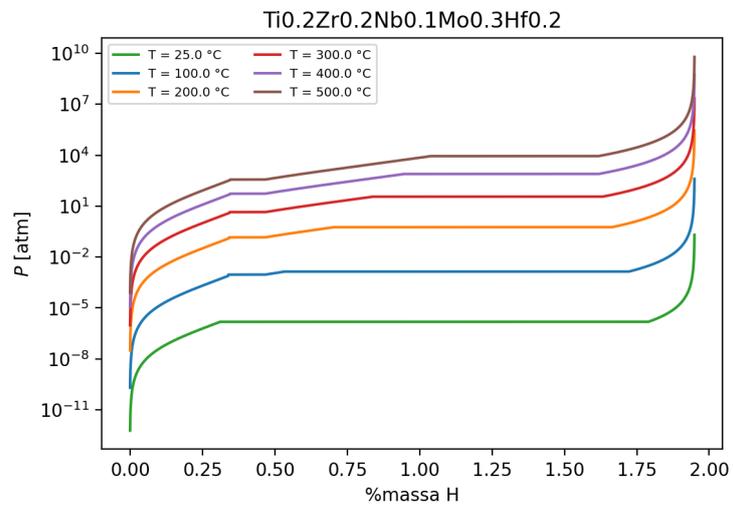
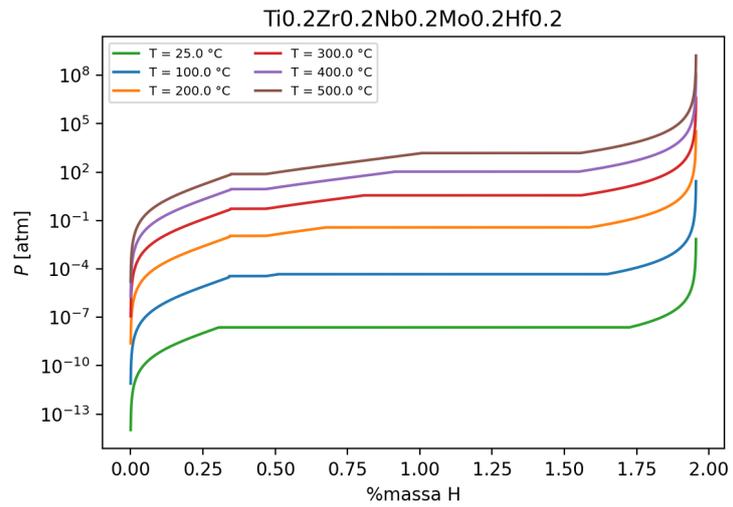
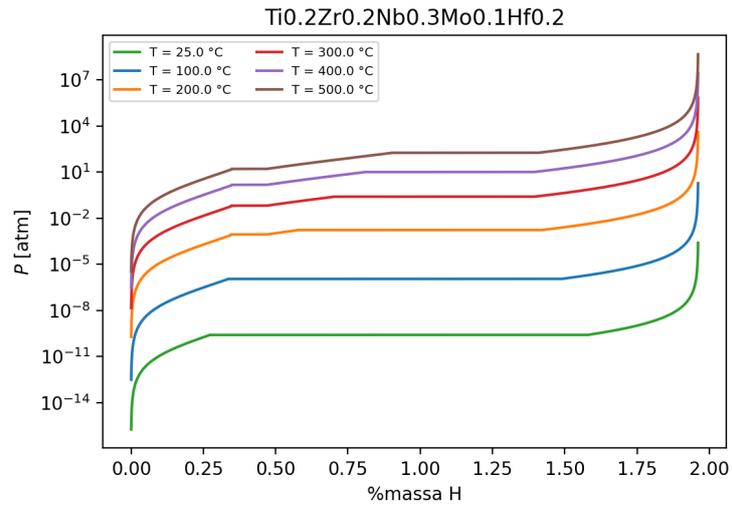
(c) TiVNbMo

Figura 5.16: PCTs de ligas reportadas do sistema TiVCrNbMo obtidos através do código desenvolvido.

As ligas desse sistema apresentam altas pressões de platô, independente dele ser entre as fases $\alpha - \delta$ ou $\beta - \delta$. Dentre as três ligas, a liga TiVCrMo se destaca por apresentar um platô mais extenso, ou seja, uma capacidade reversível maior, porém, a alta pressão do platô à temperatura ambiente é proibitiva já que está na ordem de 800 atm.

5.3.5 Sistema TiZrMoNbHf

Na figura 5.17 é possível ver os diagramas PCT calculados para as ligas relatadas do sistema TiZrMoNbHf, que não houve PCI experimental relatado.



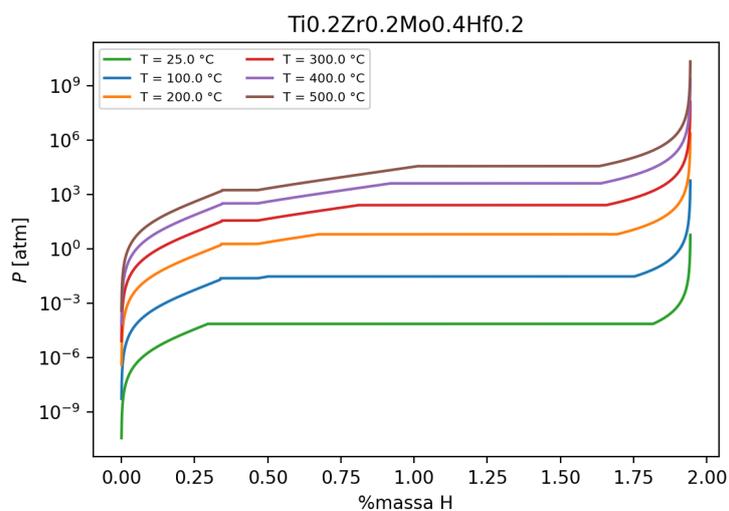
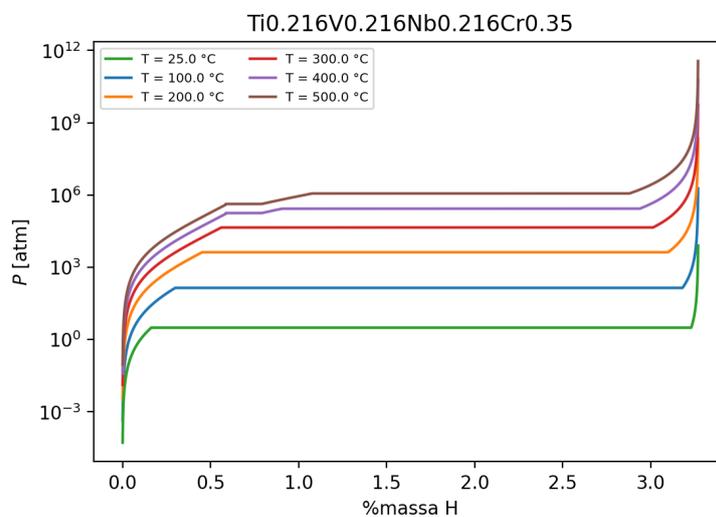
(g) TiZrMo_2Hf

Figura 5.17: Diagramas PCT de ligas reportadas do sistema TiZrMoNbHf obtidos através do código desenvolvido.

As ligas do sistema apresentaram uma transição única para a temperatura de 25 °C e dois platôs surgem com o aumento da temperatura, além disso, há o aumento da pressão dos platôs com o aumento da razão $[\text{Mo}]/[\text{M}]$ na composição das ligas do sistema, sendo então, a liga TiZrMo_2Hf a que apresentou maiores pressões de platô.

5.3.6 Sistema TiVNbCr

O sistema TiVNbCr teve quatro ligas reportadas, duas delas por mais de um autor, e em todas tiveram o PCI experimental reportado.

(a) $(\text{TiVNb})_{65}\text{Cr}_{35}$

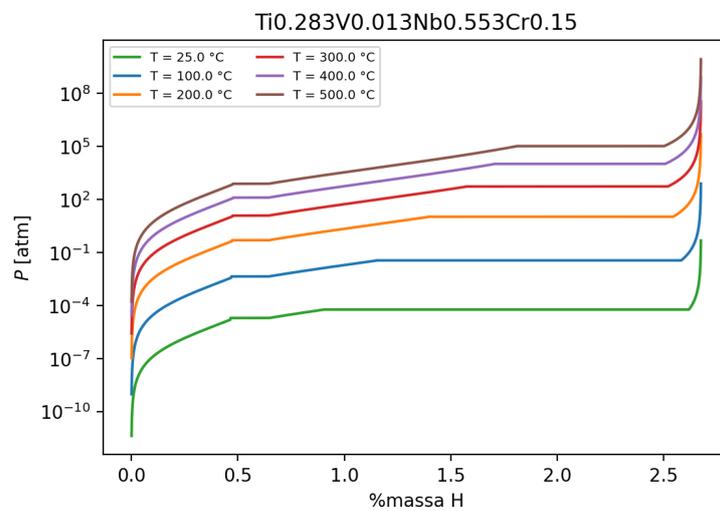
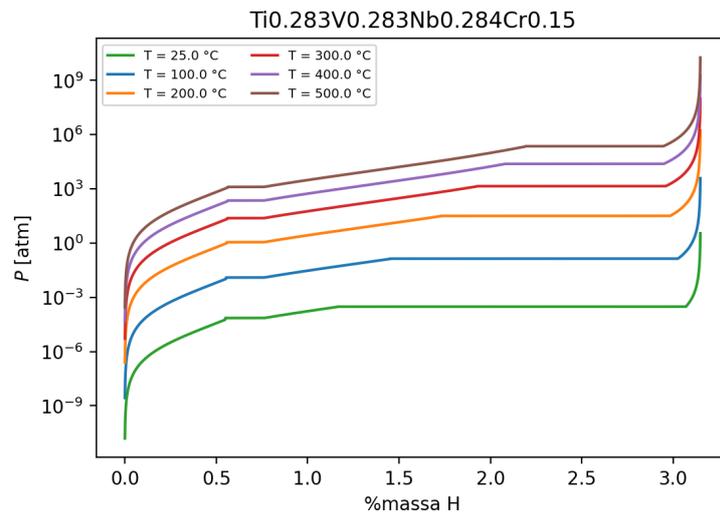
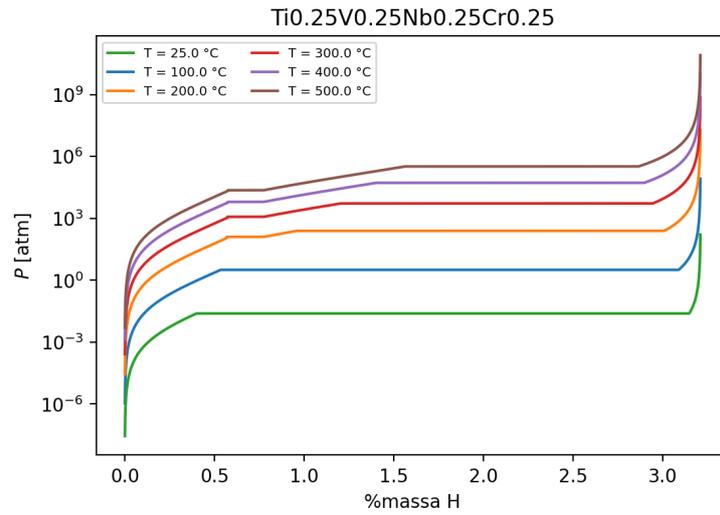
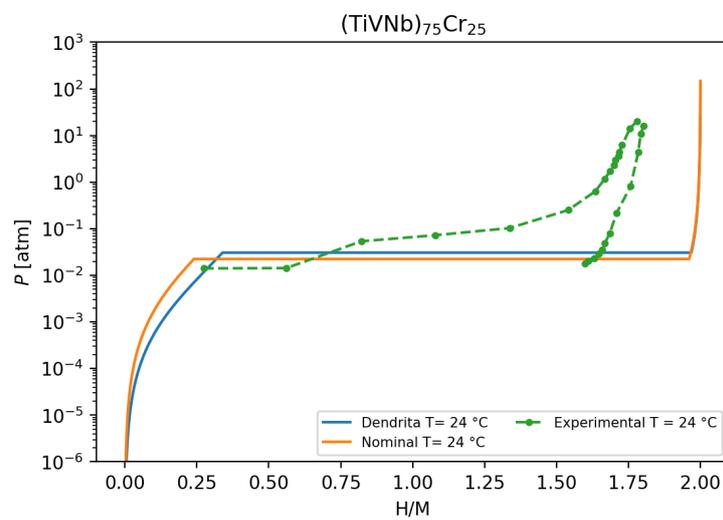
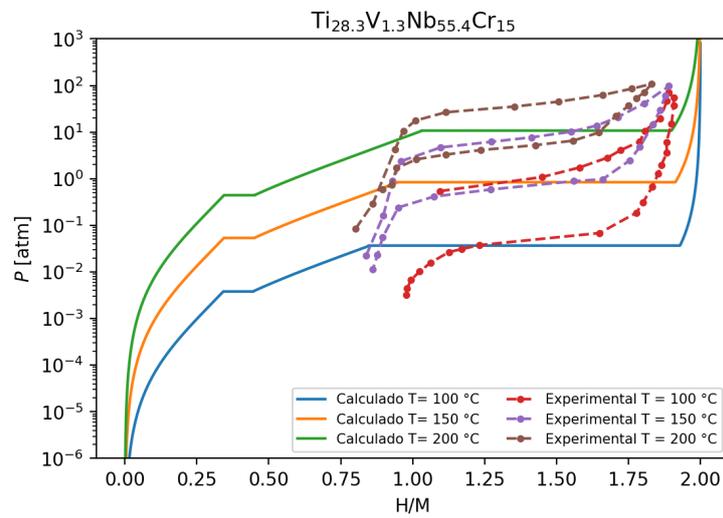
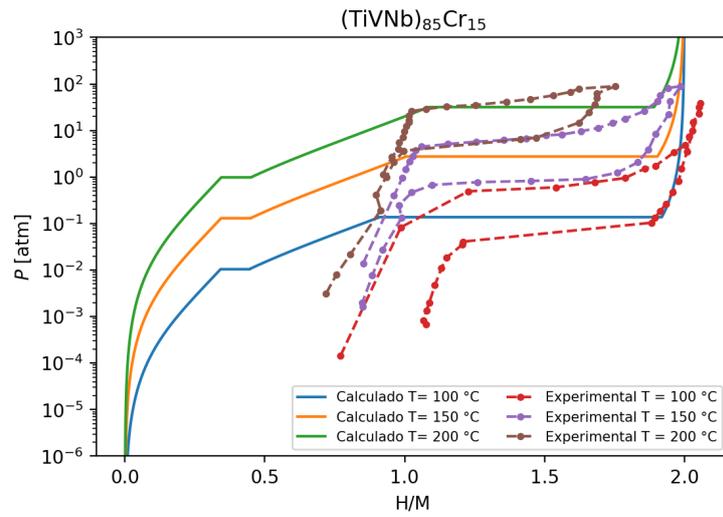


Figura 5.18: Diagramas PCT de ligas reportadas do sistema TiVNbCr obtidos através do código desenvolvido.

A figura 5.18 apresenta os diagramas PCT calculados para as ligas reportadas. Observando os gráficos da figura 5.18, podemos ver que há um comportamento de mudança na quantidade de platôs conforme a razão $[Cr]/[M]$ diminui. Além de que para cada liga há um comportamento de mudança com o aumento da temperatura.

A liga $(TiVNb)_{85}Cr_{15}$ teve o diagrama PCT reportado em [12]. Na figura 5.19a é apresentado a comparação entre o diagrama obtido para a composição nominal da liga com o diagrama PCT experimental. A liga $Ti_{28.3}V_{1.3}Nb_{55.4}Cr_{15}$ foi reportada em [5] e a comparação entre o diagrama PCT calculado com os dados experimentais é apresentada na figura 5.19b. Nessas comparações é possível ver que o modelo descreve bem as pressões de platô e a quantidade de platôs para as ligas em todas as temperaturas.

Já as ligas $(TiVNb)_{75}Cr_{25}$ e $(TiVNb)_{65}Cr_{35}$ tiveram o PCI reportado em [11]. Ambas as ligas apresentaram uma microestrutura majoritariamente CCC, porém, com uma segunda fase em pequena quantidade rica em cromo. Desta forma, foram calculados os PCI considerando tanto a composição nominal da liga como a composição medida por EDS no interior da dendrita da fase CCC, que foi considerada como sendo a composição “real” da fase CCC. Essas comparações estão presentes nas figuras 5.19c e 5.19d. Para a liga $(TiVNb)_{75}Cr_{25}$, apesar de não apresentar dois platôs, ambas composições descrevem bem a pressão do platô experimental das fases $\beta - \delta$. Enquanto para a liga $(TiVNb)_{65}Cr_{35}$, o valor nominal apresenta a mesma ordem de grandeza, mas não se ajusta bem à histerese. Já a pressão de equilíbrio calculada com a composição do interior da dendrita da fase CCC se ajustou melhor ao PCI experimental. Essa comparação mostra a importância de se reportar não apenas os diagramas PCT das ligas como também o estudo de suas microestruturas. Muitos artigos de ligas multi-componente que trazem dados experimentais de PCI ou diagramas PCT omitem a caracterização estrutural. Isto dificulta a comparação com o modelo proposto, visto que a segregação e formação de segundas fases, mesmo que em pequena quantidade, pode levar a alteração da composição química da fase CCC que é a responsável pela armazenagem de hidrogênio e que é calculada pelo modelo.



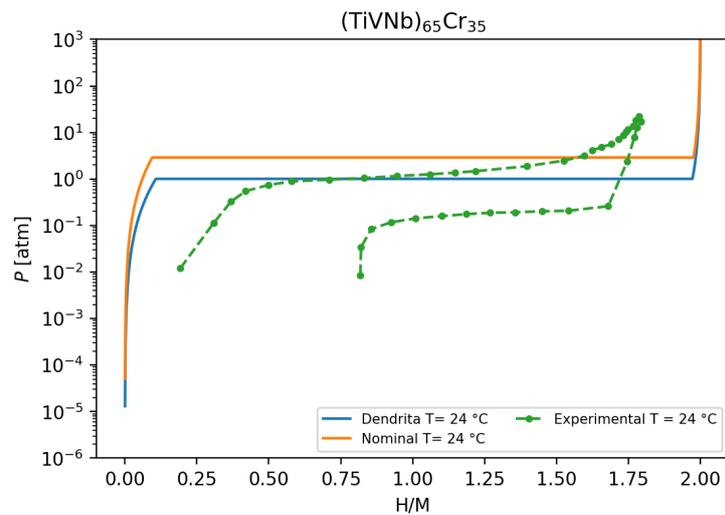
(d) $(\text{TiVNb})_{65}\text{Cr}_{35}$

Figura 5.19: Comparação entre PCIs calculados com os experimentais reportados. Dados experimentais retirados de [5, 11, 12]

5.3.7 Análise global dos sistemas

Para realizar uma análise global das ligas reportadas foi feito o mapa de pressão do platô em função da capacidade reversível para 2 temperaturas. A capacidade reversível foi considerada como a extensão do platô em %massa H. Para as ligas com equilíbrio entre as fases $\alpha - \delta$ foi utilizado a extensão do único platô, enquanto para as ligas com uma transição intermediária foi utilizado o segundo platô, que ocorre no equilíbrio entre as fases $\beta - \delta$. Por fim, foi utilizado o platô $\alpha - \beta$ quando a fase possuía apenas este equilíbrio. O mapa pode ser visto na figura 5.20.

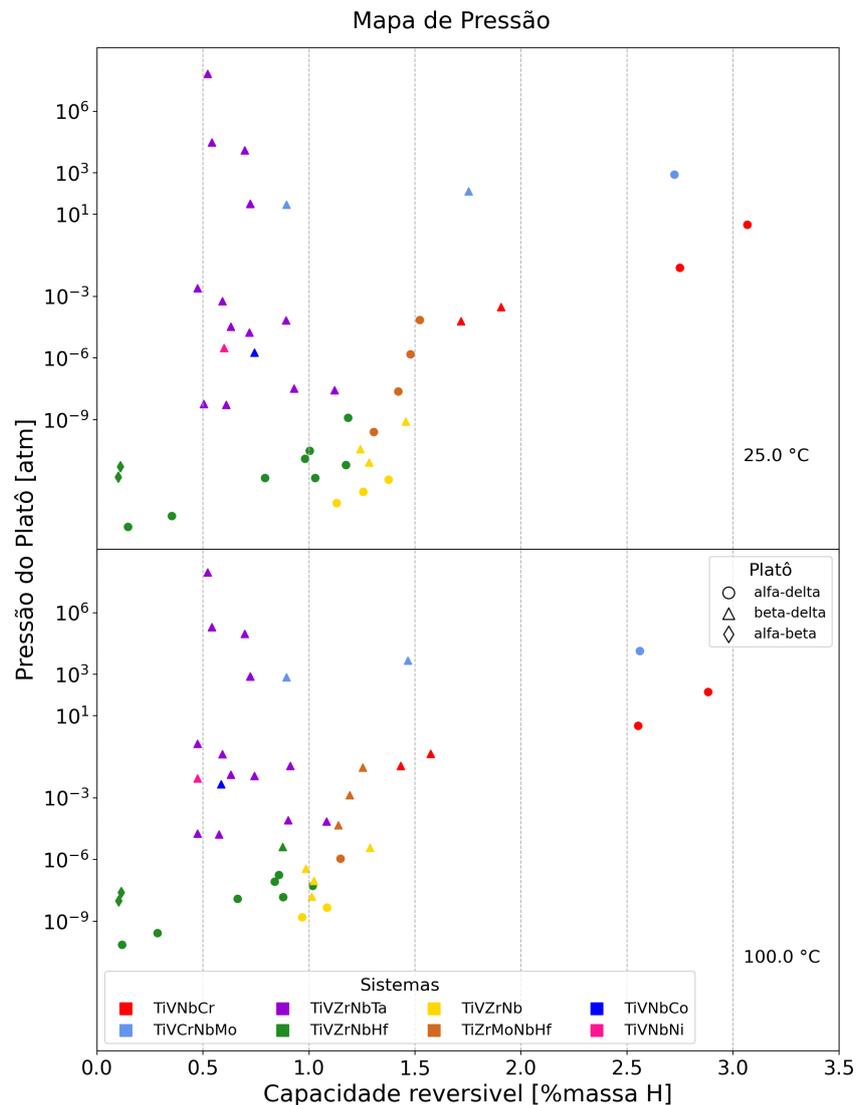


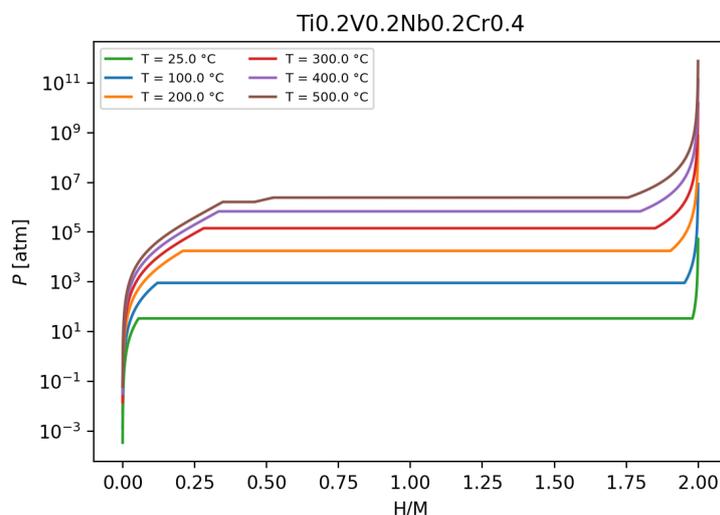
Figura 5.20: Mapa da Pressão do platô em função da capacidade reversível em %massa H, destacando os sistemas e as transições de fase.

É possível ver nos gráficos presentes na figura 5.20 que as ligas que possuem TiVNbCr em sua composição em maioria possuem mais de 1,5 %massa H, apresentando assim, as maiores capacidades reversíveis para o armazenamento de hidrogênio, sendo um possível sistema a ser aprofundado. Além disso, as ligas do sistema TiVNbCr apresentam pressões de equilíbrio entre 0,1 e 100 atm entre as temperaturas de 25 °C e 100 °C, indicando que estas ligas podem ser promissoras para aplicação em baixas temperaturas. Os demais sistemas estão situados em maioria com uma capacidade abaixo de 1,5 %massa H. É notável que a adição de Mo no sistema TiVNbCr aumentou a pressão dos platôs, então,

apesar de duas ligas desse sistema apresentarem uma capacidade reversível semelhante às ligas do sistema TiVNbCr, o aumento da pressão de platô pode tornar sua aplicação proibitiva. Pode-se notar que as ligas dos sistemas TiVZrNb e TiVZrNbHf apresentam pressões de equilíbrio extremamente baixas em 25 °C e 100 °C, indicando uma grande estabilidade do hidreto em baixas temperaturas e que estas ligas devem ser investigadas para aplicações de armazenagem em temperaturas elevadas. Feita a análise acima, o sistema $(\text{TiVNb})_{1-x}\text{Cr}_x$ foi selecionado para um estudo de caso por se mostrar promissor para o armazenamento de hidrogênio em temperaturas próximas da temperatura ambiente e por apresentar uma boa concordância com os dados experimentais.

5.4 Estudo sistemático do teor de Cr na liga (TiVNb)

O sistema $(\text{TiVNb})_{1-x}\text{Cr}_x$ se mostrou promissor por ter as melhores capacidades reversíveis calculadas pelo modelo dos sistemas reportados até o momento. Além da capacidade reversível, também apresentaram uma pressão de platô acessível em baixas temperaturas. Com o intuito de entender melhor a influência do teor de Cr nesse sistema, diagramas PCT com teores de Cr diferentes dos reportados foram calculados para o estudo de caso. Na figura 5.21 estão os diagramas PCT calculados para essas ligas do sistema.



(a) $(\text{TiVNb})_{60}\text{Cr}_{40}$

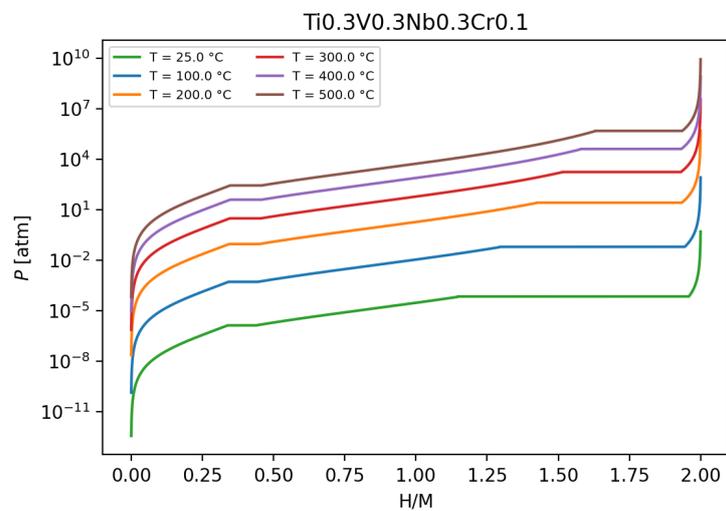
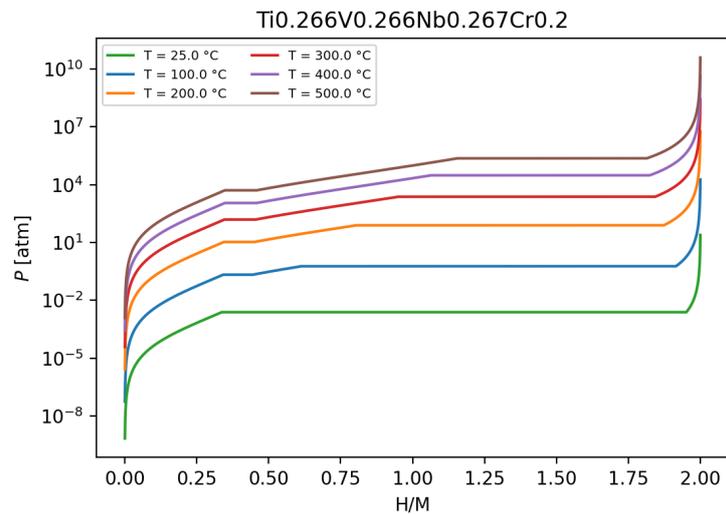
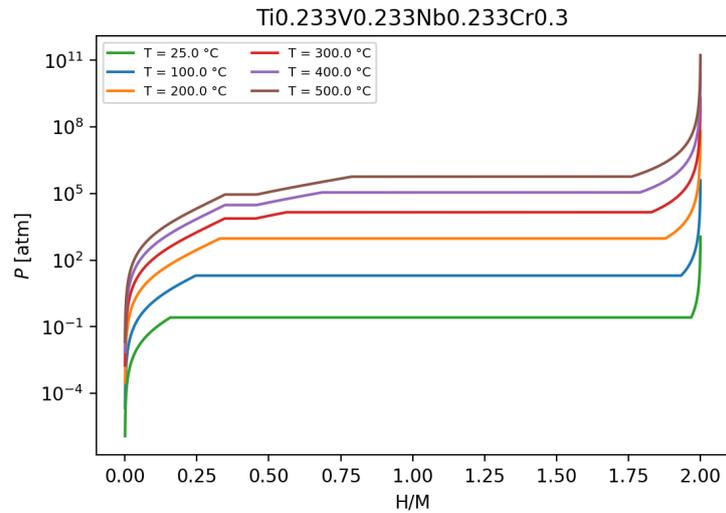


Figura 5.21: Diagramas PCT calculados para teores diferentes de cromo no sistema $(\text{TiVNb})_{1-x}\text{Cr}_x$ não reportados.

Os diagramas PCT da figura 5.21 em conjunto com os diagramas da figura 5.18 reforçam a mudança do comportamento dos platôs com a diminuição do teor de Cr, onde quanto menor a razão $[Cr]/[M]$ menor a temperatura em que ocorre dois platôs de equilíbrio.

Além disso, foram calculados os diagramas PCT do sistema $(TiVNb)_{1-x}Cr_x$ com $x = 0.1, 0.15, 0.2, 0.25, 0.3, 0.35, 0.4$, com esse conjunto de dados é possível analisar melhor como a pressão de platô e a capacidade reversível varia com o teor de Cr. Na figura 5.22, é ilustrado a comparação entre as pressões de platô em função do teor de Cr para temperaturas entre 25 °C e 500 °C.

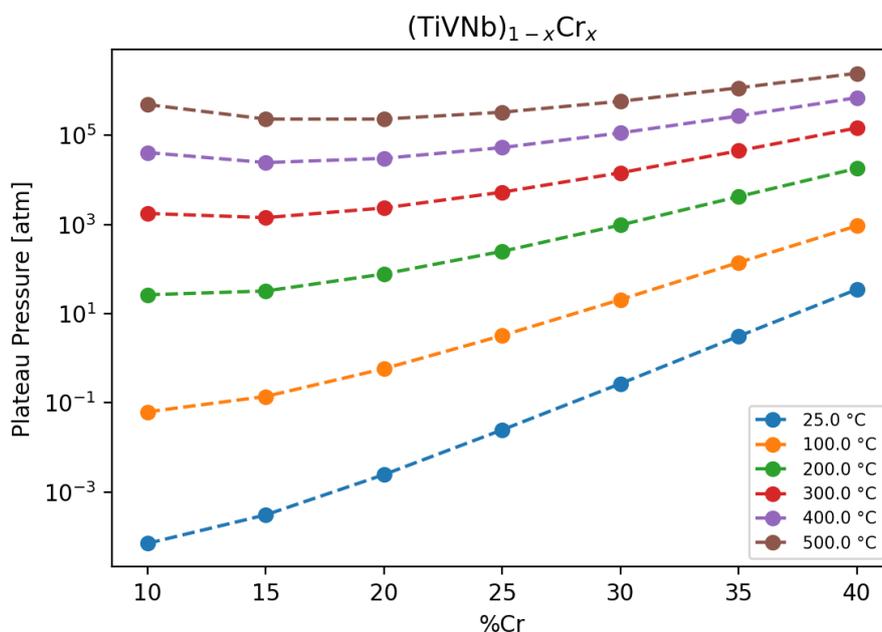


Figura 5.22: Comparação da pressão do platô em função do teor de Cr nas ligas.

É possível ver na comparação da figura 5.22, que há dois comportamentos nas ligas do sistema $(TiVNb)_{1-x}Cr_x$ em relação a pressão de platô. O primeiro é que quanto maior o teor de Cr, maior a pressão de equilíbrio, enquanto isso, para um mesmo teor de Cr, quanto maior a temperatura, maior a pressão de platô, sendo altas pressões para temperaturas acima de 300 °C.

Enquanto isso, a figura 5.23, é ilustrado a comparação entre as capacidades reversíveis em função do teor de Cr para temperaturas entre 25 °C e 500 °C.

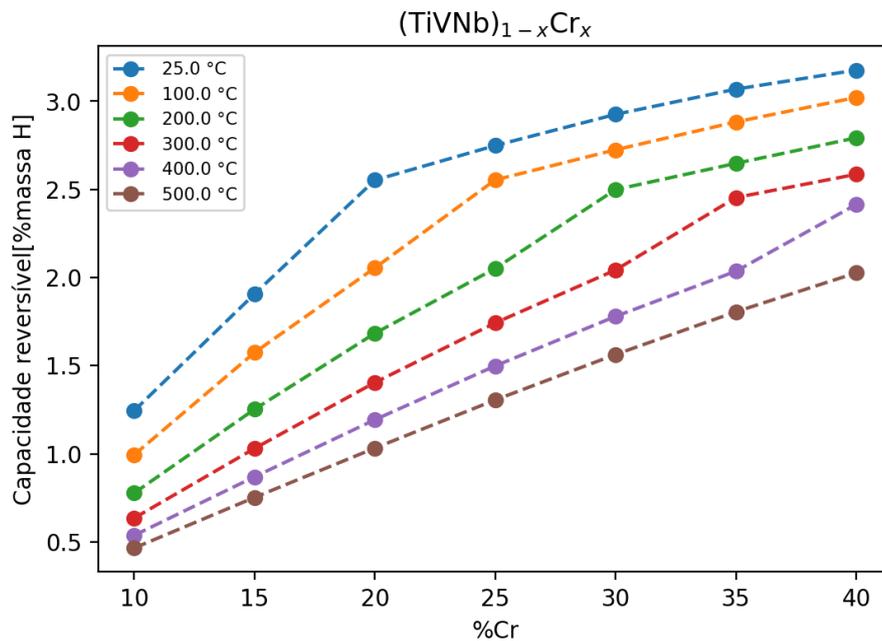


Figura 5.23: Comparação da capacidade reversível em função do teor de Cr nas ligas.

Nessa comparação, podemos ver que quanto maior a temperatura para um mesmo teor de Cr, menor a capacidade reversível. Além de que, quanto maior a quantidade de Cr na liga, maior a capacidade reversível.

Essa variação na capacidade reversível está ligada ao comportamento da variação da quantidade de platôs de equilíbrio em função do teor de Cr e da temperatura, visto que quando ocorrem dois platôs, a capacidade reversível tende a ser menor do que para uma liga que possua um platô apenas. Isso pode ser visto através de uma comparação visual entre os diagramas PCT das figuras 5.18 e 5.21.

5.5 Análise das ligas multicomponentes contendo Mg

Dada algumas particularidades no comportamento das ligas multicomponentes contendo Mg, optou-se por apresentar os resultados referentes a este tipo de liga nesta seção separada.

Analisando as ligas contendo Mg da tabela 2.2 com o modelo proposto, é possível observar que em sua maioria possuem $H_M < 0$, i.e., são previstas estáveis com estrutura CFC. Na figura 5.24 é possível ver o gráfico da variação da

energia livre de Gibbs para as três fases, onde é possível ver que a fase δ (CFC) possui a menor energia livre.

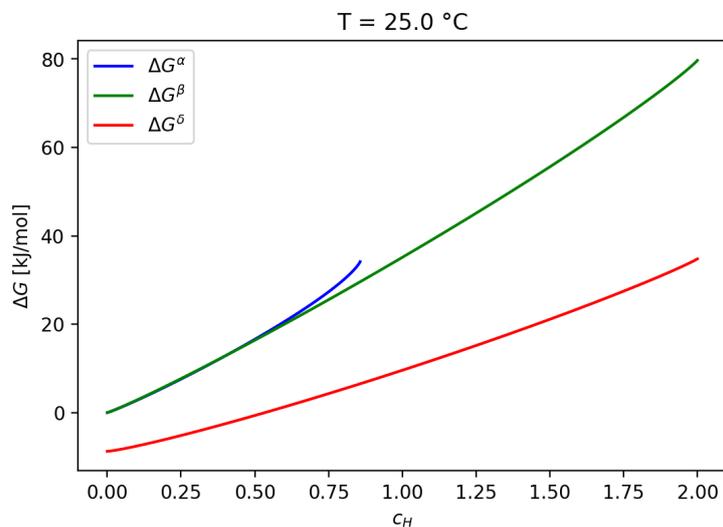
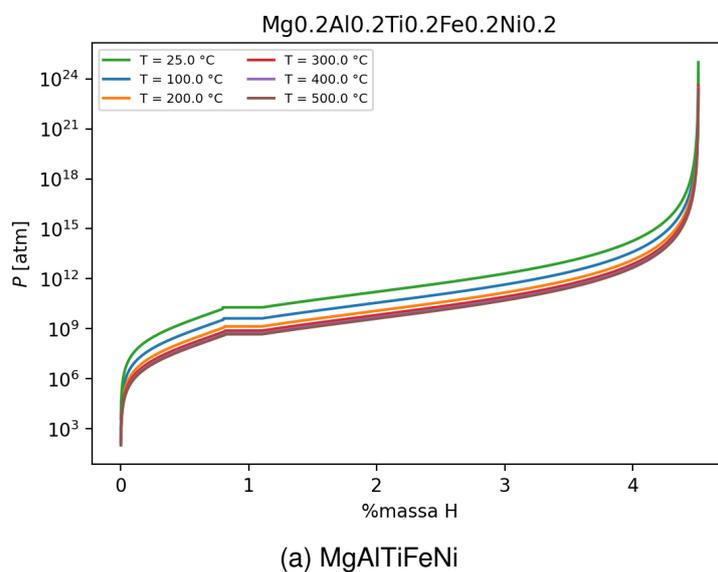
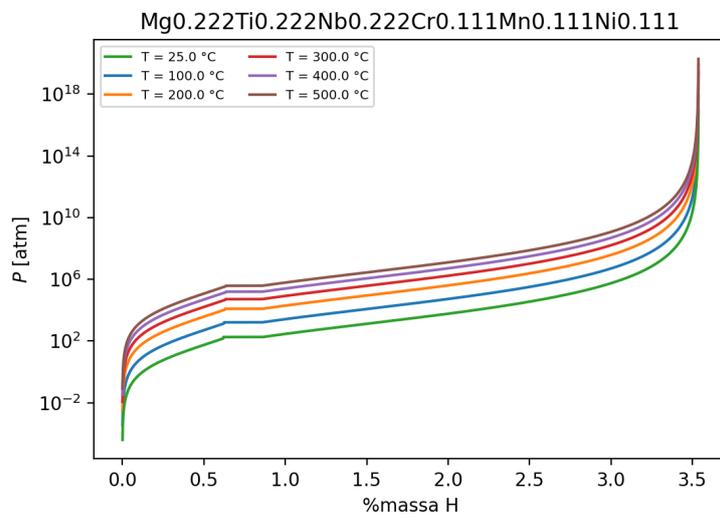
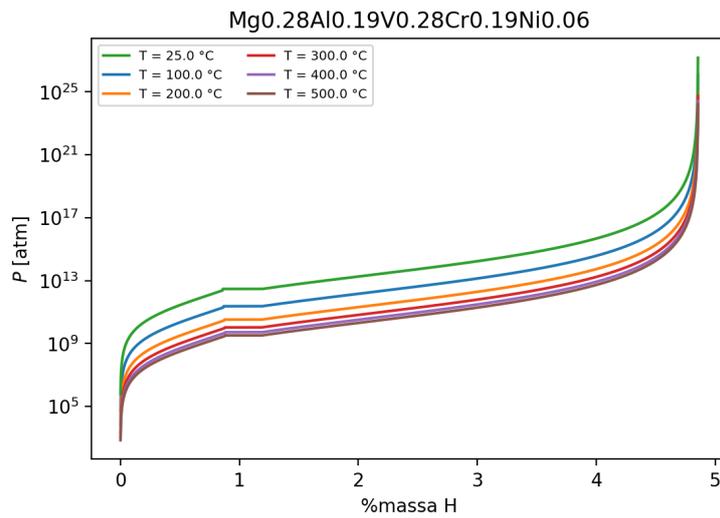
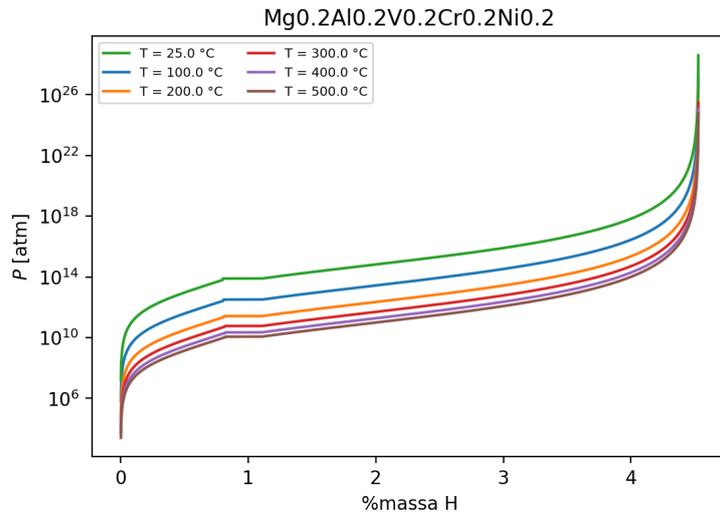


Figura 5.24: Gráfico de variação da energia livre de Gibbs da liga $Mg_{0.28}Al_{0.19}V_{0.28}Cr_{0.19}Ni_{0.06}$, sendo exemplo para as ligas com Mg que apresentam $H_M < 0$.

Como o modelo prevê que a liga terá estrutura CFC, não há a possibilidade de encontrar um platô com a fase δ , porém, essas ligas foram reportadas com estrutura CCC. Portanto, o código desenvolvido ignora a fase δ e busca o equilíbrio entre as fases α e β . No caso de não encontrar um equilíbrio é apresentado apenas o gráfico PCI da fase α . Na figura 5.25 estão ilustrados os diagramas PCT para maioria das ligas onde $H_M < 0$ e que não possuem PCI reportado.





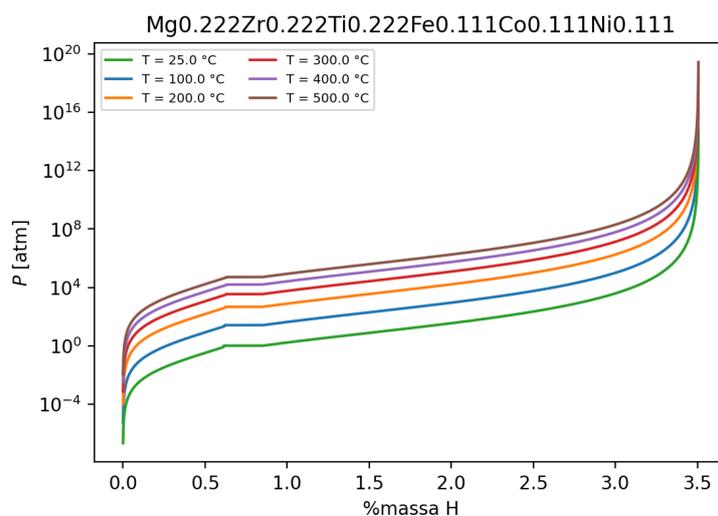
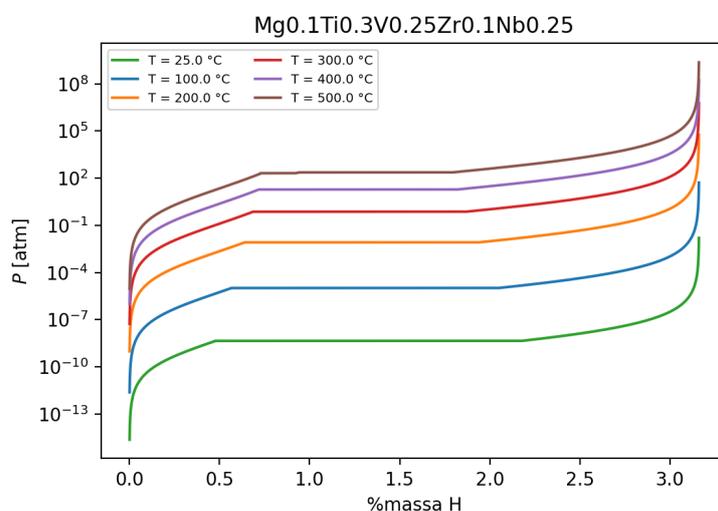
(e) $\text{MgZrTiFe}_{0.5}\text{Co}_{0.5}\text{Ni}_{0.5}$

Figura 5.25: Diagramas PCT calculados para as ligas com Mg que não tiveram PCI reportado.

Em todos os gráficos da figura 5.25 há um equilíbrio entre as fases α e β , e em sua maioria, esses platôs ocorreram em pressões de equilíbrio elevadas, com exceção das ligas $\text{MgZrTiFe}_{0.5}\text{Co}_{0.5}\text{Ni}_{0.5}$ e $\text{MgTiNbCr}_{0.5}\text{Mn}_{0.5}\text{Ni}_{0.5}$ que apresenta pressões de platô em 25 °C de 1,04 e 178 atm.

Os diagramas PCT para as duas ligas com PCI reportado estão presentes na figura 5.26.

(a) $\text{Mg}_{0.1}\text{Ti}_{0.3}\text{V}_{0.25}\text{Zr}_{0.1}\text{Nb}_{0.25}$

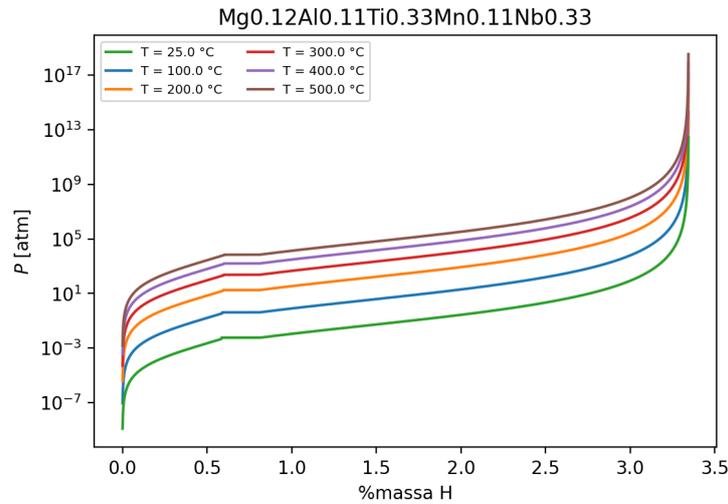
(b) $Mg_{0.12}Al_{0.11}Ti_{0.33}Mn_{0.11}Nb_{0.33}$

Figura 5.26: Diagramas PCT calculados para as ligas com Mg que tiveram PCI reportado.

Entre as ligas com PCI reportado temos um exemplo onde foi encontrado $H_M > 0$, a liga $Mg_{0.1}Ti_{0.3}V_{0.25}Zr_{0.1}Nb_{0.25}$. Nela ocorre o equilíbrio entre as fases α e δ . Enquanto na liga $Mg_{0.12}Al_{0.11}Ti_{0.33}Mn_{0.11}Nb_{0.33}$ há o equilíbrio entre as fases α e β , pois para essa liga $H_M < 0$. Como exemplo de liga com Mg e $H_M > 0$, na figura 5.27 é apresentado o gráfico da variação da energia livre de Gibbs da liga $Mg_{0.1}Ti_{0.3}V_{0.25}Zr_{0.1}Nb_{0.25}$.

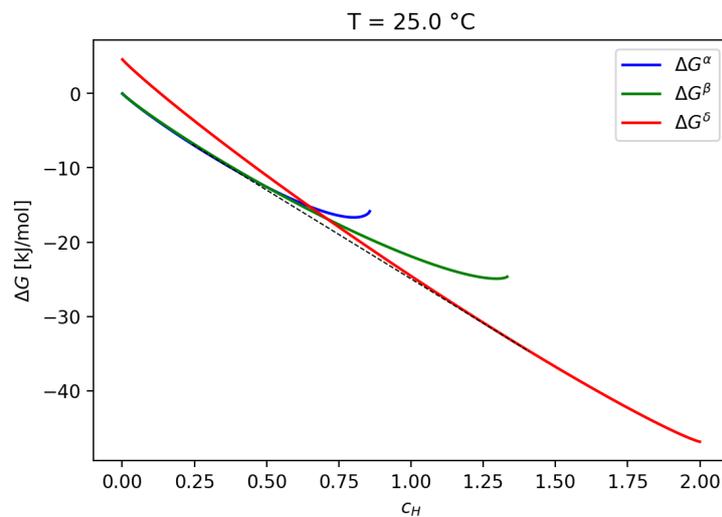


Figura 5.27: Gráfico de variação da energia livre de Gibbs da liga $Mg_{0.1}Ti_{0.3}V_{0.25}Zr_{0.1}Nb_{0.25}$, sendo exemplo para as ligas com Mg que apresentam $H_M > 0$.

A liga $\text{Mg}_{0.1}\text{Ti}_{0.3}\text{V}_{0.25}\text{Zr}_{0.1}\text{Nb}_{0.25}$ teve o PCI reportado com uma transição direta, i.e., entre as fases α e δ para a temperatura de 25 °C [24]. O platô acontece em baixas pressões, comportamento que também ocorre com o PCI obtido através do modelo. A comparação entre os gráficos PCI calculado e experimental estão presentes na figura 5.28, para essa liga os parâmetros da entropia configuracional da fase β tiveram que ser alterados para $\theta_\beta = 4$ e $r = 3$. Percebe-se que a pressão do platô experimental foi limitado pelo limite de detecção do equipamento (10^{-2} atm), indicando que a pressão baixa do platô em temperatura ambiente calculado pode estar correto. Interessantemente, o tamanho do platô foi previsto corretamente pelo modelo. Porém, o modelo previu que a concentração de hidrogênio na fase delta aumentaria com o aumento da pressão, o que não ocorreu experimentalmente.

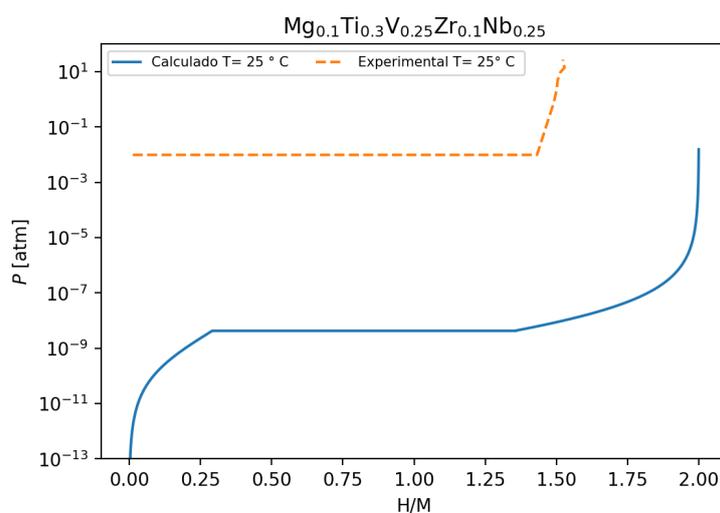


Figura 5.28: Comparação entre os gráficos PCI experimental e calculado para a liga $\text{Mg}_{0.1}\text{Ti}_{0.3}\text{V}_{0.25}\text{Zr}_{0.1}\text{Nb}_{0.25}$. Dados experimentais tirados de [24]

A liga $\text{Mg}_{0.12}\text{Al}_{0.11}\text{Ti}_{0.33}\text{Mn}_{0.11}\text{Nb}_{0.33}$ foi reportada sem apresentar uma segunda fase na temperatura de 275 °C [26]. Experimentalmente foi observado apenas a presença do hidreto intermediário de estrutura CCC distorcida, enquanto isso o modelo termodinâmico descreve a liga com $H_M < 0$, logo, de estrutura CFC. Portanto, foi necessário buscar o equilíbrio entre as fases α e β . A figura 5.29 apresenta a comparação do gráfico PCI calculado com o obtido experimentalmente. Neste caso, as pressões de equilíbrio calculadas foram ordens de gran-

deza superior do que as medidas experimentalmente, indicando que ainda há a necessidade de se melhorar a descrição termodinâmica das ligas CCC multi-componentes contendo Mg.

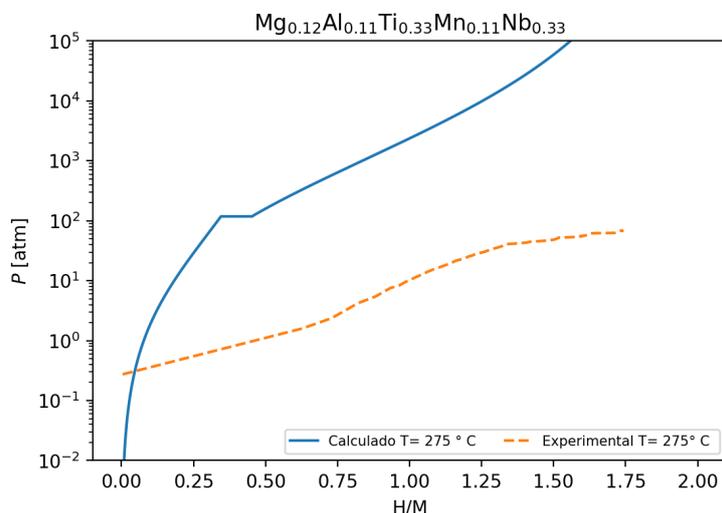


Figura 5.29: Comparação entre os gráficos PCI experimental e calculado para a liga $Mg_{0.12}Al_{0.11}Ti_{0.33}Mn_{0.11}Nb_{0.33}$. Dados experimentais tirados de [26]

5.6 Considerações gerais sobre o modelo

Nas análises realizadas para as ligas é possível ver que o modelo descreve melhor alguns sistemas do que outros. As discrepâncias vistas podem ter sua fonte tanto do modelo como nas energias que são utilizadas para os cálculos da entalpia.

Como as entalpias parciais molares de hidrogênio e energias de ligação foram obtidas através de simulações de DFT, os cálculos do modelo terão uma incerteza atrelada já que esses cálculos são realizados em 0 Kelvin. Essa é uma desvantagem do DFT para a obtenção dessas energias, já que as ligas que serão estudadas através do modelo terão sua aplicação em temperaturas acima da qual suas energias de ligação são calculadas.

Além disso, não foi considerado o caráter magnético dos elementos Co, Fe e Ni, o que deve levar a uma alteração nas energias de ligação obtidas. Outros pontos que podem ser fonte de incerteza nos valores obtidos é que não houve uma análise a respeito do caráter da ligação dos átomos de hidrogênio com os elementos da liga multicomponente e nem dos elementos da liga entre

si, e da consideração de que os átomos de hidrogênio ocupam exatamente a posição central dos interstícios. Também não houve uma consideração de como a presença de outros elementos afeta a energia da ligação entre os pares de elementos e dos elementos com hidrogênio. Outro ponto a se considerar, é que para as fases α e β , foram considerados os valores de entalpia de dissolução em diluição infinita dos elementos puros para estimar a entalpia parcial molar destas fases. Estes valores podem ser também determinados através de cálculos de DFT de supercélulas no qual estruturas com baixa concentração de hidrogênio são calculadas. Contudo, para se calcular a energia total de soluções sólidas diluídas, o número de átomos na supercélula aumenta podendo chegar a 100 ou 1000, para simular uma condição de solução sólida aleatória, aumentando drasticamente o tempo de cálculo e o poder computacional demandado. Por fim, vale lembrar que os resultados do modelo são fortemente influenciados pelos parâmetros θ e r do modelo de entropia configuracional, que ainda não é claramente definido para os hidretos intermediários (fase β) dos diferentes sistemas. Todos estes pontos deverão ser objeto de estudos futuros do nosso grupo de pesquisa e seus resultados poderão ser facilmente incorporados no código desenvolvido neste trabalho.

6 CONCLUSÕES

O projeto desenvolvido abre caminho para que o modelo termodinâmico proposto por Zepon et al. para o cálculo de diagramas PCT possa ser uma ferramenta no design de ligas multicomponente de estrutura CCC para armazenamento de hidrogênio. Para isso, foram adicionados ao modelo os elementos Mg, Al, Sc, Mn, Fe, Co, Cu, Zn, Mo e Pd. Além da implementação do modelo em um programa de código aberto desenvolvido na linguagem Python. Para avaliar tanto o modelo quanto o código desenvolvido, foram analisadas as ligas reportadas na literatura, onde foi possível concluir:

- Os elementos Mg, Al, Sc, Mn, Fe, Co, Cu, Zn, Mo e Pd foram adicionados com sucesso ao modelo;
- O modelo foi implementado em um software de código aberto, que realiza os cálculos e a obtenção dos gráficos das grandezas termodinâmicas e de diagramas PCT das ligas multicomponentes desejadas pelo usuário;
- O sistema $(\text{TiVNb})_{1-x}\text{Cr}_x$ foi o melhor descrito pelo modelo, que descreveu bem a pressão de platô de equilíbrio com a fase δ das ligas pertencentes a esse sistema;
- As ligas do sistema $(\text{TiVNb})_{1-x}\text{Cr}_x$ se mostraram promissoras para o armazenamento de hidrogênio, apresentando boas propriedades de capacidade reversível e pressão de platô;
- As ligas multicomponente contendo Mg apresentaram discrepância entre a estrutura prevista e as obtidas do modelo;
- O estudo estrutural de ligas reportadas é fundamental para a avaliação e comparação com resultados calculados pelo modelo;

7 SUGESTÕES PARA FUTUROS TRABALHOS

- Como foi visto algumas discrepâncias em alguns resultados do modelo quando comparado com resultados experimentais, é necessário investigar pontos para melhoria no modelo indicados na seção 5.6.
- Como o intuito do modelo é prever os diagramas PCT para o design de novas ligas, é necessário desenvolver um método para prever o tipo de ocupação intersticial do hidrogênio, logo, os parâmetros da entropia configuracional, em função da composição química.
- Considerando a boa descrição do sistema TiVNbCr pelo modelo e a adição de Mo a esse sistema apresentar capacidades reversíveis similares, é interessante investigar experimentalmente as propriedades que surgem com a adição de outros elementos a esse sistema.

8 REFERÊNCIAS BIBLIOGRÁFICAS

- [1] O sistema interligado nacional. c2021. URL: <http://www.ons.org.br/paginas/sobre-o-sin/o-que-e-o-sin>; acessado em 14/02/2021.
- [2] Lai, Q., Sun, Y., Wang, T., Modi, P., Cazorla, C., Demirci, U.B., et al. How to design hydrogen storage materials? fundamentals, synthesis, and storage tanks. *Advanced Sustainable Systems* 2019;3(9):1900043.
- [3] Dantzer, P. Metal-hydride technology: a critical review. *Hydrogen in Metals III* 1997;;279–340.
- [4] Chen, P., Zhu, M.. Recent progress in hydrogen storage. *Materials today* 2008;11(12):36–43.
- [5] Zepon, G., Silva, B.H., Zlotea, C., Botta, W.J., Champion, Y.. Thermodynamic modelling of hydrogen-multicomponent alloy systems: Calculating pressure-composition-temperature diagrams. *Acta Materialia* 2021;215:117070. doi:<https://doi.org/10.1016/j.actamat.2021.117070>.
- [6] Garcés, J.. The configurational entropy of mixing of interstitials solid solutions. *Applied Physics Letters* 2010;96(16):161904.
- [7] Felderhoff, M., Marques, F., Balcerzak, M., Winkelmann, F., Zepon, G.. Review and outlook on high-entropy alloys for hydrogen storage. *Energy & Environmental Science* 2021;.
- [8] Sahlberg, M., Karlsson, D., Zlotea, C., Jansson, U.. Superior hydrogen storage in high entropy alloys. *Scientific reports* 2016;6(1):1–6.
- [9] Zlotea, C., Sow, M., Ek, G., Couzinié, J.P., Perrière, L., Guillot, I., et al. Hydrogen sorption in tizrn bhfta high entropy alloy. *Journal of Alloys and Compounds* 2019;775:667–674.
- [10] Karlsson, D., Ek, G., Cedervall, J., Zlotea, C., Møller, K.T., Hansen, T.C., et al. Structure and hydrogenation properties of a hfnbtivzr high-entropy alloy. *Inorganic chemistry* 2018;57(4):2103–2110.
- [11] Strozi, R.B., Leiva, D.R., Zepon, G., Botta, W.J., Huot, J.. Effects of the chromium content in (tivnb) 100- xcrx body-centered cubic high entropy alloys

- designed for hydrogen storage applications. *Energies* 2021;14(11):3068.
- [12] Design of tivnb-(cr, ni or co) multicomponent alloys with the same valence electron concentration for hydrogen storage. *Journal of Alloys and Compounds* 2021;865:158767. doi:<https://doi.org/10.1016/j.jallcom.2021.158767>.
- [13] Ek, G., Nygård, M.M., Pavan, A.F., Montero, J., Henry, P.F., Sørby, M.H., et al. Elucidating the effects of the composition on hydrogen sorption in tivzrnbhf-based high-entropy alloys. *Inorganic Chemistry* 2020;60(2):1124–1132.
- [14] Kuncce, I., Polanski, M., Bystrzycki, J.. Microstructure and hydrogen storage properties of a tizrnbmov high entropy alloy synthesized using laser engineered net shaping (lens). *International Journal of Hydrogen Energy* 2014;39(18):9904–9910.
- [15] Nygård, M.M., Ek, G., Karlsson, D., Sørby, M.H., Sahlberg, M., Hauback, B.C.. Counting electrons-a new approach to tailor the hydrogen sorption properties of high-entropy alloys. *Acta Materialia* 2019;175:121–129.
- [16] Nygård, M.M., Ek, G., Karlsson, D., Sahlberg, M., Sørby, M.H., Hauback, B.C.. Hydrogen storage in high-entropy alloys with varying degree of local lattice strain. *International Journal of Hydrogen Energy* 2019;44(55):29140–29149.
- [17] Shen, H., Zhang, J., Hu, J., Zhang, J., Mao, Y., Xiao, H., et al. A novel tizrhfmonb high-entropy alloy for solar thermal energy storage. *Nanomaterials* 2019;9(2):248.
- [18] Montero, J., Ek, G., Laversenne, L., Nassif, V., Zepon, G., Sahlberg, M., et al. Hydrogen storage properties of the refractory ti–v–zr–nb–ta multi-principal element alloy. *Journal of Alloys and Compounds* 2020;835:155376.
- [19] Shen, H., Hu, J., Li, P., Huang, G., Zhang, J., Zhang, J., et al. Compositional dependence of hydrogenation performance of ti-zr-hf-mo-nb high-entropy alloys for hydrogen/tritium storage. *Journal of Materials Science & Technology* 2020;55:116–125.

- [20] Zhang, C., Song, A., Yuan, Y., Wu, Y., Zhang, P., Lu, Z., et al. Study on the hydrogen storage properties of a tizrnbt high entropy alloy. *International Journal of Hydrogen Energy* 2020;45(8):5367–5374.
- [21] Nygård, M.M., Fjellvåg, Ø.S., Sørby, M.H., Sakaki, K., Ikeda, K., Armstrong, J., et al. The average and local structure of tivcrnbdx ($x=0, 2.2, 8$) from total scattering and neutron spectroscopy. *Acta Materialia* 2021;205:116496.
- [22] Sleiman, S., Huot, J.. Effect of particle size, pressure and temperature on the activation process of hydrogen absorption in tivzrhfnb high entropy alloy. *Journal of Alloys and Compounds* 2021;861:158615.
- [23] Montero, J., Zlotea, C., Ek, G., Crivello, J.C., Laversenne, L., Sahlberg, M.. Tivzrn multi-principal-element alloy: synthesis optimization, structural, and hydrogen sorption properties. *Molecules* 2019;24(15):2799.
- [24] Montero, J., Ek, G., Sahlberg, M., Zlotea, C.. Improving the hydrogen cycling properties by mg addition in ti-v-zr-nb refractory high entropy alloy. *Scripta Materialia* 2021;194:113699. URL: <https://www.sciencedirect.com/science/article/pii/S1359646220308216>. doi:<https://doi.org/10.1016/j.scriptamat.2020.113699>.
- [25] Cardoso, K.R., Roche, V., Jorge Jr, A.M., Antigueira, F.J., Zepon, G., Champion, Y.. Hydrogen storage in mgaltifeni high entropy alloy. *Journal of Alloys and Compounds* 2021;858:158357. URL: <https://www.sciencedirect.com/science/article/pii/S0925838820347204>. doi:<https://doi.org/10.1016/j.jallcom.2020.158357>.
- [26] Strozi, R., Leiva, D., Huot, J., Botta, W., Zepon, G.. An approach to design single bcc mg-containing high entropy alloys for hydrogen storage applications. *International Journal of Hydrogen Energy* 2021;.
- [27] Strozi, R., Leiva, D., Huot, J., Botta, W., Zepon, G.. Synthesis and hydrogen storage behavior of mg–v–al–cr–ni high entropy alloys. *International Journal of Hydrogen Energy* 2021;46(2):2351–2361.
- [28] Marques, F., Pinto, H.C., Figueroa, S.J.A., Winkelmann, F., Fel-

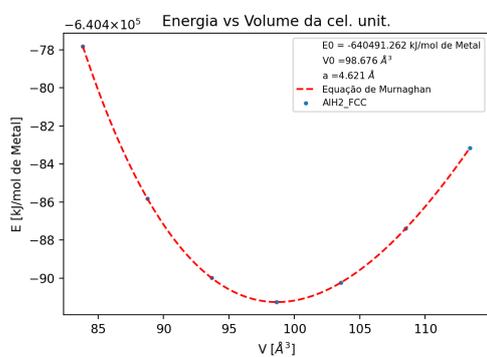
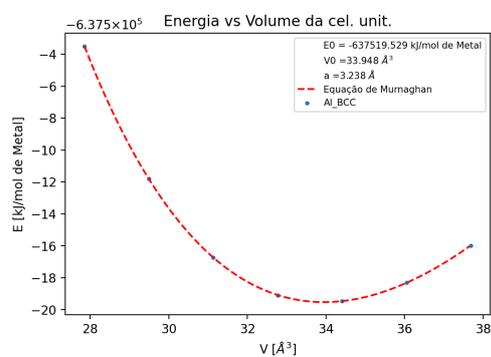
- derhoff, M., Botta, W.J., et al. Mg-containing multi-principal element alloys for hydrogen storage: A study of the $\text{Mg}_{0.5}\text{Ti}_{0.5}\text{Ni}_{0.5}$ and $\text{Mg}_{0.68}\text{Ti}_{0.55}\text{Ni}_{0.5}$ compositions. *International Journal of Hydrogen Energy* 2020;45(38):19539–19552.
- [29] Zepon, G., Leiva, D., Strozi, R., Bedoch, A., Figueroa, S., Ishikawa, T., et al. Hydrogen-induced phase transition of $\text{Mg}_{0.5}\text{Zr}_{0.5}\text{Co}_{0.5}\text{Ni}_{0.5}$ high entropy alloy. *International Journal of Hydrogen Energy* 2018;43(3):1702–1708. URL: <https://www.sciencedirect.com/science/article/pii/S0360319917344865>. doi:<https://doi.org/10.1016/j.ijhydene.2017.11.106>.
- [30] Flanagan, T., Oates, W.. Some thermodynamic aspects of metal hydrogen systems. *Journal of alloys and compounds* 2005;404:16–23.
- [31] Hohenberg, P., Kohn, W.. Inhomogeneous electron gas. *Phys Rev* 1964;136:B864–B871. doi:10.1103/PhysRev.136.B864.
- [32] Giustino, F.. *Materials Modelling using Density Functional Theory Properties and Predictions*. Oxford University Press; 2014.
- [33] Perdew, J., Burke, K., Ernzerhof, M.. D. of physics and nol 70118 j. quantum theory group tulane university. *Phys Rev Lett* 1996;77:3865–3868.
- [34] Fuchs, M.. Comparison of exchange–correlation functionals: from lda to gga and beyond. *Density-Functional Theory Calculations for Modeling Materials and Bio-Molecular Properties and Functions-A Hands-On Computer Course*, Los Angeles 2005;.
- [35] Tyuterev, V., Vast, N.. Murnaghan’s equation of state for the electronic ground state energy. *Computational materials science* 2006;38(2):350–353.
- [36] Blaha, P., Schwarz, K., Tran, F., Laskowski, R., Madsen, G.K., Marks, L.D.. Wien2k: An apw+ lo program for calculating the properties of solids. *The Journal of chemical physics* 2020;152(7):074101.
- [37] Blaha, P., Schwarz, K., Madsen, G.K.H., Kvasnicka, D., Luitz, J., Laskowski, R., et al. WIEN2k: An Augmented Plane Wave plus Local Orbitals Program for Calculating Crystal Properties. Austria: Vienna University

of Technology; 2020. URL: http://www.wien2k.at/reg_user/textbooks/usersguide.pdf.

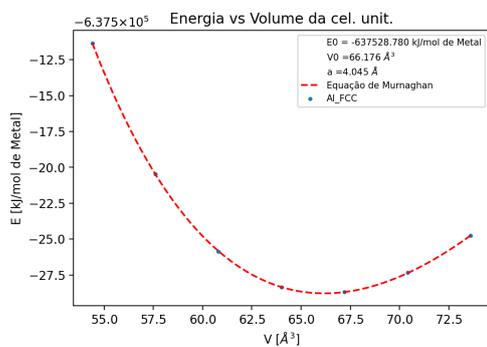
- [38] Griessen, R., Riesterer, T.. Heat of formation models. Hydrogen in inter-metallic compounds I 1988;:219–284.

APÊNDICE A: Dados obtidos do DFT

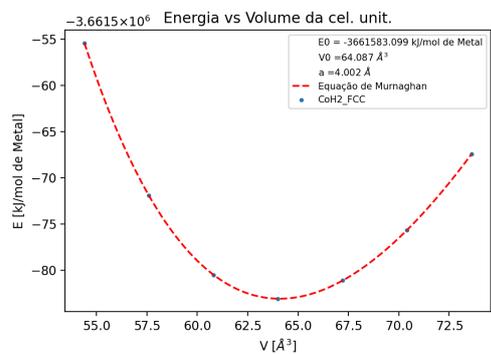
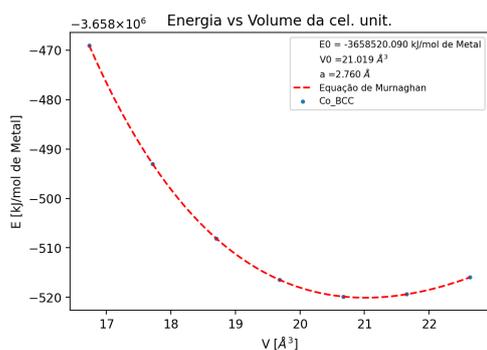
DFT das estruturas CCC, CFC e MH₂.

(a) AlH₂ - FCC

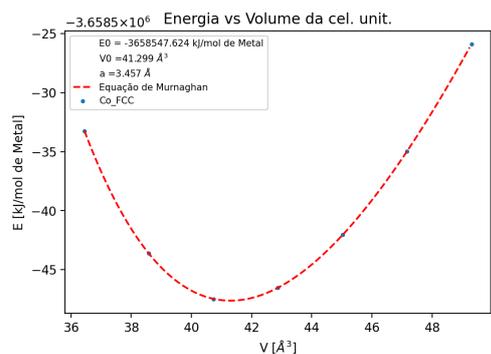
(b) Al - BCC



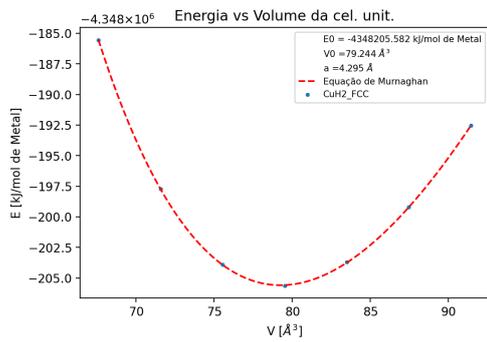
(c) Al - FCC

(d) CoH₂ - FCC

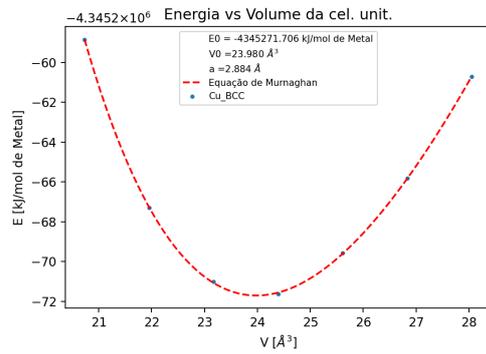
(e) Co - BCC



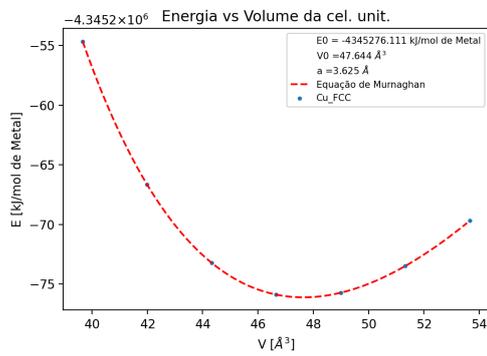
(f) Co - FCC



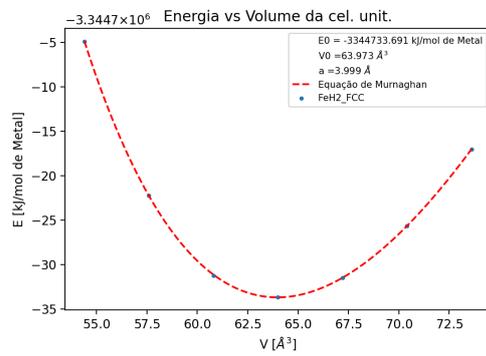
(g) CuH2 - FCC



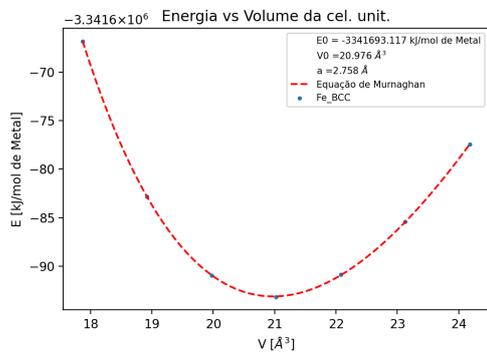
(h) Cu - BCC



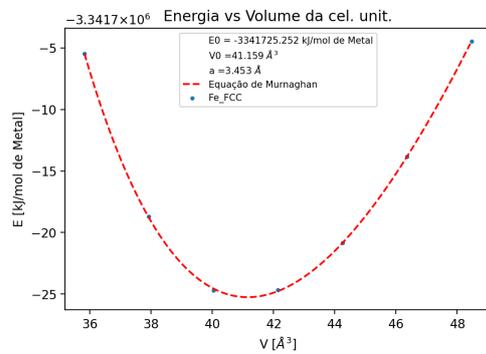
(i) Cu - FCC



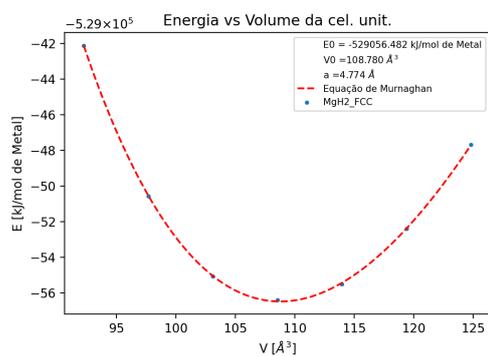
(j) FeH2 - FCC



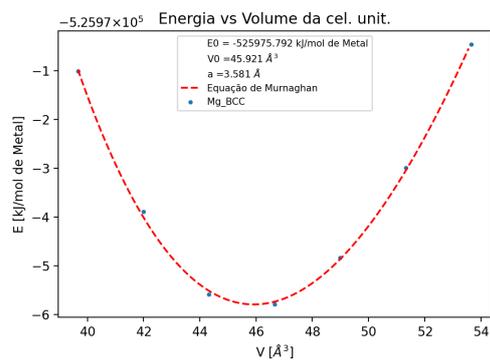
(k) Fe - BCC



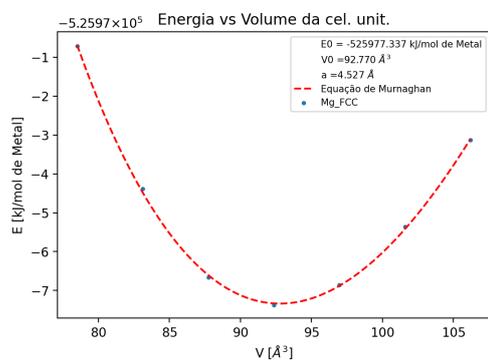
(l) Fe - FCC



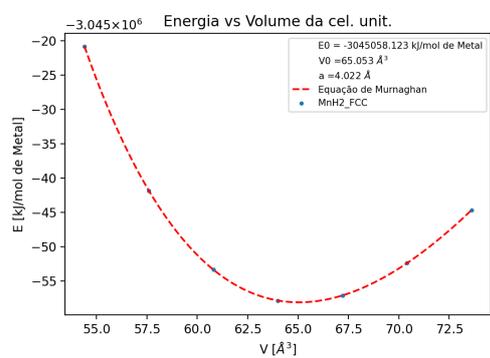
(m) MgH2 - FCC



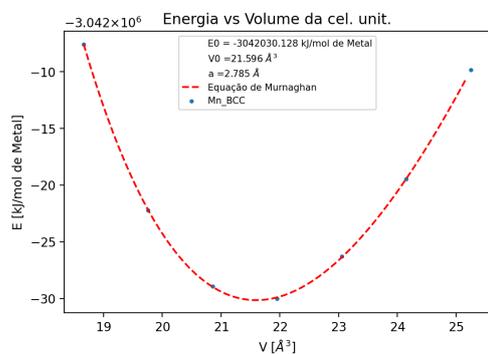
(n) Mg - BCC



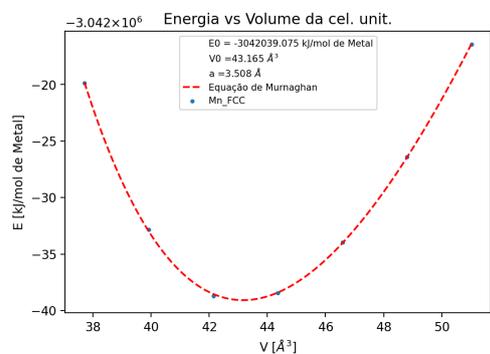
(o) Mg - FCC



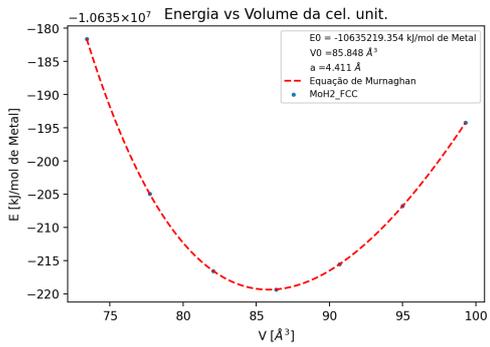
(p) MnH2 - FCC



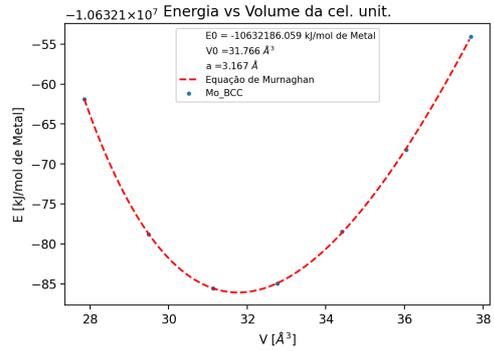
(q) Mn - BCC



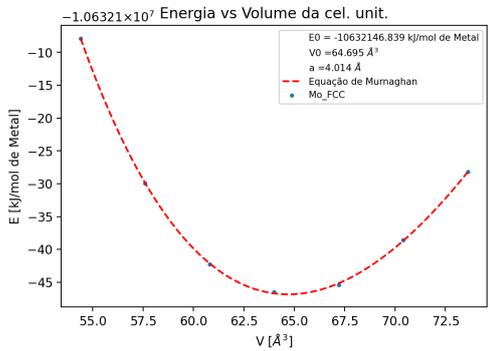
(r) Mn - FCC



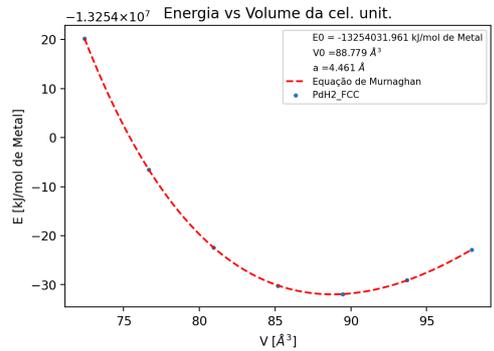
(s) MoH2 - FCC



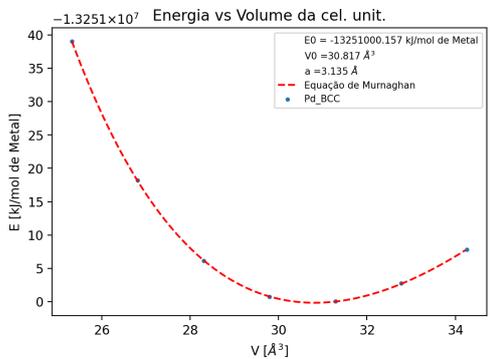
(t) Mo - BCC



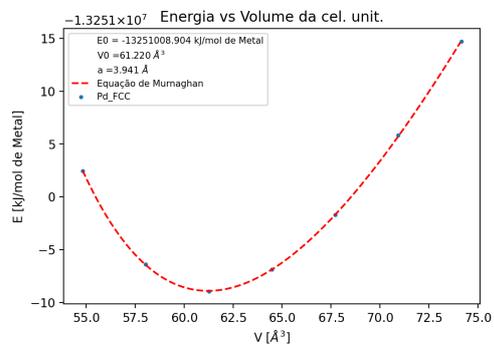
(u) Mo - FCC



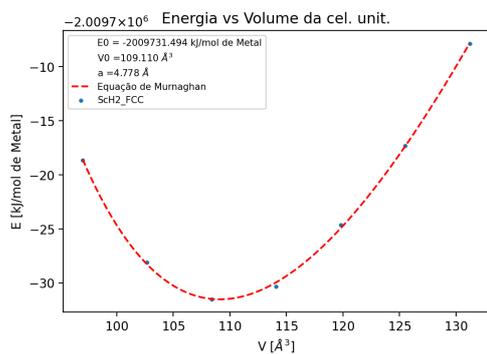
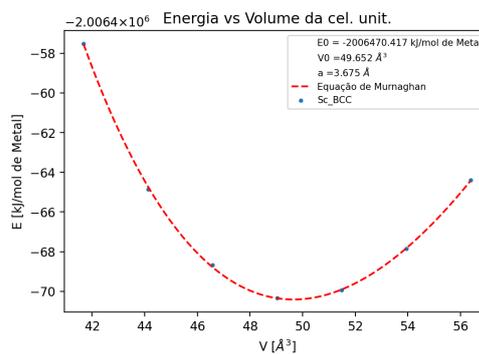
(v) PdH2 - FCC



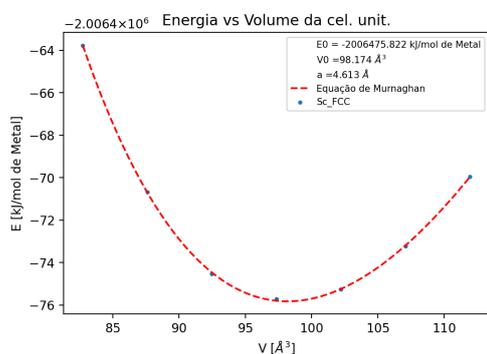
(w) Pd - BCC



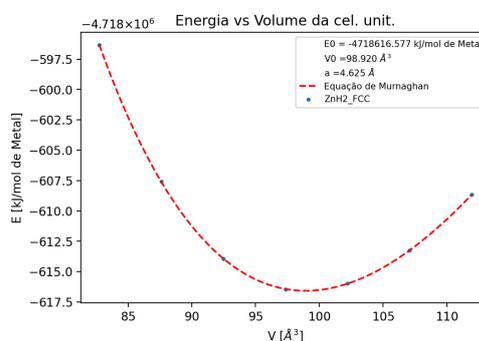
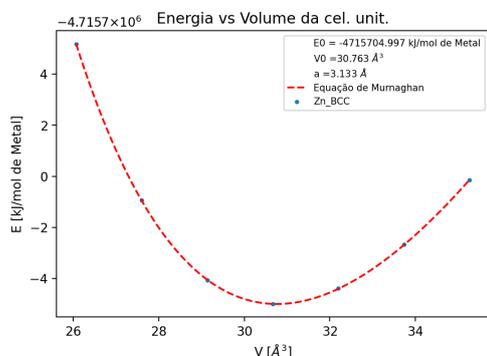
(x) Pd - FCC

(y) ScH₂ - FCC

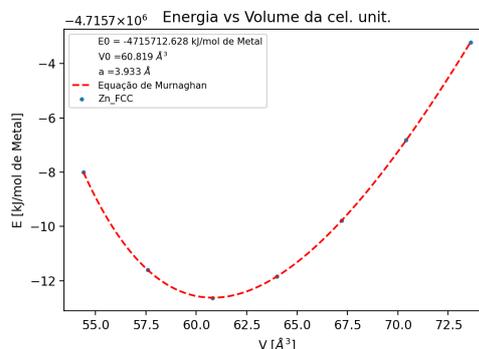
(z) Sc - BCC



(aa) Sc - FCC

(ab) ZnH₂ - FCC

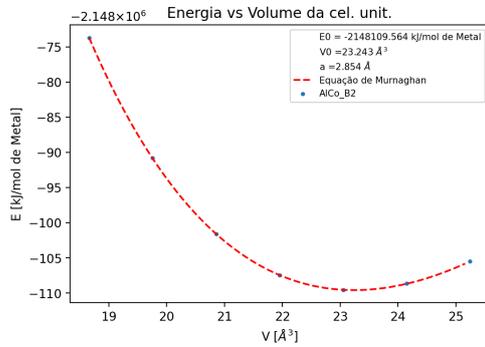
(ac) Zn - BCC



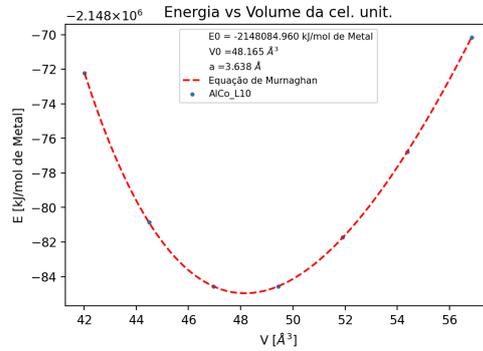
(ad) Zn - FCC

Gráficos das estruturas CCC, CFC e MH₂ para os elementos Al, Co, Cu, Fe, Mg, Mn, Mo, Pd, Sc e Zn

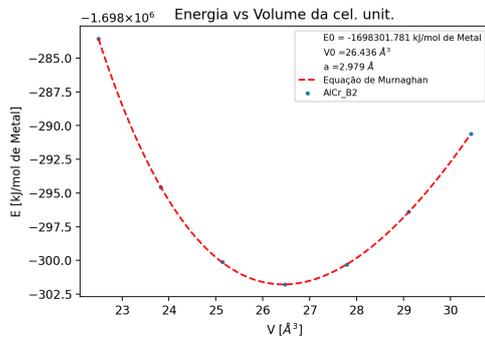
DFT das estruturas B2 e L10



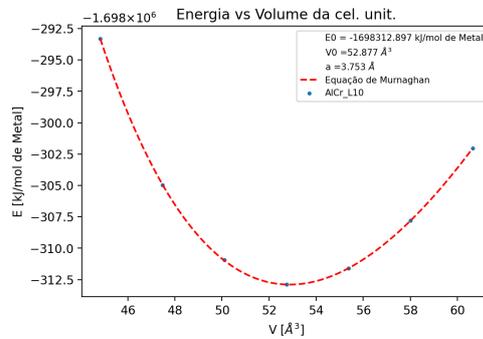
(a) AlCo - B2



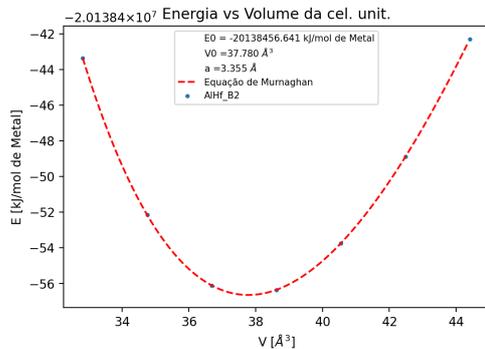
(b) AlCo - L10



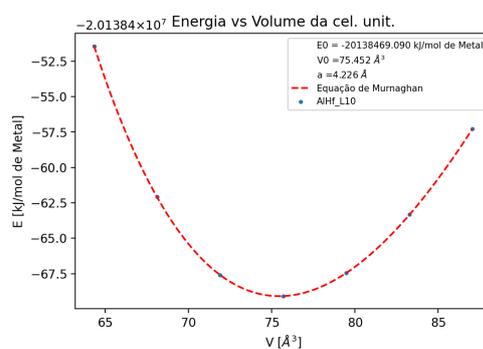
(c) AlCr - B2



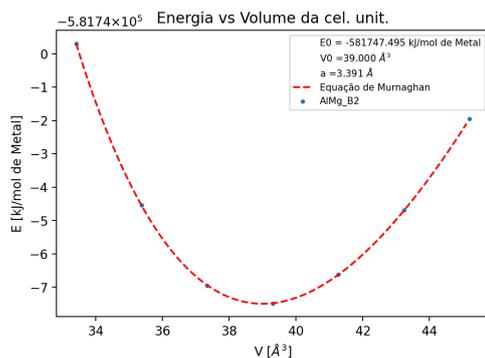
(d) AlCr - L10



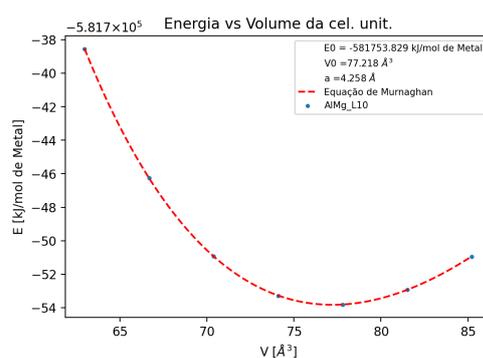
(e) AlHf - B2



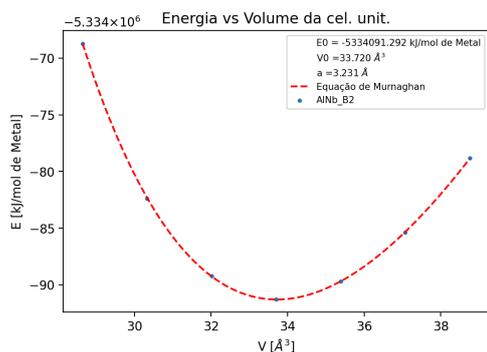
(f) AlHf - L10



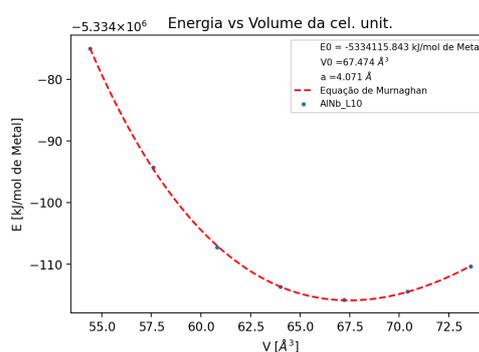
(g) AlMg - B2



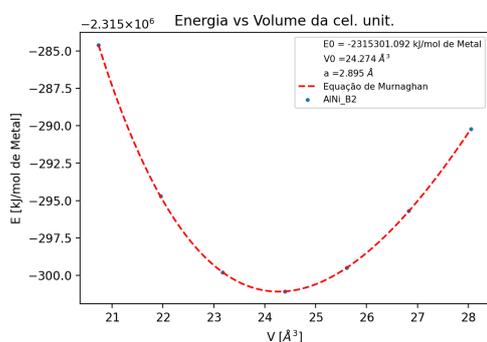
(h) AlMg - L10



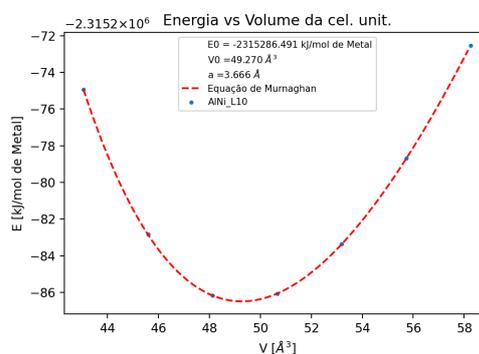
(i) AlNb - B2



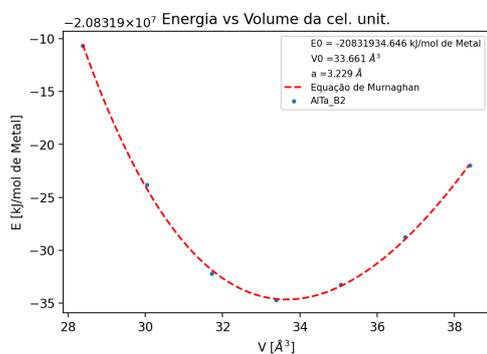
(j) AlNb - L10



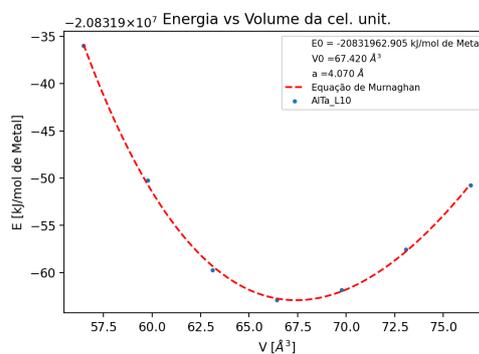
(k) AlNi - B2



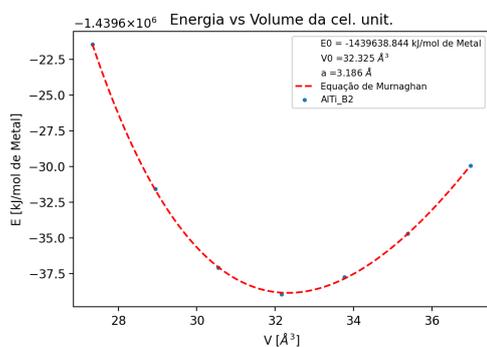
(l) AlNi - L10



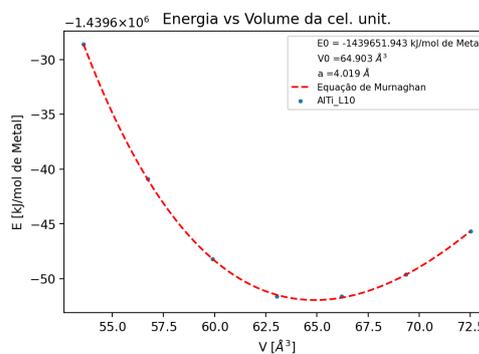
(m) AlTa - B2



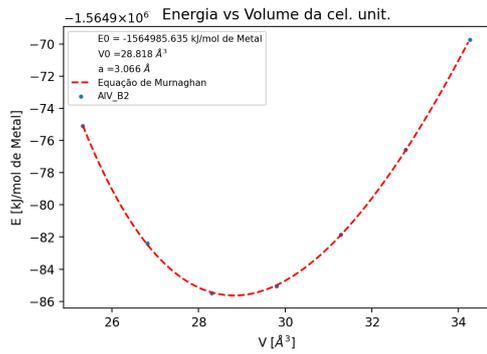
(n) AlTa - L10



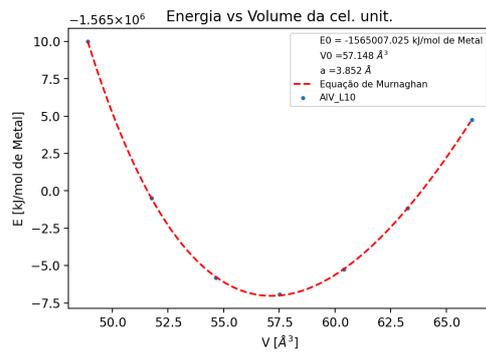
(o) AlTi - B2



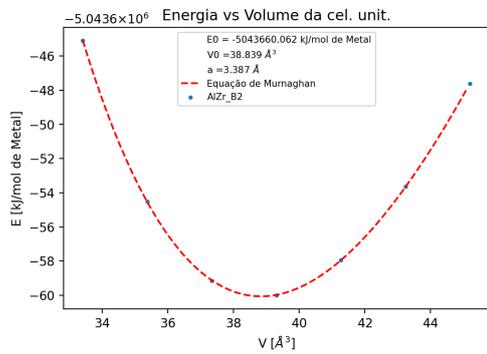
(p) AlTi - L10



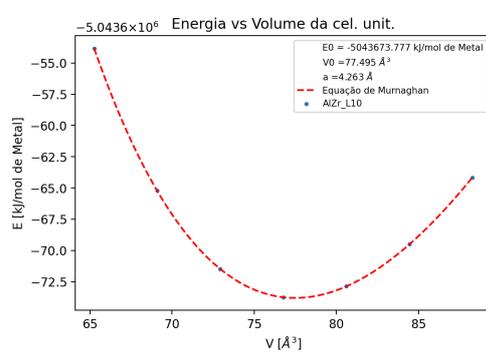
(q) AlV - B2



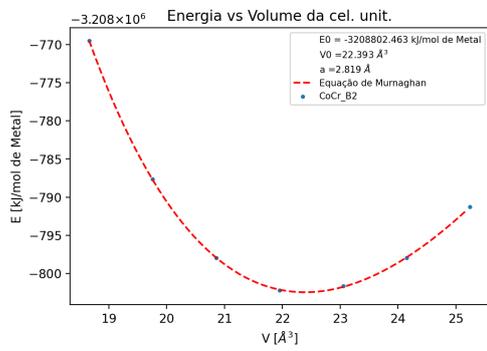
(r) AlV - L10



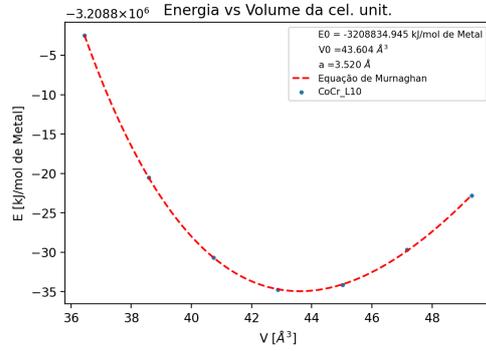
(s) AlZr - B2



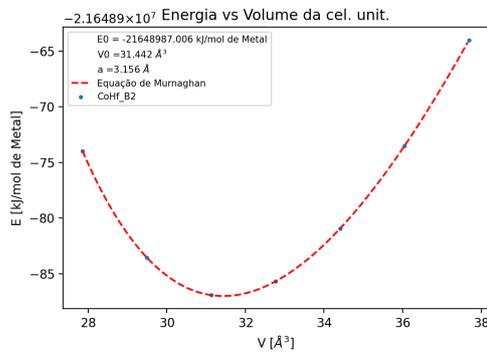
(t) AlZr - L10



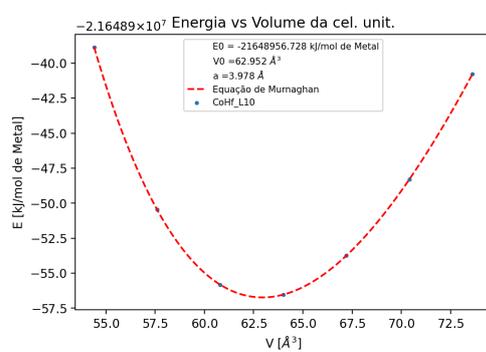
(u) CoCr - B2



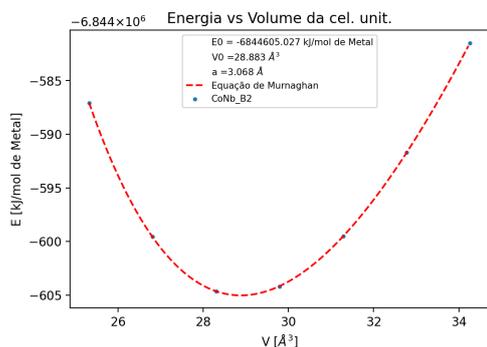
(v) CoCr - L10



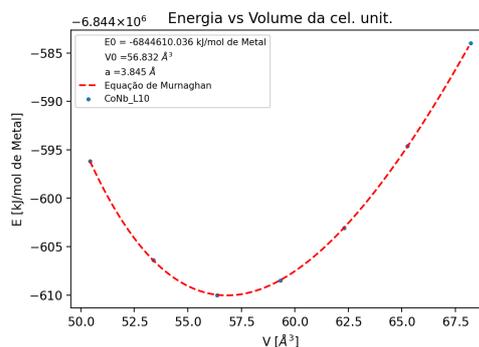
(w) CoHf - B2



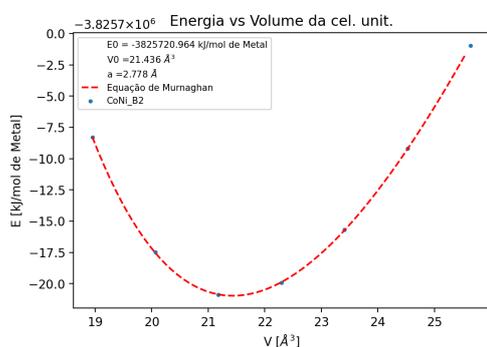
(x) CoHf - L10



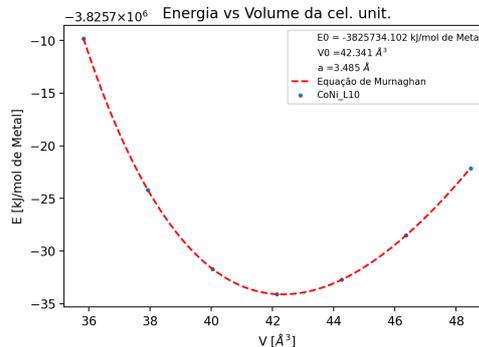
(y) CoNb - B2



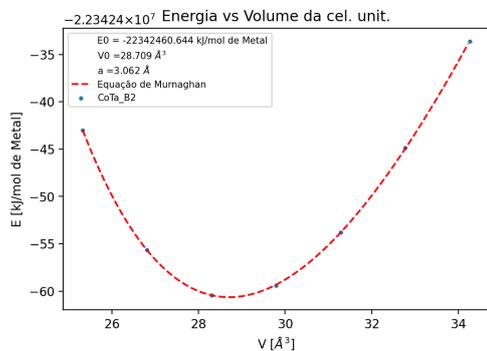
(z) CoNb - L10



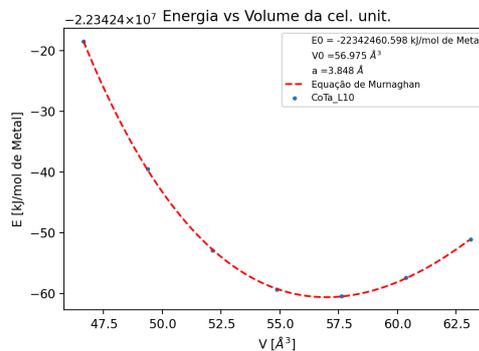
(aa) CoNi - B2



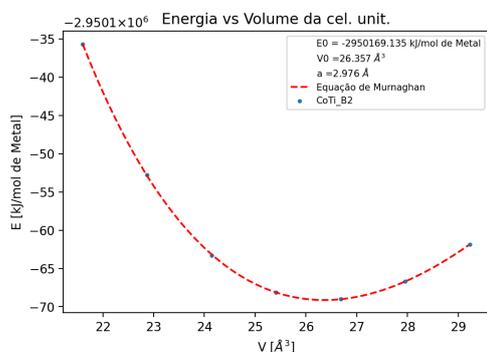
(ab) CoNi - L10



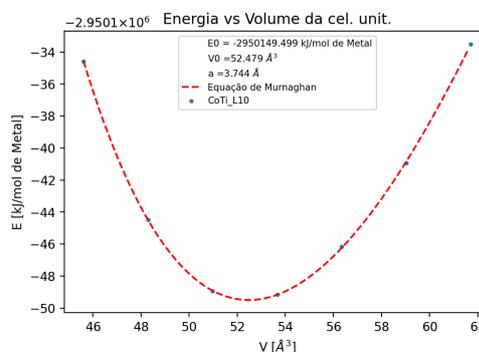
(ac) CoTa - B2



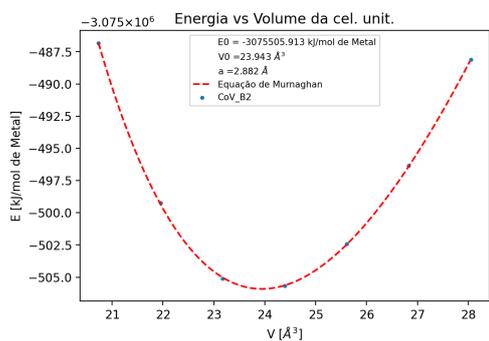
(ad) CoTa - L10



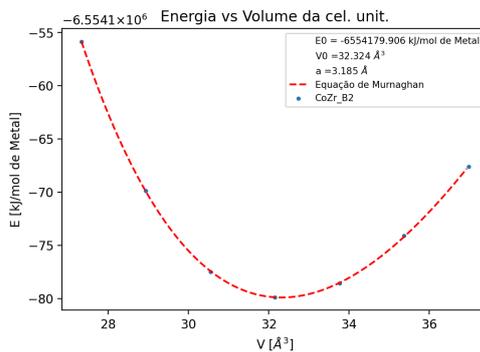
(ae) CoTi - B2



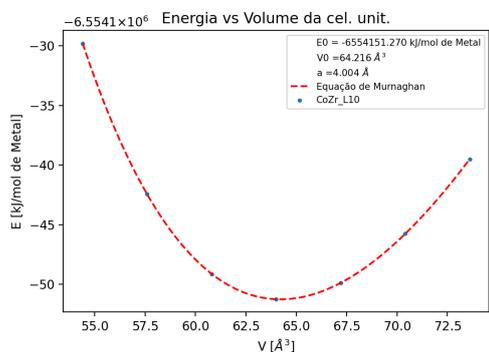
(af) CoTi - L10



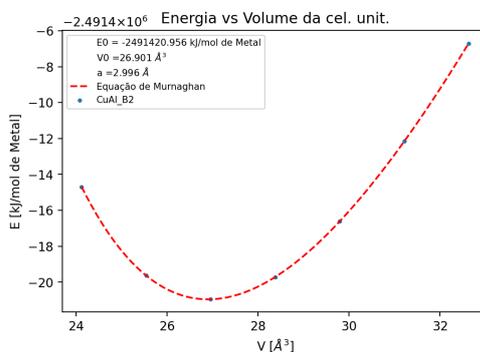
(ag) CoV - B2



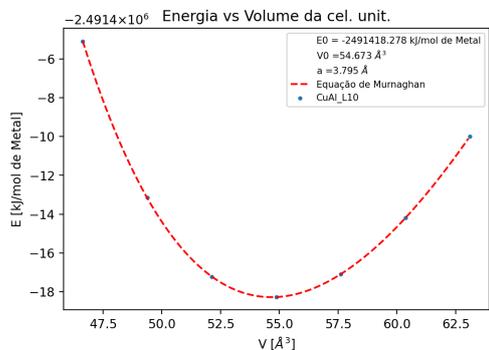
(ah) CoZr - B2



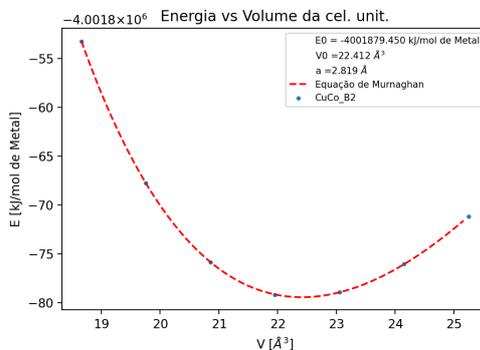
(ai) CoZr - L10



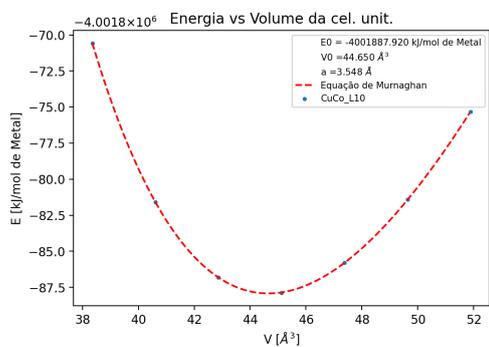
(aj) CuAl - B2



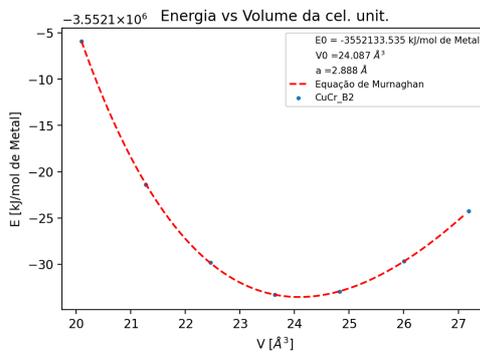
(ak) CuAl - L10



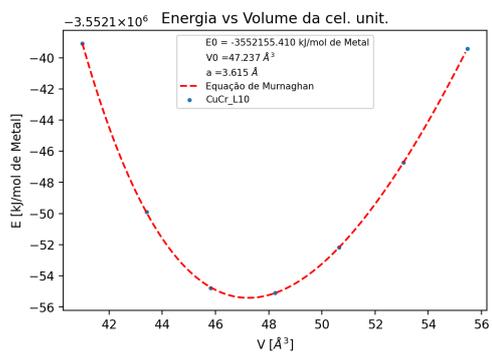
(al) CuCo - B2



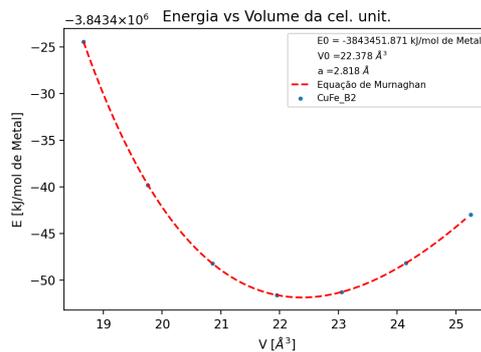
(am) CuCo - L10



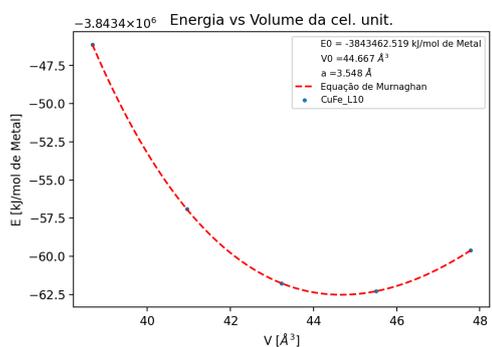
(an) CuCr - B2



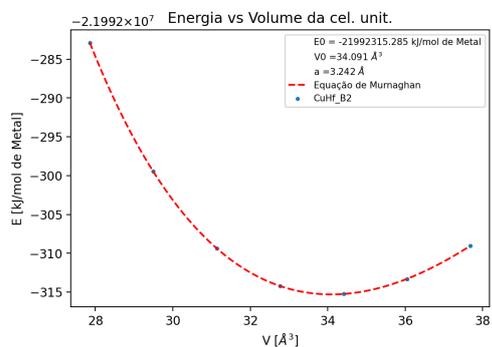
(ao) CuCr - L10



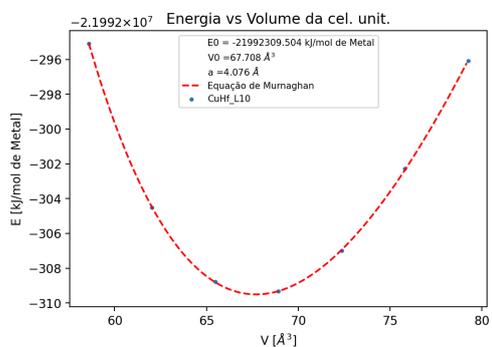
(ap) CuFe - B2



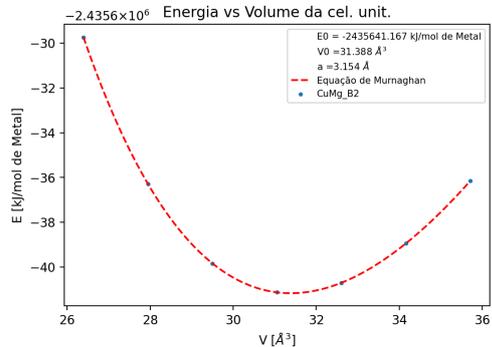
(aq) CuFe - L10



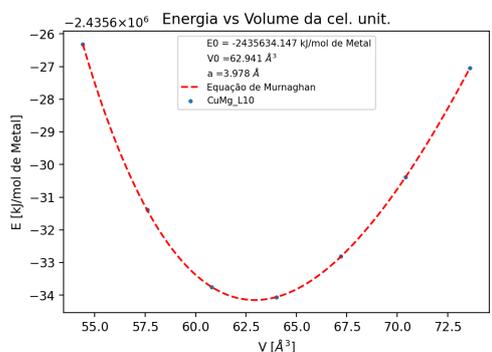
(ar) CuHf - B2



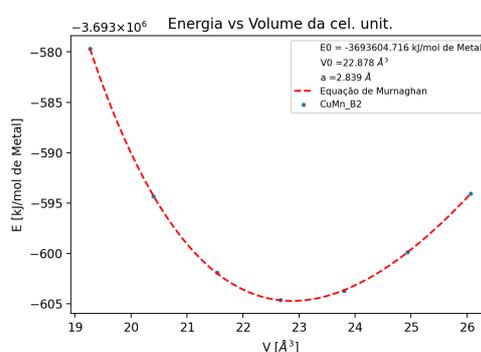
(as) CuHf - L10



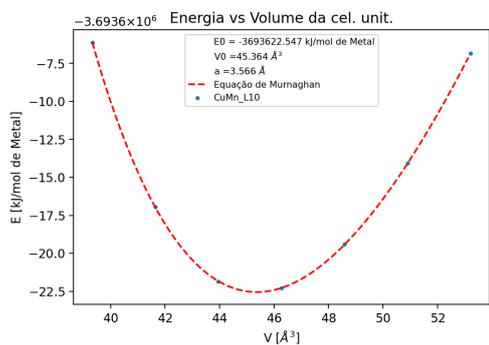
(at) CuMg - B2



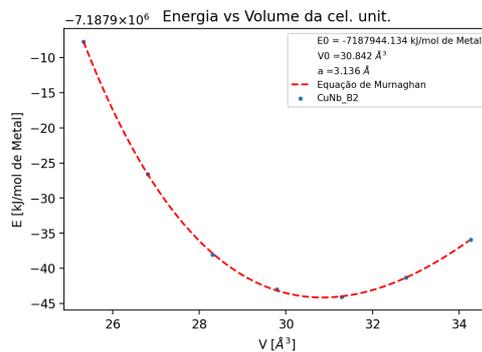
(au) CuMg - L10



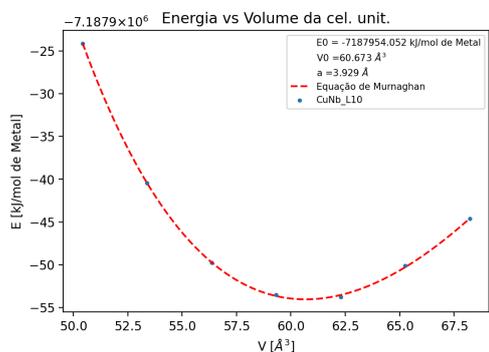
(av) CuMn - B2



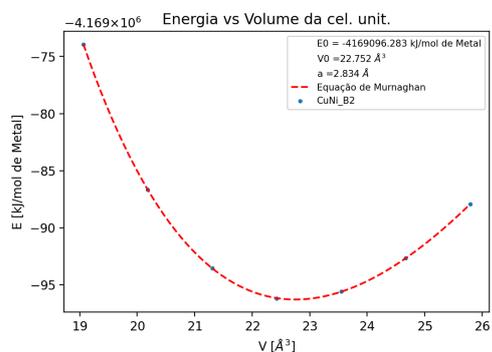
(aw) CuMn - L10



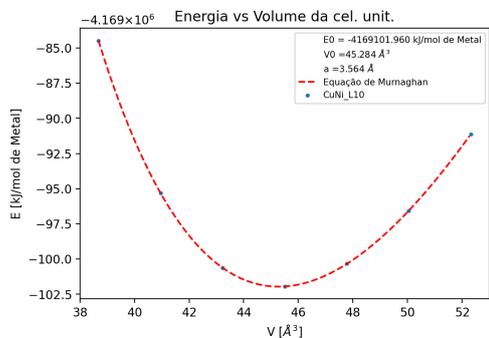
(ax) CuNb - B2



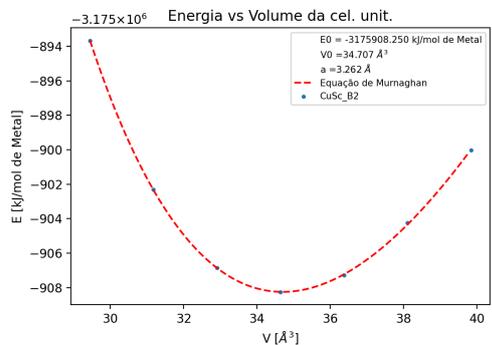
(ay) CuNb - L10



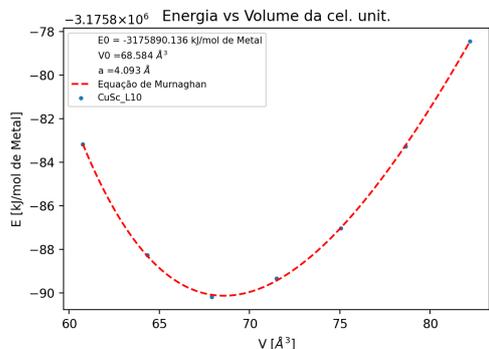
(az) CuNi - B2



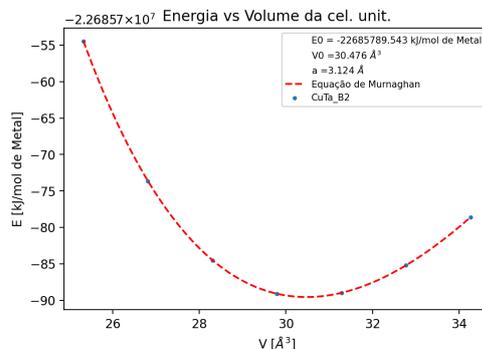
(ba) CuNi - L10



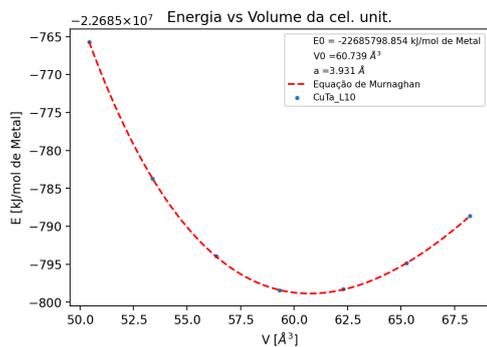
(bb) CuSc - B2



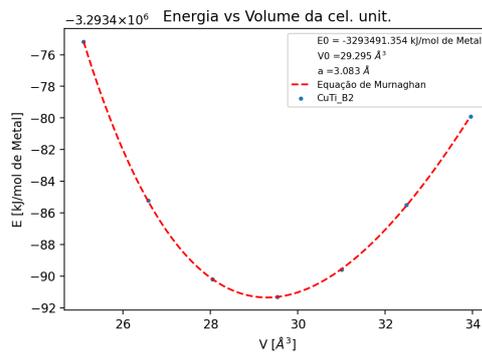
(bc) CuSc - L10



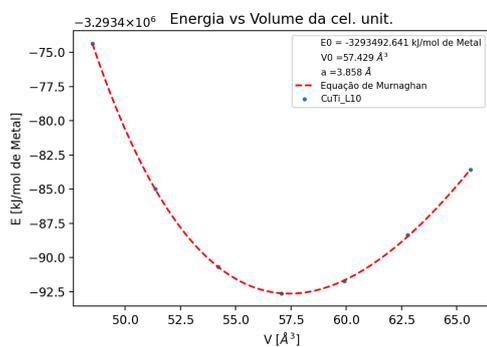
(bd) CuTa - B2



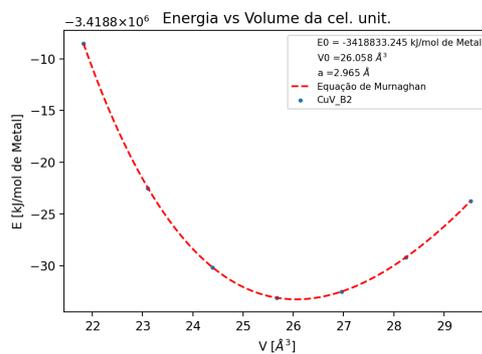
(be) CuTa - L10



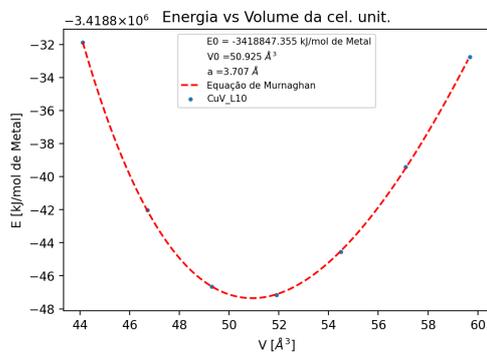
(bf) CuTi - B2



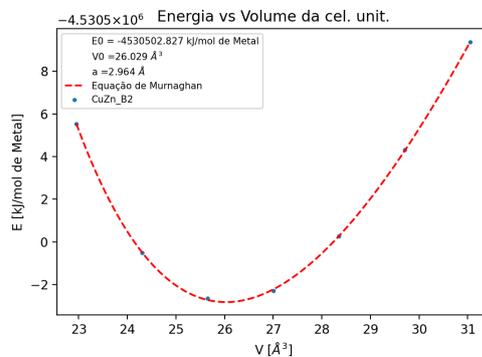
(bg) CuTi - L10



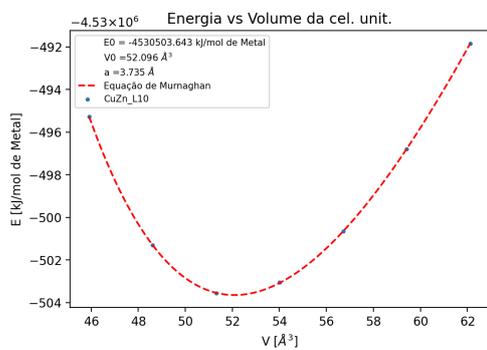
(bh) CuV - B2



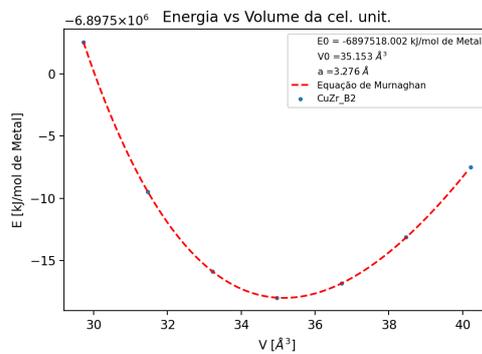
(bi) CuV - L10



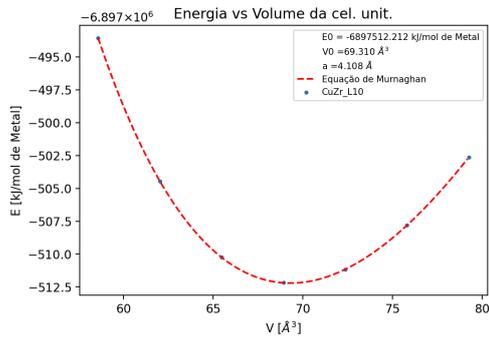
(bj) CuZn - B2



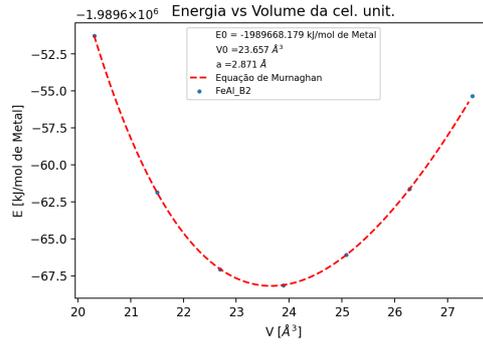
(bk) CuZn - L10



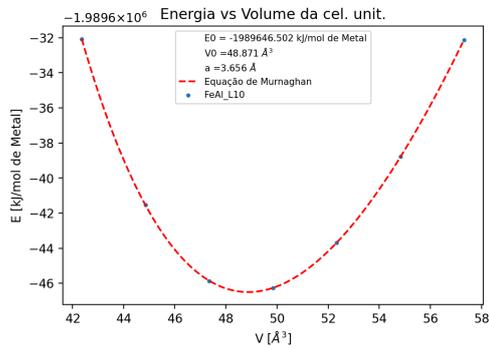
(bl) CuZr - B2



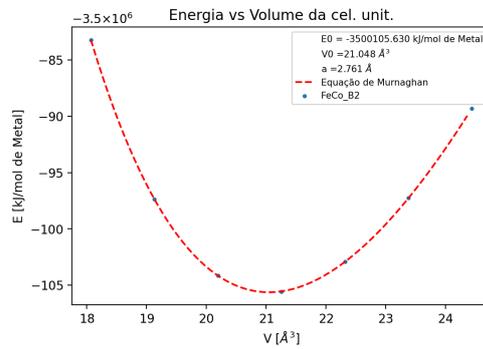
(bm) CuZr - L10



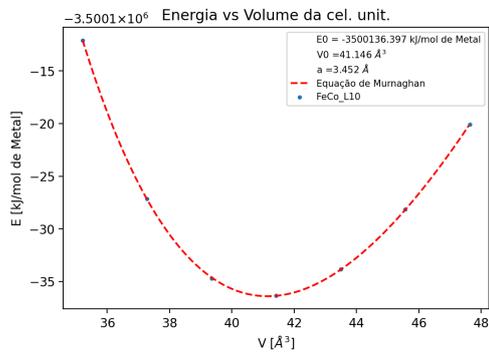
(bn) FeAl - B2



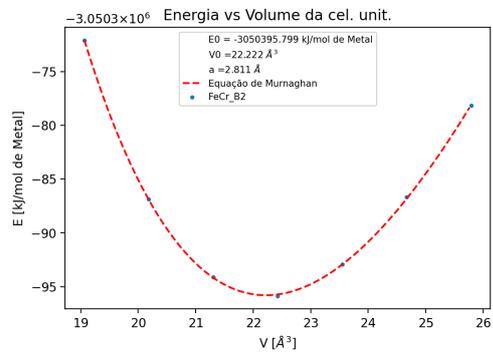
(bo) FeAl - L10



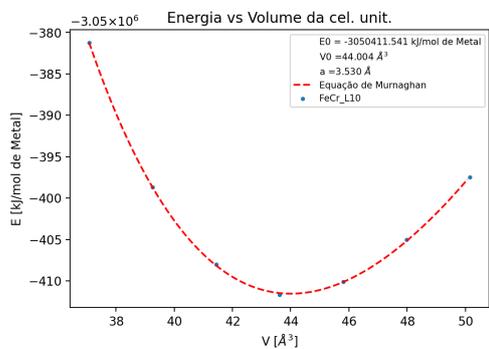
(bp) FeCo - B2



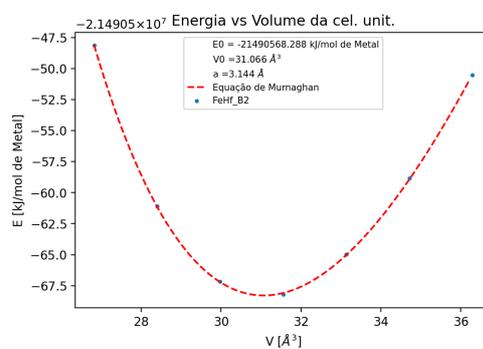
(bq) FeCo - L10



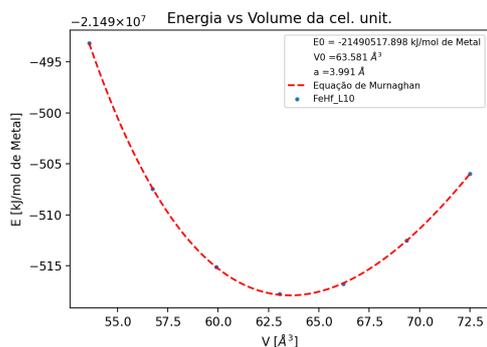
(br) FeCr - B2



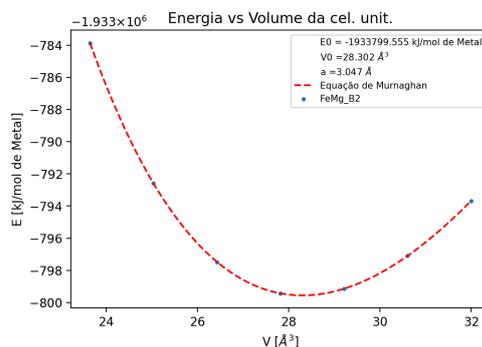
(bs) FeCr - L10



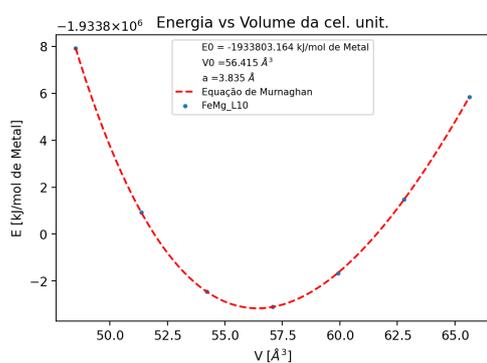
(bt) FeHf - B2



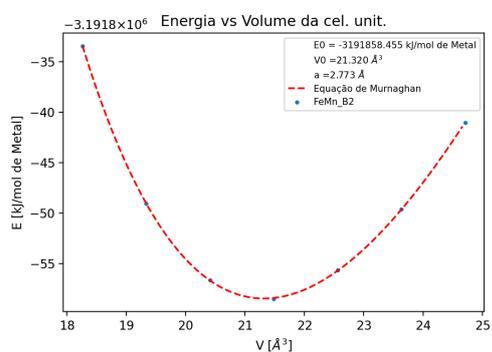
(bu) FeHf - L10



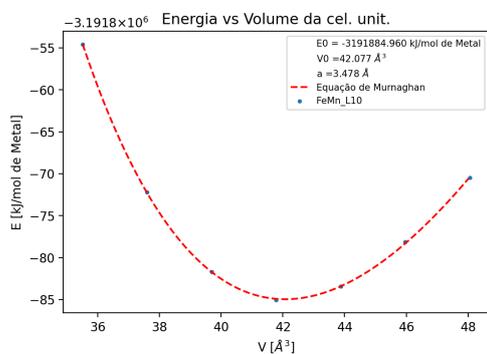
(bv) FeMg - B2



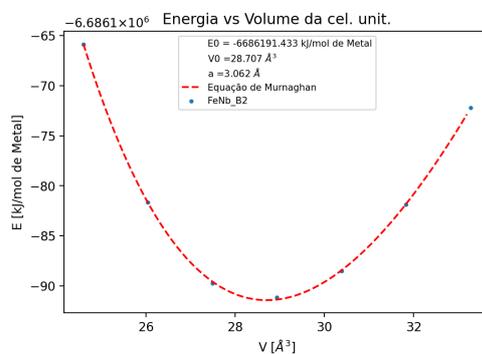
(bw) FeMg - L10



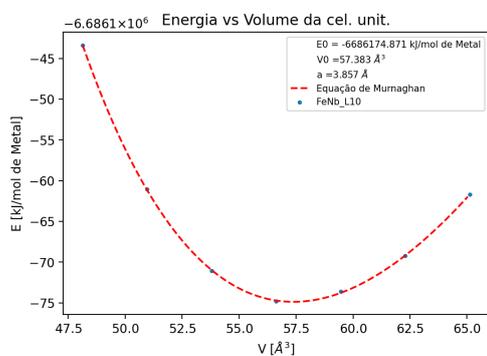
(bx) FeMn - B2



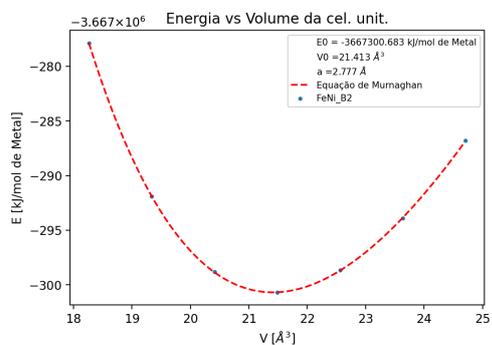
(by) FeMn - L10



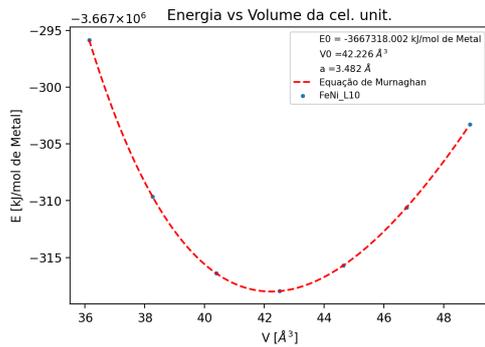
(bz) FeNb - B2



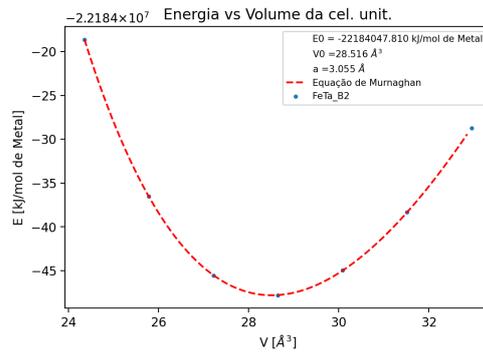
(ca) FeNb - L10



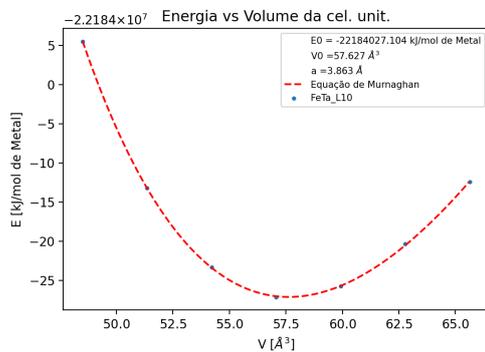
(cb) FeNi - B2



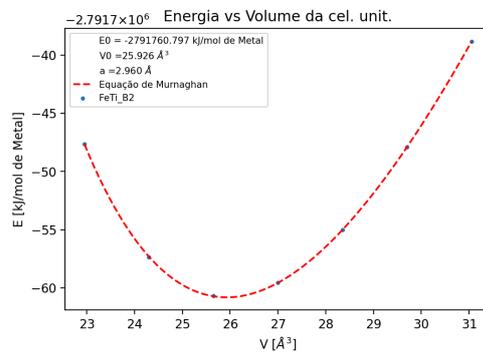
(cc) FeNi - L10



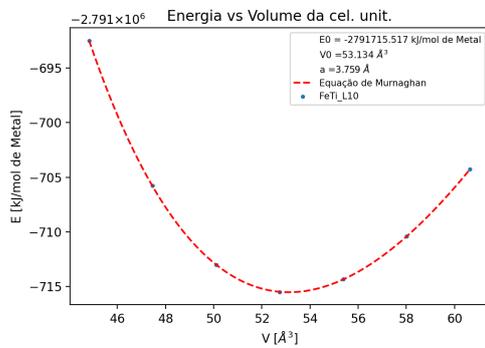
(cd) FeTa - B2



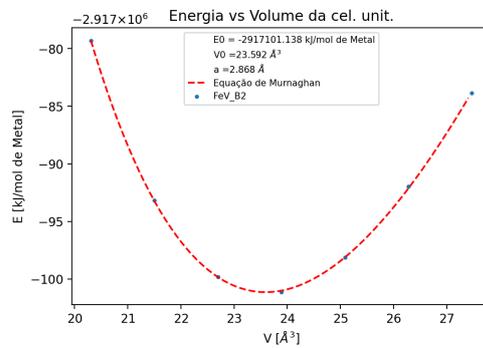
(ce) FeTa - L10



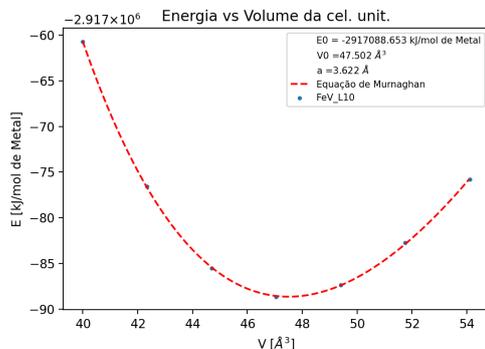
(cf) FeTi - B2



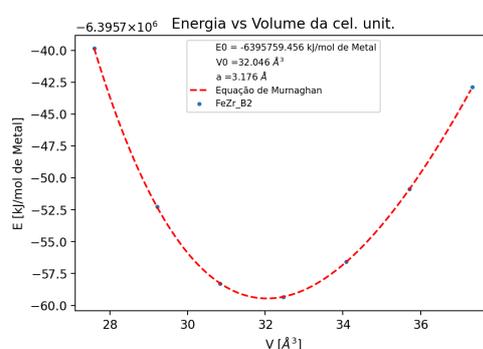
(cg) FeTi - L10



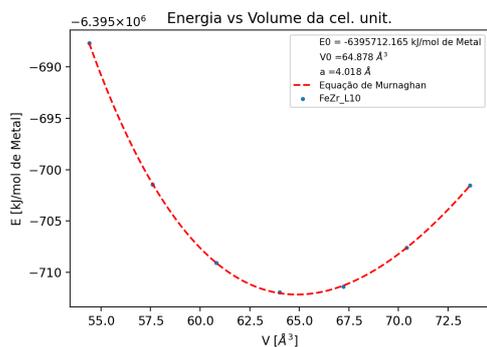
(ch) FeV - B2



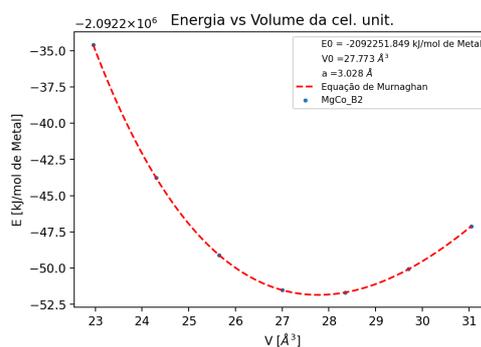
(ci) FeV - L10



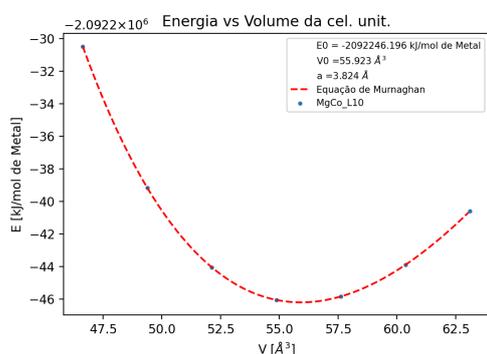
(cj) FeZr - B2



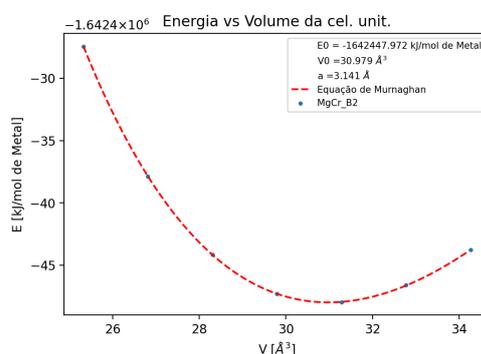
(ck) FeZr - L10



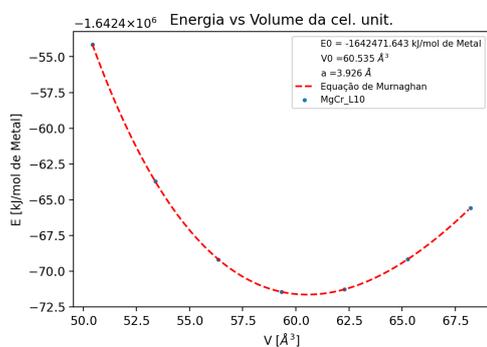
(cl) MgCo - B2



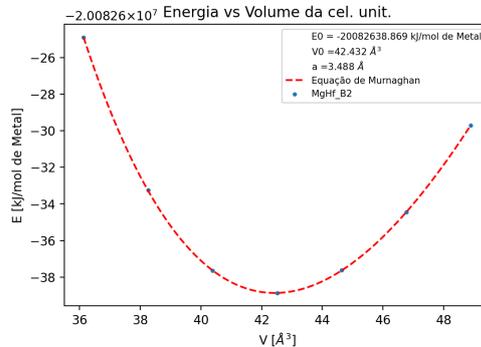
(cm) MgCo - L10



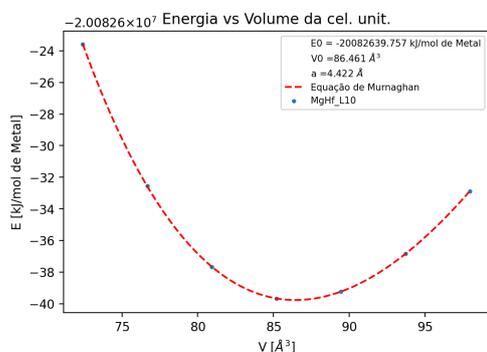
(cn) MgCr - B2



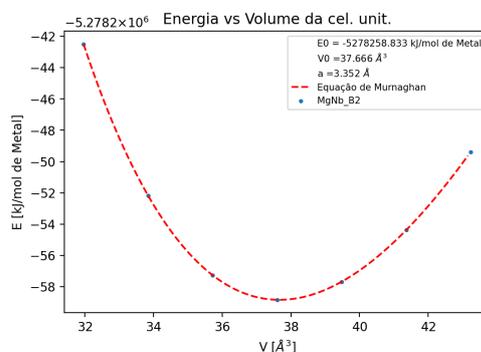
(co) MgCr - L10



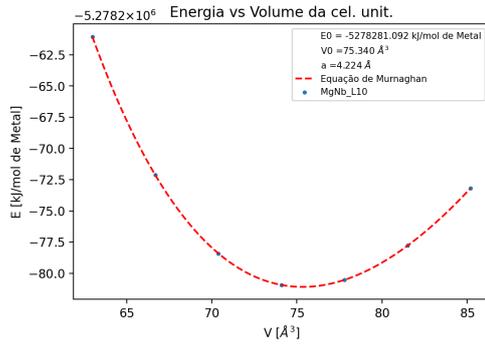
(cp) MgHf - B2



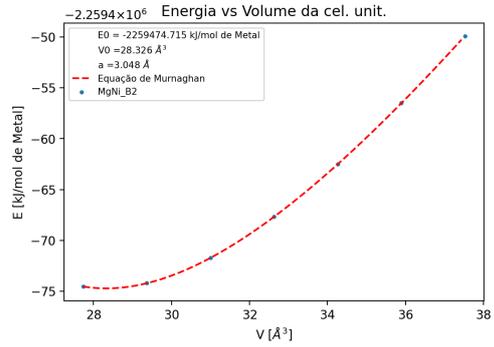
(cq) MgHf - L10



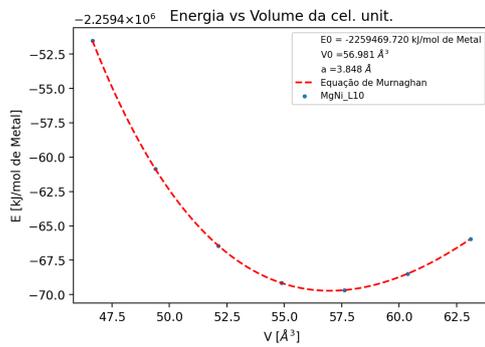
(cr) MgNb - B2



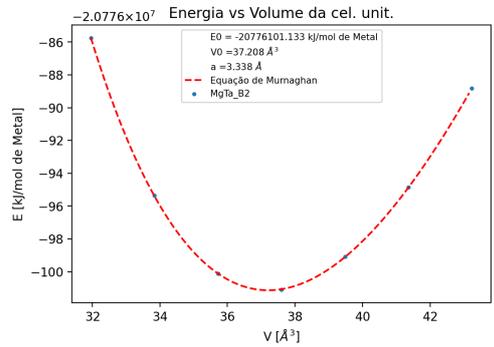
(cs) MgNb - L10



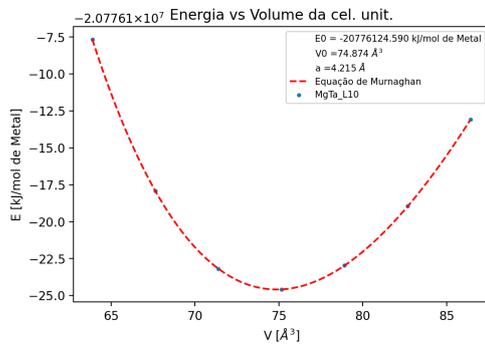
(ct) MgNi - B2



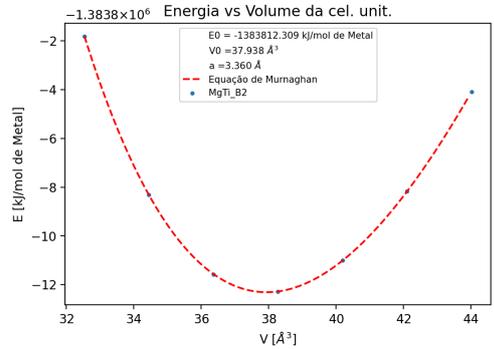
(cu) MgNi - L10



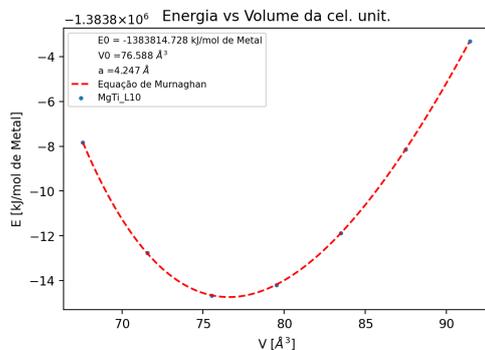
(cv) MgTa - B2



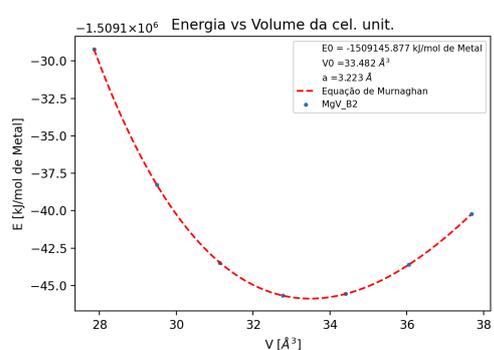
(cw) MgTa - L10



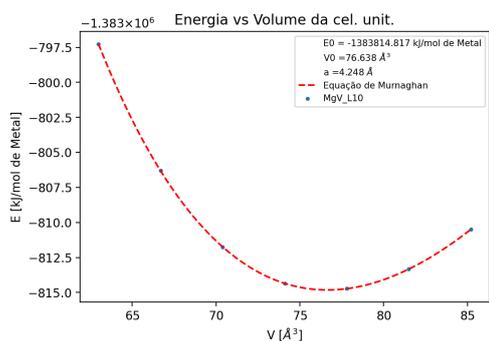
(cx) MgTi - B2



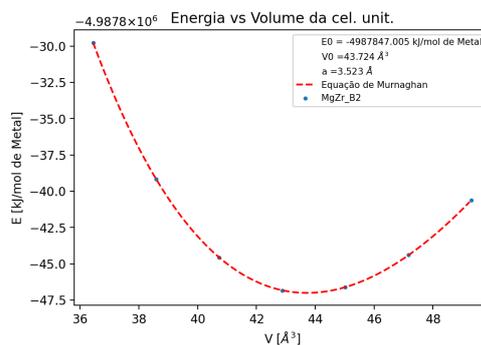
(cy) MgTi - L10



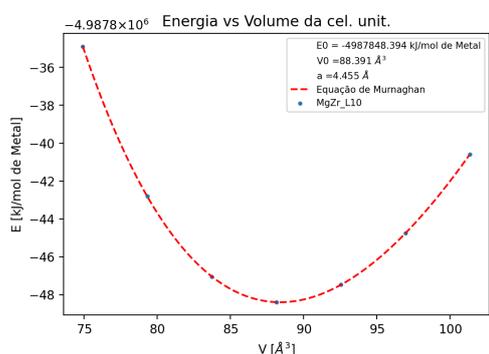
(cz) MgV - B2



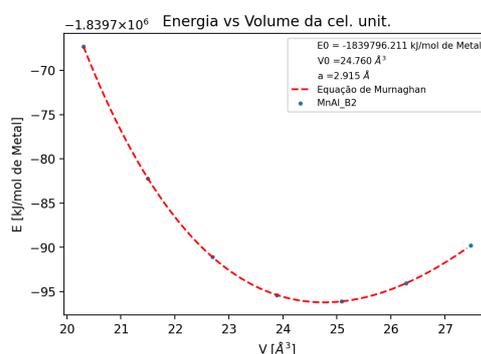
(da) MgV - L10



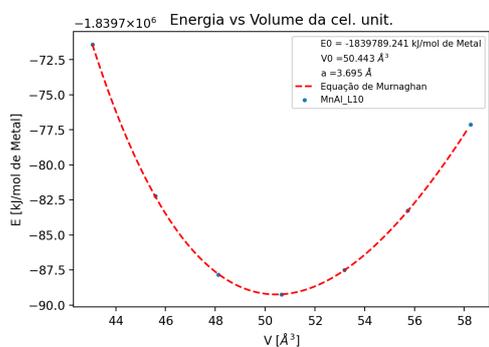
(db) MgZr - B2



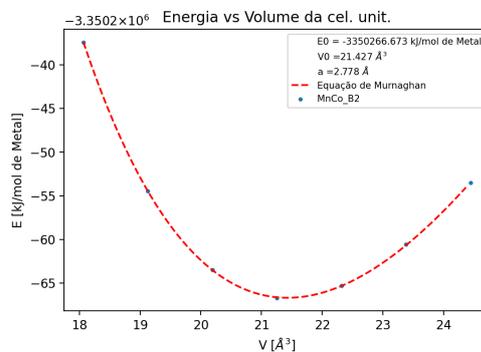
(dc) MgZr - L10



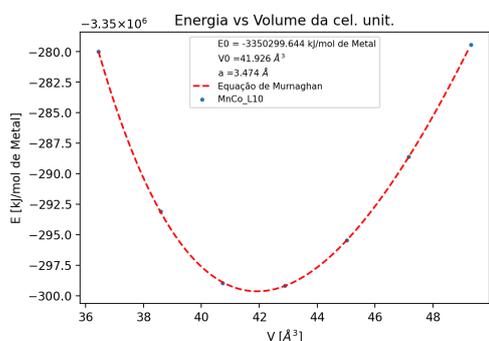
(dd) MnAl - B2



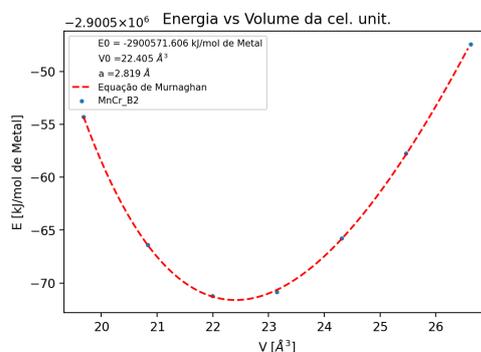
(de) MnAl - L10



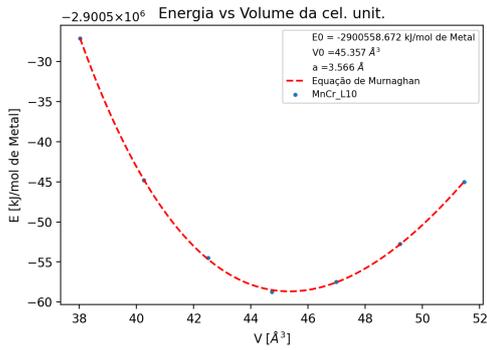
(df) MnCo - B2



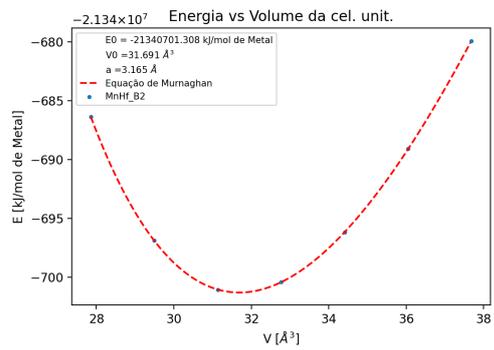
(dg) MnCo - L10



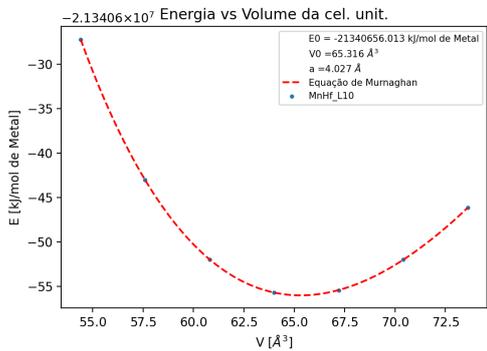
(dh) MnCr - B2



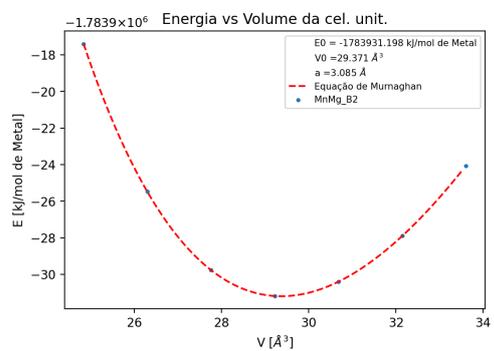
(di) MnCr - L10



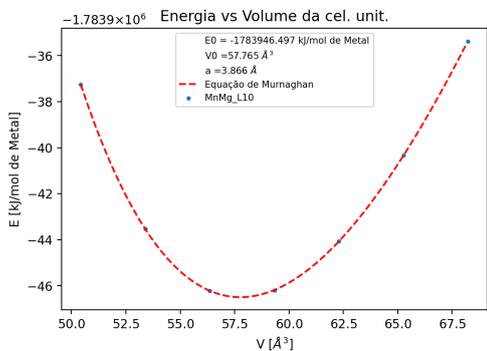
(dj) MnHf - B2



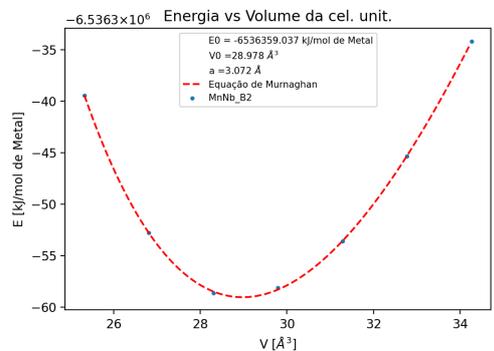
(dk) MnHf - L10



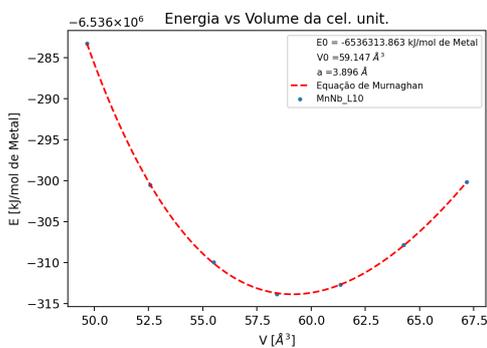
(dl) MnMg - B2



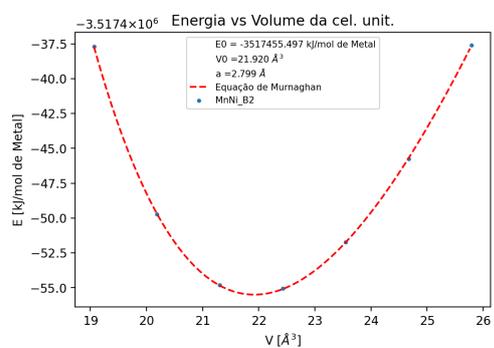
(dm) MnMg - L10



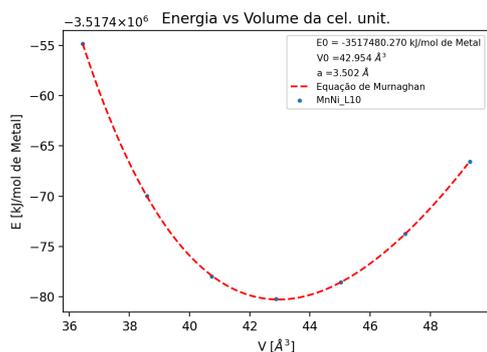
(dn) MnNb - B2



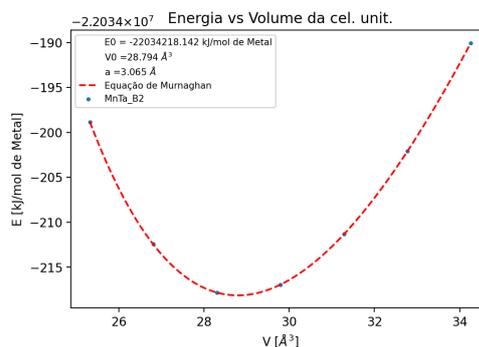
(do) MnNb - L10



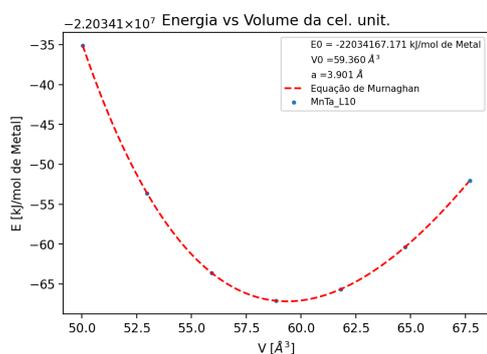
(dp) MnNi - B2



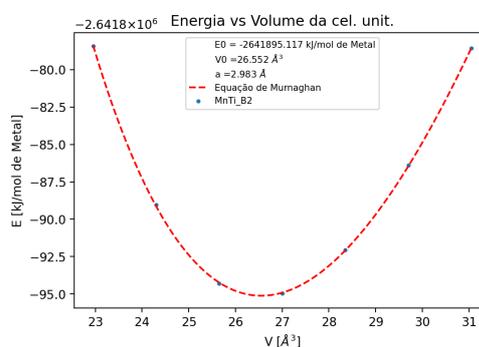
(dq) MnNi - L10



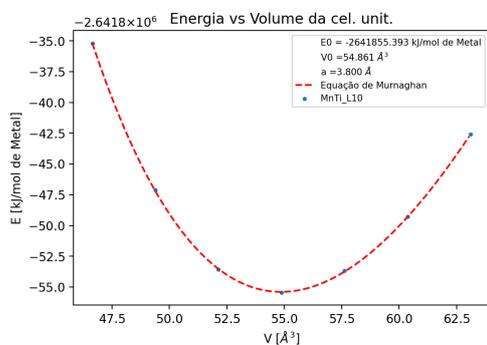
(dr) MnTa - B2



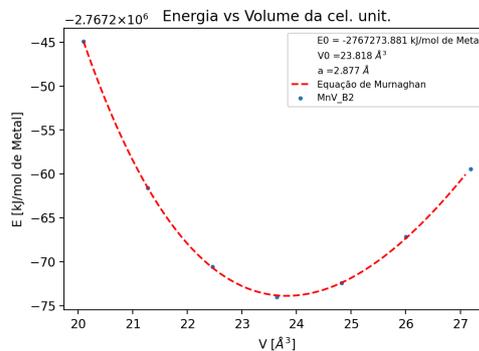
(ds) MnTa - L10



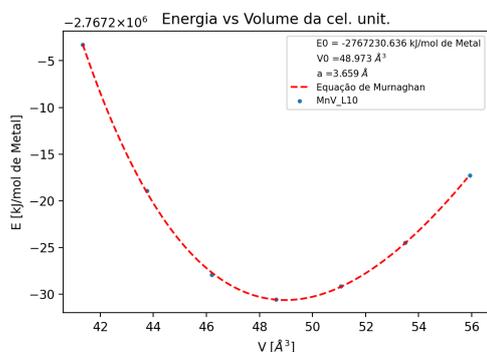
(dt) MnTi - B2



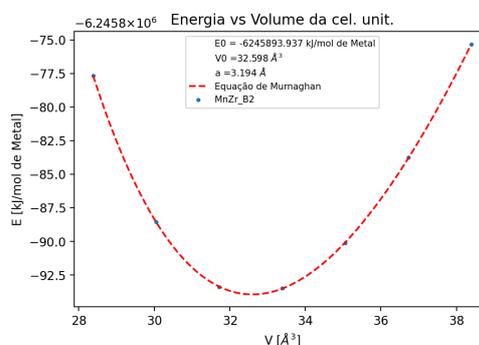
(du) MnTi - L10



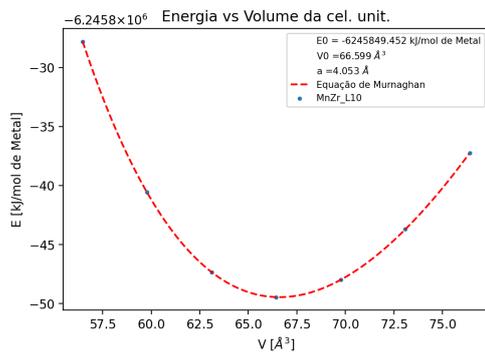
(dv) MnV - B2



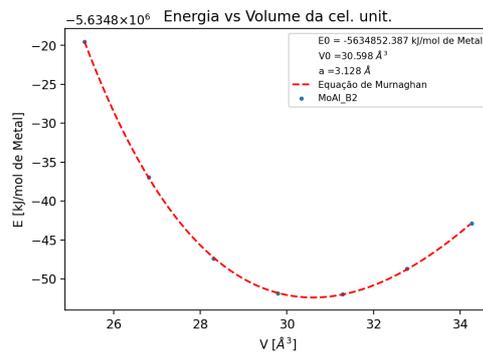
(dw) MnV - L10



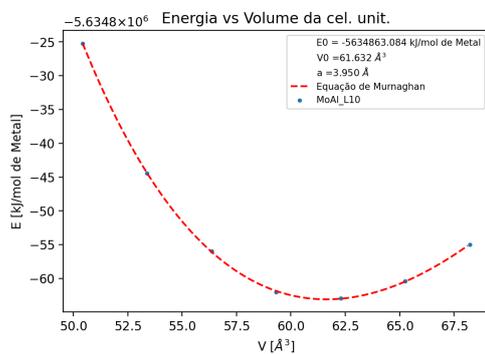
(dx) MnZr - B2



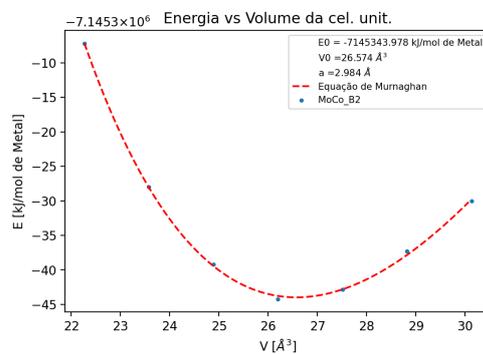
(dy) MnZr - L10



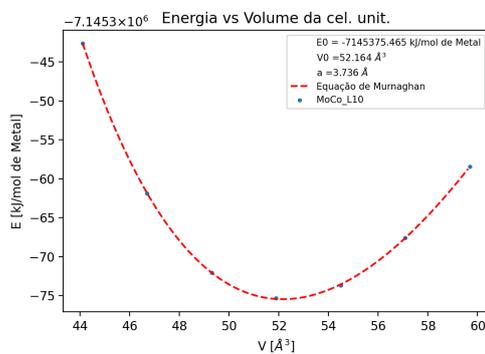
(dz) MoAl - B2



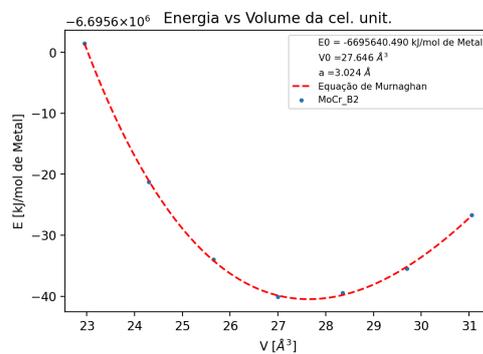
(ea) MoAl - L10



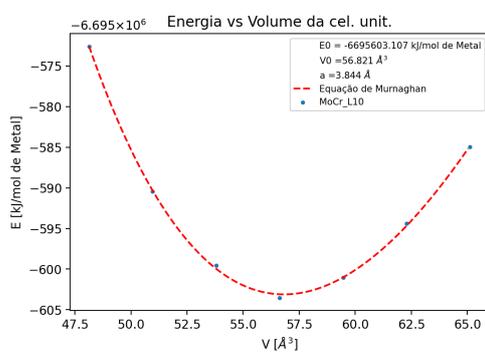
(eb) MoCo - B2



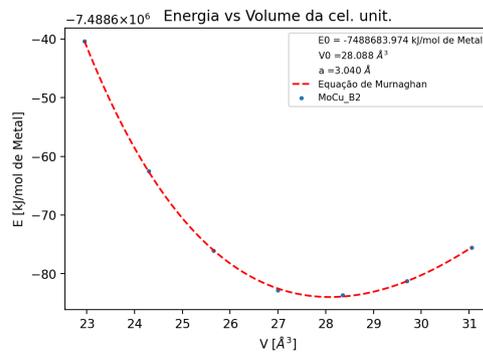
(ec) MoCo - L10



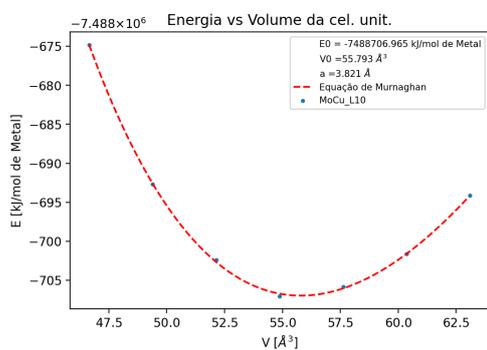
(ed) MoCr - B2



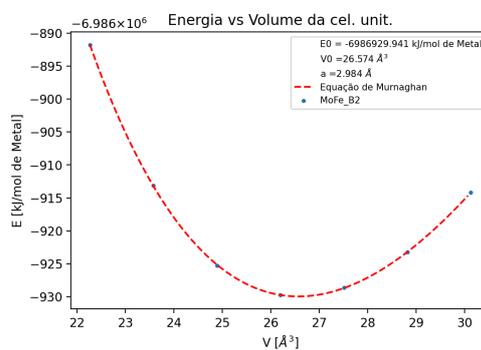
(ee) MoCr - L10



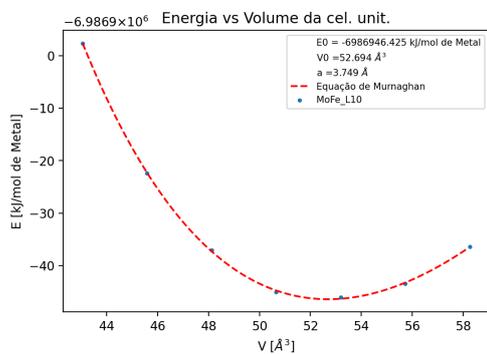
(ef) MoCu - B2



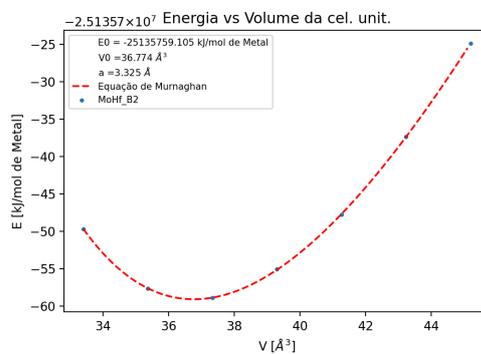
(eg) MoCu - L10



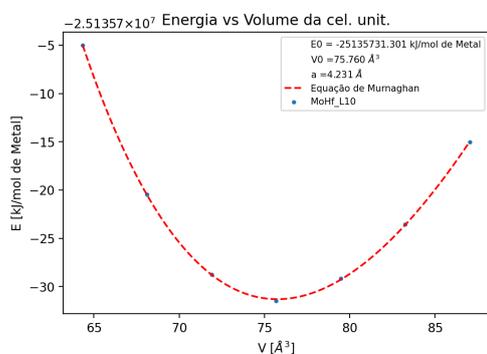
(eh) MoFe - B2



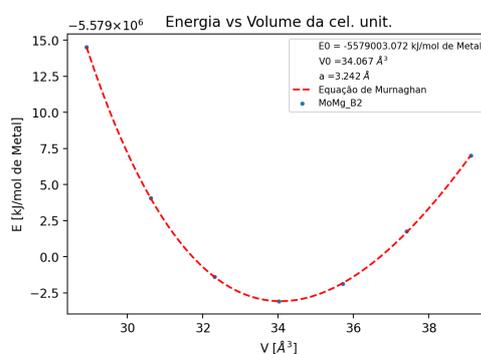
(ei) MoFe - L10



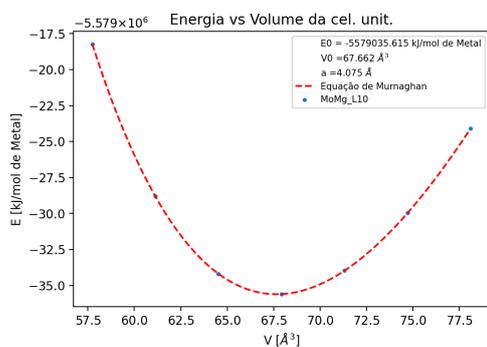
(ej) MoHf - B2



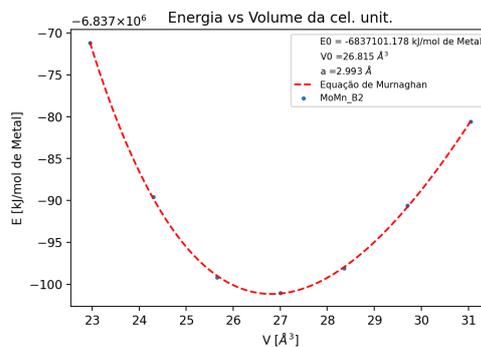
(ek) MoHf - L10



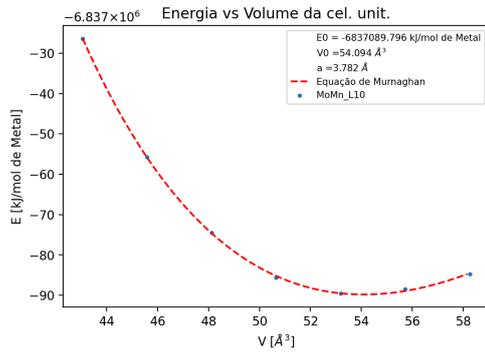
(el) MoMg - B2



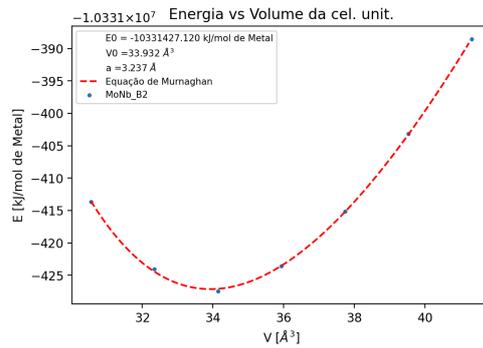
(em) MoMg - L10



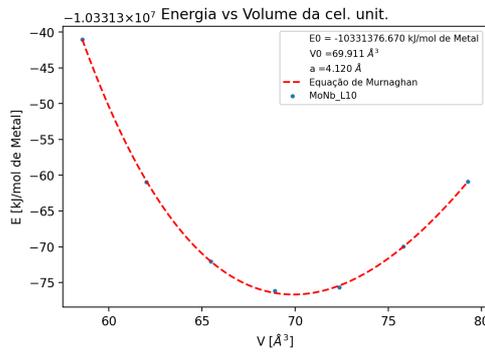
(en) MoMn - B2



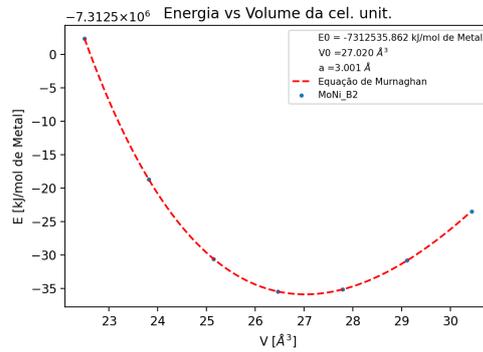
(eo) MoMn - L10



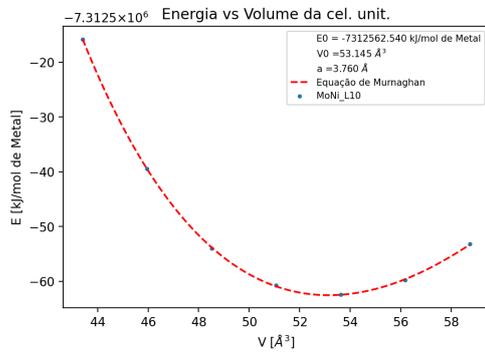
(ep) MoNb - B2



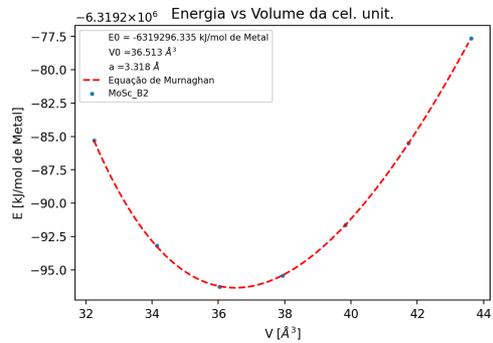
(eq) MoNb - L10



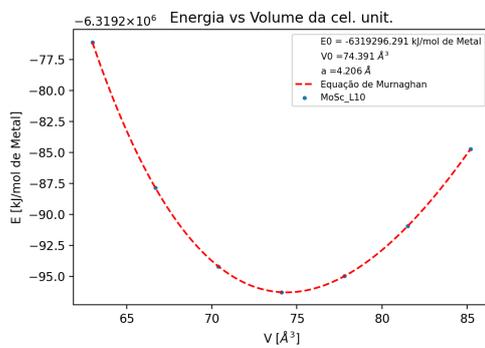
(er) MoNi - B2



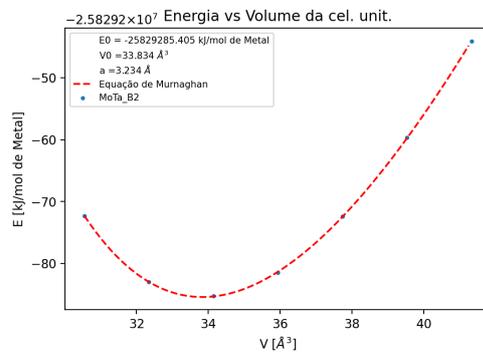
(es) MoNi - L10



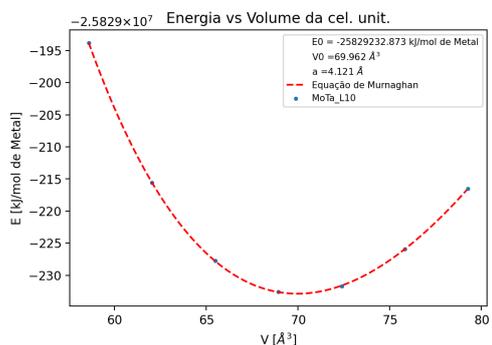
(et) MoSc - B2



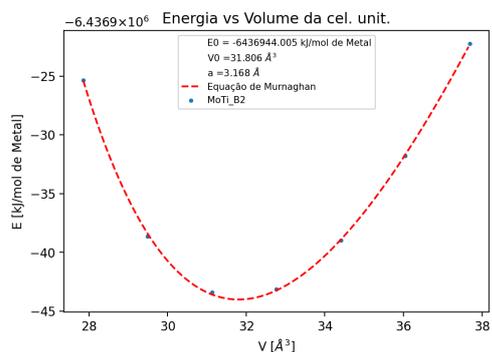
(eu) MoSc - L10



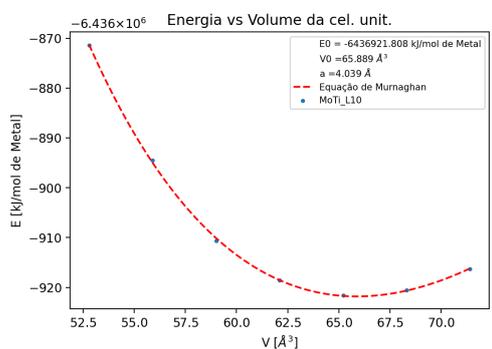
(ev) MoTa - B2



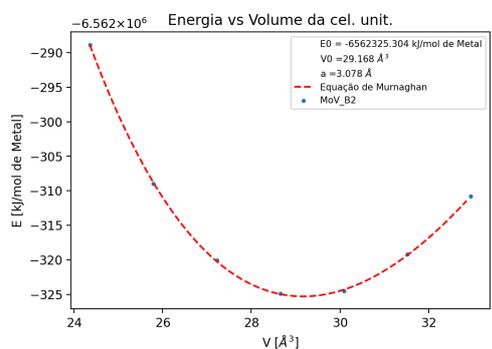
(ew) MoTa - L10



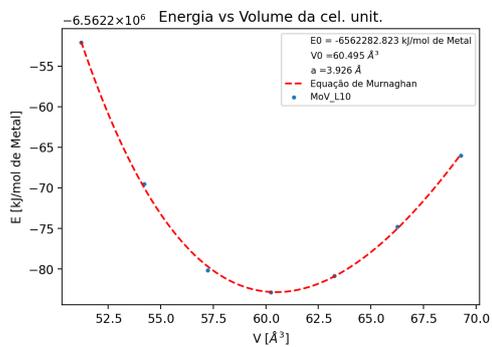
(ex) MoTi - B2



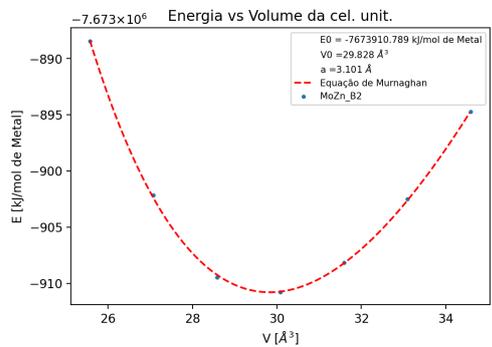
(ey) MoTi - L10



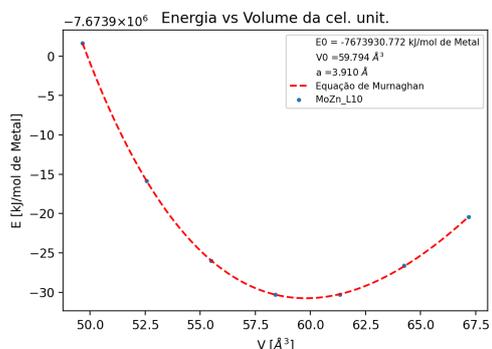
(ez) MoV - B2



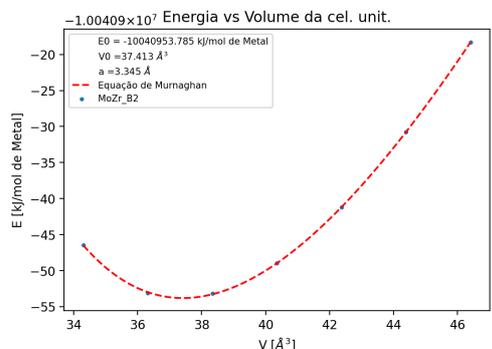
(fa) MoV - L10



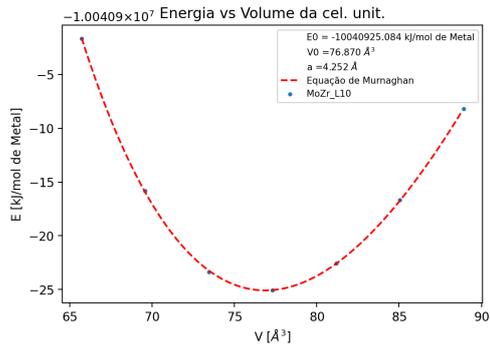
(fb) MoZn - B2



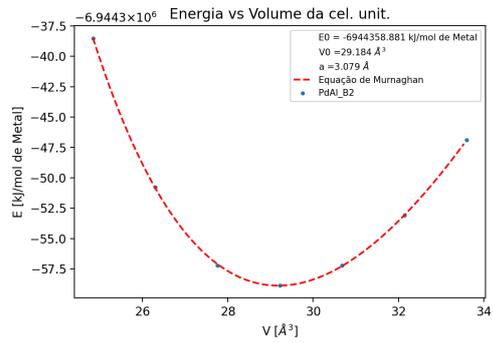
(fc) MoZn - L10



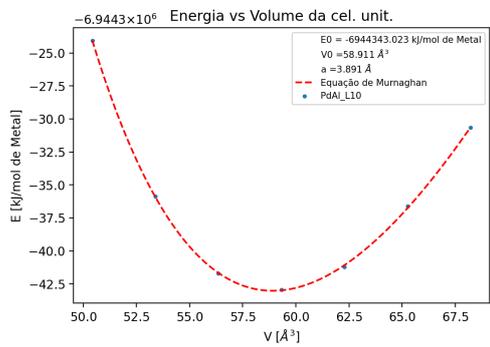
(fd) MoZr - B2



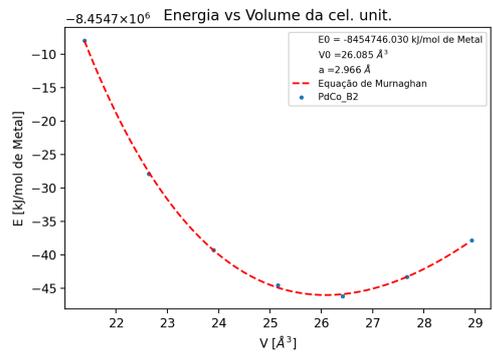
(fe) MoZr - L10



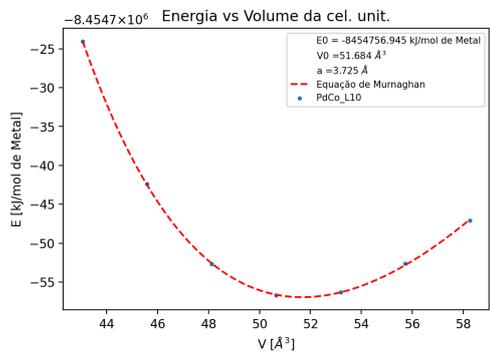
(ff) PdAl - B2



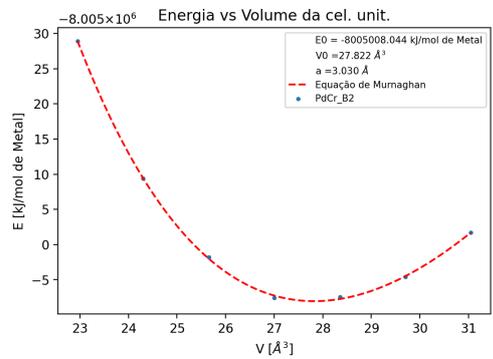
(fg) PdAl - L10



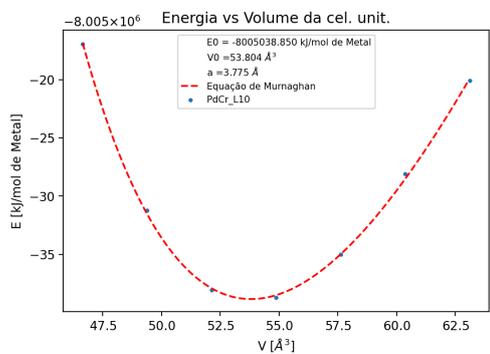
(fh) PdCo - B2



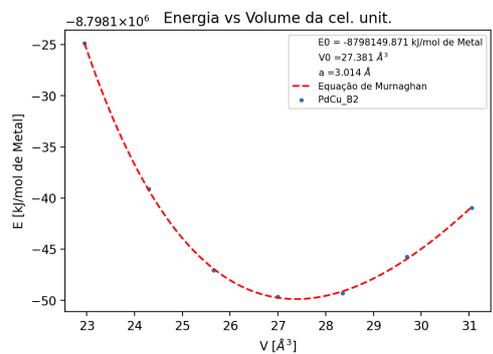
(fi) PdCo - L10



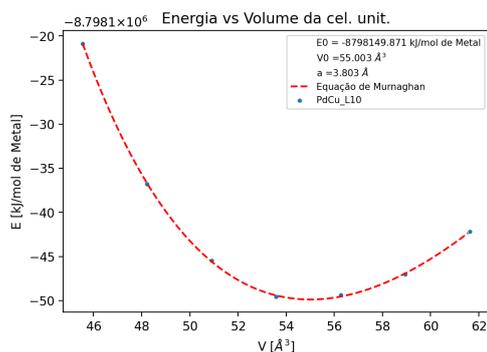
(fj) PdCr - B2



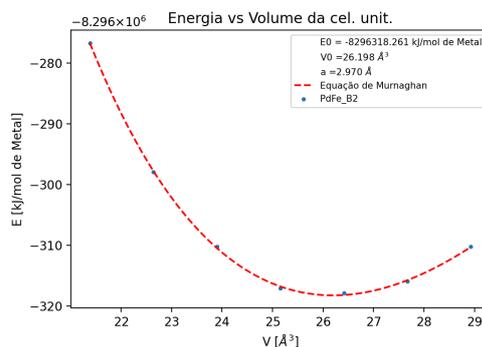
(fk) PdCr - L10



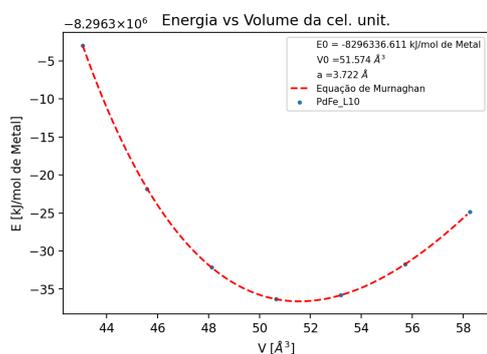
(fl) PdCu - B2



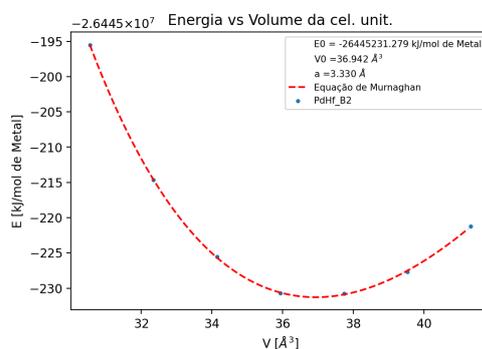
(fm) PdCu - L10



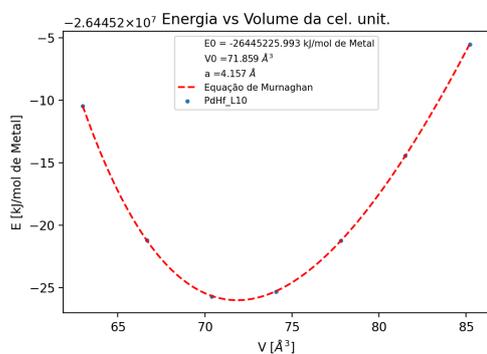
(fn) PdFe - B2



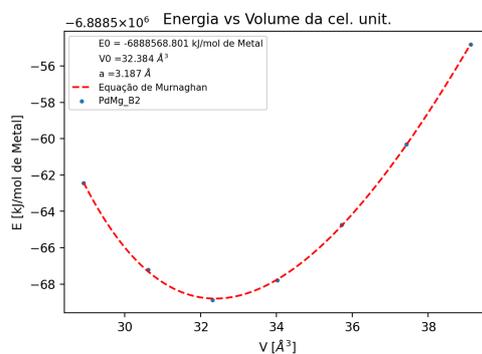
(fo) PdFe - L10



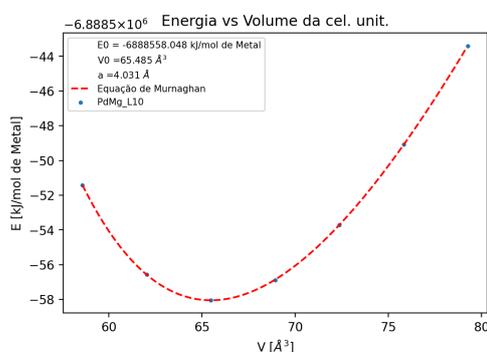
(fp) PdHf - B2



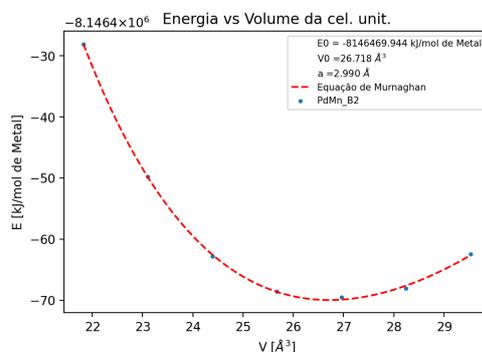
(fq) PdHf - L10



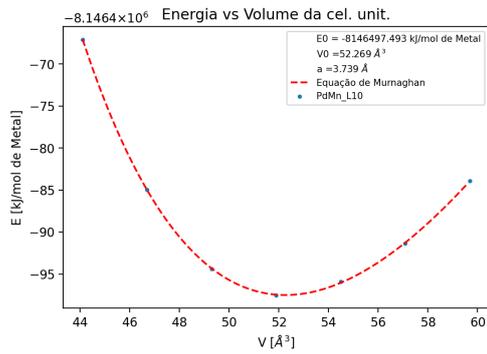
(fr) PdMg - B2



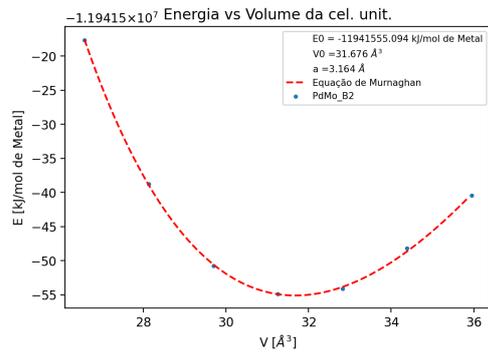
(fs) PdMg - L10



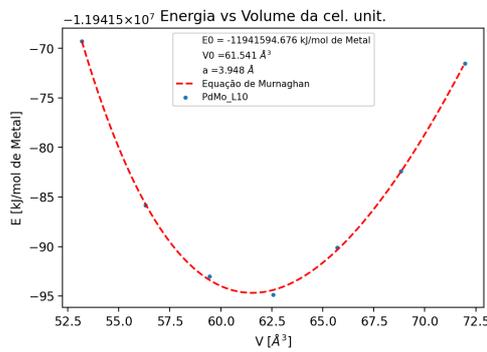
(ft) PdMn - B2



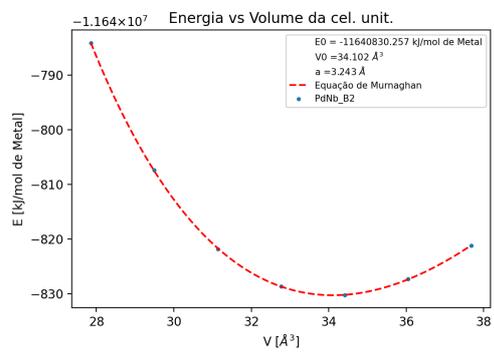
(fu) PdMn - L10



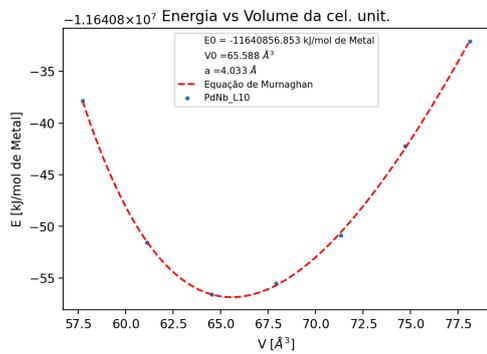
(fv) PdMo - B2



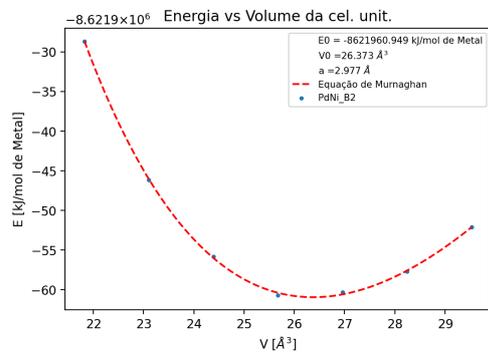
(fw) PdMo - L10



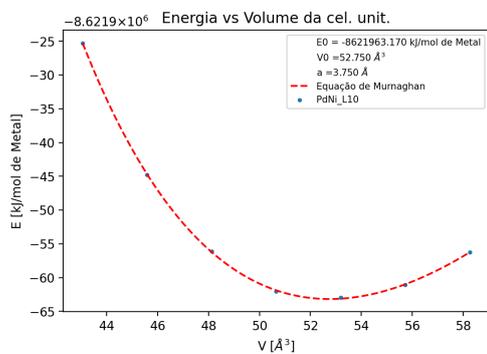
(fx) PdNb - B2



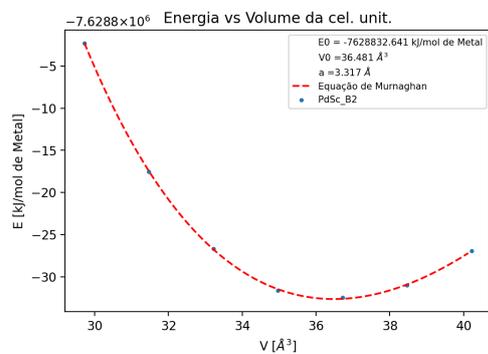
(fy) PdNb - L10



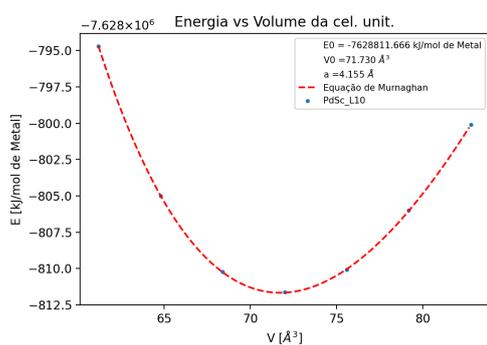
(fz) PdNi - B2



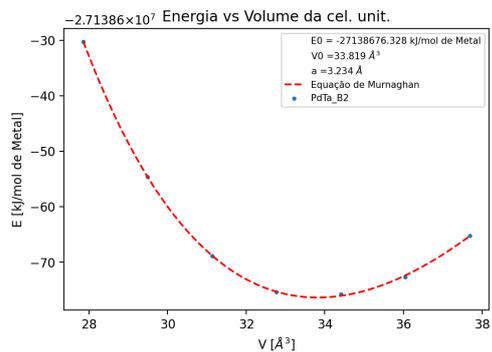
(ga) PdNi - L10



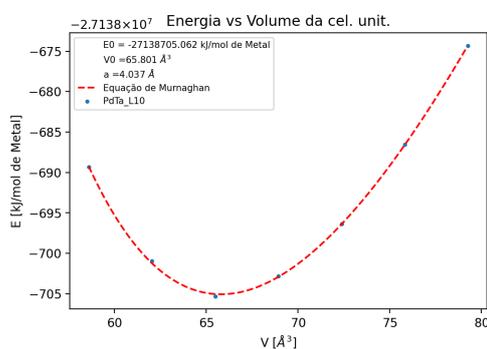
(gb) PdSc - B2



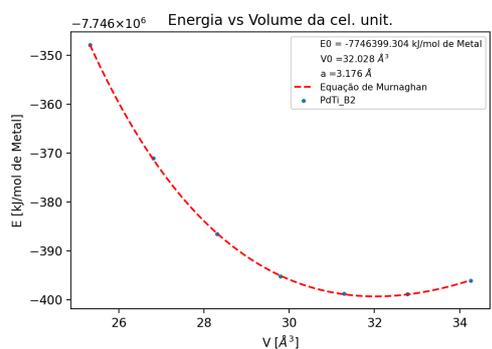
(gc) PdSc - L10



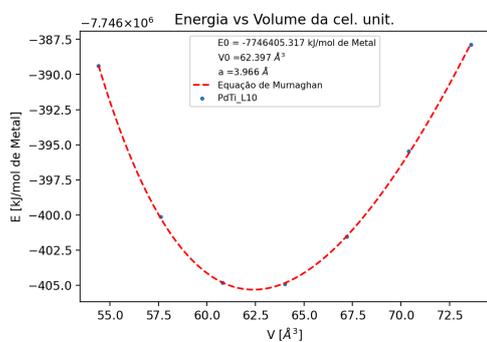
(gd) PdTa - B2



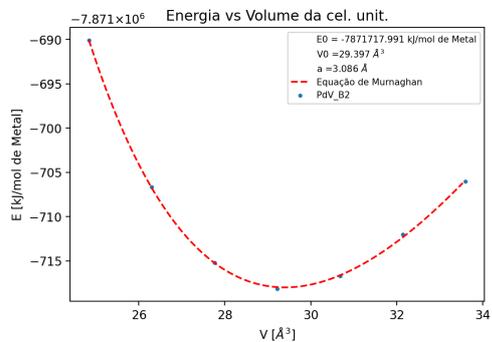
(ge) PdTa - L10



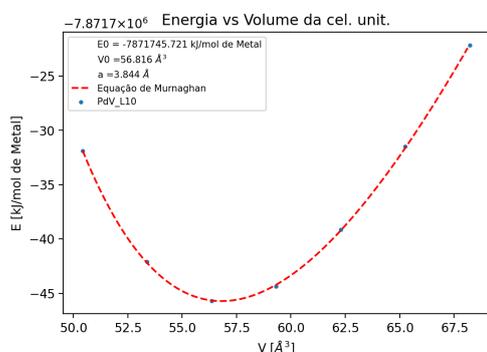
(gf) PdTi - B2



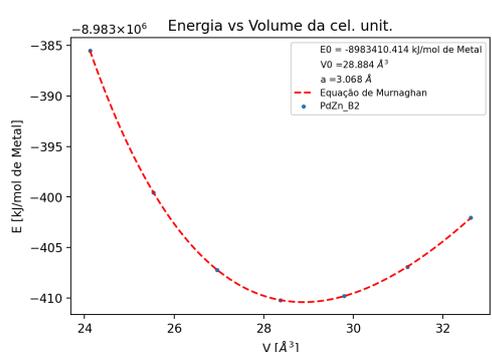
(gg) PdTi - L10



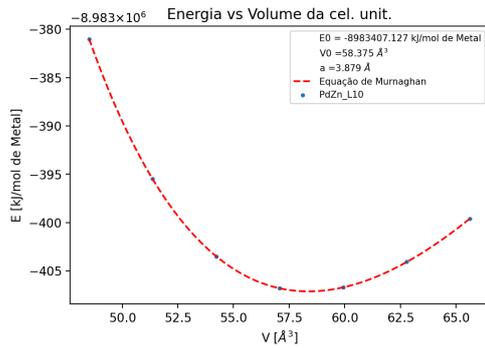
(gh) PdV - B2



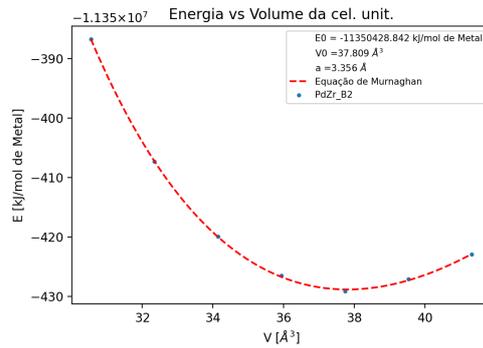
(gi) PdV - L10



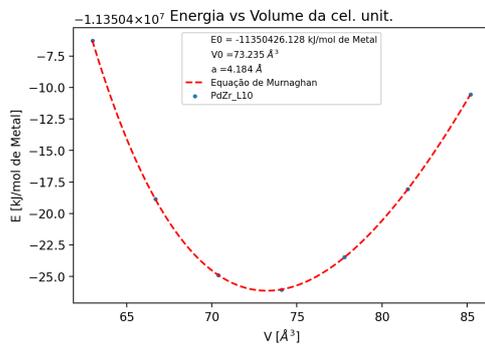
(gj) PdZn - B2



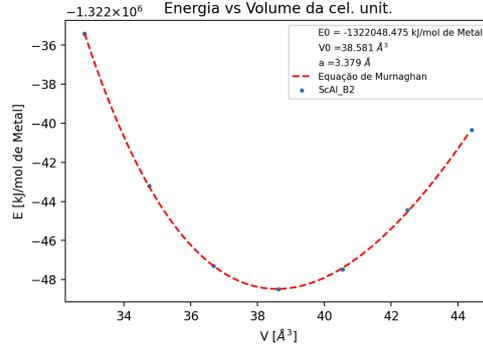
(gk) PdZn - L10



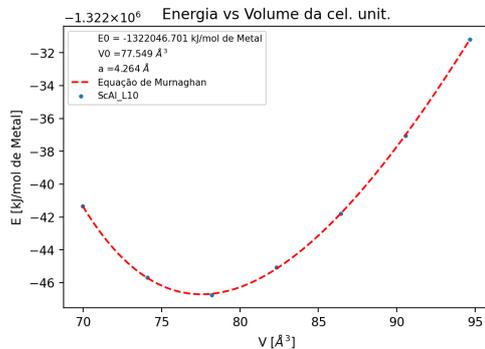
(gl) PdZr - B2



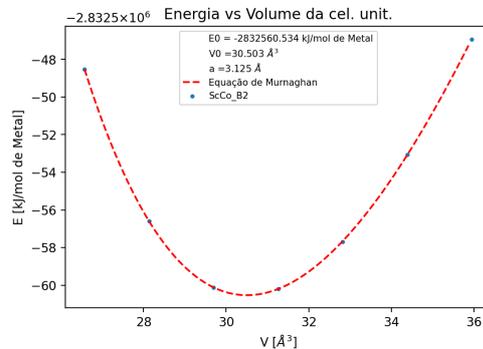
(gm) PdZr - L10



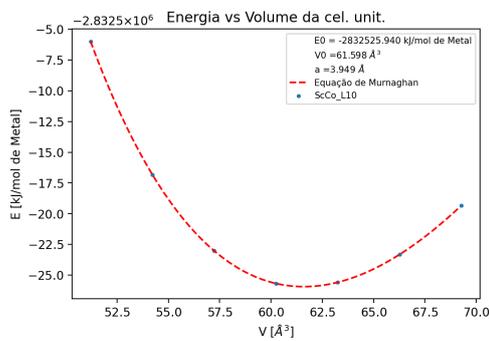
(gn) ScAl - B2



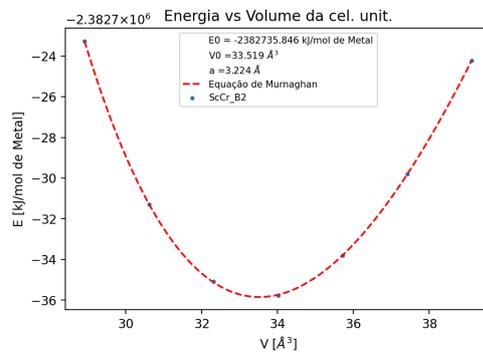
(go) ScAl - L10



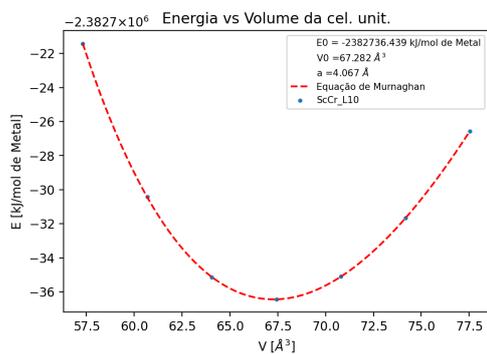
(gp) ScCo - B2



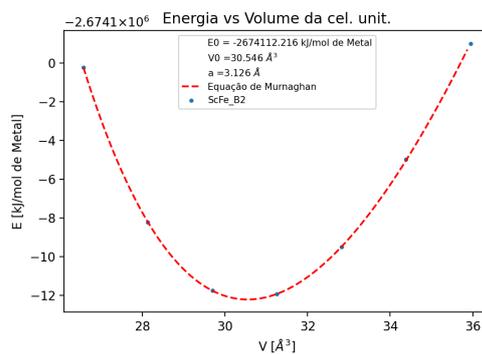
(gq) ScCo - L10



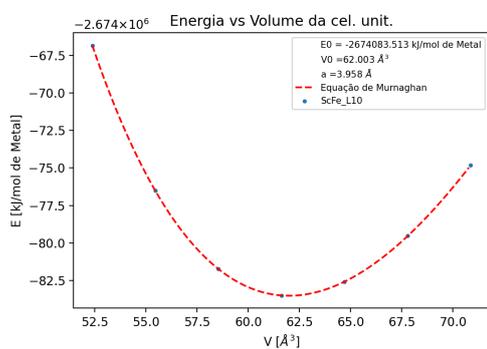
(gr) ScCr - B2



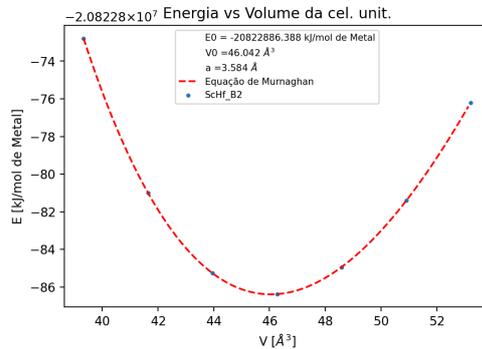
(gs) ScCr - L10



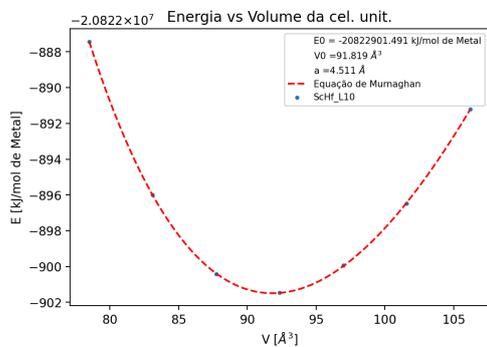
(gt) ScFe - B2



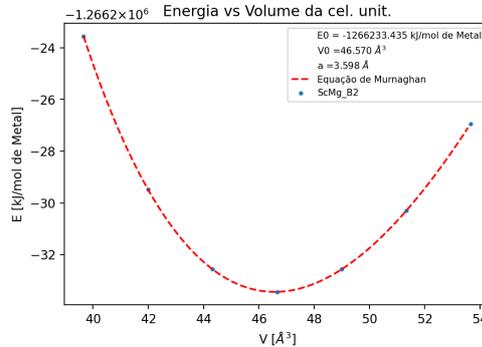
(gu) ScFe - L10



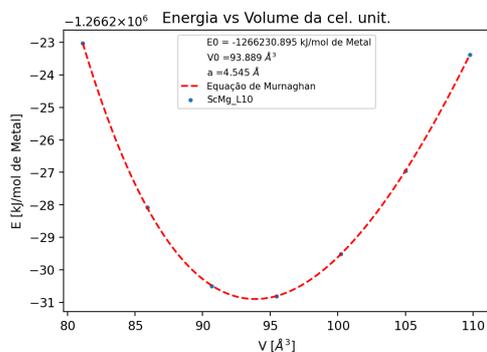
(gv) ScHf - B2



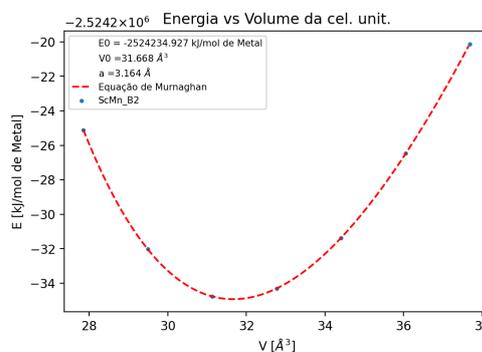
(gw) ScHf - L10



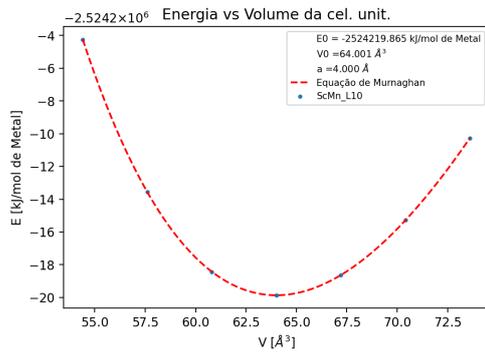
(gx) ScMg - B2



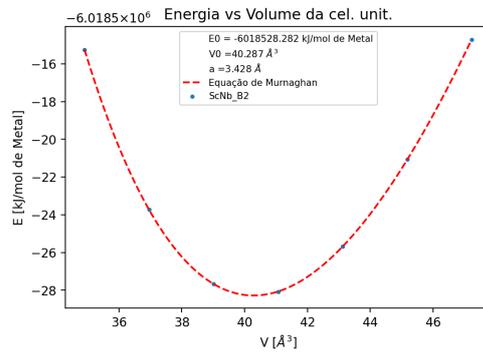
(gy) ScMg - L10



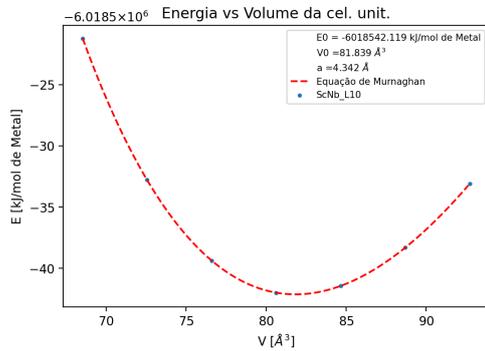
(gz) ScMn - B2



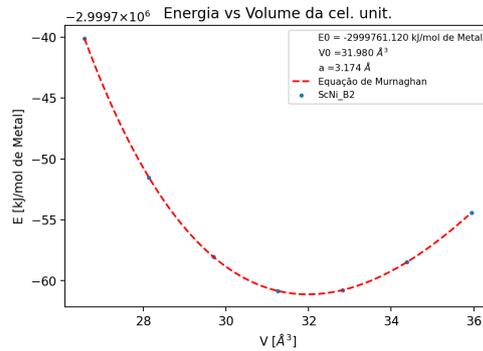
(ha) ScMn - L10



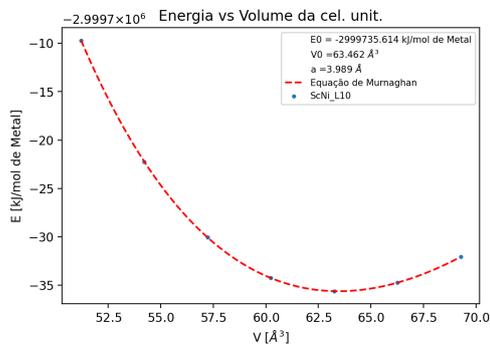
(hb) ScNb - B2



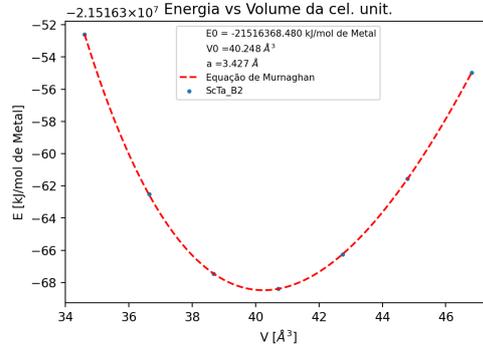
(hc) ScNb - L10



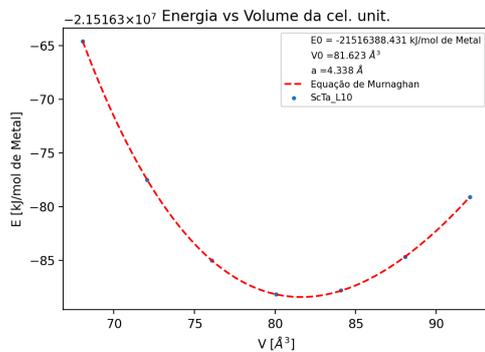
(hd) ScNi - B2



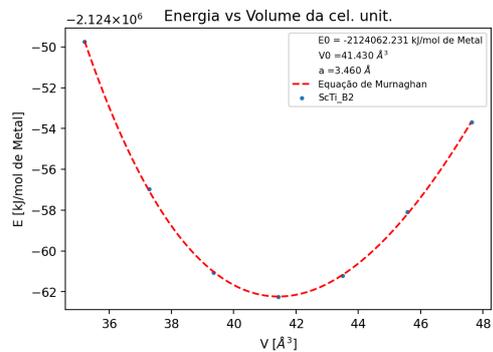
(he) ScNi - L10



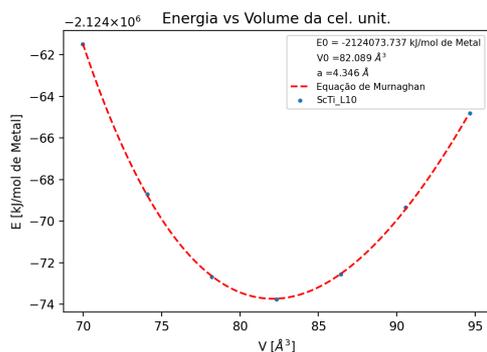
(hf) ScTa - B2



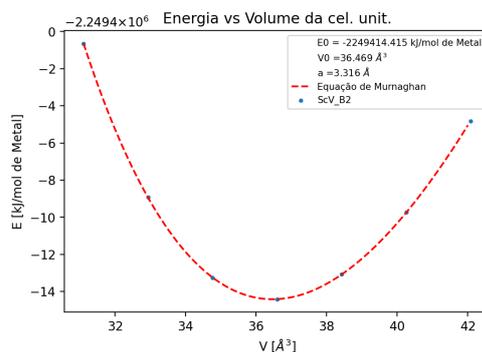
(hg) ScTa - L10



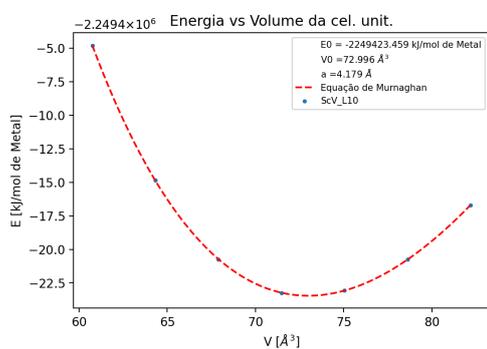
(hh) ScTi - B2



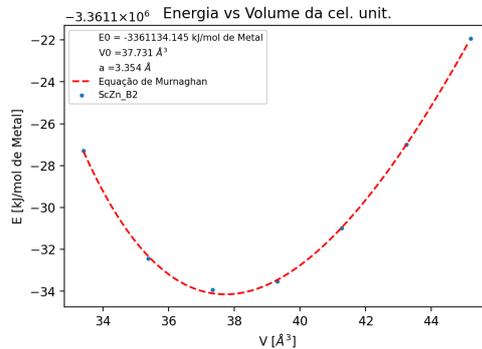
(hi) ScTi - L10



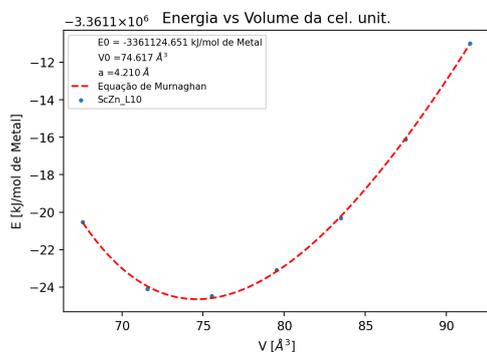
(hj) ScV - B2



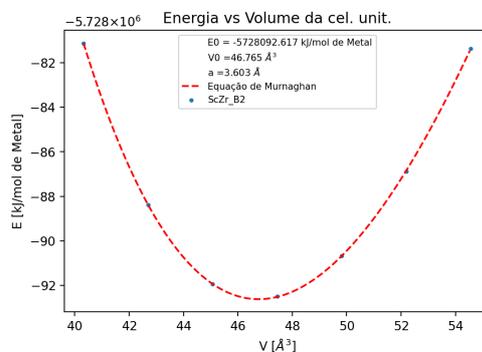
(hk) ScV - L10



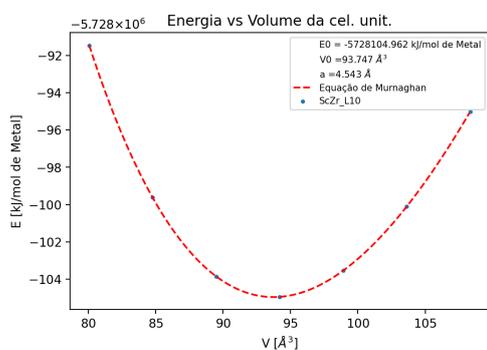
(hl) ScZn - B2



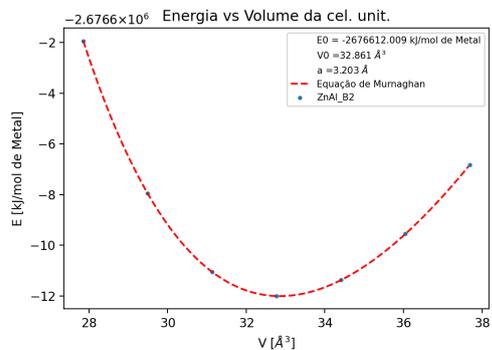
(hm) ScZn - L10



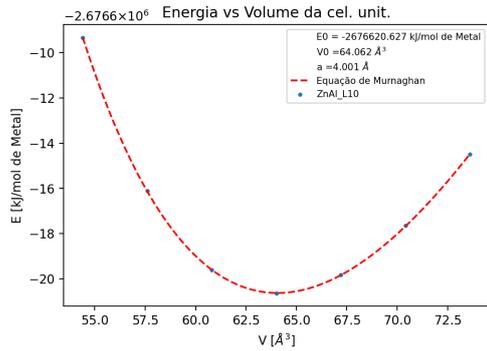
(hn) ScZr - B2



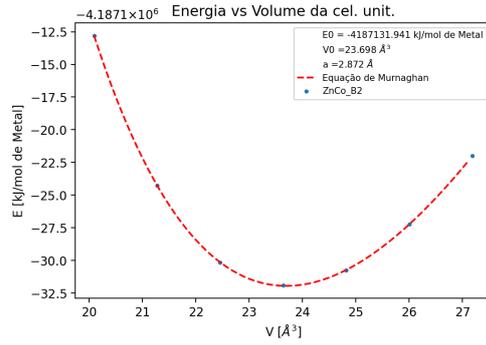
(ho) ScZr - L10



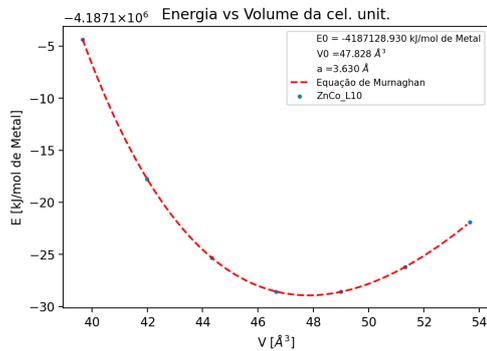
(hp) ZnAl - B2



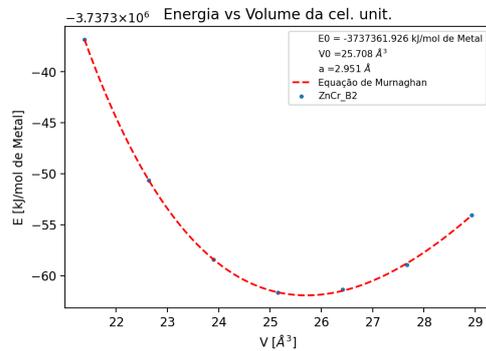
(hq) ZnAl - L10



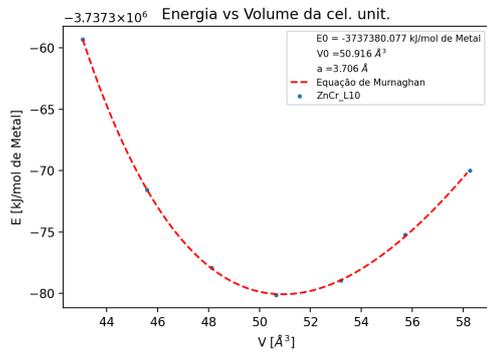
(hr) ZnCo - B2



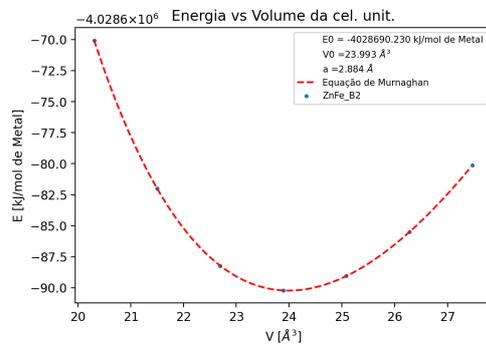
(hs) ZnCo - L10



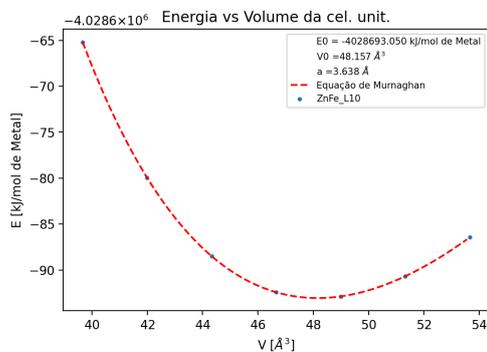
(ht) ZnCr - B2



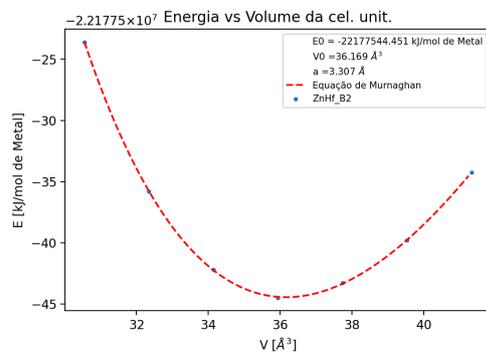
(hu) ZnCr - L10



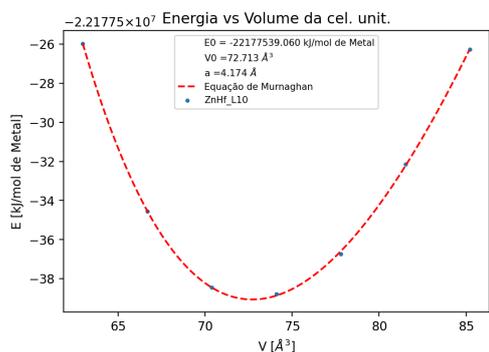
(hv) ZnFe - B2



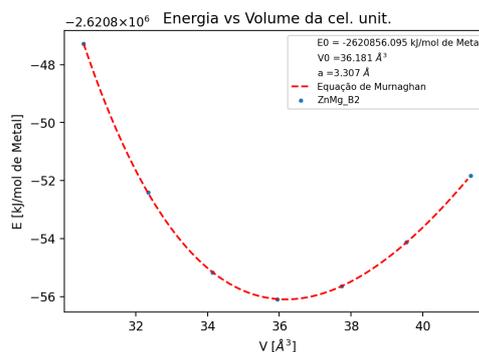
(hw) ZnFe - L10



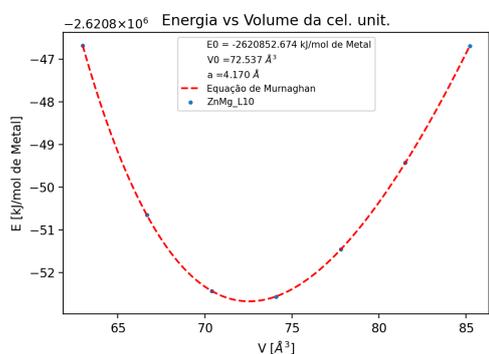
(hx) ZnHf - B2



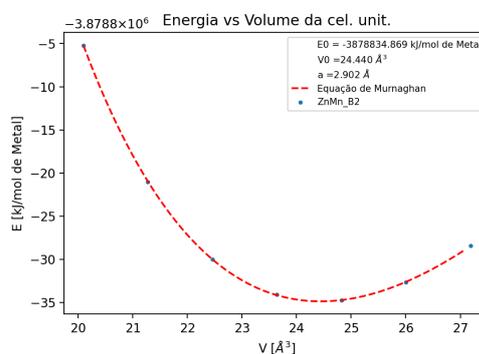
(hy) ZnHf - L10



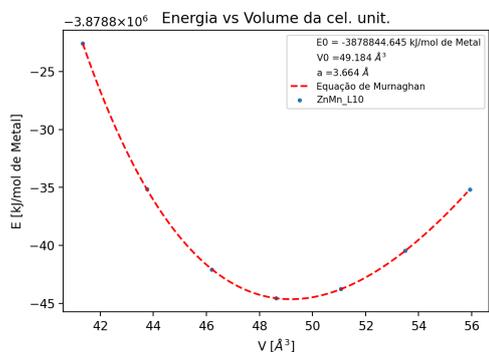
(hz) ZnMg - B2



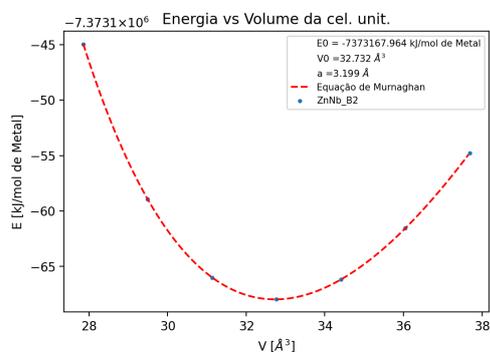
(ia) ZnMg - L10



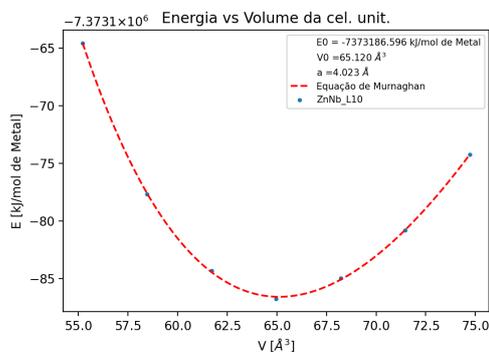
(ib) ZnMn - B2



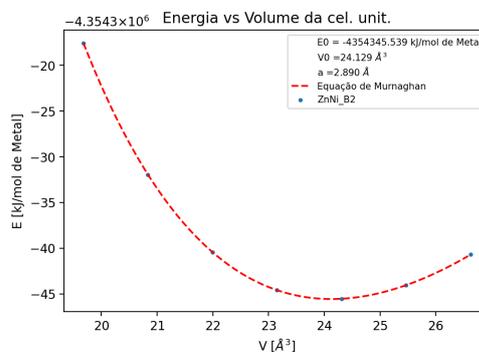
(ic) ZnMn - L10



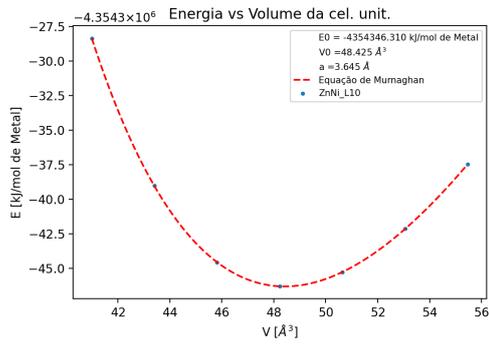
(id) ZnNb - B2



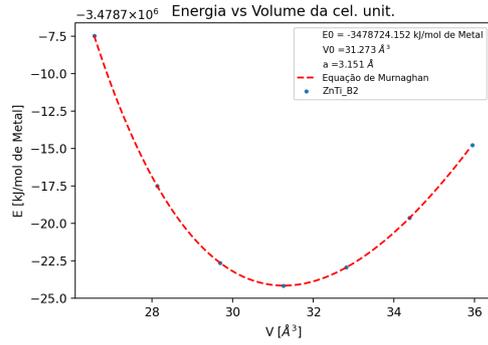
(ie) ZnNb - L10



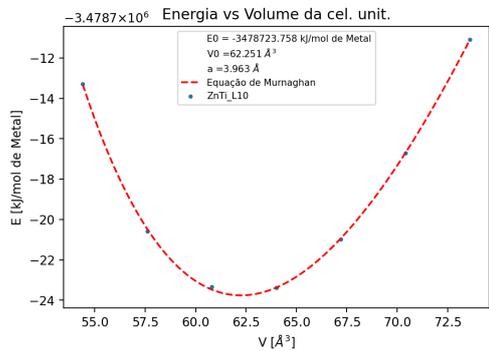
(if) ZnNi - B2



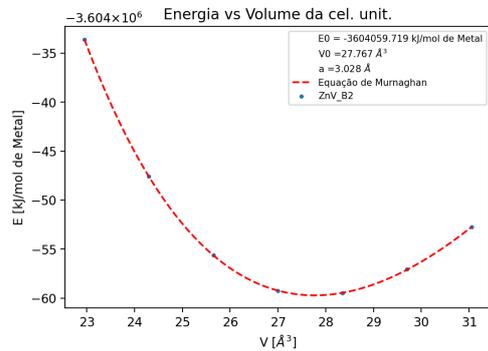
(ig) ZnNi - L10



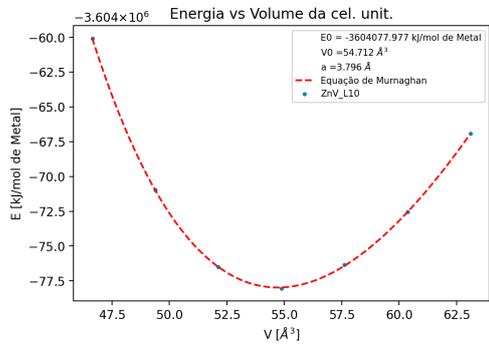
(ih) ZnTi - B2



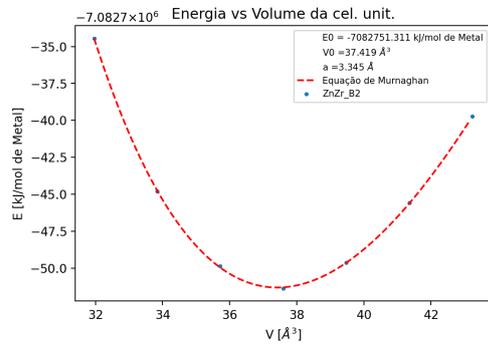
(ii) ZnTi - L10



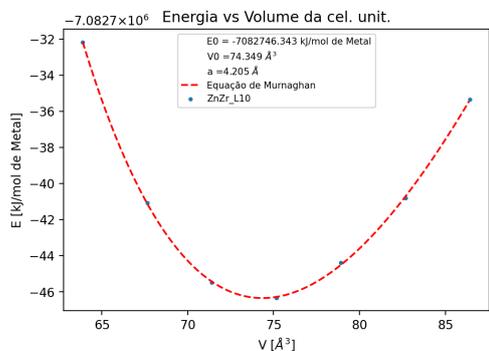
(ij) ZnV - B2



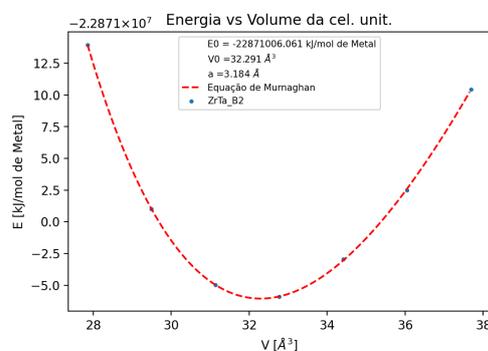
(ik) ZnV - L10



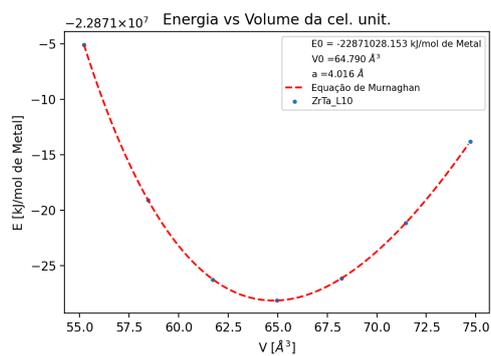
(il) ZnZr - B2



(im) ZnZr - L10



(in) ZrTa - B2



(io) ZrTa - L10

Gráficos das estruturas B2 e L10 para as ligas binárias utilizadas para a obtenção das energias de ligação entre os diferentes elementos.

APÊNDICE B: Código desenvolvido para os cálculos do modelo

```
1
2 #!/usr/bin/env python
3 # coding: utf-8
4
5 import pandas as pd
6 import numpy as np
7 import matplotlib.pyplot as plt
8 from numpy import log as ln
9 from numpy import exp
10 import os
11 import math
12 from itertools import zip_longest
13 from datetime import datetime
14 import time
15
16 hi_values = pd.read_csv('./hi.csv', sep = ';').to_numpy() #importa
    os valores de hi do arquivo csv e transforma em uma matriz
17 bonding_alpha = pd.read_csv('./bonding_alpha.csv', sep = ';').
    to_numpy() #importa os valores da energia de liga o da
    estrutura alpha
18 bonding_delta = pd.read_csv('./bonding_delta.csv', sep = ';').
    to_numpy()#importa os valores da energia de liga o da
    estrutura delta
19 atomic_mass = pd.read_csv('./atomic_mass.csv', sep = ';').to_numpy
    ()#importa os valores da energia de liga o da estrutura delta
20 composition_vetor = np.zeros(len(hi_values)) #cria um vetor de
    zeros com a mesma quantidade de coordenadas que entradas de
    elementos na matriz hi_values
21
22 def truncate(n, decimals=0):
23     multiplier = 10 ** decimals
24     return int(n * multiplier) / multiplier
```

```

25
26
27 def round_up(n, decimals=0):
28     multiplier = 10 ** decimals
29     return math.ceil(n * multiplier) / multiplier
30
31 def Hplat(H1,c_H1,H2,c_H2,T):
32     return float(H2[T][np.where(H2[T]==c_H2)[0],1] - H1[T][np.where
33     (H1[T]==c_H1)[0],1]) / (c_H2 - c_H1)
34
35 def Splat(S1,c_H1,S2,c_H2,T):
36     return float(S2[T][np.where(S2[T]==c_H2)[0],1] - S1[T][np.where
37     (S1[T]==c_H1)[0],1]) / (c_H2 - c_H1)
38
39 def Hconvert(p,mass,cH):
40     p[np.where(p==cH)[0][0],0] = cH * 1.01 *100/ (mass+ cH*1.01)
41     return
42
43 #####Defini o de variaveis#####
44 elements = []
45 composition = []
46 alloy_composition ={}
47 stop = False
48 R = 0.00831446261815324
49 #####Entrada de dados pelo usu rio#####
50 while (stop!= True):
51     y = input("Digite um elemento da liga: ")
52     elements.append(y) #adiciona a entrada y no fim da lista de
53     elementos
54     x = float(input("Digite a composi o do elemento {} na liga:
55     " .format(y)))
56     composition.append(x) #adiciona a entrada w no fim da lista de
57     composi es
58     if y in alloy_composition:
59         alloy_composition[y]= alloy_composition[y] + x

```

```

57     else:
58         alloy_composition[y]=x
59     #if sum(composition)>1:
60         # print("A soma composi o deve ser igual a 1, reinicie o
61         programa")
62     z = input("Deseja adicionar mais um elemento? Y/n ")
63     if z == "n":
64         stop = True
65         print(sum(composition))
66 for i in alloy_composition:
67     alloy_composition[i] = alloy_composition[i]/sum(composition)
68
69
70 #####Informa ao usu rio a liga que foi inserida#####
71 alloy_name = []
72 for element in alloy_composition.keys():
73     alloy_name.append(element)
74     alloy_name.append(str(truncate(alloy_composition[element],3)))
75
76 separator = ''
77 alloy = separator.join(alloy_name)
78 print(f"A liga inserida : {alloy}")
79 #print(f"Com concentra o:{alloy_composition}")
80
81
82 #####C lculo de hi para cada estrutura #####
83
84 for i in elements:
85     composition_vetor[np.where(hi_values==i)[0]] =
86     alloy_composition[i]
87
88 hm_alpha = composition_vetor * hi_values[0:,1]
89 hm_beta = composition_vetor * hi_values[0:,2]
90 hm_delta = composition_vetor * hi_values[0:,3]
91 hm_alpha= sum(hm_alpha)
92 hm_beta = sum(hm_beta)

```

```
92 hm_delta = sum(hm_delta)
93
94
95 #print(hm_alpha)
96 #print(hm_beta)
97 #print(hm_delta)
98
99
100 # In[13]:
101
102
103
104 #####C lculo de H #####
105
106 alpha_vetor = np.zeros([len(bonding_alpha),len(bonding_alpha)])
107
108 for i in elements:
109     for j in elements:
110         alpha_vetor[np.where(bonding_alpha==i)[0],np.where(
111             bonding_alpha==j)[0]]=alloy_composition[i]*alloy_composition[j]
112
113 average_bond_alpha = alpha_vetor * bonding_alpha[0:,1:]
114 average_bond_alpha = sum(sum(average_bond_alpha))
115
116 E_total_alpha = average_bond_alpha * 4
117
118 delta_vetor = np.zeros([len(bonding_delta),len(bonding_delta)])
119
120 for i in elements:
121     for j in elements:
122         delta_vetor[np.where(bonding_delta==i)[0],np.where(
123             bonding_delta==j)[0]]=alloy_composition[i]*alloy_composition[j]
124
125 average_bond_delta = delta_vetor * bonding_delta[0:,1:]
126 average_bond_delta = sum(sum(average_bond_delta))
127 E_total_delta = average_bond_delta * 6
```

```

127
128 H_delta = E_total_delta - E_total_alpha
129 H_alpha = 0
130 H_beta = 0
131 #print(H_delta)
132
133
134 ##### Input de temperatura
135 temperature = []
136 stop = False
137 while (stop!= True):
138     y = float(input("Digite uma temperatura em graus celsius para
os c lculos: "))
139     y= y + 273.15
140     temperature.append(y) #adiciona a entrada y no fim da lista de
elementos
141     z = input("Deseja adicionar mais uma temperatura? Y/n ")
142     if z == "n":
143         stop = True
144
145 temperature.sort() #organiza as temperaturas em ordem crescente
146 #####C lculos termodin micos#####
147 val_S_alpha = {}
148 val_S_beta = {}
149 val_S_delta = {}
150 val_H_alpha = {}
151 val_H_beta = {}
152 val_H_delta = {}
153 val_G_alpha = {}
154 val_G_beta = {}
155 val_G_delta = {}
156 val_mu_M_alpha = {}
157 val_mu_M_beta = {}
158 val_mu_M_delta = {}
159 val_mu_H_alpha = {}
160 val_mu_H_beta = {}
161 val_mu_H_delta = {}

```

```

162 valoresalpha = {}
163 valoresbeta = {}
164 valoresdelta = {}
165 p_total = {}
166 p_total_wt = {}
167 p_plat = {}
168 H_plat = {}
169 S_plat = {}
170 verify_equilibrium = []
171 C_H = []
172 p_equilibrium_ad = []
173 p_equilibrium_ab = []
174 p_equilibrium_bd = []
175 p_notequilibrium = []
176 ch = input("Digite o passo na composi o de hidrog nio da liga (
        Default:0.0005) \ninput:")
177 if not ch or ch == None:
178     cH_step = 0.0005
179 else:
180     cH_step = float(ch)
181
182 print("Descreva os parametros r e theta para cada estrutura no
        formato: \ntheta_alpha,r_alpha,theta_beta,r_beta,theta_delta,
        r_delta")
183 rt = (input("O padr o      6,7,4,2,2,1 \ninput:"))
184 if not rt or rt == None:
185     rt = (6,7,4,2,2,1)
186 if type(rt) == str:
187     rt = rt.split(',')
188
189 ##### Setting the Hydrogen presentation in Graphs#####
190 Hydrogen_presentation = '0'
191 while any(i for i in range(1,4) if i == int(Hydrogen_presentation))
        != True:
192     Hydrogen_presentation = input('1:c_H (Default) | 2: wt% H | 3:
        c_H and wt% H \ninput:')
193     if not Hydrogen_presentation or Hydrogen_presentation == None:

```

```

194     Hydrogen_presentation = '1'
195     #####
196
197
198 #
199     #####
200
201 # Cria pasta para salvar os arquivos da liga e arquivo txt com os
202     dados principais #
203 #
204     #####
205
206 if not os.path.isdir(f'./{alloy}'): #Verifica se a pasta j existe
207     e se n o existe cria ela com a proxima linha
208     os.mkdir(f'./{alloy}')
209
210 dnow = datetime.now()
211 my_path = os.path.abspath(alloy)
212 my_file = f'./{alloy}.txt'
213 if not os.path.isfile(os.path.join(my_path, my_file)):
214     file = open(os.path.join(my_path, my_file), 'w+')
215     file.close()
216 file = open(os.path.join(my_path, my_file), 'a')
217 file.write(f"Alloy: {alloy} | {dnow.strftime('%d/%m/%Y %H:%M')} | \
218     n")
219 file.write(f"hm[alpha]= {hm_alpha} | hm[beta]= {hm_beta} | hm[delta
220     ]= {hm_delta} |\n")
221 file.write(f"HM[alpha] = {H_alpha}| HM[beta] = {H_beta} | HM[delta
222     ] = {H_delta} |\n")
223 file.write(" Values for Theta and r is in the format: structure(
224     theta,r)\n")
225 file.write(f" alpha{float(rt[0]),float(rt[1])}, beta{float(rt[2]),
226     float(rt[3])}, delta{float(rt[4]),float(rt[5])}\n")
227 file.write(f"cH step: {cH_step}\n")
228 file.close()
229 #
230     #####

```

```
219
220
221
222 ##### Begin of calculations #####
223 begin = time.time()
224 for i in range(1,int(2/cH_step +1)):
225     C_H.append(truncate(cH_step*i,4))
226 for T in temperature:
227     alpha_S = []
228     beta_S = []
229     delta_S = []
230     alpha_H = []
231     beta_H = []
232     delta_H = []
233     alpha_G = []
234     beta_G = []
235     delta_G = []
236     alpha_mu_M = []
237     beta_mu_M = []
238     delta_mu_M = []
239     alpha_mu_H = []
240     beta_mu_H = []
241     delta_mu_H = []
242     for c_H in C_H:
243         #c_H = 0.001 * c_H
244
245         t_alpha = float(rt[0])
246         r_alpha = float(rt[1])
247         t_beta = float(rt[2])
248         r_beta = float(rt[3])
249         t_delta = float(rt[4])
250         r_delta = float(rt[5])
251         t=T/1000
252         #A = 33.066178
253         #B = -11.363417
254         #C = 11.432816
```

```

255     #D = -2.772874
256     #E = -0.158558
257     #G = 172.707974
258     #S_0 = A*ln(t) + B*t + C*(t**2)/2 + D*(t**3)/3 - E/(2*t**2)
+ G
259     #S_0 = S_0/1000 #kJ/mol
260     S_0 = R * 7/2 * (1 + ln(T/9.2)) # Calculation of Hydrogen
standard entropy
261
262
263     if (c_H < t_alpha/r_alpha):
264         dH_m_alpha = H_alpha + c_H * hm_alpha
265         dS_m_alpha = - R * (c_H * ln(c_H/(t_alpha - (c_H*(
r_alpha - 1))))+ (t_alpha - r_alpha * c_H)*ln((t_alpha - r_alpha
* c_H)/(t_alpha - (c_H*(r_alpha-1))))) - (c_H/2)*S_0
266         dG_m_alpha = dH_m_alpha - T * dS_m_alpha
267         mu_H_alpha = hm_alpha - T * (- R* ln(c_H*(t_alpha - (
c_H*(r_alpha-1)))*(r_alpha-1)/(t_alpha - (c_H*r_alpha)))*
r_alpha)- S_0/2)
268         mu_M_alpha = dG_m_alpha - c_H * mu_H_alpha
269         alpha_H.append([c_H, dH_m_alpha])
270         alpha_S.append([c_H, dS_m_alpha])
271         alpha_G.append([c_H, dG_m_alpha])
272         alpha_mu_H.append([c_H, mu_H_alpha])
273         alpha_mu_M.append([c_H, mu_M_alpha])
274         valoresalpha[c_H] = [mu_H_alpha, mu_M_alpha]
275
276
277
278     if (c_H < t_beta/r_beta):
279         dH_m_beta = H_beta + c_H * hm_beta
280         dS_m_beta = - R * (c_H * ln(c_H/(t_beta - (c_H*(r_beta
-1))))+ (t_beta - r_beta * c_H)*ln((t_beta - r_beta * c_H)/(
t_beta - (c_H*(r_beta-1))))) - (c_H/2)*S_0
281         dG_m_beta = dH_m_beta - T * dS_m_beta
282         mu_H_beta = hm_beta - T * (- R* ln(c_H*(t_beta - (c_H*(
r_beta-1)))*(r_beta-1)/(t_beta - (c_H*r_beta)))*r_beta)- S_0/2)

```

```

283     mu_M_beta = dG_m_beta - c_H * mu_H_beta
284     beta_H.append([c_H,dH_m_beta])
285     beta_S.append([c_H,dS_m_beta])
286     beta_G.append([c_H,dG_m_beta])
287     beta_mu_H.append([c_H,mu_H_beta])
288     beta_mu_M.append([c_H,mu_M_beta])
289     valoresbeta[c_H] = [mu_H_beta,mu_M_beta]
290
291
292     if (c_H < t_delta/r_delta):
293         dH_m_delta = H_delta + c_H * hm_delta
294         dS_m_delta = - R * (c_H * ln(c_H/(t_delta - (c_H*(
r_delta-1))))+ (t_delta - r_delta * c_H)*ln((t_delta - r_delta *
c_H)/(t_delta - (c_H*(r_delta-1)))) - (c_H/2)*S_0
295         dG_m_delta = dH_m_delta - T * dS_m_delta
296         mu_H_delta = hm_delta - T * (- R* ln(c_H*(t_delta - (
c_H*(r_delta-1)))*(r_delta-1)/(t_delta - (c_H*r_delta))*(
r_delta)- S_0/2)
297         mu_M_delta = dG_m_delta - c_H * mu_H_delta
298         delta_H.append([c_H,dH_m_delta])
299         delta_S.append([c_H,dS_m_delta])
300         delta_G.append([c_H,dG_m_delta])
301         delta_mu_H.append([c_H,mu_H_delta])
302         delta_mu_M.append([c_H,mu_M_delta])
303         valoresdelta[c_H] = [mu_H_delta,mu_M_delta]
304
305
306
307
308     val_S_alpha[T]= np.array(alpha_S)
309     val_S_beta[T]= np.array(beta_S)
310     val_S_delta[T]= np.array(delta_S)
311     val_H_alpha[T]= np.array(alpha_H)
312     val_H_beta[T]= np.array(beta_H)
313     val_H_delta[T]= np.array(delta_H)
314     val_G_alpha[T]= np.array(alpha_G)
315     val_G_beta[T]= np.array(beta_G)

```

```

316     val_G_delta[T]= np.array(delta_G)
317     val_mu_H_alpha[T]= np.array(alpha_mu_H)
318     val_mu_H_beta[T]= np.array(beta_mu_H)
319     val_mu_H_delta[T]= np.array(delta_mu_H)
320     val_mu_M_alpha[T]= np.array(alpha_mu_M)
321     val_mu_M_beta[T]= np.array(beta_mu_M)
322     val_mu_M_delta[T]= np.array(delta_mu_M)
323
324
325     ##### Searching phase equilibrium #####3
326     saveab = []
327     savead = []
328     savebd = []
329     converge = 0.01
330     for c_H1 in C_H:
331         for c_H2 in C_H:
332             if (valoresalpha.__contains__(c_H1) and valoresbeta.
__contains__(c_H2) and ((c_H1 < c_H2))):
333                 if (truncate(abs(valoresalpha[c_H1][0] -
valoresbeta[c_H2][0]),2)<=converge and truncate(abs(valoresalpha
[c_H1][1] - valoresbeta[c_H2][1]),2)<=converge):
334                     saveab.append([c_H1,c_H2])
335                 if (valoresbeta.__contains__(c_H1) and valoresdelta.
__contains__(c_H2)):
336                     if (truncate(abs(valoresbeta[c_H1][0] -
valoresdelta[c_H2][0]),2)<=converge and truncate(abs(valoresbeta
[c_H1][1] - valoresdelta[c_H2][1]),2)<=converge):
337                         savebd.append([c_H1,c_H2])
338                 if (valoresalpha.__contains__(c_H1) and valoresdelta.
__contains__(c_H2) and ((c_H1 < c_H2)>0.01)):
339                     if (truncate(abs(valoresalpha[c_H1][0] -
valoresdelta[c_H2][0]),2)<=converge and truncate(abs(
valoresalpha[c_H1][1] - valoresdelta[c_H2][1]),2)<=converge):
340                         savead.append([c_H1,c_H2])
341
342     equilibrium_all = {'ab': saveab, 'ad' : savead, 'bd' : savebd}
343     if not savead and any(saveab):

```

```

344     equilibrium_real = 'ab'
345 elif not saveab and any(savead):
346     equilibrium_real = 'ad'
347 elif not saveab and not savead:
348     equilibrium_real = 'non equilibrium'
349 else:
350     if any(savebd) and saveab[0][1] > savebd[0][0]:
351         equilibrium_real = 'ad'
352     else:
353         minimo = min(saveab[0][0], savead[0][0])
354         if minimo == saveab[0][0]:
355             equilibrium_real = 'ab'
356         if minimo == savead[0][0]:
357             equilibrium_real = 'ad'
358
359 print(equilibrium_real)
360 verify_equilibrium.append(equilibrium_real)
361 p_1 = []
362 p_2 = []
363 p_3 = []
364 p_4 = []
365 p_5 = []
366 P_0 = 1
367 lim_delta = 1.99
368
369 if H_delta < 0:
370     if not equilibrium_all['ab'] and not equilibrium_all['ad']:
371         for c_H in C_H:
372             if c_H < t_alpha/r_alpha: #intervalo fase alpha
373                 mu_alpha = float(val_mu_H_alpha[T][np.where(
val_mu_H_alpha[T]==c_H)[0],1])
374                 lnPH = 2* mu_alpha/(R*T)
375                 peq = P_0*exp(lnPH)
376                 p_1.append([c_H, peq])
377                 p_2 = [[c_H, peq]]
378                 p_3 = [[c_H, peq]]
379                 p_4 = [[c_H, peq]]

```

```

380         p_5 = [[c_H,peq]]
381     p_total[T] = np.array(p_1)
382     p_1_wt = p_1
383     p_2_wt = p_2
384     p_3_wt = p_3
385     p_4_wt = p_4
386     p_5_wt = p_5
387     p_1 = np.array(p_1)
388     H_plat_ab = val_H_alpha[T][np.where(val_H_alpha[T]==
t_alpha/(2*r_alpha))[0],1]
389     S_plat_ab = val_S_alpha[T][np.where(val_S_alpha[T]==
t_alpha/(2*r_alpha))[0],1]
390     H_plat[T] = [H_plat_ab,None]
391     S_plat[T] = [S_plat_ab,None]
392     p_notequilibrium.append([1/T,ln(p_1[np.where(p_1
[0:,0]==t_alpha/(2*r_alpha))[0],1])])
393
394
395
396     if any(equilibrium_all['ab']) or any(equilibrium_all['ad']):
397
398         if equilibrium_real == 'ad':
399             #####Reta tangentes dG#####
400             line_ad_coef = [valoresalpha[equilibrium_all[
equilibrium_real][0][0]][0],valoresalpha[equilibrium_all[
equilibrium_real][0][0]][1]]
401             line_ad = []
402             for c_H in C_H:
403                 if equilibrium_all[equilibrium_real][0][0] - .05 <=
c_H <= equilibrium_all[equilibrium_real][0][1] + 0.05:
404                     y = line_ad_coef[0] * c_H + line_ad_coef[1]
405                     x = c_H
406                     line_ad.append([x,y])
407             line_ad = np.array(line_ad)
408             #####C lculo PCI#####3
409             for c_H in C_H:
410                 if c_H < equilibrium_all[equilibrium_real][0][0]: #

```

```

intervalo fase alpha
411         mu_alpha = float(val_mu_H_alpha[T][np.where(
val_mu_H_alpha[T]==c_H)[0],1])
412         ln pH = 2* mu_alpha/(R*T)
413         peq = P_0*exp(ln pH)
414         p_1.append([c_H,peq])
415         if equilibrium_all[equilibrium_real][0][0]<c_H<
equilibrium_all[equilibrium_real][0][1]: # plato alpha - delta
416             p_2.append([c_H,peq])
417         if equilibrium_all[equilibrium_real][0][1]<c_H<
C_H[-1]: # intervalo fase delta
418             #print(c_H,val_mu_H_delta[T][np.where(
val_mu_H_delta[T]==c_H)[0],1])
419             mu_delta = float(val_mu_H_delta[T][np.where(
val_mu_H_delta[T]==c_H)[0],1])
420             ln pH = 2* mu_delta/(R*T)
421             peq = P_0*exp(ln pH)
422             p_3.append([c_H,peq])
423             p_4 = [[c_H,peq]]
424             p_5 = [[c_H,peq]]
425         p_total[T] = np.array(p_1 + p_2 + p_3)
426         p_1_wt = p_1
427         p_2_wt = p_2
428         p_3_wt = p_3
429         p_4_wt = p_4
430         p_5_wt = p_5
431         H_plat[T] = [Hplat(val_H_alpha, equilibrium_all[
equilibrium_real][0][0], val_H_delta, equilibrium_all[
equilibrium_real][0][1], T), None]
432         S_plat[T] = [Splat(val_S_alpha, equilibrium_all[
equilibrium_real][0][0], val_S_delta, equilibrium_all[
equilibrium_real][0][1], T), None]
433         p_plateau_ad = exp(2* ((H_plat[T][0]/(R*T) - (S_plat[T
][0]/R))))
434         for i in range(0, len(p_2)):
435             p_2[i][1] = p_plateau_ad
436         for i in range(0, len(p_1)):

```

```

437         if p_1[i][1] > p_plateau_ad:
438             p_1[i][1] = p_plateau_ad
439     for i in range(0, len(p_3)):
440         if p_3[i][1] < p_plateau_ad:
441             p_3[i][1] = p_plateau_ad
442     p_1 = np.array(p_1)
443     p_2 = np.array(p_2)
444     p_3 = np.array(p_3)
445     p_equilibrium_ad.append([1/T, ln(p_plateau_ad)])
446     p_plat[T] = [p_plateau_ad, None]
447
448
449     if equilibrium_real == 'ab':
450         #equilibrium_ab = equilibrium_real
451         lim_ab = equilibrium_all[equilibrium_real][0][1]
452
453         if not equilibrium_all['bd'] or saveab[0][1] > savebd
[0][0]:
454             for c_H in C_H:
455                 if c_H < equilibrium_all[equilibrium_real
] [0][0]: #intervalo fase alpha
456                     mu_alpha = float(val_mu_H_alpha[T][np.where
(val_mu_H_alpha[T]==c_H)[0], 1])
457                     ln pH = 2* mu_alpha/(R*T)
458                     peq = P_0*exp(ln pH)
459                     p_1.append([c_H, peq])
460                     if equilibrium_all[equilibrium_real][0][0] < c_H
<equilibrium_all[equilibrium_real][0][1]: #plato alpha-beta
461                         p_2.append([c_H, peq])
462                     if lim_ab < c_H < t_beta/r_beta: #intervalo
fase beta
463                         mu_beta = float(val_mu_H_beta[T][np.where(
val_mu_H_beta[T]==c_H)[0], 1])
464                         ln pH = 2* mu_beta/(R*T)
465                         peq = P_0*exp(ln pH)
466                         p_3.append([c_H, peq])
467                         lastcH_beta= c_H

```

```

468         if H_delta < 0:
469             p_4.append([c_H,peq])
470             p_5.append([c_H,peq])
471         if t_beta/r_beta < c_H < C_H[-1]: # plato beta-
delta
472             p_4.append([c_H,peq])
473             p_5 = [[c_H,peq]]
474         lastcH_delta = C_H[-2]
475         if H_delta > 0:
476             if t_beta/r_beta == C_H[-1]:
477                 print("You must change the values of
Theta and r for Beta phase")
478                 H_plat_bd = Hplat(val_H_beta,lastcH_beta,
val_H_delta,lastcH_delta,T)
479                 S_plat_bd = Splat(val_S_beta,lastcH_beta,
val_S_delta,lastcH_delta,T)
480                 p_plateau_bd = exp(2* ((H_plat_bd/(R*T) - (
S_plat_bd/R))))
481                 p_5[0][1] = p_plateau_bd
482                 for i in range(0,len(p_4)):
483                     p_4[i][1] = p_plateau_bd
484                 for i in range(0,len(p_3)):
485                     if p_3[i][1] > p_plateau_bd:
486                         p_3[i][1] = p_plateau_bd
487
488         else:
489
490             equilibrium_bd = min(equilibrium_all['bd'])
491
492             #####Retas tangentes#####
493             line_ab_coef = [valoresalpha[equilibrium_all[
equilibrium_real][0][0]][0],valoresalpha[equilibrium_all[
equilibrium_real][0][0]][1]]
494             line_ab = []
495             for c_H in C_H:
496                 if equilibrium_all[equilibrium_real][0][0] -
.05 <= c_H <= equilibrium_all[equilibrium_real][0][1] + 0.05:

```

```

497         y = line_ab_coef[0] * c_H + line_ab_coef[1]
498         x = c_H
499         line_ab.append([x,y])
500
501         line_bd_coef = [valoresbeta[equilibrium_bd[0]][0],
valoresbeta[equilibrium_bd[0]][1]]
502         line_bd = []
503         for c_H in C_H:
504             if equilibrium_bd[0] - .05 <= c_H <=
equilibrium_bd[1] + 0.05:
505                 y = line_bd_coef[0] * c_H + line_bd_coef[1]
506                 x = c_H
507                 line_bd.append([x,y])
508         line_ab = np.array(line_ab)
509         line_bd = np.array(line_bd)
510         #####C lculos PCT#####
511
512         for c_H in C_H:
513             if c_H < equilibrium_all[equilibrium_real
] [0][0]: #intervalo fase alpha
514                 mu_alpha = float(val_mu_H_alpha[T][np.where
(val_mu_H_alpha[T]==c_H)[0],1])
515                 ln pH = 2* mu_alpha/(R*T)
516                 peq = P_0*exp(lnpH)
517                 p_1.append([c_H,peq])
518             if equilibrium_all[equilibrium_real][0][0]<c_H
<equilibrium_all[equilibrium_real][0][1]: #plato alpha-beta
519                 p_2.append([c_H,peq])
520             if lim_ab < c_H < equilibrium_bd[0]: #intervalo
fase beta
521                 mu_beta = float(val_mu_H_beta[T][np.where(
val_mu_H_beta[T]==c_H)[0],1])
522                 ln pH = 2* mu_beta/(R*T)
523                 peq = P_0*exp(lnpH)
524                 p_3.append([c_H,peq])
525             if equilibrium_bd[0] < c_H < equilibrium_bd[1]:
# plato beta-delta

```

```

526         p_4.append([c_H,peq])
527         if equilibrium_bd[1] <c_H< C_H[-1]: # intervalo
fase delta
528             mu_delta = float(val_mu_H_delta[T][np.where
(val_mu_H_delta[T]==c_H)[0],1])
529             ln pH = 2* mu_delta/(R*T)
530             peq = P_0*exp(ln pH)
531             p_5.append([c_H,peq])
532             H_plat_bd = Hplat(val_H_beta, equilibrium_bd[0],
val_H_delta, equilibrium_bd[1], T)
533             S_plat_bd = Splat(val_S_beta, equilibrium_bd[0],
val_S_delta, equilibrium_bd[1], T)
534             p_plateau_bd = exp(2* ((H_plat_bd/(R*T) - (
S_plat_bd/R))))
535             for i in range(0, len(p_4)):
536                 p_4[i][1] = p_plateau_bd
537             for i in range(0, len(p_3)):
538                 if p_3[i][1] > p_plateau_bd:
539                     p_3[i][1] = p_plateau_bd
540             for i in range(0, len(p_5)):
541                 if p_5[i][1] < p_plateau_bd:
542                     p_5[i][1] = p_plateau_bd
543
544             if H_delta>0:
545                 p_total[T] = np.array(p_1 + p_2 + p_3 + p_4 + p_5)
546             elif H_delta<0:
547                 p_total[T] = np.array(p_1 + p_2 + p_3)
548             p_1_wt = p_1
549             p_2_wt = p_2
550             p_3_wt = p_3
551             p_4_wt = p_4
552             p_5_wt = p_5
553
554             H_plat_ab = Hplat(val_H_alpha, equilibrium_all[
equilibrium_real][0][0], val_H_beta, equilibrium_all[
equilibrium_real][0][1], T)
555             S_plat_ab = Splat(val_S_alpha, equilibrium_all[

```

```

equilibrium_real][0][0], val_S_beta, equilibrium_all[
equilibrium_real][0][1], T)
556         if not H_plat_ab == False and not S_plat_ab == False:
557             p_plateau_ab = exp(2* ((H_plat_ab/(R*T) - (
S_plat_ab/R))))
558             for i in range(0, len(p_2)):
559                 p_2[i][1] = p_plateau_ab
560             for i in range(0, len(p_1)):
561                 if p_1[i][1] > p_plateau_ab:
562                     p_1[i][1] = p_plateau_ab
563             for i in range(0, len(p_3)):
564                 if p_3[i][1] < p_plateau_ab:
565                     p_3[i][1] = p_plateau_ab
566             p_1 = np.array(p_1)
567             p_2 = np.array(p_2)
568             p_3 = np.array(p_3)
569             if H_delta > 0:
570                 p_4 = np.array(p_4)
571                 p_5 = np.array(p_5)
572                 p_equilibrium_ab.append([1/T, ln(p_plateau_ab)])
573                 p_equilibrium_bd.append([1/T, ln(p_plateau_bd)])
574                 p_plat[T] = [p_plateau_ab, p_plateau_bd]
575                 H_plat[T] = [H_plat_ab, H_plat_bd]
576                 S_plat[T] = [S_plat_ab, S_plat_bd]
577             elif H_delta < 0:
578                 if any(equilibrium_all['ab']):
579                     p_plat[T] = [p_plateau_ab, None]
580                     H_plat[T] = [H_plat_ab, None]
581                     S_plat[T] = [S_plat_ab, None]
582                 else:
583                     None
584
585
586 ##### Convert c_H to wt% H
#####
587
588 p_1_wt = np.array(p_1_wt)

```

```

589     p_2_wt = np.array(p_2_wt)
590     p_3_wt = np.array(p_3_wt)
591     p_4_wt = np.array(p_4_wt)
592     p_5_wt = np.array(p_5_wt)
593
594     if type(p_4_wt) != type(p_1_wt) or type(p_5_wt) != type(p_1_wt):
595         p_4_wt = np.array([[0,0]])
596         p_5_wt = np.array([[0,0]])
597
598     alloy_atomic_mass = sum(composition_vector * atomic_mass[0:,1])
599
600
601     for i,j,k,l,m in zip_longest(p_1_wt[0:,0], p_2_wt[0:,0], p_3_wt
602     [0:,0], p_4_wt[0:,0], p_5_wt[0:,0]):
603         if i != None: Hconvert(p_1_wt, alloy_atomic_mass, i)
604         if j != None: Hconvert(p_2_wt, alloy_atomic_mass, j)
605         if k != None: Hconvert(p_3_wt, alloy_atomic_mass, k)
606         if l != None and any(p_4_wt[0]) != [0,0]: Hconvert(p_4_wt,
607     alloy_atomic_mass, l)
608         if m != None and any(p_5_wt[0]) != [0,0]: Hconvert(p_5_wt,
609     alloy_atomic_mass, m)
610
611
612     if not equilibrium_all['ab'] and not equilibrium_all['ad']:
613         p_total_wt[T] = p_1_wt
614
615     if any(equilibrium_all['ab']) or any(equilibrium_all['ad']):
616         if equilibrium_real == "ab":
617             if H_delta > 0:
618                 p_total_wt[T] = np.concatenate((p_1_wt, p_2_wt,
619     p_3_wt, p_4_wt, p_5_wt))
620             elif H_delta < 0:
621                 p_total_wt[T] = np.concatenate((p_1_wt, p_2_wt,
622     p_3_wt))
623             elif equilibrium_real == "ad":
624                 p_total_wt[T] = np.concatenate((p_1_wt, p_2_wt, p_3_wt))
625     end = time.time()

```

```

621     tempo = end - begin
622     print(tempo)
623
624     ##### Important Infos Text
625     #####
626     my_path = os.path.abspath(alloy)
627     my_file = f'./{alloy}.txt'
628     file = open(os.path.join(my_path, my_file), 'a')
629     file.write("
----- \n")
630     file.write(f"Temperature = {T - 273.15} \u00B0C | {T} K \n")
631     if equilibrium_real == 'ab':
632         cHalpha = equilibrium_all[equilibrium_real][0][0]
633         cHbeta1 = equilibrium_all[equilibrium_real][0][1]
634         if not equilibrium_all['bd'] or saveab[0][1] > savebd
[0][0]:
635             cHbeta2 = lastcH_beta
636             cHdelta = lastcH_delta
637         else:
638             cHbeta2 = equilibrium_bd[0]
639             cHdelta = equilibrium_bd[1]
640         wtalpha = cHalpha * 1.01 * 100 / (alloy_atomic_mass + cHalpha
*1.01)
641         wtbeta1 = cHbeta1 * 1.01 * 100 / (alloy_atomic_mass + cHbeta1
*1.01)
642         wtbeta2 = cHbeta2 * 1.01 * 100 / (alloy_atomic_mass + cHbeta2
*1.01)
643         wtdelta = cHdelta * 1.01 * 100 / (alloy_atomic_mass + cHdelta
*1.01)
644     if any(equilibrium_all['ab']):
645         file.write("Plateau alfa-beta \n")
646         file.write(f"c_H[alpha] = {cHalpha} \n" )
647         file.write(f"c_H[beta] = {cHbeta1} \n")
648         file.write(f"wt% H[alpha] = {wtalpha} \n" )
649         file.write(f"wt% H[beta] = {wtbeta1} \n")
650         file.write(f"Pressure Plateau = {p_plat[T][0]:e} atm \n
")

```

```

650         file.write(f"Entalpy Plateau = {H_plat[T][0]} kJ/mol H
        \n")
651         file.write(f"Entropy Plateau = {S_plat[T][0]} kJ/mol H
        \n")
652         if H_delta>0:
653             file.write("Plato beta-delta \n")
654             file.write(f"c_H[beta] = {cHbeta2} \n")
655             file.write(f"c_H[delta] = {cHdelta} \n" )
656             file.write(f"wt% H[beta] = {wtbeta2} \n" )
657             file.write(f"wt% H[delta] = {wtdelta} \n")
658             file.write(f"Pressure Plateau = {p_plat[T][1]:e} atm \n
        ")
659             file.write(f"Entalpy Plateau = {H_plat[T][1]} kJ/mol H
        \n")
660             file.write(f"Entropy Plateau = {S_plat[T][1]} kJ/mol H
        \n")
661         elif equilibrium_real == 'ad':
662             cHalpha = equilibrium_all[equilibrium_real][0][0]
663             cHdelta = equilibrium_all[equilibrium_real][0][1]
664             wtalpha = cHalpha * 1.01 *100/ (alloy_atomic_mass+ cHalpha
        *1.01)
665             wtdelta = cHdelta * 1.01 *100/ (alloy_atomic_mass+ cHdelta
        *1.01)
666             file.write("Plateau alfa-delta \n")
667             file.write(f"c_H[alpha] = {cHalpha} \n" )
668             file.write(f"c_H[delta] = {cHdelta} \n")
669             file.write(f"wt% H[alpha] = {wtalpha} \n" )
670             file.write(f"wt% H[delta] = {wtdelta} \n")
671             file.write(f"Pressure Plateau = {p_plat[T][0]:e} atm \n")
672             file.write(f"Entalpy Plateau = {H_plat[T][0]} kJ/mol H \n"
        )
673             file.write(f"Entropy Plateau = {S_plat[T][0]} kJ/mol H \n")
674         elif H_delta < 0 and not equilibrium_all['ab'] and not
        equilibrium_all['ad']:
675             file.write("Equilibrium not found")
676         file.write("
        ----- \n")

```

```

677     file.close()
678
679
680 ##### Plot graphs and their data txt
        files #####3
681
682
683     if not os.path.isdir(f'./{alloy}/{T - 273.15}'): #Verifica se a
        pasta j existe e se n o existe cria ela com a proxima linha
684         os.mkdir(f'./{alloy}/{T - 273.15}')
685     my_path = os.path.abspath(alloy+ f"/{T - 273.15}") # Figures
        out the absolute path for you in case your working directory
        moves around.
686
687
688
689
690     ##### Entropy Variation #####
691     my_file = f'{alloy}_Sall_T{T - 273.15}C.png'
692     # Create Figure (empty canvas)
693     fig = plt.figure()
694
695     # Add set of axes to figure
696     axes = fig.add_axes([0.1, 0.1, 0.8, 0.8]) # left, bottom, width
        , height (range 0 to 1)
697
698     # Plot on that set of axes
699     axes.plot(val_S_alpha[T][0:,0], val_S_alpha[T][0:,1], 'b',
        label = r'$\Delta S^{\alpha}$')
700     axes.plot(val_S_beta[T][0:,0], val_S_beta[T][0:,1], 'g', label
        = r'$\Delta S^{\beta}$')
701     axes.plot(val_S_delta[T][0:,0], val_S_delta[T][0:,1], 'r',
        label = r'$\Delta S^{\delta}$')
702     axes.set_xlabel('$c_H$') # Notice the use of set_ to begin
        methods
703     axes.set_ylabel(r"$\Delta S$ [kJ/mol]")
704     axes.set_title(f'T = {T- 273.15} \u00B0C')

```

```

705     axes.legend(loc = 0)
706     fig.savefig(os.path.join(my_path, my_file), dpi=200,
707     bbox_inches='tight')
708
709
710     ##### Entalpy Variation #####
711     my_file = f'{alloy}_Hall_T{T - 273.15}C.png'
712     # Create Figure (empty canvas)
713     fig = plt.figure()
714
715     # Add set of axes to figure
716     axes = fig.add_axes([0.1, 0.1, 0.8, 0.8]) # left, bottom, width
717     , height (range 0 to 1)
718
719     # Plot on that set of axes
720     axes.plot(val_H_alpha[T][0:,0], val_H_alpha[T][0:,1], 'b',
721     label = r'$\Delta H^{\alpha}$')
722     axes.plot(val_H_beta[T][0:,0], val_H_beta[T][0:,1], 'g', label
723     = r'$\Delta H^{\beta}$')
724     axes.plot(val_H_delta[T][0:,0], val_H_delta[T][0:,1], 'r',
725     label = r'$\Delta H^{\delta}$')
726     axes.set_xlabel('$c_H$') # Notice the use of set_ to begin
727     methods
728     axes.set_ylabel(r"$\Delta H$ [kJ/mol]")
729     axes.set_title(f'T = {T- 273.15} \u00B0C')
730     axes.legend(loc = 0)
731     fig.savefig(os.path.join(my_path, my_file), dpi=200,
732     bbox_inches='tight')
733
734
735     ##### Varia  o da Energia livre de Gibbs
736     #####
737     if equilibrium_real == "ad":
738
739         my_file = f'{alloy}_Gall_T{T - 273.15}C.png'
740         # Create Figure (empty canvas)

```

```

734     fig = plt.figure()
735
736     # Add set of axes to figure
737     axes = fig.add_axes([0.1, 0.1, 0.8, 0.8]) # left, bottom,
width, height (range 0 to 1)
738
739     # Plot on that set of axes
740     axes.plot(val_G_alpha[T][0:,0], val_G_alpha[T][0:,1], 'b',
label = r'\Delta G^{\alpha}$')
741     axes.plot(val_G_beta[T][0:,0], val_G_beta[T][0:,1], 'g',
label = r'\Delta G^{\beta}$')
742     axes.plot(val_G_delta[T][0:,0], val_G_delta[T][0:,1], 'r',
label = r'\Delta G^{\delta}$')
743     axes.plot(line_ad[0:,0], line_ad[0:,1], 'k--', lw =0.7)
744     axes.set_xlabel('$c_H$') # Notice the use of set_ to begin
methods
745     axes.set_ylabel(r"\Delta G$ [kJ/mol]")
746     axes.set_title(f'T = {T- 273.15} \u00B0C')
747     axes.legend(loc = 0)
748     fig.savefig(os.path.join(my_path, my_file), dpi=200,
bbox_inches='tight')
749
750     elif equilibrium_real == "ab":
751
752     my_file = f'{alloy}_Gall_T{T - 273.15}C.png'
753     # Create Figure (empty canvas)
754     fig = plt.figure()
755
756     # Add set of axes to figure
757     axes = fig.add_axes([0.1, 0.1, 0.8, 0.8]) # left, bottom,
width, height (range 0 to 1)
758
759     # Plot on that set of axes
760     axes.plot(val_G_alpha[T][0:,0], val_G_alpha[T][0:,1], 'b',
label = r'\Delta G^{\alpha}$')
761     axes.plot(val_G_beta[T][0:,0], val_G_beta[T][0:,1], 'g',
label = r'\Delta G^{\beta}$')

```

```

762     axes.plot(val_G_delta[T][0:,0], val_G_delta[T][0:,1], 'r',
label = r'$\Delta G^{\delta}$')
763     if not equilibrium_all['bd'] or saveab[0][1] > savebd
[0][0]:
764         None
765     else:
766         axes.plot(line_ab[0:,0], line_ab[0:,1], 'k--', lw =0.7)
767         axes.plot(line_bd[0:,0], line_bd[0:,1], 'k--', lw =
0.7)
768     axes.set_xlabel('$c_H$') # Notice the use of set_ to begin
methods
769     axes.set_ylabel(r"$\Delta G$ [kJ/mol]")
770     axes.set_title(f'T = {T- 273.15} \u00B0C')
771     axes.legend(loc = 0)
772     fig.savefig(os.path.join(my_path, my_file), dpi=200,
bbox_inches='tight')
773     elif equilibrium_real == "non equilibrium":
774
775     my_file = f'{alloy}_Gall_T{T - 273.15}C.png'
776     # Create Figure (empty canvas)
777     fig = plt.figure()
778
779     # Add set of axes to figure
780     axes = fig.add_axes([0.1, 0.1, 0.8, 0.8]) # left, bottom,
width, height (range 0 to 1)
781
782     # Plot on that set of axes
783     axes.plot(val_G_alpha[T][0:,0], val_G_alpha[T][0:,1], 'b',
label = r'$\Delta G^{\alpha}$')
784     axes.plot(val_G_beta[T][0:,0], val_G_beta[T][0:,1], 'g',
label = r'$\Delta G^{\beta}$')
785     axes.plot(val_G_delta[T][0:,0], val_G_delta[T][0:,1], 'r',
label = r'$\Delta G^{\delta}$')
786     axes.set_xlabel('$c_H$') # Notice the use of set_ to begin
methods
787     axes.set_ylabel(r"$\Delta G$ [kJ/mol]")
788     axes.set_title(f'T = {T- 273.15} \u00B0C')

```

```

789     axes.legend(loc = 0)
790     fig.savefig(os.path.join(my_path, my_file), dpi=200,
bbox_inches='tight')
791
792
793     #####Potencial qu mico do Metal#####
794     my_file = f'{alloy}_miMall_T{T - 273.15}C.png'
795     # Create Figure (empty canvas)
796     fig = plt.figure()
797
798     # Add set of axes to figure
799     axes = fig.add_axes([0.1, 0.1, 0.8, 0.8]) # left, bottom, width
, height (range 0 to 1)
800
801     # Plot on that set of axes
802     axes.plot(val_mu_M_alpha[T][0:,0], val_mu_M_alpha[T][0:,1], 'b'
, label = r'$\mu_{M}^{\alpha}$')
803     axes.plot(val_mu_M_beta[T][0:,0], val_mu_M_beta[T][0:,1], 'g',
label = r'$\mu_{M}^{\beta}$')
804     axes.plot(val_mu_M_delta[T][0:,0], val_mu_M_delta[T][0:,1], 'r'
, label = r'$\mu_{M}^{\delta}$')
805     axes.set_xlabel('$c_H$') # Notice the use of set_ to begin
methods
806     axes.set_ylabel(r"$\mu_{M}$ [kJ/mol]")
807     axes.set_title(f'T = {T- 273.15} \u00B0C')
808     axes.legend(loc = 0)
809     fig.savefig(os.path.join(my_path, my_file), dpi=200,
bbox_inches='tight')
810
811
812     #####Potencial qu mico do Hidrog nio
#####
813     my_file = f'{alloy}_miHall_T{T - 273.15}C.png'
814     # Create Figure (empty canvas)
815     fig = plt.figure()
816
817     # Add set of axes to figure

```

```

818     axes = fig.add_axes([0.1, 0.1, 0.8, 0.8]) # left, bottom, width
      , height (range 0 to 1)
819
820     # Plot on that set of axes
821     axes.plot(val_mu_H_alpha[T][0:,0], val_mu_H_alpha[T][0:,1], 'b'
      , label = r'$\mu_{H}^{\alpha}$')
822     axes.plot(val_mu_H_beta[T][0:,0], val_mu_H_beta[T][0:,1], 'g',
      label = r'$\mu_{H}^{\beta}$')
823     axes.plot(val_mu_H_delta[T][0:,0], val_mu_H_delta[T][0:,1], 'r'
      , label = r'$\mu_{H}^{\delta}$')
824     axes.set_xlabel('$c_H$') # Notice the use of set_ to begin
      methods
825     axes.set_ylabel(r"$\mu_{H}$ [kJ/mol]")
826     axes.set_title(f'T = {T- 273.15} \u00B0C')
827     axes.legend(loc = 0)
828     fig.savefig(os.path.join(my_path, my_file), dpi=200,
      bbox_inches='tight')
829
830
831
832     ##### PCI Diagrams #####
833
834     if Hydrogen_presentation == '1' or Hydrogen_presentation == '3'
      :
835         if H_delta < 0:
836             if not equilibrium_all['ab'] and not equilibrium_all['
      ad']:
837                 my_file = f'{alloy}_PCI_T{T - 273.15}C_cH.png'
838                 # Create Figure (empty canvas)
839                 fig = plt.figure()
840
841                 # Add set of axes to figure
842                 axes = fig.add_axes([0.1, 0.1, 0.8, 0.8]) # left,
      bottom, width, height (range 0 to 1)
843
844                 # Plot on that set of axes
845                 axes.plot(p_1[0:,0], p_1[0:,1], 'b', label = r'$\

```

```

alpha$')
846         plt.plot([], [], ' ', label= f"Plato = {p_2[0,1]:e}
           atm")
847         plt.yscale("log")
848         axes.set_xlabel(r'$c_H$') # Notice the use of set_
to begin methods
849         axes.set_ylabel(r"$P$ [atm]")
850         axes.set_title(f'T = {T- 273.15} \u00B0C')
851         axes.legend(loc = 0,prop={'size': 7.5})
852         fig.savefig(os.path.join(my_path, my_file), dpi
=200, bbox_inches='tight')
853
854         if any(equilibrium_all['ab']) or any(equilibrium_all['ad'])
:
855
856             if equilibrium_real == 'ad':
857
858
859                 my_file = f'{alloy}_PCI_T{T - 273.15}C_cH.png'
860                 # Create Figure (empty canvas)
861                 fig = plt.figure()
862
863                 # Add set of axes to figure
864                 axes = fig.add_axes([0.1, 0.1, 0.8, 0.8]) # left,
bottom, width, height (range 0 to 1)
865
866                 # Plot on that set of axes
867                 axes.plot(p_1[0:,0], p_1[0:,1], 'b', label = r'$\
alpha$')
868                 axes.plot(p_2[0:,0], p_2[0:,1], 'g', label = r'$\
alpha - \delta$')
869                 axes.plot(p_3[0:,0], p_3[0:,1], 'r', label = r'$\
delta$')
870         plt.plot([], [], ' ', label= f"Plato = {p_2[0,1]:e}
           atm")
871         plt.yscale("log")
872         axes.set_xlabel(r'$c_H$') # Notice the use of set_

```

```

to begin methods
873     axes.set_ylabel(r"$P$ [atm]")
874     axes.set_title(f'T = {T- 273.15} \u00B0C')
875     axes.legend(loc = 0,prop={'size': 7.5})
876     fig.savefig(os.path.join(my_path, my_file), dpi
=200, bbox_inches='tight')
877
878
879     if equilibrium_real == 'ab':
880
881         my_file = f'{alloy}_PCI_T{T - 273.15}C_cH.png'
882         # Create Figure (empty canvas)
883         fig = plt.figure()
884
885         # Add set of axes to figure
886         axes = fig.add_axes([0.1, 0.1, 0.8, 0.8]) # left,
bottom, width, height (range 0 to 1)
887
888         # Plot on that set of axes
889         axes.plot(p_1[0:,0], p_1[0:,1], 'b', label = r'$\
alpha$')
890         axes.plot(p_2[0:,0], p_2[0:,1], 'g', label = r'$\
alpha - \beta$')
891         axes.plot(p_3[0:,0], p_3[0:,1], 'r', label = r'$\
beta$')
892         if H_delta >0:
893             axes.plot(p_4[0:,0], p_4[0:,1], 'y', label = r'\
$\beta - \delta$')
894             axes.plot(p_5[0:,0], p_5[0:,1], 'tab:orange',
label = r'$\delta$')
895         plt.plot([], [], ' ', label= f"Plato 1= {p_2[0,1]:e
} atm")
896         if H_delta >0:
897             plt.plot([], [], ' ', label= f"Plato 2= {p_4
[0,1]:e} atm")
898         plt.yscale("log")
899         axes.set_xlabel(r'$c_H$') # Notice the use of set_

```

```

to begin methods
900     axes.set_ylabel(r"$P$ [atm]")
901     axes.set_title(f'T = {T- 273.15} \u00B0C')
902     axes.legend(loc = 0,prop={'size': 7.5})
903     fig.savefig(os.path.join(my_path, my_file), dpi
=200, bbox_inches='tight')
904
905
906     if Hydrogen_presentation == '2' or Hydrogen_presentation == '3'
:
907         if H_delta <0:
908             if not equilibrium_all['ab'] and not equilibrium_all['
ad']:
909                 my_file = f'{alloy}_PCI_T{T - 273.15}C_mass.png'
910                 # Create Figure (empty canvas)
911                 fig = plt.figure()
912
913                 # Add set of axes to figure
914                 axes = fig.add_axes([0.1, 0.1, 0.8, 0.8]) # left,
bottom, width, height (range 0 to 1)
915
916                 # Plot on that set of axes
917                 axes.plot(p_1_wt[0:,0], p_1_wt[0:,1], 'b', label =
r'$\alpha$')
918                 plt.plot([], [], ' ', label= f"Plato = {p_2[0,1]:e}
atm")
919                 plt.yscale("log")
920                 axes.set_xlabel('wt% H') # Notice the use of set_
to begin methods
921                 axes.set_ylabel(r"$P$ [atm]")
922                 axes.set_title(f'T = {T- 273.15} \u00B0C')
923                 axes.legend(loc = 0,prop={'size': 7.5})
924                 fig.savefig(os.path.join(my_path, my_file), dpi
=200, bbox_inches='tight')
925
926                 if any(equilibrium_all['ab']) or any(equilibrium_all['ad'])
:

```

```

927         if equilibrium_real == 'ad':
928
929             my_file = f'{alloy}_PCI_T{T - 273.15}C_mass.png'
930             # Create Figure (empty canvas)
931             fig = plt.figure()
932
933             # Add set of axes to figure
934             axes = fig.add_axes([0.1, 0.1, 0.8, 0.8]) # left,
bottom, width, height (range 0 to 1)
935
936             # Plot on that set of axes
937             axes.plot(p_1_wt[0:,0], p_1_wt[0:,1], 'b', label =
r'\alpha$')
938             axes.plot(p_2_wt[0:,0], p_2_wt[0:,1], 'g', label =
r'\alpha - \delta$')
939             axes.plot(p_3_wt[0:,0], p_3_wt[0:,1], 'r', label =
r'\delta$')
940             plt.plot([], [], ' ', label= f"Plato = {p_2[0,1]:e}
atm")
941             plt.yscale("log")
942             axes.set_xlabel('wt% H') # Notice the use of set_
to begin methods
943             axes.set_ylabel(r"$P$ [atm]")
944             axes.set_title(f'T = {T - 273.15} \u00B0C')
945             axes.legend(loc = 0,prop={'size': 7.5})
946             fig.savefig(os.path.join(my_path, my_file), dpi
=200, bbox_inches='tight')
947
948
949         if equilibrium_real == 'ab':
950
951
952             my_file = f'{alloy}_PCI_T{T - 273.15}C_mass.png'
953             # Create Figure (empty canvas)
954             fig = plt.figure()
955
956             # Add set of axes to figure

```

```

957         axes = fig.add_axes([0.1, 0.1, 0.8, 0.8]) # left,
bottom, width, height (range 0 to 1)
958
959         # Plot on that set of axes
960         axes.plot(p_1_wt[0:,0], p_1_wt[0:,1], 'b', label =
r'\alpha$')
961         axes.plot(p_2_wt[0:,0], p_2_wt[0:,1], 'g', label =
r'\alpha - \beta$')
962         axes.plot(p_3_wt[0:,0], p_3_wt[0:,1], 'r', label =
r'\beta$')
963         if H_delta >0:
964             axes.plot(p_4_wt[0:,0], p_4_wt[0:,1], 'y',
label = r'\beta - \delta$')
965             axes.plot(p_5_wt[0:,0], p_5_wt[0:,1], 'tab:
orange', label = r'\delta$')
966             plt.plot([], [], ' ', label= f"Plato 1= {p_2[0,1]:e
} atm")
967             if H_delta >0:
968                 plt.plot([], [], ' ', label= f"Plato 2= {p_4
[0,1]:e} atm")
969             plt.yscale("log")
970             axes.set_xlabel('wt% H') # Notice the use of set_
to begin methods
971             axes.set_ylabel(r"$P$ [atm]")
972             axes.set_title(f'T = {T- 273.15} \u00B0C')
973             axes.legend(loc = 0,prop={'size': 7.5})
974             fig.savefig(os.path.join(my_path, my_file), dpi
=200, bbox_inches='tight')
975
976
977
978
979
980         #####Arquivos fonte .txt dos dados obtidos
#####
981
982         if not os.path.isdir(f'./{alloy}/{T - 273.15}/data'): #Verifica

```

```

se a pasta j existe e se n o existe cria ela com a proxima
linha
983     os.mkdir(f'./{alloy}/{T - 273.15}/data')
984     my_path = os.path.abspath(alloy+ f"/{T - 273.15}/data") #
Figures out the absolute path for you in case your working
directory moves around.
985
986
987     my_file = f'{alloy}_Sa_T{T - 273.15}C.txt'
988     dataframe = pd.DataFrame(val_S_alpha[T])
989     dataframe = dataframe.rename(columns = { 0 : "c_H", 1 : "
S_alpha"})
990     dataframe.to_csv(os.path.join(my_path, my_file), sep = '\t',
encoding = "utf-8", index = False, columns=['c_H', 'S_alpha'])
991
992     my_file = f'{alloy}_Sb_T{T - 273.15}C.txt'
993     dataframe = pd.DataFrame(val_S_beta[T])
994     dataframe = dataframe.rename(columns = { 0 : "c_H", 1 : "S_beta
"})
995     dataframe.to_csv(os.path.join(my_path, my_file), sep = '\t',
encoding = "utf-8", index = False, columns=['c_H', 'S_beta'])
996
997     my_file = f'{alloy}_Sd_T{T - 273.15}C.txt'
998     dataframe = pd.DataFrame(val_S_delta[T])
999     dataframe = dataframe.rename(columns = { 0 : "c_H", 1 : "
S_delta"})
1000     dataframe.to_csv(os.path.join(my_path, my_file), sep = '\t',
encoding = "utf-8", index = False, columns=['c_H', 'S_delta'])
1001
1002     my_file = f'{alloy}_Ha_T{T - 273.15}C.txt'
1003     dataframe = pd.DataFrame(val_H_alpha[T])
1004     dataframe = dataframe.rename(columns = { 0 : "c_H", 1 : "
H_alpha"})
1005     dataframe.to_csv(os.path.join(my_path, my_file), sep = '\t',
encoding = "utf-8", index = False, columns=['c_H', 'H_alpha'])
1006
1007     my_file = f'{alloy}_Hb_T{T - 273.15}C.txt'

```

```
1008     dataframe = pd.DataFrame(val_H_beta[T])
1009     dataframe = dataframe.rename(columns = { 0 : "c_H", 1 : "H_beta
1010     dataframe.to_csv(os.path.join(my_path, my_file), sep = '\t',
encoding = "utf-8", index = False, columns=['c_H', 'H_beta'])
1011
1012     my_file = f'{alloy}_Hd_T{T - 273.15}C.txt'
1013     dataframe = pd.DataFrame(val_H_delta[T])
1014     dataframe = dataframe.rename(columns = { 0 : "c_H", 1 : "
H_delta"})
1015     dataframe.to_csv(os.path.join(my_path, my_file), sep = '\t',
encoding = "utf-8", index = False, columns=['c_H', 'H_delta'])
1016
1017     my_file = f'{alloy}_Ga_T{T - 273.15}C.txt'
1018     dataframe = pd.DataFrame(val_G_alpha[T])
1019     dataframe = dataframe.rename(columns = { 0 : "c_H", 1 : "
G_alpha"})
1020     dataframe.to_csv(os.path.join(my_path, my_file), sep = '\t',
encoding = "utf-8", index = False, columns=['c_H', 'G_alpha'])
1021
1022     my_file = f'{alloy}_Gb_T{T - 273.15}C.txt'
1023     dataframe = pd.DataFrame(val_G_beta[T])
1024     dataframe = dataframe.rename(columns = { 0 : "c_H", 1 : "G_beta
1025     dataframe.to_csv(os.path.join(my_path, my_file), sep = '\t',
encoding = "utf-8", index = False, columns=['c_H', 'G_beta'])
1026
1027     my_file = f'{alloy}_Gd_T{T - 273.15}C.txt'
1028     dataframe = pd.DataFrame(val_G_delta[T])
1029     dataframe = dataframe.rename(columns = { 0 : "c_H", 1 : "
G_delta"})
1030     dataframe.to_csv(os.path.join(my_path, my_file), sep = '\t',
encoding = "utf-8", index = False, columns=['c_H', 'G_delta'])
1031
1032     my_file = f'{alloy}_miMa_T{T - 273.15}C.txt'
1033     dataframe = pd.DataFrame(val_mu_M_alpha[T])
1034     dataframe = dataframe.rename(columns = { 0 : "c_H", 1 : "
```

```

mi_M_alpha"})
1035     dataframe.to_csv(os.path.join(my_path, my_file), sep = '\t',
encoding = "utf-8", index = False, columns=['c_H', 'mi_M_alpha'
])
1036
1037     my_file = f'{alloy}_miMb_T{T - 273.15}C.txt'
1038     dataframe = pd.DataFrame(val_mu_M_beta[T])
1039     dataframe = dataframe.rename(columns = { 0 : "c_H", 1 : "
mi_M_beta"})
1040     dataframe.to_csv(os.path.join(my_path, my_file), sep = '\t',
encoding = "utf-8", index = False, columns=['c_H', 'mi_M_beta'])
1041
1042     my_file = f'{alloy}_miMd_T{T - 273.15}C.txt'
1043     dataframe = pd.DataFrame(val_mu_M_delta[T])
1044     dataframe = dataframe.rename(columns = { 0 : "c_H", 1 : "
mi_M_delta"})
1045     dataframe.to_csv(os.path.join(my_path, my_file), sep = '\t',
encoding = "utf-8", index = False, columns=['c_H', 'mi_M_delta'
])
1046
1047     my_file = f'{alloy}_miHa_T{T - 273.15}C.txt'
1048     dataframe = pd.DataFrame(val_mu_H_alpha[T])
1049     dataframe = dataframe.rename(columns = { 0 : "c_H", 1 : "
mi_H_alpha"})
1050     dataframe.to_csv(os.path.join(my_path, my_file), sep = '\t',
encoding = "utf-8", index = False, columns=['c_H', 'mi_H_alpha'
])
1051
1052     my_file = f'{alloy}_miHb_T{T - 273.15}C.txt'
1053     dataframe = pd.DataFrame(val_mu_H_beta[T])
1054     dataframe = dataframe.rename(columns = { 0 : "c_H", 1 : "
mi_H_beta"})
1055     dataframe.to_csv(os.path.join(my_path, my_file), sep = '\t',
encoding = "utf-8", index = False, columns=['c_H', 'mi_H_beta'])
1056
1057     my_file = f'{alloy}_miHd_T{T - 273.15}C.txt'
1058     dataframe = pd.DataFrame(val_mu_H_delta[T])

```

```

1059     dataframe = dataframe.rename(columns = { 0 : "c_H", 1 : "
mi_H_delta"})
1060     dataframe.to_csv(os.path.join(my_path, my_file), sep = '\t',
encoding = "utf-8", index = False, columns=['c_H', 'mi_H_delta'
])
1061
1062
1063     my_file = f'{alloy}_PCI_T{T - 273.15}C_cH.txt'
1064     dataframe = pd.DataFrame(p_total[T])
1065     dataframe = dataframe.rename(columns = { 0 : "c_H", 1 : "P(atm)
"})
1066     dataframe.to_csv(os.path.join(my_path, my_file), sep = '\t',
encoding = "utf-8", index = False, columns=['c_H', 'P(atm)'])
1067
1068     my_file = f'{alloy}_PCI_T{T - 273.15}C_mass.txt'
1069     dataframe = pd.DataFrame(p_total_wt[T])
1070     dataframe = dataframe.rename(columns = { 0 : "wt% H", 1 : "P(
atm)"})
1071     dataframe.to_csv(os.path.join(my_path, my_file), sep = '\t',
encoding = "utf-8", index = False, columns=['wt% H', 'P(atm)'])
1072 #####Van't Hoff plot #####
1073
1074 if len(temperature)>1:
1075     my_path = os.path.abspath(alloy)
1076
1077     if all( i == 'ad' for i in verify_equilibrium):
1078
1079         p_equilibrium_ad = np.array(p_equilibrium_ad)
1080         my_file = f'{alloy}_VantHoff.png'
1081         # Create Figure (empty canvas)
1082         fig = plt.figure()
1083
1084         # Add set of axes to figure
1085         axes = fig.add_axes([0.1, 0.1, 0.8, 0.8]) # left, bottom,
width, height (range 0 to 1)
1086
1087         # Plot on that set of axes

```

```

1088     axes.plot(p_equilibrium_ad[0:,0], p_equilibrium_ad[0:,1], '
k', label = r'$\alpha - \delta$')
1089     for T in [temperature[0], temperature[-1]]:
1090         plt.plot([], [], ' ', label= r"$\Delta H_{Plat} ^{\alpha - \delta}$$(%.2f C) = %.4f kJ/mol" %(T-273.15, float(
H_plat[T][0])))
1091         plt.plot([], [], ' ', label= r"$\Delta S_{Plat} ^{\alpha - \delta}$$(%.2f C) = %.4f kJ/mol" %(T-273.15, float(
S_plat[T][0])))
1092     axes.set_xlabel('1/T [K-1 $]') # Notice the use of set_
to begin methods
1093     axes.set_ylabel(r"$\ln(P_{eq}(atm))$")
1094     axes.set_title(f"Van't Hoff {alloy}")
1095     axes.legend(loc = 0, prop={'size': 5})
1096     fig.savefig(os.path.join(my_path, my_file), dpi=200,
bbox_inches='tight')
1097
1098     elif all( i == 'ab' for i in verify_equilibrium) and H_delta>0:
1099
1100     p_equilibrium_ab = np.array(p_equilibrium_ab)
1101     p_equilibrium_bd = np.array(p_equilibrium_bd)
1102     my_file = f'{alloy}_VantHoff.png'
1103     # Create Figure (empty canvas)
1104     fig = plt.figure()
1105
1106     # Add set of axes to figure
1107     axes = fig.add_axes([0.1, 0.1, 0.8, 0.8]) # left, bottom,
width, height (range 0 to 1)
1108
1109     # Plot on that set of axes
1110     axes.plot(p_equilibrium_ab[0:,0], p_equilibrium_ab[0:,1], '
k', label = r'$\alpha - \beta$')
1111     for T in [temperature[0], temperature[-1]]:
1112         plt.plot([], [], ' ', label= r"$\Delta H_{Plat} ^{\alpha - \beta}$$(%.2f C) = %.4f kJ/mol" %(T-273.15, float(H_plat
[T][0])))
1113         plt.plot([], [], ' ', label= r"$\Delta S_{Plat} ^{\alpha - \beta}$$(%.2f C) = %.4f kJ/mol" %(T-273.15, float(S_plat
[T][0])))

```

```

alpha - \beta}{$(%.2f C) = %.4f kJ/mol" %(T-273.15, float(S_plat
[T][0]))))
1114     axes.plot(p_equilibrium_bd[0:,0], p_equilibrium_bd[0:,1], '
r', label = r'$\beta - \delta$')
1115     for T in [temperature[0], temperature[-1]]:
1116         plt.plot([], [], ' ', label= r"$\Delta H_{Plat} \sim \beta
- \delta}{$(%.2f C) = %.4f kJ/mol" %(T-273.15, float(H_plat[T
][1]))))
1117         plt.plot([], [], ' ', label= r"$\Delta S_{Plat} \sim \beta
- \delta}{$(%.2f C) = %.4f kJ/mol" %(T-273.15, float(S_plat[T
][1]))))
1118     axes.set_xlabel('1/T [K-1 $]') # Notice the use of set_
to begin methods
1119     axes.set_ylabel(r"$\ln(P_{eq}(atm))$")
1120     axes.set_title(f"Van't Hoff {alloy}")
1121     axes.legend(loc = 0, prop={'size': 4})
1122     fig.savefig(os.path.join(my_path, my_file), dpi=200,
bbox_inches='tight')
1123     elif H_delta>0:
1124
1125         if not equilibrium_all['ab'] and not equilibrium_all['bd']:
1126             p_notequilibrium = np.array(p_notequilibrium)
1127             p_equilibrium_ad = np.array(p_equilibrium_ad)
1128             p_equilibrium_ab = np.array(p_equilibrium_ab)
1129             p_equilibrium_bd = np.array(p_equilibrium_bd)
1130             p_equilibrium = p_equilibrium_ab + p_equilibrium_bd
1131             my_file = f'{alloy}_VantHoff.png'
1132             # Create Figure (empty canvas)
1133             fig = plt.figure()
1134
1135             # Add set of axes to figure
1136             axes = fig.add_axes([0.1, 0.1, 0.8, 0.8]) # left, bottom,
width, height (range 0 to 1)
1137
1138             # Plot on that set of axes
1139
1140             if not equilibrium_all['ab'] and not equilibrium_all['bd']:

```

```

1141         axes.plot(p_notequilibrium[0:,0], p_notequilibrium
1142 [0:,1], 'b', label = r'$\alpha - \delta$')
1143
1144         if any(equilibrium_all['ab']) or any(equilibrium_all['ad'])
1145 :
1146             axes.plot(p_equilibrium_ad[0:,0], p_equilibrium_ad
1147 [0:,1], 'b', label = r'$\alpha - \delta$')
1148             for T in temperature:
1149                 if H_plat[T][1] == None:
1150                     plt.plot([], [], ' ', label= r"$\Delta H_{Plat}
1151 ^{\alpha - \delta}$(%.2f C) = %.4f kJ/mol" %(T-273.15, float(
1152 H_plat[T][0])))
1153                     plt.plot([], [], ' ', label= r"$\Delta S_{Plat}
1154 ^{\alpha - \delta}$(%.2f C) = %.4f kJ/mol" %(T-273.15, float(
1155 S_plat[T][0])))
1156             axes.plot(p_equilibrium_ab[0:,0], p_equilibrium_ab
1157 [0:,1], 'k', label = r'$\alpha - \beta$')
1158             for T in temperature:
1159                 if not H_plat[T][1] == None:
1160                     plt.plot([], [], ' ', label= r"$\Delta H_{Plat}
1161 ^{\alpha - \beta}$(%.2f C) = %.4f kJ/mol" %(T-273.15, float(
1162 H_plat[T][0])))
1163                     plt.plot([], [], ' ', label= r"$\Delta S_{Plat}
1164 ^{\alpha - \beta}$(%.2f C) = %.4f kJ/mol" %(T-273.15, float(
1165 S_plat[T][0])))
1166             axes.plot(p_equilibrium_bd[0:,0], p_notequilibrium[0:,1],
1167 'r', label = r'$\beta - \delta$')
1168             for T in temperature:
1169                 if not H_plat[T][1] == None:
1170                     plt.plot([], [], ' ', label= r"$\Delta H_{Plat}
1171 ^{\beta - \delta}$(%.2f C) = %.4f kJ/mol" %(T-273.15, float(
1172 H_plat[T][1])))
1173                     plt.plot([], [], ' ', label= r"$\Delta S_{Plat}
1174 ^{\beta - \delta}$(%.2f C) = %.4f kJ/mol" %(T-273.15, float(
1175 S_plat[T][1])))
1176             axes.set_xlabel('1/T [K-1 $]') # Notice the use of set_

```

```

to begin methods
1161     axes.set_ylabel(r"$\ln(P_{eq}(atm))$")
1162     axes.set_title(f"Van't Hoff {alloy}")
1163     axes.legend(loc = 0,prop={'size': 4})
1164     fig.savefig(os.path.join(my_path, my_file), dpi=200,
bbox_inches='tight')
1165
1166     #####PCT Diagram#####
1167     if Hydrogen_presentation == '1' or Hydrogen_presentation == '3'
:
1168         my_file = f'{alloy}_PCT_cH.png'
1169         # Create Figure (empty canvas)
1170         fig = plt.figure()
1171
1172         # Add set of axes to figure
1173         axes = fig.add_axes([0.1, 0.1, 0.8, 0.8]) # left, bottom,
width, height (range 0 to 1)
1174
1175         # Plot on that set of axes
1176         for T in temperature:
1177             axes.plot(p_total[T][0:,0], p_total[T][0:,1], label = f
"{T-273.15} \u00B0C")
1178             if any(equilibrium_all['ab']) or any(equilibrium_all['
ad']):
1179                 if p_plat[T][1] == None:
1180                     plt.plot([], [], ' ', label= f"Plato = {p_plat[
T][0]:e} atm")
1181                 else:
1182                     plt.plot([], [], ' ', label= f"Plato 1 = {
p_plat[T][0]:e} atm")
1183                     plt.plot([], [], ' ', label= f"Plato 2 = {
p_plat[T][1]:e} atm")
1184                 plt.yscale("log")
1185                 axes.set_xlabel('$c_H$') # Notice the use of set_ to begin
methods
1186                 axes.set_ylabel(r"$P$ [atm]")
1187                 axes.set_title(f'PCT {alloy}')

```

```

1188     axes.legend(loc = 0,prop={'size': 5})
1189     fig.savefig(os.path.join(my_path, my_file), dpi=200,
bbox_inches='tight')
1190     if Hydrogen_presentation == '2' or Hydrogen_presentation == '3'
:
1191         my_file = f'{alloy}_PCT_mass.png'
1192         # Create Figure (empty canvas)
1193         fig = plt.figure()
1194
1195         # Add set of axes to figure
1196         axes = fig.add_axes([0.1, 0.1, 0.8, 0.8]) # left, bottom,
width, height (range 0 to 1)
1197
1198         # Plot on that set of axes
1199         for T in temperature:
1200             axes.plot(p_total_wt[T][0:,0], p_total_wt[T][0:,1],
label = f"{T-273.15} \u00B0C")
1201             if any(equilibrium_all['ab']) or any(equilibrium_all['
ad']):
1202                 if p_plat[T][1] == None:
1203                     plt.plot([], [], ' ', label= f"Plato = {p_plat[
T][0]:e} atm")
1204                 else:
1205                     plt.plot([], [], ' ', label= f"Plato 1 = {
p_plat[T][0]:e} atm")
1206                     plt.plot([], [], ' ', label= f"Plato 2 = {
p_plat[T][1]:e} atm")
1207             plt.yscale("log")
1208             axes.set_xlabel('wt% H') # Notice the use of set_ to begin
methods
1209             axes.set_ylabel(r"$P$ [atm]")
1210             axes.set_title(f'PCT {alloy}')
1211             axes.legend(loc = 0,prop={'size': 5})
1212             fig.savefig(os.path.join(my_path, my_file), dpi=200,
bbox_inches='tight')
1213 end = time.time()
1214 tempo = end - begin

```

```
1215 print(tempo)
```