GEOMETRIC ACCELERATION OF COMPLEX CHEMICAL EQUILIBRIUM CALCULATIONS FOR INCLUSION IN MULTIPHYSICS AND PROCESS MODELS

by

Willem Abraham Roos

submitted in partial fulfilment of the requirements for the degree

Philosophiae Doctor (Metallurgical Engineering)

in the

Faculty of Engineering, Built Environment and Information Technology

of the

University of Pretoria

on

July 27, 2023

Abstract

Incorporating multi-component, multi-phase, high-temperature, complex chemical equilibrium calculations into multiphysics and process models can provide valuable insights into industrial processes and equipment that current modelling methods and measurements cannot. The equilibrium state of a thermochemical system is determined by minimising the Gibbs energy for a given set of system component concentrations, temperature, and pressure, and becomes computationally expensive when a large number of these calculations have to be performed. This makes direct integration of chemical equilibrium calculations into models infeasible.

There have been many attempts to, in one way or another, accelerate these calculations. The strengths of these existing acceleration methods, together with fundamental thermochemical theory, were used to conceptualise and develop a new accelerator algorithm. The accelerator algorithm uses a system's phase diagram and the Gibbs phase rule to map the thermochemical system to geometric space by storing calculated physical and thermochemical properties in-situ for later recall and interpolation. Linear interpolation with the lever rule in geometric space is less computationally expensive than Gibbs energy minimisation. The advantage of populating a database in-situ is that data is only generated and stored in the regions accessed by the model as it is being solved. The accelerator algorithm is based on established thermochemical theory, and the generality thereof allows the accelerator to be used in any system, regardless of the number of components.

The performance of the accelerator algorithm was tested on a number of two- and three-component systems as well as on two industry-related processes; a simplified four-component ilmenite smelting system and a simplified five-component iron- and steelmaking system. As the number of system components increase, so does the computational expense of equilibrium calculations. This translated to larger acceleration factors being achieved as the number of system components increased – from as high as 20 in two-component systems to 1000 in the four- and five-component systems. Interpolation errors made on phase compositions were in the order of 10^{-2} mol mol^{-1} and less. This would translate to an interpolated phase composition being accurate to within 99 % of the calculated phase composition. The majority of interpolation errors made on physical and thermochemical properties were in the order of 1 % and less.

The developed algorithm showed noteworthy acceleration of equilibrium calculations when tested on the two industry-related processes while maintaining acceptable levels of accuracy. There is great potential for the accelerator algorithm to make the inclusion of equilibrium calculations in models with many system components feasible. The performance of the accelerator can be improved by transferring the algorithm to a more computationally efficient compiled programming language and utilising a more performant database system.

List of Publications

The following articles have been published in partial fulfilment of the requirements for the degree Philosophiae Doctor in Metallurgical Engineering together with a short summary of each:

Publication 1: W.A. Roos and J.H. Zietsman. "Accelerating Complex Chemical Equilibrium Calculations – A Review". In: *CALPHAD: Computer Coupling of Phase Diagrams and Thermochemistry* 77 (2022). (Roos and Zietsman 2022a) (doi: 10.1016/j.calphad.2021.102380)

The computational expense of including equilibrium calculations into process and multiphysics models were illustrated. A few applications were discussed where the inclusion of equilibrium calculations into models provided valuable insight, showing the importance thereof. A review was done of a number of methods found in literature that were developed to improve the efficiency of equilibrium calculations and make it more feasible to include into multiphysics and process models. The advantages and pitfalls of the reviewed methods were identified and suggestions made of the methods that could be combined to create a new accelerator algorithm that would benefit from as many of the advantages as possible.

Publication 2: W.A. Roos and J.H. Zietsman. "Geometric Acceleration of Complex Chemical Equilibrium Calculations – Algorithm and Application to Two- and Three-component Systems". In: *CALPHAD: Computer Coupling of Phase Diagrams and Thermochemistry* 77 (2022). (Roos and Zietsman 2022b) (doi: 10.1016/j.calphad.2022.102420)

A new generic accelerator algorithm was conceptualised based on the suggestions from the first publication. Foundational concepts required for the algorithm development were discussed with a short demonstration of the concepts applied to a two- and three-component system. The developed accelerator algorithm was presented in detail together with a nomenclature of all the symbols and concepts used. Performance tests were performed in a number of two- and three-component systems and the results presented. The performance of the accelerator algorithm was satisfactory for the most part, except for the acceleration achieved in two-phase regions in three-component systems due to a computationally expensive sub-routine in the interpolation algorithm that solved a system of non-linear equations.

Publication 3: W.A. Roos, A.E.J. Bogaers, and J.H. Zietsman. "Geometric Acceleration of Complex Chemical Equilibrium Calculations – Performance in Two- to Five-component Systems". In: *CALPHAD: Computer Coupling of Phase Diagrams and Thermochemistry* 82 (2023). (Roos, Bogaers, and Zietsman 2023) (doi: 10.1016/j.calphad.2023.102584)

A change was made to the implementation of the troublesome interpolation algorithm sub-routine that improved the efficiency of solving the system of non-linear equations. Acceleration was now achieved in all phase regions of the systems tested in the previous publication. Functionality and performance tests were performed on a number of two- and three-component systems and the findings illustrated and results presented as heatmaps on phase diagrams. Distribution plots of the results were presented to provide a better indication of the performance achieved. Thereafter, the performance of the accelerator algorithm was tested in two industry-related processes; a simplified four-component ilmenite smelting system and a simplified five-component iron- and steelmaking system. It was demonstrated that the accelerator algorithm has potential of achieving noteworthy levels of acceleration in systems with a large number of components while maintaining acceptable levels of accuracy.

Dedication

In the hopes that this work may in some way contribute to new insights into their complex systems, this work is dedicated to all members of the modelling community that have a desire to include more detailed thermochemical properties into their models.

Acknowledgements

Glory to God for providing me with all the opportunities to make this work possible and the perseverance to complete this work to the best of my abilities.

A special thank you to my parents, Johan and Lizette Roos, for a lifetime of lessons learnt from you and the countless opportunities you gave me. Thank you for your support, guidance, and encouragement through all the challenges I face. I am eternally grateful for what you have done for me; without you, I would not be the person I am today.

To my supervisor, Dr Johan Zietsman, who not only shared the excitement of the discoveries made over the duration of this project but also helped overcome the hurdles we faced — I am grateful for the opportunity I had to work with you. Thank you for your support, guidance, insight, suggestions and methods you shared with me; it made me a better engineer, and for that I am grateful. Your enthusiasm and vision for the pyrometallurgy industry, especially in South Africa, is inspiring and I truly hope that the work we did in this project will help fulfil this vision.

Thank you Marinda Zietsman for your support, encouragement, and especially your administrative assistance throughout this project.

Thank you Dr Alfred Bogaers for all the fruitful discussions and valuable feedback during the development phase as well as your guidance regarding the numerical methods implemented in the algorithm.

Thank you to everyone at Ex Mente for all the interest you showed in the project, for your support and encouragement. I would like to thank Ruan Theron, Dr Johan Heyns, Heine Weitz, Rudie Theron, Stefan van der Merwe, and Max Dittmar for all the discussions, feedback, and assistance. I am also grateful to Ex Mente for allowing me to use their ChemAppPy equilibrium calculation software throughout this project.

I want to thank my family and friends for the interest you showed and your support. I would like to thank my wife Sané in particular for all your support, encouragement, and patience throughout this project.

"One of the principal objects of theoretical research is to find the point of view from which the subject appears in its greatest simplicity."

Josiah Willard Gibbs

Table of Contents

List of Figures

List of Tables

Glossary

acceleration A notable reduction in the duration of time that an equilibrium state is provided to a model by an accelerator algorithm in comparison to the equivalent direct equilibrium calculation.

acceleration factor The factor by which the computation time of an equilibrium state has been reduced by an accelerator algorithm in comparison to the equivalent direct equilibrium calculation.

acceleration method A method used or an algorithm employed to reduce the computation time of an equilibrium calculation.

accelerator algorithm An algorithm employed to reduce the computation time of an equilibrium calculation.

accuracy The degree to which the interpolated equilibrium state is similar to the correct calculated equilibrium state. This applies to phase compositions, fractions, and properties, as well as system properties.

addition The process of adding data to a database.

barycentric coordinate weight A weight applied to a vector between two vertices of a simplex $\in [0, 1]$. A set of barycentric coordinate weights are used to describe a point in a simplex as sum of weighted vectors between the vertices of the simplex.

barycentric coordinate system A coordinate system that express the location of a point contained within a simplex as the sum of weighted vectors between the vertices of the simplex.

calculated phase composition The composition of a phase as determined by a direct equilibrium calculation.

calculated equilibrium state Equilibrium state directly calculated by equilibrium calculation software. This includes the equilibrium state at the system state as well as the stable phases.

Cayley-Menger determinant The determinant of a modified distance matrix of the vertices of a simplex that is used to determine the content of a simplex.

chemical potential The potential for a chemical substance to undergo a chemical change due to differences in concentrations.

chemical element Any chemical substance that cannot be decomposed into simpler substances by a chemical process.

composition The set of system components and concentrations they are present at. Can be the composition of a phase or of the system state.

compositional potential index The index of a system component concentration in the uniform structured reference frame. The system component concentration can be part of the composition of the system state or a phase.

compositional potential index array The set of compositional potential indices of all system component concentrations in a composition, either of the system state or a phase.

composition tolerance The tolerance applied to phase compositions to determine valid tie simplices that can be combined into a phase region cell. The composition tolerance is also used to define the intervals of the uniform structured reference frame used in the database.

composition vertex The composition of a stable phase, as a coordinate in compositional space, that is associated with the vertex of a tie simplex where the vertex is located on the phase's OPF boundary.

content The measure of the space contained within the vertices of a simplex.

convex hull Smallest subset of coordinates that contain all the coordinates of the entire set within a convex envelope.

database A collection of organised data that can be easily accessed, managed, updated, new data stored, and existing data recalled.

database population The process of adding data to a database.

deceleration A notable increase in the time that an equilibrium state is provided to a model by an accelerator algorithm in comparison to a direct equilibrium calculation. The opposite to acceleration.

degree of freedom An indication of how many intensive variables can be changed independently without altering the number of phases that exists at thermochemical equilibrium.

dimensional characteristic The combination of the three dimensionalities that characterise an OPF feature; the system geometric dimensionality as well as the OPF feature's geometric and functional dimensionalities.

direct equilibrium calculation An equilibrium calculation performed when equilibrium calculation software is directly integrated into a model – no intermediary accelerators or databases are present.

direct integration When a model and equilibrium calculation software communicate directly with each other; the model provides the system state and the equilibrium calculation software returns the equilibrium state.

discretisation node A vertex of a discretisation segment.

discretisation by simplices A method to subdivide a geometrical feature into smaller discrete simplices; a line into 1-simplices, a surface into 2-simplices, a volume into 3-simplices, etc.

discretisation segment The discrete part that is the result of discretising a geometrical feature.

discretisation algorithm The sequence of rules and instructions followed to discretise a phase region into discrete phase region cells.

discretise To subdivide a geometrical feature into smaller discrete parts.

efficiency The state or ability to be efficient.

efficient To perform a task with minimum amount of time wasted and optimum usage of resources.

eligible A stored tie simplex or phase region cell is considered eligible if it meets specific database search and recall requirements.

equilibrium The unquestionable state of a system when no net macroscopic flow of matter or energy occurs. This state is reached when all compositional and non-compositional potentials are balanced and there is no tendency for spontaneous change to occur. Also referred to as thermochemical equilibrium.

equilibrium state The state consisting of the thermochemical properties of the system and all stable phases reached at equilibrium.

equilibrium calculation accelerator An accelerator algorithm employed to provide an equilibrium state to a model faster than the equivalent direct equilibrium calculation can.

equilibrium calculation software A software package employed to directly calculate an equilibrium state.

equilibrium reactor A component in a process model that represent a unit operation where a number of material streams undergo chemical reactions which are assumed to reach equilibrium.

equilibrium calculation A calculation performed to determine the equilibrium state for a given system state. Abbreviation for multi-component, multi-phase, high-temperature, complex chemical equilibrium calculation.

foundational concept Important theories and concepts that are key to the development and functionality of the accelerator algorithm.

functionality The quality of the accelerator algorithm being suitable to serve its intended purpose well. Investigated during a functionality test.

functionality and performance test Collective for functionality test and performance test.

functionality test A test that is performed to determine if the accelerator algorithm functions as intended.

fundamental thermochemical theory Established thermochemical theory that is foundational to the functionality of the accelerator algorithm, ensuring that it is thermodynamically consistent to provide security for the decisions taken during the discretisation and interpolation algorithms.

generality The quality of being generic.

generic To function regardless of the number of system components or number of phases.

Gibbs phase rule A relationship between the number of phases that will co-exist within a system at equilibrium and the number of intensive variables that can be varied independently. Examples of intensive variables are temperature, pressure, and independent system component concentrations.

Gibbs energy minimisation A method employed to calculate thermochemical equilibrium by determining the equilibrium state with the lowest Gibbs energy for a given system state.

independent intensive variable A variable that does not depend on the system size, and can be varied independently from all other intensive variables. This include non-compositional variables such as temperature and pressure, and compositional variables such as system component concentrations.

index The position in the uniform structured reference frame.

in-situ Being performed while the model is being solved.

interpolate To estimate new data within a discrete set of known data.

interpolated phase composition The composition of a phase as determined by the interpolation algorithm.

interpolated equilibrium calculation The process of estimating the equilibrium state by performing interpolation between calculated equilibrium states stored in a database.

interpolated system equilibrium state Equilibrium state of the system determined by the accelerator algorithm from interpolated phase equilibrium states.

interpolated tie simplex A tie simplex interpolated from stored tie simplices that contains the specified system state. The vertices of this tie simplex are the interpolated phase equilibrium states, which are used in the interpolation routine to determine the interpolated system equilibrium state.

interpolated equilibrium state The estimated equilibrium state determined by the accelerator algorithm by performing an interpolated equilibrium calculation from stored data. This includes the interpolated equilibrium state of each stable phase as well as the system.

interpolation error The difference between a calculated property value and the corresponding interpolated property value.

interpolation algorithm The sequence of rules and instructions followed to obtain the interpolated equilibrium state from a set of calculated equilibrium states.

invariant point A point in a phase diagram that has a degree of freedom of zero; a slight change in any compositional or non-compositional independent intensive variable would lead to the change in number of stable phases at equilibrium.

iso-non-compositional-potential At constant non-compositional potentials such as temperature and pressure. Abbreviated by iso-ψ.

isobar A geometrical feature that connects all states with the same pressure.

iso-chemical-potential At constant chemical potential.

isobaric At constant pressure.

isopleth A geometrical feature that connects all states with the same system component concentrations.

isotherm A geometrical feature that connects all states with the same temperature.

isothermal At constant temperature.

lever rule A method of calculating phase fractions at a system composition within a phase region by evaluating the geometric relations of the phase's compositions to the composition of interest.

limit Upper and lower bounds applied to compositional and non-compositional potentials to define the space within a phase diagram where the accelerator algorithm has to operate.

mass fraction The fraction of a composition that consists of a system component, expressed in terms of mass.

material property model A surrogate model employed to determine a material property based on a system state or an equilibrium state.

material property A physical or thermochemical property that is not provided by equilibrium calculation software, but can be estimated by a surrogate material property model.

mesh cell When the domain of a multiphysics model is subdivided into many small discrete regions, each region is called a mesh cell.

model Mathematical representation of a physical system. Collective for multiphysics and process models and does not refer to thermochemical solution model, unless stated.

molar ratio The ratio between the amounts, measured in moles, of chemical substances.

mole fraction The ratio of a given system component's amount to the amount of all system components in a composition. The ratio is expressed as a fraction $\in [0, 1]$ and the amounts are measured in moles.

multi-phase In the presence of multiple phases.

multi-component In the presence of multiple system components.

multiphysics model A mathematical model that describes multiple coupled physical phenomena within a geometrically discretised representation of a physical system. Physical phenomena include, but are not limited to heat transfer, fluid flow, structural dynamics, electromagnetism, and chemical potential and is typically solved using methods such as the finite element or finite volume methods.

non-compositional potential tie simplex A tie simplex that has the same non-compositional potentials as the system state.

non-compositional potential index The index of a non-compositional potential in the uniform structured reference frame.

non-compositional potential index array The set of indices of all non-compositional potentials in the uniform structured reference frame.

non-compositional potential The ability for a chemical substance to undergo a chemical change due to differences in non-compositional properties such as temperature and pressure.

OPF feature functional dimensionality The dimensionality of the space that an OPF feature can traverse – functions in.

OPF feature geometric dimensionality The dimensionality of the geometrical feature that represents the OPF feature.

OPF boundary Abbreviation for one-phase fraction boundary. A phase region boundary of a multi-phase region that is associated with only one of the stable phases and is the only phase that has a non-zero amount at the phase region boundary. The phase therefore has a phase fraction of one.

OPF region Abbreviation for one-phase fraction region. An entire phase region where there is only one stable phase and it has a non-zero amount. The phase therefore has a phase fraction of one.

OPF feature An OPF (one-phase fraction) feature is a single-phase region or single-phase boundary of a multi-phase region. In these features more than one phase can be stable, but only one phase can have a non-zero amount and therefore a phase fraction of one.

performance The measure of the acceleration achieved and level of accuracy of an accelerator algorithm in comparison to direct equilibrium calculations. Measured during a performance test.

performance requirement The level of performance, both acceleration and accuracy, that is required for the accelerator algorithm to be considered successful.

performance criteria A set of standards by which performance of the accelerator algorithm is measured.

performance test Test that is performed to determine the performance of the accelerator algorithm, both acceleration achieved and the level of accuracy.

phase A homogeneous quantity of matter that has uniform physical and chemical characteristics. A phase can consist of a single substance (pure substance) or a mixture of substances (solution phase).

phase region section A section made through a phase region cell at the non-compositional potentials of the system state.

phase fraction difference The difference between a phase's calculated and interpolated phase fraction in the calculated and interpolated equilibrium states.

phase composition difference The difference between a phase's calculated and interpolated composition in the calculated and interpolated equilibrium states.

phase region cell A discrete portion of a phase region created during the discretisation algorithm by combining tie simplices that are within a specified tolerance.

phase property Physical or thermochemical property associated with a stable phase.

phase component A component in addition to the system components that a phase can consist of, such as an electron.

phase constituent The base chemical substances that a phase is composed of.

phase-simplex A simplex within a tie simplex constructed between the system composition and the compositions of all phases of the tie simplex except for the phase to whom the phase simplex belongs to. A phase simplex is used in the lever rule to determine the phase fraction of the phase.

phase composition constraint A conservation constraint between the concentrations of elements that governs the composition of a phase.

phase region boundary The boundary of a phase region wherein the same combination of phases are found to be in equilibrium.

phase region A region in a system where the same combination of phases are found to be in equilibrium. This usually spans over a range of system component concentrations, temperatures, and pressures.

phase diagram geometry The geometric features and the rules, such as the Gibbs phase rule, that governs the dimensionality of these features, that describes the relationship between a system state and the resulting equilibrium state.

phase diagram A projection of Gibbs energy relationships onto temperature-pressure-composition space as geometric maps of the limits of stability of various phases in a thermochemical system. It can be used to geometrically indicate what the equilibrium state of a system will be for a given system state.

phase composition The set of system component concentrations that a phase is composed of.

phase fraction The ratio of a given phase's amount to the amount of all phases present at equilibrium. The ratio is expressed as a fraction $\in [0, 1]$.

physical property error Error made on a physical property between a direct equilibrium calculation and interpolated equilibrium calculation.

physical property A material property that describes the relationship between the composition, temperature, and pressure of a substance and its physical attributes.

populate To add data to a database.

process model A mathematical model that describes the energy and mass flows of a thermochemical process.

property Collective for physical and thermochemical properties.

property error Collective for the error made on a physical or thermochemical property

pure substance A chemical substance with both definite and constant composition and distinct chemical properties.

recall To search and retrieve data from a database that satisfies certain criteria.

recall set A set of data that have been recalled from the database that met a specific search requirement.

recall algorithm The sequence of rules and instructions followed to search and retrieve data from a database that satisfies certain criteria.

reference frame cell A single discrete cell in the uniform structured reference frame.

region Refers to phase region.

simplex The generalisation of a triangle in kD space, which has $\mathrm{k}+1$ vertices.

simplex geometry The geometric principles of a simplex.

single-phase boundary A phase region boundary of a multi-phase region that is associated with only one of the stable phases and is the only phase that has a non-zero amount at the phase region boundary. More commonly referred to as a OPF boundary.

single-phase region An entire phase region where there is only one stable phase and it has a non-zero amount. More commonly referred to as a OPF region.

solution A homogeneous mixture of two or more chemical substances.

stable A phase is considered stable when its activity is equal to one: $a^{\sigma}_{\varphi} = 1$

storage space The physical space required on computer hardware to store data.

storage algorithm The sequence of rules and instructions followed to store data so that it can be recalled efficiently at a later stage.

store To add data to a database.

stored data Data that has been stored in a database.

system Refers to a thermochemical system.

system equilibrium state The equilibrium state at the specified system state.

system physical property Refers to a physical property associated with the system state.

system property Collective for physical and thermochemical properties associated with the system state.

system thermochemical property Refers to a thermochemical property associated with the system state.

system geometric dimensionality The number of dimensions of the phase diagram that describes the thermochemical system.

system compositional constraint A constraint between the concentrations of elements that govern the compositions of independent system components.

system state The combination of compositional and non-compositional potentials that defines the state of the system where the equilibrium state is to be calculated.

system component The base chemical substances that participate in an equilibrium reaction. A pure substance or compound can be a system component.

system composition Refers to the composition, or concentration of system components, of the system state.

temperature-phase composition coordinate A coordinate created at the vertex of a tie simplex, consisting of the tie simplex's temperature and the composition of the phase present at the vertex.

temperature tolerance The tolerance applied to tie simplex temperatures to determine valid tie simplices that can be combined into a phase region cell. The temperature tolerance is also used to define the intervals of the the uniform structured reference frame used in the database.

test Collective for functionality and performance tests.

thermochemical property error Error made on a thermochemical property between a direct equilibrium calculation and interpolated equilibrium calculation.

thermochemical system A series of phases, consisting of the same system components, that are found in different combinations with one another under varying temperature, pressure, and system component concentrations.

thermochemical database A self-consistent database that contains thermochemical data for thermochemical solution models of all potentially stable phases in the system. A combination of thermodynamic solution models and thermochemical data is used to mathematically describe the Gibbs energy of a system.

thermochemical property A material property that describes the relationship between the heat applied or the temperature of a substance and its chemical properties.

thermochemical solution model A model employed in equilibrium calculation software that describes the Gibbs energy relationship of a system for given temperatures, pressures, and system component concentrations.

thermodynamic property A material property that describes the relationship between the heat or work applied to a substance and its physical attributes and chemical properties. Considered as a collective for physical and thermochemical properties.

tie simplex A simpex within a phase region whose vertices correspond to the compositions of all phases present at equilibrium. All the phase compositions are at the same compositional and non-compositional potentials and are tied together by the simplex.

tolerance Collective for temperature and composition tolerances.

triangulation A method to subdivide a feature into smaller discrete triangular shapes; a surface into triangles, a volume into tetrahedrons, etc.

uniform structured reference frame A grid that divides a phase diagram into cells of uniform size so that the cells can be used to assign indices to features such as tie simplices and phase region cells. These indices are used as a reference frame when the tie simplices or and phase region cells are stored to the database or need to be recalled at a later stage. The cell size is determined by the temperature and composition tolerances.

univariant A geometrical feature in a phase diagram that has a degree of freedom of one.

Acronyms

OPF one-phase fraction

Nomenclature

Thermochemical System Objects

Identifier	Count	
σ		the system
ε	$\hat{\varepsilon}$	chemical element
ζ^{σ}	$\hat{\zeta}^{\sigma}$	system compositional constraint
ϵ	$\hat{\epsilon}$	independent system component
φ	$\hat{\varphi}$	phase
θ^{φ}	$\hat{\theta}^{\varphi}$	phase constituent
ε^{φ}	$\hat{\varepsilon}^{\varphi}$	phase component
ζ^{φ}	$\hat{\zeta}^{\varphi}$	phase compositional constraint
ϵ^{φ}	$\hat{\epsilon}^{\varphi}$	independent phase component
ς	$\hat{\varsigma}$	chemical specie

Compositional Quantities

Scalar	Array	
n	$\mathbf{n}$	amount (mol)
x	$\mathbf{x}$	amount fraction (mol mol^{-1})
a	$\mathbf{a}$	activity
μ	$\boldsymbol{\mu}$	chemical potential (J mol^{-1})
ι^{ϵ}	$\mathbf{I}^{\epsilon}$	storage reference frame index

Degrees of Freedom

f	of phase region
f'	of phase region with one fixed ψ
f''	of phase region with two fixed ψ
f^{φ}	of phase φ considered in isolation

Non-compositional Potentials

ψ	non-compositional potential identifier
$\hat{\psi}$	number of non-compositional potentials
$\boldsymbol{\Psi}$	$\begin{bmatrix} T & p & \dots & \psi^{\sigma}_{\hat{\psi}} \end{bmatrix}$ array
$\mathbf{I}^{\boldsymbol{\Psi}}$	$\begin{bmatrix} \iota_T & \iota_p & \dots & \iota_{\psi^{\sigma}_{\hat{\psi}}} \end{bmatrix}$ storage reference frame index array

T	Temperature (K)
p	Pressure (Pa)

Thermochemical Quantities

τ	thermochemical quantity identifier
$\hat{\tau}$	number of thermochemical quantities
$\mathbf{T}$	$\begin{bmatrix}\bar{V} & \bar{H} & \bar{S} & \bar{G}\end{bmatrix}$ array

$\bar{V}$	molar volume ($\mathrm{m^3\,mol^{-1}}$)
$\bar{H}$	molar enthalpy ($\mathrm{J\,mol^{-1}}$)
$\bar{S}$	molar entropy ($\mathrm{J\,K^{-1}\,mol^{-1}}$)
$\bar{G}$	molar Gibbs energy ($\mathrm{J\,mol^{-1}}$)

Physical Properties

ρ	physical property identifier
$\hat{\rho}$	number of physical properties
$\mathbf{P}$	$\begin{bmatrix}\bar{C}_p & \rho & \mu & \kappa & \ldots & \rho_{\hat{\rho}}\end{bmatrix}$ array

$\bar{C}_p$	molar heat capacity, constant pressure ($\mathrm{J\,K^{-1}\,mol^{-1}}$)
ρ	mass density ($\mathrm{kg\,m^{-3}}$)
μ	viscosity (P)
κ	thermal conductivity ($\mathrm{W\,m^{-1}\,K^{-1}}$)

Dimensionality

$\mathbb{D}^{\sigma}_{\mathrm{g}}$	system geometric dimensionality
$\mathbb{D}^{\mathcal{B}}$	OPF boundary dimensional characteristics
$\mathbb{D}^{\mathcal{B}}_{\mathrm{f}}$	OPF boundary functional dimensionality
$\mathbb{D}^{\mathcal{B}}_{\mathrm{g}}$	OPF boundary geometric dimensionality

Storage Reference Frame

$\mathcal{D}$	system's phase diagram
$\delta^{\mathcal{D}}$	reference frame cell of phase diagram

Thermochemical Data Structures

base notation

$\mathbf{X}$		specified state (sp., independent)
$\mathbf{Y}$	$\tilde{\mathbf{Y}}$	equilibrium state (eq., dependent)

Column 1: directly calculated values

Column 2: interpolated equilibrium values indicated with tilde

Interpolated values in arrays only apply to interpolated data structures.

E.g $\tilde{\mathbf{x}}^{\varphi_i}_{\epsilon}$ applies to $\tilde{\mathbf{Y}}^{\varphi_i}$, and $\mathbf{x}^{\varphi_i}_{\epsilon}$ to $\mathbf{Y}^{\varphi_i}$.

system (σ)

$\mathbf{X}^{\sigma}$		$\begin{bmatrix}\mathbf{x}^{\sigma}_{\epsilon} & \mathbf{\Psi}^{\sigma}\end{bmatrix}$
$\mathbf{Y}^{\sigma}$	$\tilde{\mathbf{Y}}^{\sigma}$	$\begin{bmatrix}\tilde{\mathbf{x}}^{\sigma}_{\varphi} & \tilde{\boldsymbol{\mu}}^{\sigma}_{\epsilon} & \tilde{\mathbf{T}}^{\sigma} & \tilde{\mathbf{P}}^{\sigma} & \begin{bmatrix}\tilde{\mathbf{Y}}^{\varphi_1} & \ldots & \tilde{\mathbf{Y}}^{\varphi_{\hat{\varphi}}}\end{bmatrix}\end{bmatrix}$

phase i (φ_i)

$\mathbf{Y}^{\varphi_i}$	$\tilde{\mathbf{Y}}^{\varphi_i}$	$\begin{bmatrix}\tilde{\mathbf{x}}^{\varphi_i}_{\epsilon} & \tilde{\mathbf{T}}^{\varphi_i} & \tilde{\mathbf{P}}^{\varphi_i}\end{bmatrix}$

Geometric Data Structures

simplex $({}_{\mathrm{k}}\Delta)$

$\nu({}_{\mathrm{k}}\Delta)$	content of simplex

tie simplex $(\Delta^{\mathcal{R}_{\mathrm{i}}})$ in phase region i

t		tie simplex identifier
$\Delta^{\mathcal{R}_{\mathrm{i}}}$	$\tilde{\Delta}^{\mathcal{R}_{\mathrm{i}}}$	$\left[\boldsymbol{\Psi}^{\sigma} \quad \tilde{\boldsymbol{\mu}}_{\epsilon}^{\sigma} \quad \tilde{\mathbf{x}}_{\epsilon}^{\mathcal{R}_{\mathrm{i}}} \quad \tilde{\mathbf{T}}^{\mathcal{R}_{\mathrm{i}}} \quad \tilde{\mathbf{P}}^{\mathcal{R}_{\mathrm{i}}}\right]$
$\hat{\Delta}^{\mathcal{R}_{\mathrm{i}}}$		number of tie simplices
$\mathbf{x}_{\epsilon}^{\mathcal{R}_{\mathrm{i}}}$	$\tilde{\mathbf{x}}_{\epsilon}^{\mathcal{R}_{\mathrm{i}}}$	$\left[\tilde{\mathbf{x}}_{\epsilon}^{\varphi_1} \quad \ldots \quad \tilde{\mathbf{x}}_{\epsilon}^{\varphi_{\hat{\varphi}^{\mathcal{R}_{\mathrm{i}}}}}\right]$ composition vertices
$\mathbf{T}^{\mathcal{R}_{\mathrm{i}}}$	$\tilde{\mathbf{T}}^{\mathcal{R}_{\mathrm{i}}}$	$\left[\tilde{\mathbf{T}}^{\varphi_1} \quad \ldots \quad \tilde{\mathbf{T}}^{\varphi_{\hat{\varphi}^{\mathcal{R}_{\mathrm{i}}}}}\right]$ thermochemical quantities
$\mathbf{P}^{\mathcal{R}_{\mathrm{i}}}$	$\tilde{\mathbf{P}}^{\mathcal{R}_{\mathrm{i}}}$	$\left[\tilde{\mathbf{P}}^{\varphi_1} \quad \ldots \quad \tilde{\mathbf{P}}^{\varphi_{\hat{\varphi}^{\mathcal{R}_{\mathrm{i}}}}}\right]$ physical properties
$\{\Delta^{\mathcal{R}_{\mathrm{i}}}\}_{\gamma}$		set of tolerable tie simplices
$\tilde{\Delta}_{\psi}^{\mathcal{R}_{\mathrm{i}}}$		interpolated iso-ψ tie simplex
$\hat{\Delta}_{\psi}^{\mathcal{R}_{\mathrm{i}}}$		number of interpolated iso-ψ tie simplices
$\Delta_{\varphi_{\mathrm{j}}}^{\mathcal{R}_{\mathrm{i}}}$		phase simplex of phase j in the lever rule

phase region i $(\mathcal{R}_{\mathrm{i}})$

r	discretisation cell identifier
$\delta^{\mathcal{R}_{\mathrm{i}}}$	$\left[\Delta_1^{\mathcal{R}_{\mathrm{i}}} \quad \ldots \quad \Delta_{\mathrm{f}+1}^{\mathcal{R}_{\mathrm{i}}}\right]$ discretisation cell
$\delta_{\psi}^{\mathcal{R}_{\mathrm{i}}}$	$\left[\tilde{\Delta}_1^{\mathcal{R}_{\mathrm{i}}} \quad \ldots \quad \tilde{\Delta}_{\mathrm{f}+1-\hat{\psi}}^{\mathcal{R}_{\mathrm{i}}}\right]$ interpolated constant $\boldsymbol{\Psi}^{\sigma}$ cell
$\Omega^{\delta^{\mathcal{R}_{\mathrm{i}}}}$	convex hull of $\delta^{\mathcal{R}_{\mathrm{i}}}$

OPF feature of phase j in region i $(\mathcal{B}_{\mathrm{j}}^{\mathrm{i}})$

$\delta^{\mathcal{B}_{\mathrm{j}}^{\mathrm{i}}}$	$\left[\mathbf{Y}_1^{\varphi_{\mathrm{j}}} \quad \ldots \quad \mathbf{Y}_{\mathrm{f}+1}^{\varphi_{\mathrm{j}}}\right]$ discretisation segment
$\delta_{\psi}^{\mathcal{B}_{\mathrm{j}}^{\mathrm{i}}}$	$\left[\tilde{\mathbf{Y}}_1^{\varphi_{\mathrm{j}}} \quad \ldots \quad \tilde{\mathbf{Y}}_{\mathrm{f}+1-\hat{\psi}}^{\varphi_{\mathrm{j}}}\right]$ interpolated constant $\boldsymbol{\Psi}^{\sigma}$ segment
$\tilde{\mathbf{x}}_{\epsilon_{(\psi)}}^{\mathcal{B}_{\mathrm{j}}^{\mathrm{i}}}$	$\left[(\tilde{\mathbf{x}}_{\epsilon}^{\varphi_{\mathrm{j}}})_1 \quad \ldots \quad (\tilde{\mathbf{x}}_{\epsilon}^{\varphi_{\mathrm{j}}})_{\mathrm{f}+1-\hat{\psi}}\right]$ array of $\delta_{\psi}^{\mathcal{B}_{\mathrm{j}}^{\mathrm{i}}}$ phase compositions

Barycentric Coordinate Interpolation

$\beta_{\mathrm{i}}^{\Delta}$	barycentric coordinate weight within $\tilde{\Delta}^{\mathcal{R}_{\mathrm{i}}}$
$\beta_{\mathrm{j}}^{\mathcal{B}}$	barycentric coordinate weight within $\delta_{\psi}^{\mathcal{B}_{\mathrm{j}}^{\mathrm{i}}}$
$\boldsymbol{\beta}^{\Delta}$	Array of $(\hat{\varphi} - 1)$ $\beta_{\mathrm{i}}^{\Delta}$ within $\tilde{\Delta}^{\mathcal{R}_{\mathrm{i}}}$
$\boldsymbol{\beta}^{\mathcal{B}}$	array of $(\hat{\Delta}_{\psi}^{\mathcal{R}} - 1)$ $\beta_{\mathrm{j}}^{\mathcal{B}}$ within $\delta_{\psi}^{\mathcal{B}_{\mathrm{j}}^{\mathrm{i}}}$
$\mathbf{w}$	barycentric coordinate weight vector of $\delta_{\psi}^{\mathcal{R}_{\mathrm{i}}}$
$\mathbf{B}$	matrix of $\hat{\varphi}$ of $\tilde{\mathbf{x}}_{\epsilon_{(\psi)}}^{\mathcal{B}_{\mathrm{j}}}$

Tolerances and Performance Measurement

$\Delta T^{\mathcal{D}}$	Temperature tolerance
$\Delta x_{\epsilon}^{\mathcal{D}}$	Composition tolerance
AF	Acceleration factor
$\Delta\mathrm{t}(\mathrm{direct})$	Direct calculation elapsed time
$\Delta\mathrm{t}(\mathrm{accelerator})$	Accelerated calculation elapsed time
$\mathrm{d}x_{\epsilon}^{\varphi_\mathrm{i}}$	Phase i composition error
$\mathrm{d}x_{\varphi_\mathrm{i}}^{\sigma}$	Phase i fraction error
$E\rho^{\sigma}$	Physical property error
$E\tau^{\sigma}$	Thermochemical property error

Part I

Introduction

Chapter 1

Background

1.1 Project Background

Incorporating multi-component, multi-phase, high-temperature, complex chemical equilibrium calculations, hereafter simply referred to as equilibrium calculations, into multiphysics models and process models, hereafter simply referred to as models, can provide valuable insight into real-life industry-relevant processes that current modelling or measurements cannot. The word "model" in this work does not refer to thermochemical solution models. Equilibrium calculations provide phase fractions, phase compositions, physical properties such as heat capacity and thermochemical properties such as enthalpy as functions of system temperature, pressure, and composition. These equilibrium calculations are performed with self-consistent thermochemical databases developed using experimental data. Some additional material properties that may be required by models are not provided by equilibrium calculations, however. Material property models can estimate additional properties such as viscosity, density, thermal and electrical conductivity based on equilibrium calculation results. Incorporating equilibrium calculations allow models to better represent industry-relevant systems, providing more insight into complex systems that would otherwise be difficult or infeasible to obtain. This can enable better design assumptions and more informative decisions to be made when designing new, or improving current processes and equipment, and reduce the need for a trial-and-error approach usually followed in pyrometallurgy and material science.

Although this approach can provide more accurate material properties to models, including large numbers of equilibrium calculations into models are computationally expensive and are usually omitted or incorporated in a simplified manner. As shown in Figure 1.1, there is a high-order non-linear relationship between a system's number of components and equilibrium calculation computation time. Large multiphysics models can have thousands or even millions of mesh cells and an equilibrium calculation may need to be performed in each mesh cell for every iteration. Figure 1.2a shows how single iteration computation time increases linearly as more mesh cells are used in a multiphysics model. Multiple iterations are, however, required for convergence and multiple time steps when a transient model is considered. Figure 1.2b shows how total computation time increases linearly as more iterations are required. In cases with many system components, mesh cells, iterations, and time steps, performing all of the required equilibrium calculations can result in infeasible solving times of months or even years. It should be noted that the thermochemical databases and solution models used in the equilibrium calculations, and the computer on which the calculations are performed, have an influence on the computation time, but the trend remains the same. Equilibrium calculations can however be successfully implemented into models when the number of calculations that need to be performed are small enough, with the work of Agelet de Saracibar, Chiumenti, and Cervera (2006), Blond et al. (2014), Marin-Alvarado (2015), and Corcoran, Kaye, and Piro (2016) as examples.

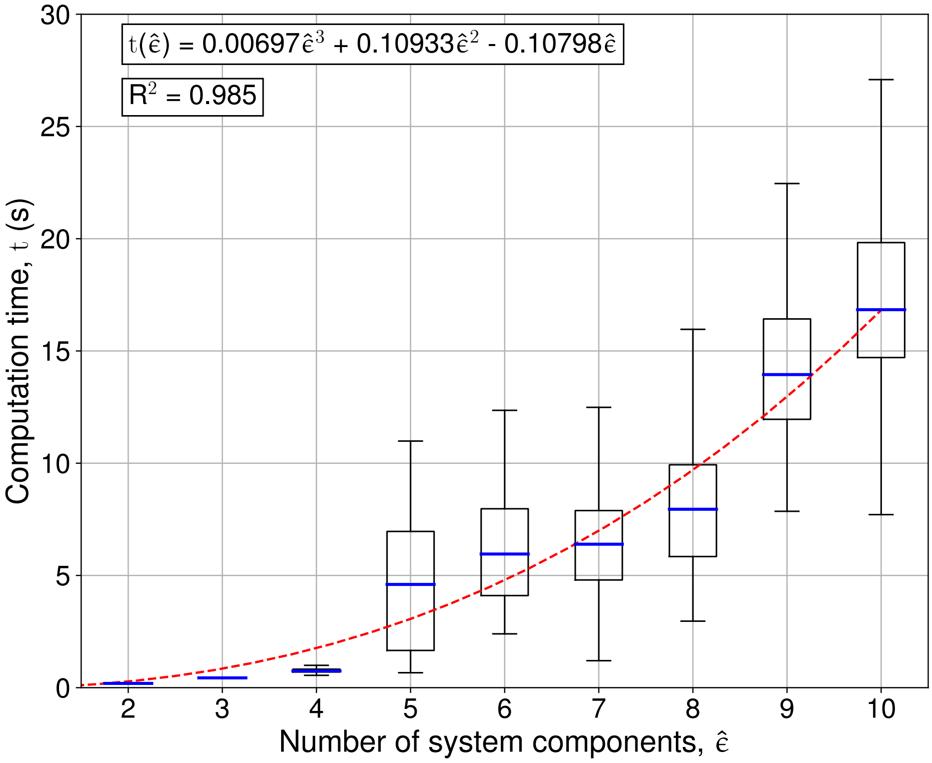
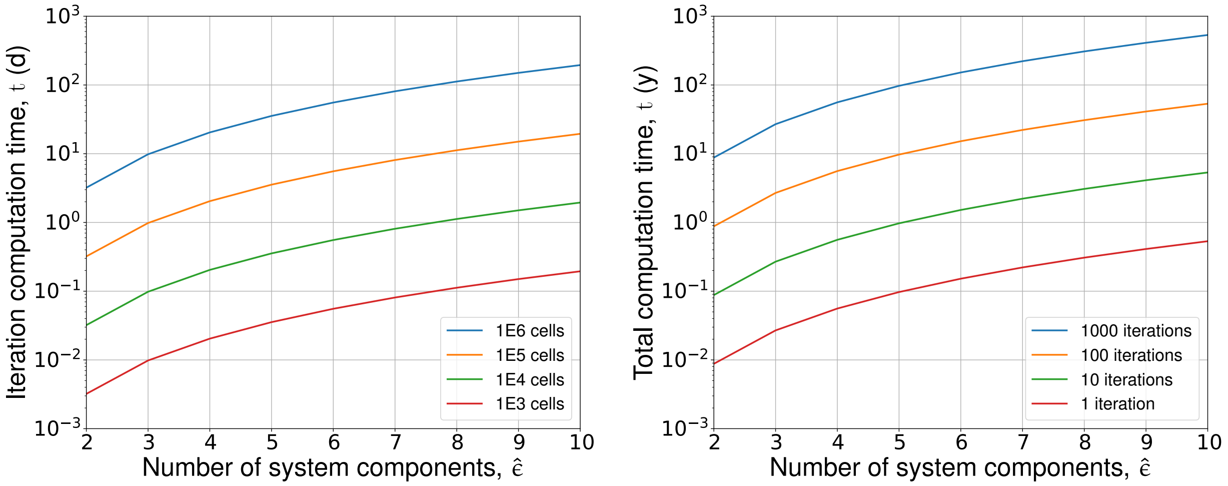


Figure 1.1: Third-order increase in equilibrium calculation computation time with number of system components. Initial system components were Fe-O, and Si, Al, Mg, Ca, N, Mn, Na, K were added. 250 randomly distributed equilibrium calculations per $\hat{\epsilon}$-component system. Equilibrium calculations were performed with ChemAppPy (Ex Mente Technologies 2019) and data from the FToxid database in FactSage 7.2.0 (Bale et al. 2016). The calculations were performed in series on an Intel Core i7-3770 CPU (4 cores) with a clock speed of 3.4 GHz, and with 8 GB of RAM.



(a) Single iteration computation time of a multiphysics model, in days, as the number of mesh cells and system components increase.

(b) Total computation time, in years, as a multiphysics model with 1 000 000 mesh cells is solved for a number of iterations.

Figure 1.2: Scale of the computation problem when equilibrium calculations are included in a multiphysics model.

The same high-order non-linear relationship between a system's number of components and equilibrium calculation computation time is seen in process models. The difference is that these models do not have mesh cells to consider, but still need to perform many equilibrium calculations when the process is divided into multiple equilibrium reactors (Zietsman and Pistorius 2006). Many equilibrium calculations can be required to ensure convergence to steady state, and even more so for simulations of transient process models and when these models are used in optimisation routines. Solving times can easily become infeasible.

Figure 1.3 illustrates how equilibrium calculation software is directly integrated with a model. The model provides the equilibrium calculation software with a system state $\mathbf{X}^\sigma$ at which equilibrium needs to be calculated. The equilibrium calculation software returns the calculated equilibrium state $\mathbf{Y}^\sigma$. This is referred to as performing a direct equilibrium calculation. The computation time of direct equilibrium calculations need to be reduced to make it more feasible to include these calculations into models; an equilibrium calculation accelerator needs to be implemented, as illustrated in Figure 1.3. Conceptually, an equilibrium calculation accelerator, together with stored data in its database, acts as an intermediary between a model and equilibrium calculation software. In this work, the word "database" refers to the collection of calculated thermochemical properties and not to the thermochemical database that contains thermochemical solution models. The accelerator's database can be populated prior to solving the model, or while the model is being solved (in-situ). When possible, the accelerator algorithm recalls stored data and uses it in an interpolation algorithm to determine an interpolated equilibrium state $\tilde{\mathbf{Y}}^\sigma$ rather than performing the more computationally expensive direct equilibrium calculation to obtain the calculated equilibrium state $\mathbf{Y}^\sigma$.

Several acceleration methods have previously been developed to improve the computational efficiency of including equilibrium calculations into models so that more comprehensive simulations can be performed in a reasonable time. The strengths of these methods can be combined, together with fundamental thermochemical theory, to develop a generic accelerator algorithm that can be incorporated into a system with any number of components. This is the focus of this work.

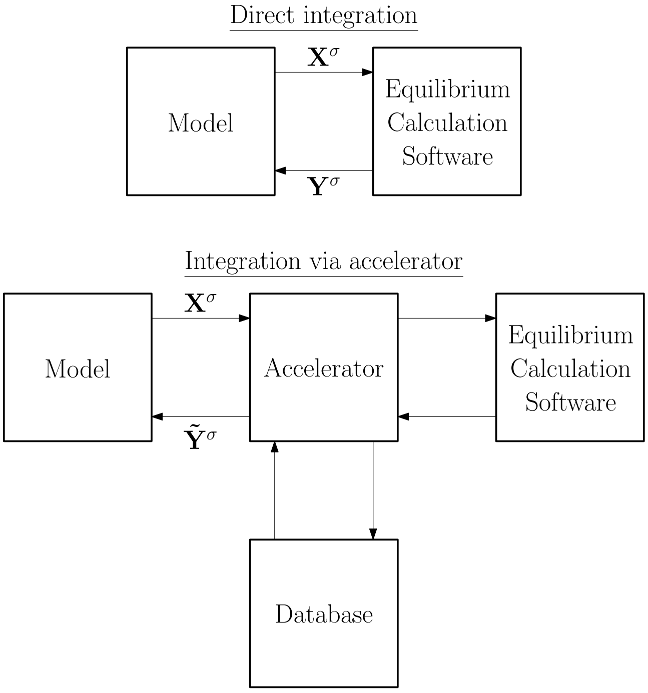

Direct integration
$\mathbf{X}^\sigma$
Model
Equilibrium Calculation Software
$\mathbf{Y}^\sigma$
Integration via accelerator
$\mathbf{X}^\sigma$
Model
Accelerator
Equilibrium Calculation Software
$\tilde{\mathbf{Y}}^\sigma$
Database

Figure 1.3: Schematic representation of direct equilibrium calculation and integration of an accelerator between a model and equilibrium calculation software. Adapted from Zietsman (2016).

1.2 Document Overview

The document is structured as follows:

Nomenclature: Because the accelerator algorithm was developed to be generic, the foundational concepts, as well as the discretisation, storage and recall, and interpolation algorithms are described with generic symbolic expressions. Throughout the text, some case-specific examples are used to illustrate the symbolic expressions in the context of applicable concepts or algorithms. A nomenclature is presented at the beginning of this document in which all symbols that are used in the text, are described.

Part I: Introduction Background on the project is provided in Chapter 1, outlining the problem at hand. The research focus is explained in Chapter 2 to direct the approach that has to be followed to address the identified problem. Foundational concepts required for the algorithm development is discussed in Chapter 3 with a short demonstration of the concepts applied to a two- and three-component system. This chapter forms part of the work in Roos and Zietsman (2022b). Chapter 4 discusses the numerical and mathematical nature of equilibrium calculations and why they are so computationally expensive.

Part II: Literature Review A few cases are discussed in Chapter 5 showing the importance of including equilibrium calculations into models. Chapter 6 discusses a number of acceleration methods found in literature that were developed to improve the efficiency of equilibrium calculations and make it more feasible to include them into multiphysics and process models. Both of these chapters form part of the work in Roos and Zietsman (2022a).

Part III: Research Approach and Methodology The approach followed during this project is outlined in Chapter 7 and the methods used to accomplish the steps are discussed in Chapter 8.

Part IV: Accelerator Algorithm The accelerator algorithm development is presented in Chapter 9 and forms part of the work in Roos and Zietsman (2022b). Results of the accelerator algorithm functionality and performance tests in two- and three-component systems are presented in Chapter 10. Performance test results of applying the accelerator algorithm to four- and five-component systems of industry-related processes are discussed in Chapter 11. The results of all tests performed in two- to five-component systems form part of the work in Roos, Bogaers, and Zietsman (2023).

Part V Closure Concluding remarks are made in Chapter 12 of the developed accelerator algorithm as well as its functionality and performance. Chapter 13 presents recommendations regarding improvements that could be made to the implementation. Alterations and additions that could be made to the developed accelerator algorithm to improve its capabilities, accuracy, and performance as well as reduce its storage space requirements are also presented.

Appendices The detailed derivation of the system of non-linear equations used in the interpolation algorithm is discussed in Appendix A. Distribution plots of the performance test results for all the investigated two- and three-component systems are displayed in Appendix B.

Chapter 2

Research Focus

Before any research and development is done, the focus and direction of the research effort in addressing the identified problem has to be established.

2.1 Problem Statement

Equilibrium calculations have been incorporated into models before but in a simplified manner where the thermochemical systems were represented by a small number of components. There is a need to include complex multi-component equilibrium calculations into models that better represent industry-relevant thermochemical systems. This would provide valuable insight into real-life processes that current modelling or measurements cannot. These equilibrium calculations are however computationally expensive and result in model solving times of months or even years which are simply infeasible. A method of including equilibrium calculations into models more efficiently is therefore required.

2.2 Research Topic

A new generic equilibrium calculation accelerator algorithm was developed that uses a system's phase diagram to map the thermochemical system to geometric space by storing calculated physical and thermochemical properties in-situ for later recall and interpolation. Linear interpolation in geometric space is less computationally expensive than Gibbs energy minimisation in thermochemical space to determine equilibrium. This will enable the efficient inclusion of equilibrium calculations in models.

2.3 Research Purpose

The purpose of this research is to develop deeper insight into pyrometallurgical processes to enable better design assumptions and more informative decisions to be made when designing new, or improving current processes and equipment with the aim of increasing throughput and reducing emissions. This can be accomplished by improving model representations of industry-relevant systems by efficiently including complex multi-component equilibrium calculations through the integration of an accelerator.

2.4 Research Significance

Being able to efficiently include equilibrium calculations into models will provide more insight into complex multi-component systems that would previously be difficult or infeasible to obtain. This will enable better design assumptions and more informative decisions to be made when designing new, or improving current processes and equipment.

The migration of the iron- and steelmaking industry to more environmentally-friendly technologies in a strict time frame is a perfect example. Regulations have been imposed to reduce emissions, which is leading to the development of new and alternative processes and technologies, some of which substitute carbon with natural gas and hydrogen as reducing agents and utilise renewable energy sources (Zhang et al. 2021). With stringent deadlines by when emissions have to be reduced and new processes be implemented, there is limited time to build pilot plants to test newly developed processes and even less time for demonstration plants. There is, however, time to build models and use them to rapidly investigate multiple new process concepts, weed out concepts that do not meet requirements, and optimise feasible concepts before new industrial plants are built.

2.5 Research Requirements

The aim of this project was to conceptualise, develop, and demonstrate the functionality and performance of a new accelerator algorithm, **not to produce and deliver optimised software that can be used in production**. For this stage of development that this work is aimed at, the following are required from the accelerator algorithm:

Generic algorithm: The accelerator algorithm should work for systems with any number of components, even though the performance tests would only extend to five-component systems in this work.

Thermodynamically consistent: The accelerator algorithm has to be strongly based on fundamental thermochemical theory to provide security for the decisions taken when the system is discretised and interpolation is performed with the stored data.

Storage requirement: The accelerator algorithm should produce and store as little data into the database as possible.

User control: The accelerator algorithm has to provide the user control over the balance between acceleration and accuracy.

Performance: The accelerator algorithm should be capable of achieving significant acceleration while maintaining acceptable accuracy.

2.6 Research Contribution

A new generic equilibrium calculation accelerator algorithm was developed that can be applied to any system, regardless of the number of components. The generality of the accelerator algorithm comes from being based on sound fundamental thermochemical theory such as phase diagram geometry, the Gibbs phase rule, and the lever rule. The accelerator algorithm was tested in a number of two- to five-component systems and it was demonstrated that the accelerator algorithm could provide physical and thermochemical properties to models at significant levels of acceleration at an acceptable level of accuracy. Other developed generic acceleration methods found in literature have not been demonstrated to work for five-component systems or higher.

Chapter 3

Foundational Concepts

Before the accelerator algorithm is discussed, the key foundational concepts that it is based on are presented in this chapter.

3.1 The Gibbs Phase Rule

For a phase region with $\hat{\varphi}$ phases in a system with $\hat{\epsilon}$ independent system components and $\hat{\psi}$ varying non-compositional potentials the Gibbs phase rule (Equation (3.1)) specifies the number of degrees of freedom f, which is the number of independent intensive variables that need to be specified, to fully define the system (Callister and Rethwisch 2011).

$$\mathrm{f} = \hat{\epsilon} - \hat{\varphi} + \hat{\psi} \tag{3.1}$$

As shown in Equation (3.2), the number of independent system components $\hat{\epsilon}$ is equal to the number of chemical elements $\hat{\varepsilon}$ minus the number of system compositional constraints $\hat{\zeta}^{\sigma}$ (Hack 2008). An example of this is the CaO$-$SiO$_2$ system in which there are $\hat{\varepsilon} = 3$ chemical elements, but the Ca:O and Si:O molar ratios are fixed at sufficiently high oxygen potentials ($\hat{\zeta}^{\sigma}$=1). This yields only $\hat{\epsilon} = 2$ independent system components.

$$\hat{\epsilon} = \hat{\varepsilon} - \hat{\zeta}^{\sigma} \tag{3.2}$$

Because of the importance of system compositional constraints to the accelerator algorithm, Equation (3.3) is used as the preferred form of the Gibbs phase rule in this work.

$$\mathrm{f} = \hat{\varepsilon} - \hat{\zeta}^{\sigma} - \hat{\varphi} + \hat{\psi} \tag{3.3}$$

Temperature and pressure are the most common non-compositional potentials, which generally means that $\hat{\psi} = 2$. When pressure is fixed the degrees of freedom are reduced by one. In this case, only temperature contributes to f, and is indicated here as f' and calculated with Equation (3.4). The accelerator developed in this work is aimed at isobaric systems and therefore uses this form of the Gibbs phase rule.

$$\mathrm{f}' = \hat{\varepsilon} - \hat{\zeta}^{\sigma} - \hat{\varphi} + 1 \tag{3.4}$$

For isothermal conditions in isobaric systems $\hat{\psi} = 0$, and the degrees of freedom are indicated as f'' and calculated with Equation (3.5). This form of the Gibbs phase rule is used in the accelerator when calculations are performed on isotherms in isobaric systems.

$$\mathrm{f}'' = \hat{\varepsilon} - \hat{\zeta}^{\sigma} - \hat{\varphi} \tag{3.5}$$

3.2 Phase Diagram Geometry

The accelerator algorithm uses phase diagrams, which are projections of Gibbs energy relationships onto temperature-composition space, as geometric maps of the limits of stability of various phases in thermochemical systems. Phase regions, phase region boundaries, univariants, and invariant points in phase diagrams are all geometric features.

The algorithm only captures data associated with single-phase features. These include single-phase regions, and single-phase boundaries of multi-phase regions. In these features more than one phase can be stable ($a_{\varphi}^{\sigma} = 1$), but only one phase can have a non-zero amount ($n_{\varphi}^{\sigma} > 0$) and therefore a phase fraction of one ($x_{\varphi}^{\sigma} = 1$). These features are referred to as one-phase fraction (OPF) features; $\mathcal{B}$. OPF features provide composition vertices of tie simplices that are used to calculate phase fractions with the lever rule.

3.2.1 Single-phase Regions

As per the definition of an OPF feature, an entire single-phase region is such a feature, and therefore an OPF region. The red shaded areas in Figure 3.1 are examples.

3.2.2 Multi-phase Regions

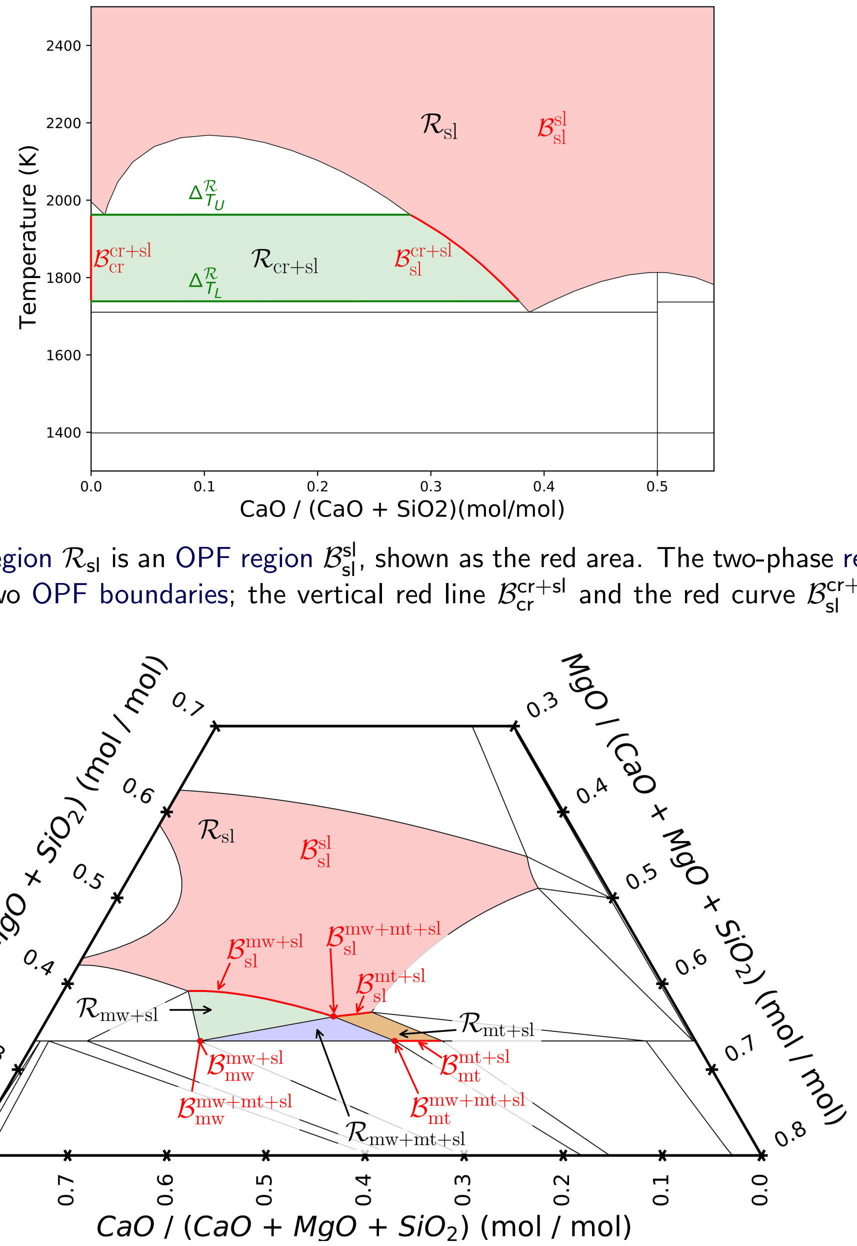
Some boundaries of a phase region satisfy the conditions of an OPF feature, and is referred to as an OPF boundary. In a two-phase region, as shown in Figures 3.1a and 3.1b, there are two OPF boundaries and in three-phase regions, as shown in Figure 3.1b, there are three OPF boundaries. Unlike OPF regions, OPF boundaries are not areas in two-component systems or volumes in three-component systems, but lower-dimensional objects such as curves or surfaces that bound their phase regions.

It should be noted that not all phase region boundaries are OPF boundaries. As seen in Figure 3.1a, there are two bounding tie simplices, $\Delta_{T_U}^{\mathcal{R}}$ and $\Delta_{T_L}^{\mathcal{R}}$. Such boundaries are not OPF boundaries seeing that more than one phase can have a non-zero amount under these conditions.

Generically, a $\hat{\varphi}$-phase region, with $\hat{\varphi} > 1$, in a $\hat{\varepsilon}$-component system has $\hat{\varphi}$ OPF boundaries. These boundaries are geometric objects such as lines, curves, surfaces, volumes, and hyper-volumes.

3.3 Dimensionality

An OPF feature's dimensional characteristics ($\mathbb{D}^{\mathcal{B}}$) determine the number of data points, and therefore the number of direct calculations, that are required for discretisation and interpolation. It is not sufficient to state that such a feature has a certain dimensionality; for example, that a phase region boundary is 1-dimensional. The system's geometric dimensionality ($\mathbb{D}_{\mathrm{g}}^{\sigma}$), and the OPF feature's geometric ($\mathbb{D}_{\mathrm{g}}^{\mathcal{B}}$) and functional dimensionalities ($\mathbb{D}_{\mathrm{f}}^{\mathcal{B}}$) need to be considered.



(a) The single-phase region $\mathcal{R}_{\text{sl}}$ is an OPF region $\mathcal{B}_{\text{sl}}^{\text{sl}}$, shown as the red area. The two-phase region $\mathcal{R}_{\text{cr+sl}}$, shown in green, has two OPF boundaries; the vertical red line $\mathcal{B}_{\text{cr}}^{\text{cr+sl}}$ and the red curve $\mathcal{B}_{\text{sl}}^{\text{cr+sl}}$. (sl: slag, cr: cristobalite)

(b) The single-phase region $\mathcal{R}_{\text{sl}}$ is an OPF region $\mathcal{B}_{\text{sl}}^{\text{sl}}$, shown as the red area. The two-phase region $\mathcal{R}_{\text{mw+sl}}$, shown as the green area, has two OPF boundaries; a red curve $\mathcal{B}_{\text{sl}}^{\text{mw+sl}}$ and a red point $\mathcal{B}_{\text{mw}}^{\text{mw+sl}}$. The two-phase region $\mathcal{R}_{\text{mt+sl}}$, shown as the orange area, has two OPF boundaries; a red curve $\mathcal{B}_{\text{sl}}^{\text{mt+sl}}$ and a red line $\mathcal{B}_{\text{mt}}^{\text{mt+sl}}$. The three-phase region $\mathcal{R}_{\text{mw+mt+sl}}$, shown as the blue area, has three OPF boundaries; three red points $\mathcal{B}_{\text{mw}}^{\text{mw+mt+sl}}$, $\mathcal{B}_{\text{mt}}^{\text{mw+mt+sl}}$, and $\mathcal{B}_{\text{sl}}^{\text{mw+mt+sl}}$. (sl: slag, mw: merwinite, mt: monticellite) (1750 K isotherm)

Figure 3.1: Example OPF features in isobaric two- and three-component systems.

3.3.1 System Geometric Dimensionality

The number of dimensions of the phase diagram that describes a thermochemical system is referred to as the system's geometric dimensionality $\mathbb{D}_{\mathrm{g}}^{\sigma}$. It is calculated with Equation (3.6).

$$\mathbb{D}_{\mathrm{g}}^{\sigma} = \hat{\varepsilon} - \hat{\zeta}^{\sigma} - 1 + \hat{\psi} \tag{3.6}$$

For the isobaric three-component system CaO−MgO−SiO$_2$, $\mathbb{D}_{\mathrm{g}}^{\sigma} = 4 - 1 - 1 + 1 = 3$. The system is therefore described by a 3D phase diagram.

3.3.2 OPF Feature Geometric Dimensionality

OPF features can be 1D lines or curves, 2D flat or curved surfaces, 3D volumes, 4D hypervolumes, etc. For the OPF feature of phase j of phase region i, $\mathcal{B}_{\mathrm{j}}^{\mathrm{i}}$, this is referred to as its geometric dimensionality $\mathbb{D}_{\mathrm{g}}^{\mathcal{B}_{\mathrm{j}}^{\mathrm{i}}}$, and it can be calculated with Equation (3.7).

$$\mathbb{D}_{\mathrm{g}}^{\mathcal{B}_{\mathrm{j}}^{\mathrm{i}}} = \mathrm{f} \tag{3.7}$$

For the $\mathcal{B}_{\text{sl}}^{\text{sl}}$ region in Figure 3.1a $\mathbb{D}_{\mathrm{g}}^{\mathcal{B}} = \mathrm{f} = 3 - 1 - 1 + 1 = 2$. This phase region is therefore a 2D surface. In the same system, the $\mathcal{B}_{\text{cr}}^{\text{cr+sl}}$ and $\mathcal{B}_{\text{sl}}^{\text{cr+sl}}$ boundaries of region $\mathcal{R}_{\text{cr+sl}}$ has $\mathbb{D}_{\mathrm{g}}^{\mathcal{B}} = 3 - 1 - 2 + 1 = 1$. The cristobalite boundary is therefore a line and the slag boundary a curve; both are 1D features.

Equation (3.7) does not describe all cases correctly for the three-component system in Figure 3.1b. For single-phase regions $\mathrm{f} = 4 - 1 - 1 + 1 = 3$, which correctly indicates that these OPF regions are represented by 3D volumes. In three-phase regions $\mathrm{f} = 4 - 1 - 3 + 1 = 1$, which also correctly shows that the OPF boundaries of these regions are represented by 1D lines or curves. In two-phase regions $\mathrm{f} = 4 - 1 - 2 + 1 = 2$, which states that the OPF boundaries of such regions are 2D surfaces. This is correct when the phases consist of 2 or 3 independent system components. Pure substances, however, exist as constant composition lines that only vary with temperature. A OPF boundary can therefore not be 2-dimensional ($\mathbb{D}_{\mathrm{g}}^{\mathcal{B}} = \mathrm{f} = 2$) if the phase only exists in 1D.

To correctly determine $\mathbb{D}_{\mathrm{g}}^{\mathcal{B}}$, the f for the phase region must be combined with f^{φ}; the degrees of freedom when considering phase φ in isolation. f^{φ} incorporates all the phase's phase composition constraints $\hat{\zeta}^{\varphi}$, and is calculated with Equation (3.8). OPF feature geometric dimensionality $\mathbb{D}_{\mathrm{g}}^{\mathcal{B}}$ is calculated correctly for all cases with Equation (3.9).

$$\mathrm{f}^{\varphi} = \hat{\varepsilon} - \hat{\zeta}^{\varphi} - 1 + \hat{\psi} \tag{3.8}$$

$$\mathbb{D}_{\mathrm{g}}^{\mathcal{B}} = \min\left(\mathrm{f}, \mathrm{f}^{\varphi}\right) \tag{3.9}$$

3.3.3 OPF Feature Functional Dimensionality

Finally, the dimensionality of the space that an OPF feature traverses needs to be considered. This is referred to as the feature's functional dimensionality $\mathbb{D}_{\mathrm{f}}^{\mathcal{B}}$, and it is calculated with Equation (3.10).

$$\mathbb{D}_{\mathrm{f}}^{\mathcal{B}} = \mathrm{f}^{\varphi} \tag{3.10}$$

For a 1D OPF boundary in a 3D system that curves in all 3 dimensions $\mathbb{D}_{\mathrm{f}}^{\mathcal{B}} = 3$. When the boundary is constrained to traverse a flat plane $\mathbb{D}_{\mathrm{f}}^{\mathcal{B}} = 2$. If the phase in question is a pure substance, it is constrained to a 1D straight line and $\mathbb{D}_{\mathrm{f}}^{\mathcal{B}} = 1$.

Region $\mathcal{R}_{\text{cr+sl}}$ in Figure 3.1a has two OPF boundaries, $\mathcal{B}_{\text{cr}}^{\text{cr+sl}}$ and $\mathcal{B}_{\text{sl}}^{\text{cr+sl}}$, both of which are 1D features, since $\mathbb{D}_{\mathrm{g}}^{\mathcal{B}} = \mathrm{f} = 1$. Cristobalite is a pure substance, which means that $\mathcal{B}_{\text{cr}}^{\text{cr+sl}}$ is constrained to a single composition coordinate and it only traverses one dimension, namely temperature; therefore, for $\mathcal{B}_{\text{cr}}^{\text{cr+sl}}$, $\mathbb{D}_{\mathrm{f}}^{\mathcal{B}} = \mathrm{f}^{\text{cr}} = 1$. In contrast, slag is a solution with two independent system components (CaO and SiO_2). The $\mathcal{B}_{\text{sl}}^{\text{cr+sl}}$ boundary's coordinates therefore vary in both composition and temperature, which means that it traverses 2D space and $\mathbb{D}_{\mathrm{f}}^{\mathcal{B}} = \mathrm{f}^{\text{sl}} = 2$.

3.3.4 Dimensional Characteristics

The three different dimensionalities that characterise an OPF feature are related through Equation (3.11). All features are constrained by the geometric dimensionality of the phase diagram.

$$\mathbb{D}^{\mathcal{B}}_{\mathrm{g}} <= \mathbb{D}^{\mathcal{B}}_{\mathrm{f}} <= \mathbb{D}^{\sigma}_{\mathrm{g}} \tag{3.11}$$

To concisely indicate the dimensional characteristics for an OPF feature, the notation $\mathbb{D}^{\mathcal{B}} = (\mathbb{D}^{\mathcal{B}}_{\mathrm{g}} : \mathbb{D}^{\mathcal{B}}_{\mathrm{f}} : \mathbb{D}^{\sigma}_{\mathrm{g}})$ is used. For the two-component system in Figure 3.1a, $\mathcal{B}^{\mathsf{sl}}_{\mathsf{sl}}$ has $\mathbb{D}^{\mathcal{B}} = (2{:}2{:}2)$, $\mathcal{B}^{\mathsf{cr\text{-}sl}}_{\mathsf{sl}}$ has $\mathbb{D}^{\mathcal{B}} = (1{:}2{:}2)$, and $\mathcal{B}^{\mathsf{cr\text{-}sl}}_{\mathsf{cr}}$ has $\mathbb{D}^{\mathcal{B}} = (1{:}1{:}2)$.

3.4 Simplices

A simplex is the generalisation of a triangle in kD space, which has $\mathrm{k}+1$ vertices and is denoted by ${}_{\mathrm{k}}\Delta$. For instance, a triangle is a 2D object, referred to as a 2-simplex and denoted by ${}_{2}\Delta$. Similarly, a 3-simplex is a tetrahedron, a 1-simplex is a line, and a 0-simplex is a point. Simplices where $\mathrm{k} > 3$ are beyond our ability to visualise, and these are simply referred to as k-simplices.

3.4.1 Content of a Simplex

The content $\nu({}_{\mathrm{k}}\Delta)$ of a k-simplex is the measure of the space contained within its vertices. Examples include the length of a line, area of a triangle, and volume of a tetrahedron. For any dimensionality k, ν is calculated with the Cayley-Menger determinant (Sommerville 1929), which only requires the simplex's vertex coordinates. The distances between each of the $\mathrm{k}+1$ vertices x_{i} of the k-simplex are used to construct the $(\mathrm{k}+1) \times (\mathrm{k}+1)$ distance matrix $\mathbf{D} = (\mathrm{d}_{\mathrm{ij}})$ where the indices d_{ij} are calculated with Equation (3.12).

$$\mathrm{d}_{\mathrm{ij}} = \|x_{\mathrm{i}} - x_{\mathrm{j}}\|^2 \quad \text{with} \quad \mathrm{i},\mathrm{j} \in \mathbb{N}[1, \mathrm{k}+1] \tag{3.12}$$

Matrix $\bar{\mathbf{D}}$ is a $(\mathrm{k}+2) \times (\mathrm{k}+2)$ matrix obtained from $\mathbf{D}$ by bordering it with a bottom row $[1, \ldots 1, 0]$ and a right column $[1, \ldots 1, 0]^{\mathsf{T}}$. The determinant of matrix $\bar{\mathbf{D}}$ is called the Cayley-Menger determinant $|\bar{\mathbf{D}}|$, shown in Equation (3.13).

$$|\bar{\mathbf{D}}| = \begin{vmatrix} 0 & \mathrm{d}_{12} & \mathrm{d}_{13} & \mathrm{d}_{14} & \cdots & \mathrm{d}_{1\mathrm{j}} & 1 \\ \mathrm{d}_{21} & 0 & \mathrm{d}_{23} & \mathrm{d}_{24} & \cdots & \mathrm{d}_{2\mathrm{j}} & 1 \\ \mathrm{d}_{31} & \mathrm{d}_{32} & 0 & \mathrm{d}_{34} & \cdots & \mathrm{d}_{3\mathrm{j}} & 1 \\ \mathrm{d}_{41} & \mathrm{d}_{42} & \mathrm{d}_{43} & 0 & \cdots & \mathrm{d}_{4\mathrm{j}} & 1 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathrm{d}_{\mathrm{i}1} & \mathrm{d}_{\mathrm{i}2} & \mathrm{d}_{\mathrm{i}3} & \mathrm{d}_{\mathrm{i}4} & \cdots & 0 & 1 \\ 1 & 1 & 1 & 1 & \cdots & 1 & 0 \end{vmatrix} \tag{3.13}$$

The Cayley-Menger determinant, together with a pre-factor determined with Heron's formula (Alexandria 100AD), is used to calculate the content $\nu({}_{\mathrm{k}}\Delta)$ of a k-simplex with Equation (3.14).

$$\nu({}_{\mathrm{k}}\Delta) = \sqrt{\frac{(-1)^{\mathrm{k}-1}}{2^{\mathrm{k}}(\mathrm{k}!)^2}|\bar{\mathbf{D}}|} \tag{3.14}$$

3.4.2 Barycentric Coordinates

The barycentric coordinate system is used to express the location of a point contained within a simplex with reference to its vertices, as shown in Figure 3.2. From one vertex x_1 of the $\mathrm{k}+1$ number of vertices of a k-simplex, k number of vectors $\mathbf{V}_\mathrm{i}$ are created to each of the k remaining vertices x_i. A barycentric coordinate weight β_i is applied to each vector and the sum of weighted vectors result in the vector from the origin vertex x_1 to the location of a point p. The coordinates of a point inside a k-simplex can therefore be described by the $\mathrm{k}+1$ vertices of a k-simplex and a set of k barycentric coordinate weights, as shown in Equation (3.15).

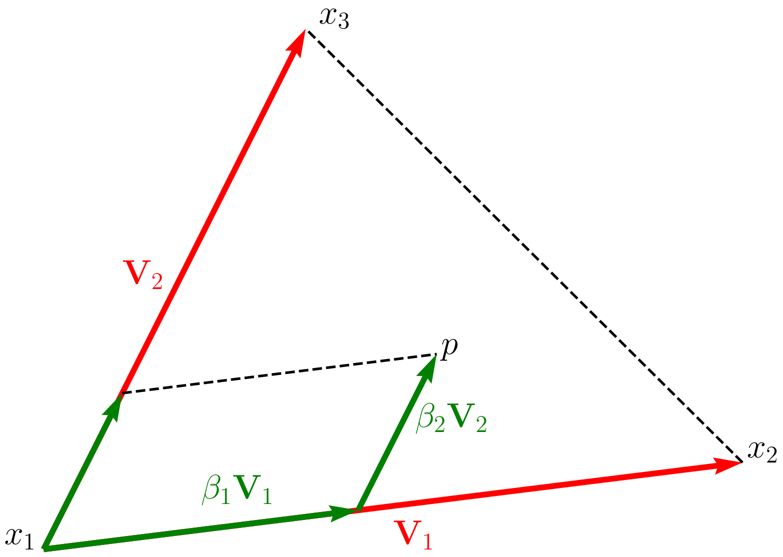


Figure 3.2: Barycentric coordinate weights describing the position of a point within a ${}_2\Delta$.

$$\begin{aligned} p - x_1 &= \sum_{\mathrm{i}=1}^{\mathrm{k}} \beta_\mathrm{i}\mathbf{V}_\mathrm{i} \\ p &= x_1 + \sum_{\mathrm{i}=1}^{\mathrm{k}} \beta_\mathrm{i}(x_{\mathrm{i}+1} - x_1) \end{aligned} \tag{3.15}$$

The conditions of Equation (3.16) have to be met for a point to be contained within the k-simplex. If any of the barycentric coordinate weights are smaller than zero or larger than one, the point is outside the simplex. If the sum of weights are larger than one, the resulting vector also ends at a point outside the simplex.

$$\beta_\mathrm{i} \in \mathbb{R}[0,1] \quad \text{and} \quad \sum_{\mathrm{i}=1}^{\mathrm{k}} \beta_\mathrm{i} <= 1 \tag{3.16}$$

3.4.3 Discretisation by Simplices

Triangulation can be used to discretise OPF features. Lines are used to discretise 1D features, triangles for 2D features, and k-simplices for features with $\mathbb{D}_\mathrm{g}^{\mathcal{B}} = \mathrm{k}$. This allows an OPF feature to be discretised into segments with the same geometric dimensionality, which have the fewest number of vertices per discretisation segment. This is referred to as discretisation by simplices.

3.5 Tie Simplices

For a system at equilibrium, all potentials (temperature, pressure, chemical potentials, etc.) are equal in all phases present. On a phase diagram at constant pressure and temperature, phase compositions on OPF boundaries associated with multi-phase region i are connected by a tie simplex $\Delta^{\mathcal{R}_\mathrm{i}}$, which is an iso-chemical-potential simplex. The dimensionality k of a tie simplex is determined by the number of phases present according to Equation (3.17), and its number of vertices is equal to the number of phases present with each vertex representing the composition of a phase.

$$\mathrm{k} = \hat{\varphi}^{\mathcal{R}_\mathrm{i}} - 1 \tag{3.17}$$

Examples include tie-lines in two-phase regions, tie-triangles in three-phase regions, tie-tetrahedra in four-phase regions, etc. In single-phase regions a "tie simplex" is a point (0-simplex), since there are not more than one phase to "tie together".

3.6 The Lever Rule

The phase fraction $x^{\sigma}_{\varphi_\mathrm{i}}$ of any phase i present at equilibrium can be determined with tie simplices and the lever rule, which is based on the inverse distance principle. The closer the specified system composition $\mathbf{x}^{\sigma}_{\epsilon}$ is to the vertex $\mathbf{x}^{\varphi_\mathrm{i}}_{\epsilon}$ of phase i in the tie simplex, the higher the fraction of phase i and the lower the fractions of the other phases.

When the system composition $\mathbf{x}^{\sigma}_{\epsilon}$ is inside a tie simplex $\Delta^{\mathcal{R}}$, it divides the tie simplex into $\hat{\varphi}$ phase-simplices $\Delta^{\mathcal{R}}_{\varphi_\mathrm{i}}$, one for each phase i, as shown in Figure 3.3. The vertices of $\Delta^{\mathcal{R}}_{\varphi_\mathrm{i}}$ include $\mathbf{x}^{\sigma}_{\epsilon}$ as well as all the vertices of $\Delta^{\mathcal{R}}$ except for $\mathbf{x}^{\varphi_\mathrm{i}}_{\epsilon}$, the composition of phase i itself. The phase fraction for a phase i in any $\hat{\varphi}$-phase region of any $\hat{\epsilon}$-component system is calculated with the generic lever rule, as shown in Equation (3.18). In single-phase regions, $x^{\sigma}_{\varphi_\mathrm{i}} = 1$, since there is only one phase (and the content of a point is $\nu({}_0\Delta) = 1$).

$$x^{\sigma}_{\varphi_\mathrm{i}} = \frac{\nu(\Delta^{\mathcal{R}}_{\varphi_\mathrm{i}})}{\sum\limits_{\mathrm{j}=1}^{\hat{\varphi}} \nu(\Delta^{\mathcal{R}}_{\varphi_\mathrm{j}})} = \frac{\nu(\Delta^{\mathcal{R}}_{\varphi_\mathrm{i}})}{\nu(\Delta^{\mathcal{R}})} \tag{3.18}$$

3.7 Calculating Thermochemical Properties

Once an equilibrium calculation is performed at temperature T and system composition $\mathbf{x}^{\sigma}_{\epsilon}$, the compositions $\mathbf{x}^{\varphi}_{\epsilon}$ of all phases and thermochemical properties (e.g. H^{φ}, S^{φ}, G^{φ}) for each phase are known and can be stored as a tie simplex. Now, for the same temperature T and any system composition $\mathbf{x}^{\sigma}_{\epsilon}$ inside the tie simplex, phase fractions $\mathbf{x}^{\sigma}_{\varphi}$ can be calculated with the lever rule whereafter system thermochemical properties (e.g. H^{σ}, S^{σ}, G^{σ}) can be calculated exactly with Equation (3.19) without performing another equilibrium calculation.

$$\tau^{\sigma} = \sum_{\mathrm{i}=1}^{\hat{\varphi}} x^{\sigma}_{\varphi_\mathrm{i}} \tau^{\varphi_\mathrm{i}} \quad \text{where } \tau \in \{H, S, G, V\} \tag{3.19}$$

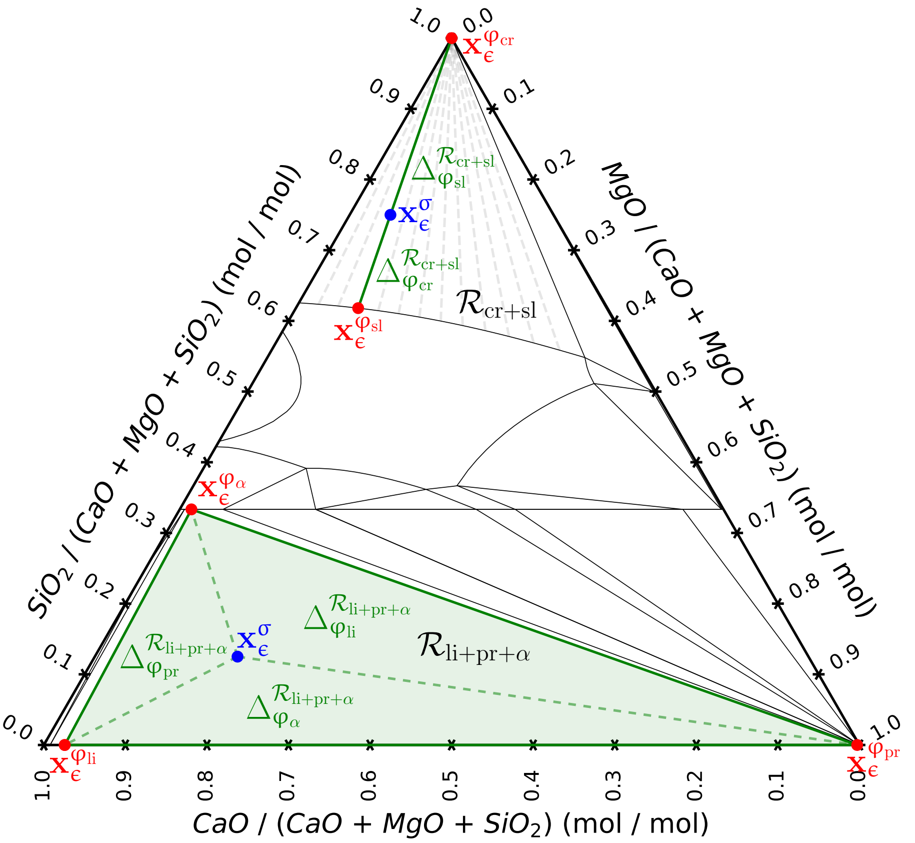

$\mathbf{x}_\epsilon^{\varphi_{\mathrm{cr}}}$
$\triangle_{\varphi_{\mathrm{sl}}}^{\mathcal{R}_{\mathrm{cr+sl}}}$
$\mathbf{x}_\epsilon^{\sigma}$
$\triangle_{\varphi_{\mathrm{cr}}}^{\mathcal{R}_{\mathrm{cr+sl}}}$
$\mathcal{R}_{\mathrm{cr+sl}}$
$\mathbf{x}_\epsilon^{\varphi_{\mathrm{sl}}}$
$\mathbf{x}_\epsilon^{\varphi_{\alpha}}$
$\triangle_{\varphi_{\mathrm{li}}}^{\mathcal{R}_{\mathrm{li+pr+}\alpha}}$
$\mathcal{R}_{\mathrm{li+pr+}\alpha}$
$\triangle_{\varphi_{\mathrm{pr}}}^{\mathcal{R}_{\mathrm{li+pr+}\alpha}}$
$\mathbf{x}_\epsilon^{\sigma}$
$\triangle_{\varphi_{\alpha}}^{\mathcal{R}_{\mathrm{li+pr+}\alpha}}$
$\mathbf{x}_\epsilon^{\varphi_{\mathrm{li}}}$
$\mathbf{x}_\epsilon^{\varphi_{\mathrm{pr}}}$
SiO_2 / (CaO + MgO + SiO_2) (mol / mol)
MgO / (CaO + MgO + SiO_2) (mol / mol)
CaO / (CaO + MgO + SiO_2) (mol / mol)
1.0 0.9 0.8 0.7 0.6 0.5 0.4 0.3 0.2 0.1 0.0
0.0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1.0
1.0 0.9 0.8 0.7 0.6 0.5 0.4 0.3 0.2 0.1 0.0

Figure 3.3: Phase-simplices in a two- and three-phase region in the CaO−MgO−SiO_2 three-component system. (sl: slag, cr: cristobalite, li: lime, pr: periclase, α: α-Ca_2SiO_4) (1750 K isotherm)

3.8 Demonstration of Foundational Concepts

In this section, the foundational concepts presented in the previous sections are demonstrated by applying them to the phase diagrams of two- and three-component systems.

3.8.1 Two-component System

The isobaric CaO−SiO_2 system under ambient atmosphere shown in Figure 3.1a is presented as a two-component system example. Since pressure is constant, only one non-compositional potential is of concern, namely temperature; this means $\hat{\psi} = 1$. The system incorporates the chemical elements Ca, O, and Si; therefore $\hat{\varepsilon} = 3$. Due to the high oxygen potential of the ambient atmosphere, the oxygen content of the system is constrained to $n_\mathrm{O} = n_\mathrm{Ca} + 2n_\mathrm{Si}$. Therefore, a single system compositional constraint exists and $\hat{\zeta}^\sigma = 1$.

Calculating the phase diagram's dimensionality with Equation (3.6) yields $\mathbb{D}_\mathrm{g}^\sigma = 2$. The thermochemical system is therefore described with a 2D phase diagram. The axes of the phase diagram are the mole fraction of one system component x_CaO and temperature.

OPF region $\mathcal{B}_\mathrm{sl}^\mathrm{sl}$ has $\mathbb{D}^\mathcal{B} = (2{:}2{:}2)$. It is a 2D surface ($\mathbb{D}_\mathrm{g}^\mathcal{B} = 2$) with coordinates varying in both x_CaO and T ($\mathbb{D}_\mathrm{f}^\mathcal{B} = 2$), existing in a 2D phase diagram ($\mathbb{D}_\mathrm{g}^\sigma = 2$).

The two-phase $\mathcal{R}_\mathrm{cr+sl}$ region's OPF boundary $\mathcal{B}_\mathrm{cr}^\mathrm{cr+sl}$ has $\mathbb{D}^\mathcal{B} = (1{:}1{:}2)$. $\mathbb{D}_\mathrm{g}^\mathcal{B} = \mathbb{D}_\mathrm{f}^\mathcal{B} = 1$ because it is a pure substance pinned to a single composition on the x_CaO axis. The OPF boundary $\mathcal{B}_\mathrm{sl}^\mathrm{cr+sl}$ of this phase region has $\mathbb{D}^\mathcal{B} = (1{:}2{:}2)$. Although it is a 1D curve ($\mathbb{D}_\mathrm{g}^\mathcal{B} = 1$), its coordinates vary in both T and x_CaO and $\mathrm{f}^\mathrm{sl} = 2$, which yields $\mathbb{D}_\mathrm{f}^\mathcal{B} = 2$.

3.8.2 Three-component System

The isobaric CaO−MgO−SiO$_2$ system under ambient atmosphere shown in Figure 3.1b is presented as a three-component system example. With pressure constant, temperature is the only non-compositional potential, and $\hat{\psi} = 1$. The system involves the chemical elements Ca, Mg, O, and Si; therefore $\hat{\varepsilon} = 4$. A high oxygen potential again constrains the system oxygen content to $n_{\mathsf{O}} = n_{\mathsf{Ca}} + n_{\mathsf{Mg}} + 2n_{\mathsf{Si}}$. This single system compositional constraint yields $\hat{\zeta}^{\sigma} = 1$.

The phase diagram has $\mathbb{D}^{\sigma}_{\mathrm{g}} = 3$, which correctly indicates it to be 3-dimensional. The axes of the phase diagram are the mole fractions of two system components x_{CaO} and x_{MgO}, and temperature.

The OPF region $\mathcal{B}^{\mathsf{sl}}_{\mathsf{sl}}$ has $\mathbb{D}^{\mathcal{B}} = (3{:}3{:}3)$. It is a 3D volume ($\mathbb{D}^{\mathcal{B}}_{\mathrm{g}} = 3$) with coordinates varying with x_{CaO}, x_{MgO} and T ($\mathbb{D}^{\mathcal{B}}_{\mathrm{f}} = 3$), existing in a 3D phase diagram ($\mathbb{D}^{\sigma}_{\mathrm{g}} = 3$).

The two-phase $\mathcal{R}_{\mathsf{mw+sl}}$ region's OPF boundary $\mathcal{B}^{\mathsf{mw+sl}}_{\mathsf{mw}}$ has $\mathbb{D}^{\mathcal{B}} = (1{:}1{:}3)$. Its geometric and functional dimensionalities are both equal to one, because it is a pure substance. The OPF boundary $\mathcal{B}^{\mathsf{mw+sl}}_{\mathsf{sl}}$ of this region has $\mathbb{D}^{\mathcal{B}} = (2{:}3{:}3)$. Although it is a 2D curved surface ($\mathbb{D}^{\mathcal{B}}_{\mathrm{g}} = 2$), its coordinates vary with temperature and both system composition coordinates, which yields $\mathbb{D}^{\mathcal{B}}_{\mathrm{f}} = 3$.

The two-phase $\mathcal{R}_{\mathsf{mt+sl}}$ region's OPF boundary $\mathcal{B}^{\mathsf{mt+sl}}_{\mathsf{mt}}$ has $\mathbb{D}^{\mathcal{B}} = (2{:}2{:}3)$. Its geometric dimensionality is equal to two, because $\mathrm{f} = 2$ for the phase region, and $\mathrm{f}^{\mathsf{mt}} = 2$ for this phase. $\mathbb{D}^{\mathcal{B}}_{\mathrm{f}} = 2$ because the phase only exists on the line connecting the Ca$_2$SiO$_4$ and Mg$_2$SiO$_4$ end members.

Because $\mathrm{f} = 1$ in the three-phase $\mathcal{R}_{\mathsf{mw+mt+sl}}$ region, $\mathbb{D}^{\mathcal{B}}_{\mathrm{g}} = 1$ for all three phases. For the $\mathcal{B}^{\mathsf{mw+mt+sl}}_{\mathsf{mw}}$ OPF boundary $\mathbb{D}^{\mathcal{B}}_{\mathrm{f}} = 1$, since it is a pure substance. For the $\mathcal{B}^{\mathsf{mw+mt+sl}}_{\mathsf{mt}}$ OPF boundary $\mathbb{D}^{\mathcal{B}}_{\mathrm{f}} = 2$ since the phase exists on the line connecting the Ca$_2$SiO$_4$ and Mg$_2$SiO$_4$ end members and therefore $\mathrm{f}^{\mathsf{mt}} = 2$. Finally, $\mathbb{D}^{\mathcal{B}}_{\mathrm{f}} = 3$ for the $\mathcal{B}^{\mathsf{mw+mt+sl}}_{\mathsf{sl}}$ OPF boundary because this phase has no additional phase composition constraints beyond those of the system.

3.8.3 General System

The method employed here to two- and three-component systems applies to $\hat{\varepsilon}$-component systems in general. This, along with the generality of simplex geometry, the Cayley-Menger determinant, the generic lever rule, and the Gibbs phase rule provides a general basis for discretisation and interpolation of OPF regions and OPF boundaries, and for the calculation of phase fractions, phase compositions, and thermochemical properties.

Chapter 4

Background on Equilibrium Calculations

Spontaneous physical change is governed by the second law of thermodynamics, which states that such change will only occur when it results in an increase in an isolated system's entropy. Gibbs energy is a useful quantity that allows change in entropy to be described in terms of properties of the system alone; the surroundings can conveniently be disregarded. Thermochemical equilibrium of a system is reached when Gibbs energy reaches a minimum; also the state of maximum entropy.

This physical optimisation problem can be cast into mathematical and numerical formulations, and ultimately into computer software that can be used to simulate a system's drive towards equilibrium. This is however an iterative process and, as displayed in Figure 1.1 become more computationally expensive as more system components are considered and the optimisation problem become more intricate.

Many different methods have been developed to calculate thermochemical equilibrium by determining the minimum Gibbs energy for a system state. Simultaneous and decoupled equation-solving methods have been developed and used. Teh and Rangaiah (2002) discussed and evaluated the various equation-solving methods in more detail. Piro et al. (2013) employed another method, Gibbs energy partitioning, in the Thermochimica library. Convex hulls have been used as a geometrical approach to determine the state with minimum Gibbs energy (Voskov, Dzuban, and Maksimov 2015). The most popular method, however, has been the Gibbs energy minimisation routine, which has been implemented in a variety of open-source (Sundman et al. 2015; Sundman, Lu, and Ohtani 2015; Otis and Liu 2017; Nichita, Gomez, and Luna 2002) and commercial (Bale et al. 2016; Shobu 2009; Davies et al. 2002; Andersson et al. 2002; Chen et al. 2002) software packages. The Open-Calphad software also provides the capability to perform multiple equilibrium calculations in parallel (Sundman, Lu, and Ohtani 2015).

To better understand why equilibrium calculations employing the Gibbs energy minimisation routine become more computationally expensive as more system components are considered, a brief overview of the mathematical formulation and numerical aspects of the routine is presented. Detailed derivations are not presented here, but references are provided throughout this chapter for interested readers.

4.1 Mathematical Formulation

The total Gibbs energy of the system G^{σ} is often expressed in terms of the amount b^{ϵ} and chemical potential μ^{ϵ} of all $\hat{\epsilon}$ number of independent system components of the system, shown in Equation (4.1) (Eriksson and Hack 1990).

$$G^{\sigma} = \sum_{i=1}^{\hat{\epsilon}} b^{\epsilon_i} \mu^{\epsilon_i} \tag{4.1}$$

With Equation (4.1) an objective function can be set up that describes the Gibbs energy for a given temperature, pressure, and composition of the system. To determine the equilibrium condition, this objective function needs to be minimised while being constrained by a mass balance for each of the $\hat{\epsilon}$ number of independent system components (Eriksson and Hack 1990). For each possible phase φ that exists in the system, a stoichiometric matrix $\mathbf{s}^{\varphi}$ is used where the coefficients describe the ratio between all the phase constituents θ^{φ} and the system components ϵ. This matrix is used, together with the amount $n^{\theta^{\varphi}}$ of each phase constituent θ^{φ} of phase φ, to determine the amount for each independent system component, as shown in Equation (4.2).

$$b^{\epsilon_i} = \sum_{j=1}^{\hat{\varphi}} \sum_{k=1}^{\hat{\theta}^{\varphi_j}} \mathbf{s}_{ik}^{\varphi_j} n^{\theta_k^{\varphi_j}} \quad i \in [1, \hat{\epsilon}] \tag{4.2}$$

Each of the $\hat{\varphi}$ number of stoichiometric matrices has a size of $\hat{\epsilon} \times \hat{\theta}^{\varphi}$. The implications when the number of system components to be considered are increased, are as follows:

1. many new phases can exist with the addition of one single system component; therefore the number of stoichiometric matrices that has to be summed over increases non-linearly,
2. the number of rows of a stoichiometric matrix increases linearly with the number of system components,
3. some phases contain components in addition to system components, such as electrons,
4. therefore, the number of columns of the stoichiometric matrix increases linearly with the exception of some phases with additional phase components.

The non-linear increase in the number of phases when more system components are considered differs from system to system; it depends on the number of new phases that can be created between the existing and newly added system components.

Described here is only one formulation used to minimize Gibbs energy. The ChemAppPy (Ex Mente Technologies 2019) software employed in this work make use of this type of formulation and is therefore considered in this investigation. There are several other formulations, as described in the review by Smith (1980).

4.2 Numerical Aspects

The Gibbs energy of the system, expressed by Equation (4.1), needs to be minimised while being constrained to the mass balance of each system component expressed by Equation (4.2). Lagrange's method of undetermined multipliers is suitable to solve the constrained minimisation problem (Eriksson 1971), resulting in the Lagrange objective function (Koukkari and Pajarre 2011) to be minimised, as shown in Equation (4.3).

$$L = G^{\sigma} - \sum_{i=1}^{\hat{\epsilon}} \pi_i \left[\sum_{j=1}^{\hat{\varphi}} \sum_{k=1}^{\hat{\theta}^{\varphi_j}} \mathbf{s}_{ik}^{\varphi_j} n^{\theta_k^{\varphi_j}} - b^{\epsilon_i} \right] \tag{4.3}$$

The Lagrangian multipliers π_i for each system component increases linearly as the number of system components increase. Partial derivatives of the Lagrange objective function is calculated for each system component, therefore, the number of partial derivatives also increase linearly as the number of system components increase. The partial derivatives has to be calculated for every iteration of the Gibbs energy minimisation routine, and can become computationally expensive when many system components are considered.

Part II

Literature Review

Chapter 5

Applications of Equilibrium Calculations

In the development of models, it is often adequate to make simplifications regarding thermochemical behaviour and physical properties. In other cases it is essential to incorporate these details, since they are core to the investigation. Here we focus on the latter where the application of equilibrium calculations to multiphysics and process models have been invaluable.

5.1 Multiphysics Models

Incorporating equilibrium calculations into models have been used extensively in **casting design** by studying the mechanisms contributing to macrosegregation (ten Cate et al. 2008). The solidification, shrinkage, air-gap development, stresses, and temperature distribution of a casting have also been modelled by incorporating equilibrium calculations into models (Agelet de Saracibar, Chiumenti, and Cervera 2006).

Refractory materials are often subjected to high temperature gradients, corrosive environments, and mechanical loads (Blond et al. 2014) in an attempt to protect equipment or operators nearby. Models have been used to investigate different materials and how they degrade due to oxidation, corrosion, vaporization, sublimation, ablation, and dissolution (Blond et al. 2014; Tabiei and Sockalingam 2012). There are clear signs in the work done by Blond et al. (2014) that the inclusion of equilibrium calculations improve insight into stresses and strains in furnace refractories. Equilibrium calculations have been included into models with adaptive meshes by Tabiei and Sockalingam (2012) to investigate the lifespan of dynamic refractory linings of hypersonic vehicles during their re-entry through Earth's atmosphere.

Biomass-based energy technologies have been improved significantly when equilibrium calculations were incorporated into multiphysics models (Pannala, Simunovic, and Frantziskonis 2010). Multi-phase reactors are widely used, but their designs are primarily based on experimental data due to process complexity. The understanding of these processes is limited because dense and erosive flows encumber measurement tools. Even when measuring tools are used, system dynamics are altered, leading to measurement inaccuracy (Pannala, Simunovic, and Frantziskonis 2010). Models that incorporate equilibrium calculations can bridge the gap between experimental data and the actual behaviour of these reactors (Pannala, Simunovic, and Frantziskonis 2010), which can be used to improve reactor design.

Equilibrium calculations allowed phase compositions, oxygen chemical potential and other thermochemical properties to be included into models that were essential for investigating and estimating performance, safety, and behaviour of **nuclear fuel** that could not be determined through experimentation during operation (Corcoran, Kaye, and Piro 2016; Samuelsson et al. 2020).

Complex chemical reactions, heat and mass transfer, phase changes, and multi-phase flows make it extremely challenging to model pyrometallurgical processes. Molten slags are usually produced, in some cases it is the desired product (Zietsman 2004), and can be very corrosive, which can cause furnace refractory lining damage. To combat this, furnace sidewalls are cooled with the intent to solidify slag onto the refractory lining, forming a **slag freeze lining** (Zietsman and Pistorius 2006). Including equilibrium calculations into models make it possible to describe chemical reactions, heat and mass transfer, phase changes, and multi-phase flow, which can then be used to estimate freeze lining condition.

5.2 Process Models

Process models of **ilmenite-smelting DC furnaces** (Zietsman 2004; Zietsman and Pistorius 2006), have been used to investigate the interaction between slag bath and freeze lining. This was done by determining the influences of thermal and chemical changes and changes in operational parameters on these interactions. Insights from these investigations were used to improve and refine the operational strategies of the process.

There is continual interaction between the flow of molten slag and freeze lining as remelting and solidification occurs. It is important to maintain a freeze lining at all times, but it is near impossible to measure the geometry within molten liquid to ensure its presence. **Monitoring freeze lining thickness** can be done by following the trends of thermocouples in the sidewalls (Zietsman and Pistorius 2005). The thermocouple data is used to calculate heat flux through the freeze lining and estimate its thickness.

The **electric arc furnace** (EAF) is the main process used to recycle iron and steel and the second-most important process in terms of global steel production (Hay, Echterhof, and Visuri 2019). Process models have been used to improve the understanding and control of the EAF process as well as optimizing its energy and resource efficiency (Hay, Reimann, and Echterhof 2019). These EAF process models include equilibrium calculations that provide composition and temperature estimations throughout the process that cannot be measured directly due to the harsh conditions within the furnace.

EAF process models can be used to **automatically create operational charts in real time** and can be adjusted based on furnace operating conditions (Hay, Echterhof, and Visuri 2019). Scenarios such as new control strategies, different injection or charging materials, and installation of new equipment can be evaluated virtually and adjusted with a process model before any changes are implemented. Hay, Echterhof, and Visuri (2019) and Hay, Reimann, and Echterhof (2019) summarised the different EAF process models that have been developed and their continual improvements.

5.3 Conclusion

Being able to include equilibrium calculations into a model that accurately represented the thermochemical systems allowed for virtual prototyping and testing of new processes, control strategies and equipment before any costly physical tests had to be performed, new plants had to be built or changes made to existing plants, or new equipment purchased. Using models where equilibrium calculations have been included to monitor and control processes where conventional monitoring methods are infeasible have been found to be invaluable. Discussed here were only a few examples of where the implementation of equilibrium calculations provide insight into systems and processes that would otherwise be difficult or impossible to obtain.

Chapter 6

Accelerating Equilibrium Calculations

Incorporating equilibrium calculations into models can provide more accurate results, but at a cost of longer computational time. To reduce equilibrium calculation computation times and make model computational times more feasible, several acceleration methods have been developed.

6.1 Uniform Discretisation of System Space

The feasibility of uniformly discretising the temperature and compositional space of a thermochemical system was investigated by ten Cate et al. (2008). At each discretisation node, an equilibrium calculation had to be performed and thermochemical properties stored for later recall. The composition in a system with $\hat{\epsilon}$ number of components can be described by the concentration of $\hat{\epsilon} - 1$ components. An isobaric four-component system was considered with varying temperature. Therefore, a thermodynamic property of the four-component system was dependent on four independent variables; temperature and the concentration of three components. The range of each independent variable was divided into 600 nodes to form a uniform grid. To store two thermodynamic properties as functions of the four independent variables with single precision (32 bit per value) on each of the grid nodes, $2 \times 4 \times 600^4 \times$ 32 bit $=$ 4 TB of storage space would have been needed (ten Cate et al. 2008). The size of a uniform grid pre-calculated database can become too large for the memory of computers to use (ten Cate et al. 2008) and searching through that amount of data can be very time-consuming. The space complexity is of order $O(n^d)$ where d is the dimension of the grid (ten Cate et al. 2008).

A similar tabulation method was developed by Saad, Gandin, and Bellet (2015). A known alloy was chosen and the composition variation intervals were identified. These intervals were not known before the model was solved and had to cover the alloy composition extremes – informed estimations had to be made. A temperature interval also had to be identified, but in general was chosen from the initial melting temperature to room temperature. Systematic checks were done within these intervals at predefined step sizes and equilibrium calculations were performed at each temperature and compositional combination. The number of phases, together with the compositions and fractions of each, were determined from each equilibrium calculation. This allowed the temperature and compositional interval of the phase region boundary associated with each phase to be determined. Thereafter, the composition and temperature intervals of each phase was systematically checked to determine thermochemical properties of the phase at different temperatures and compositions. Interpolation could then be performed between phase composition and temperature nodes to obtain phase properties for an equilibrium calculation.

6.2 Liquidus Surface Mapping

A method was used to store the liquidus and solidus surfaces of solid and eutectic phases in a three-component system (Doré, Combeau, and Rappaz 2000; Wu et al. 2013). A number of equilibrium calculations were performed prior to the simulation of the model for combinations of various temperatures, pressures, and system component concentration to be used to describe the surfaces.

The concentration of one system component was kept constant and the liquidus surface mapping along the isopleth was completed by ranging the other system component between its minimum and maximum concentration by a fixed increment. Two partial slopes of the liquidus surface; $m_{l,A} = \partial T / \partial c_{l,A}$ relative to system component A concentration and $m_{l,B} = \partial T / \partial c_{l,B}$ relative to system component B concentration, and two partition coefficients; k_A and k_B were stored for a fixed concentration $c_{l,C}$ of system component C. Partition coefficients are used to describe tie lines between the solidus and liquidus surfaces (Wu et al. 2013). The first system component's concentration was then adjusted by a fixed increment and the mapping repeated.

The primary phase's liquidus temperature, T_f, together with the liquidus surface's slopes, $m_{l,A}$ and $m_{l,B}$, are used to describe the liquidus surface for a given composition, as described by Equation (6.1) and illustrated with Figure 6.1, (Wu et al. 2013).

$$T = T_f + m_{l,A} \times c_{l,A} + m_{l,B} \times c_{l,B} \tag{6.1}$$

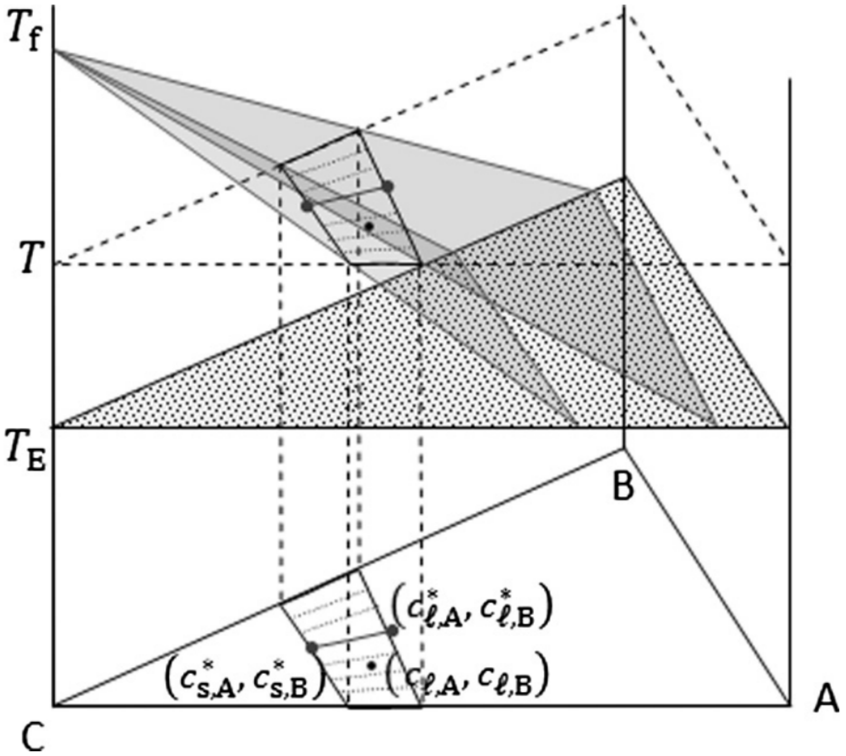


Figure 6.1: Liquidus surface of two-phase region described by stored partial derivative values of temperature with respect to system component concentrations. Tie lines described by stored partition coefficients used to determine the solidus surface. As found in (Wu et al. 2013).

Given the phase composition on the liquidus surface, the corresponding phase's composition on the solidus surface – that is found on the same tie line – is described by the partition coefficients, as described by Equation (6.2) and illustrated in Figure 6.1, (Wu et al. 2013).

$$\begin{aligned} c^*_{s,A} &= k_A \times c^*_{l,A} \\ c^*_{s,B} &= k_B \times c^*_{l,B} \end{aligned} \tag{6.2}$$

With this method a global mapping file was compiled. Bilinear interpolation was used to calculate the liquidus temperature, the slopes of the liquidus surface, and the partition coefficients for given system component concentrations.

An acceleration factor of 4 was observed in three-component systems with the mapping file requiring a few MB of storage space (Doré, Combeau, and Rappaz 2000). When a mapping file was created in a four-component system, the storage space was found to be in the order of 100 MB. As seen in the feasibility study on uniform discretisation of temperature-compositional space done by ten Cate et al. (2008) and the liquidus and solidus surface mapping by (Doré, Combeau, and Rappaz 2000), as more complex systems with more components are considered, more storage space is required and can become infeasible.

6.3 In-situ Database Population

Instead of performing a large number of equilibrium calculations and storing the results prior to solving a model, an initially empty database can be populated as the model is solving – in-situ – as presented by Pope (1997). As the model requires thermochemical properties at a specified system state, an equilibrium calculation is performed, and results stored to the database. At a later stage, when the model requires thermochemical properties near stored results, interpolation, rather than a direct equilibrium calculation, is performed and the results returned. The advantage is that the database is populated only in regions of the system space that is accessed by the model, known as the accessed regions, and no unnecessary equilibrium calculations have to be performed beforehand. This reduces the number of computationally expensive equilibrium calculations that need to be performed and reduces the storage space requirement of the database.

It was found that the use of in-situ adaptive database population completed 10^9 queries in 54 hr, compared to more than 6 yr (theoretically) when direct integration was considered – an acceleration factor of about 10^3. It roughly took 1.5 hr of the computational time to complete all the additions to the database and in the remaining computational time only database recalls were made and interpolation performed (Pope 1997).

Analysis of the in-situ adaptive database population method's performance as well as an improvement to the recall algorithm was done by Chen (2004). An improvement on the algorithm was made by Lu and Pope (2009) in terms of the database-recall strategies and the addition of error checking and correction. In comparison to the previous implementation (Pope 1997), the computational time has been halved and the storage space requirements have been reduced by a factor of five.

In-situ database population was also used in the work of Larsson and Höglund (2015) where the system space was divided into a structured reference frame but no equilibrium calculations were performed beforehand – only the nodes' temperature and compositions were determined. When thermochemical properties were required at a specified system state, the nodes of the structured reference frame nearest to the specified system state were identified, equilibrium calculations performed at their corresponding temperature and compositions, and the results stored to the database. From the results obtained, interpolation is performed to determine thermochemical properties at the specified system state. When another specified system state is found between reference frame nodes where equilibrium calculations have been performed, they are simply used again to interpolate towards the specified system state. In the case where some identified reference frame nodes do not have any results available, equilibrium calculations are performed, results stored to the database, and interpolation performed. A database storage space size limit is also specified and when new results have to be stored, the results that have not been used for the longest period is overwritten.

The in-situ database populations scheme of Larsson and Höglund (2015) was implemented by Pillai et al. (2016) and its performance evaluated. In one test, an acceleration factor of 2 was observed. In another test, the compositional space accessed by the model was larger and therefore more equilibrium calculations were required, and an acceleration factor of 0.6 was observed.

As noticed in the work by Pope (1997), initially only computationally expensive equilibrium calculations are performed and little to no acceleration is observed, or even a deceleration due to the extra overhead computations required to store the data. As the database is populated more densely, the frequency of recalls and interpolations increase up to the point where only recalls and interpolation is performed and large accelerations are observed.

6.4 Re-initialisation

Multiphysics models can have thousands or even millions of mesh cells and an equilibrium calculation needs to be performed in each mesh cell for every iteration. A re-initialisation acceleration method was developed by Poschmann, Piro, and Simunovic (2020) to include equilibrium calculations from Thermochimica (Piro et al. 2013) into multiphysics models more efficiently. An equilibrium calculation is performed in each mesh cell of the multiphysics model for the first iteration and the calculated equilibrium state stored with the associated cell. During the next iteration, the stored equilibrium state of a mesh cell is used as an accurate initial estimate for its next equilibrium calculation. The more accurate the estimated initial condition, the less iterations are required in the Gibbs energy minimisation routine for the equilibrium calculation to reach convergence, reducing the computation time. The newly calculated equilibrium state is stored for the next iteration.

The performance of the re-initialisation method was tested in a number of systems with varying complexity based on realistic nuclear fuel applications. In a system with two components an acceleration factor of 2.5 was achieved. The acceleration factor was small because the overhead of storing and recalling equilibrium states was similar to the gain from the reduced number of Gibbs energy minimisation routine iterations (Poschmann et al. 2021). In more complex systems where 13 and 23 components were considered acceleration factors in the range of 5 to 40 was achieved (Poschmann et al. 2021). An application was demonstrated where an acceleration factor of 7.38 was achieved and was attributed to a 96.1 % reduction in the number of iterations performed during the Gibbs energy minimisation routine (Poschmann, Piro, and Simunovic 2020).

6.5 Parallelisation

Although continuous improvement of the different accelerator algorithms reduce the computational time of equilibrium calculations, another method of achieving large acceleration is by performing these calculations in parallel across multiple central processing unit (CPU) cores. A comparison between single-core and parallel computed equilibrium calculations were performed (Pillai et al. 2016) where a number of equilibrium calculations were performed with the Thermo-calc (Andersson et al. 2002) software. These equilibrium calculations were divided equally between a number of CPU cores with the MPI (Message Passing Interface) protocol and a close-to-linear scaling was observed (Pillai et al. 2016). This scaling continued up to a number of cores where the time required to transfer data between all cores became comparable to equilibrium calculations themselves – no longer advantageous to use more cores.

In one case where the equilibrium calculations were less computationally expensive, a linear scaling was found up to 24 cores. In another case where the equilibrium calculations were more computationally expensive, a linear scaling was found up to 48 cores (Pillai et al. 2016). Acceleration of equilibrium calculations are possible by dividing them amongst numerous CPU cores, but there are limits to the scaling and it differs from case to case. This method focuses on distributing the equilibrium calculations between a number of cores rather than storing the data for later recall and potentially avoiding performing each equilibrium calculation.

The in-situ database population and interpolation scheme of Larsson and Höglund (2015), as discussed in Section 6.3, was implemented and parallelised with the MPI protocol by Pillai et al. (2016). Performance tests were done on 4, 12 and 48 cores where the parallelised in-situ database population and interpolation algorithm was compared against direct equilibrium calculations on the same number of cores. The one test showed that the parallelised in-situ database population and interpolation algorithms was about 40 % to 50 % faster than parallelised direct equilibrium calculations. In another test it was 25 % to 50 % slower as the compositional space accessed by the model was larger and therefore more equilibrium calculations were required to populate the database (Pillai et al. 2016). In either case, acceleration was achieved in comparison to single-core equilibrium calculations, although the scaling is less than linear.

Some CPUs can have as many as 48 cores or even more, but graphics processing units (GPUs) can have much more; where it is not uncommon for GPUs to have several thousands of cores. Although a GPU has a substantial amount of cores more than a CPU, its cores has much less on-chip storage capabilities, a smaller instruction set, and slower clock speeds. This means that a single GPU core can not perform as computational demanding calculations as a CPU core and is not as fast as a CPU core, but because of the vast amount of cores at its disposal, a GPU can massively parallelise operations and perform them much quicker than a CPU can.

GPU parallelisation of equilibrium calculations was done by Gandham et al. (2016). To avoid latency in the equilibrium calculation being performed on a CPU or GPU, the data required to perform the calculation is stored within the on-chip cache which is present on both CPUs and GPUs. However, the cache size on a CPU is in general much larger than that of a GPU. As the number of system components increase, so does the number of registers required to store the data required to efficiently perform an equilibrium calculation. On a GPU, when this exceeds the maximum number of registers per thread, the GPU's dynamic random access memory (DRAM) is used for storage, leading to a reduction in performance. In contrast, the cache of a CPU is large enough to store all the data required for an equilibrium calculation for the test cases considered.

To reduce the demand for the GPU on-chip resources, a mixed-precision approach was taken where equilibrium calculations benefited from the performance of single-precision floating point computations and the accuracy of double-precision computations; majority of computations are performed with single-precision, requiring less on-chip storage, and double-precision only used to refine the solution to the desired accuracy when needed. It was found in the test cases that only 1 % to 2 % of equilibrium calculations that were performed required double-precision calculations for refinement. An acceleration factor of 3 was observed when this mixed-precision approach was used compared to using only double-precision computations for every equilibrium calculation.

To compare the performance between CPU and GPU parallelisation, Gandham et al. (2016) implemented the same accelerator algorithm for both processing units. The CPU used was a Intel Xeon E5-2630 v3 with eight cores clocked at 2.4 GHz and the GPU was an NVIDIA Tesla K80 board. Across two test cases, the average acceleration factor obtained by the GPU compared to the 8-core CPU (also performing equilibrium calculations in parallel) ranges from 5 for systems with 30 components to 17 for systems with 6 components.

6.6 Polynomial Fit to Thermodynamic properties

Thermodynamic properties can display discontinuities, as seen in Figure 6.2, with varying temperature as well as variations in system component concentrations. Instead of using a uniform grid to capture thermochemical property data, less nodes can be used if only these discontinuities are captured and stored (ten Cate et al. 2008). However, the locations of these discontinuities need to be known.

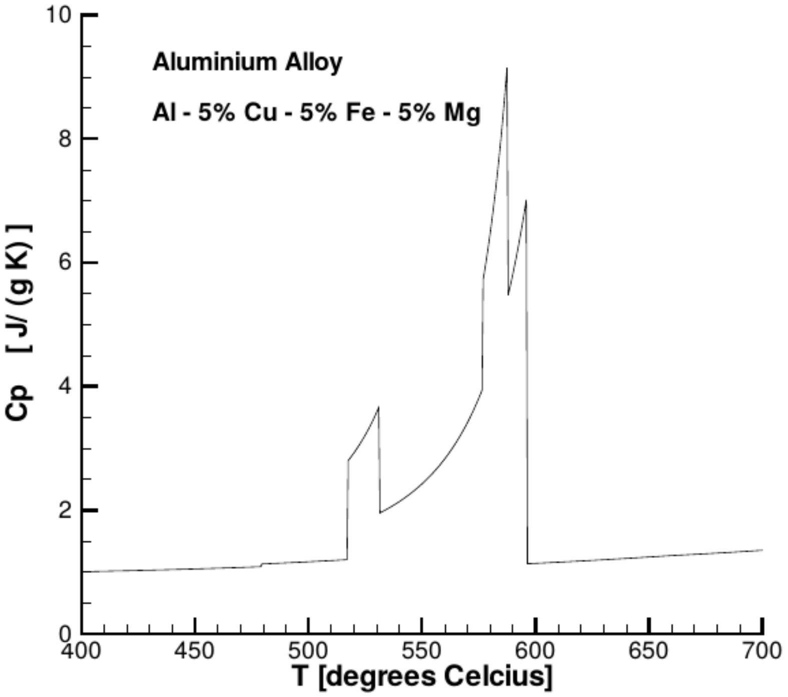


Figure 6.2: Dependency of heat capacity on temperature for an Aluminium alloy with fixed composition of 5 % Copper, Iron, and Magnesium. As found in ten Cate et al. (2008).

A diagram showing the compositions of all stable phases of a thermodynamic system present at equilibrium as a function of system component concentrations, temperature, and pressure is referred to as a phase diagram. A phase diagram is a geometrical representation of a thermodynamic system (Gibbs 1873). Locations of the observed discontinuities in thermodynamic properties correspond to phase region boundaries found on a phase diagram, and when grid nodes are adapted to the phase diagram shape in an unstructured non-uniform manner, the number of nodes used to discretise the temperature and compositional axes are expected to decrease.

The positions of discontinuities seen in Figure 6.2 were stored and first-order piecewise polynomials fitted between those nodes. First-order polynomials might not have been the most accurate simplification, but with the addition of only one node, more accurate second-order piecewise polynomials could be used.

To discretise and store the heat capacity, as depicted in Figure 6.2, for varying temperature and a fixed composition, only 17 nodes were needed if first-order piecewise polynomials were considered and 23 nodes for second-order piecewise polynomials (ten Cate et al. 2008). A reduction factor of around 30 was found for storage space requirement when this method was implemented (without a significant loss in the accuracy) in comparison to the uniform grid approach where 600 nodes were used to discretise the temperature axis – discussed in Section 6.1. The mapping was done for a fixed composition, but the composition will seldom be constant throughout the entire domain and model. The method of fitting piecewise polynomials to thermodynamic properties was therefore applied at a number of different system component concentrations. Instead of discretising the concentration of each component into a uniform grid, unstructured meshing was applied by clustering more nodes at important concentrations (ten Cate et al. 2008). It was estimated that a reduction factor for storage space requirement of about 3 can be obtained by applying this method to the discretisation of a single component's concentration range (ten Cate et al. 2008).

These reductions were applied to the same four-component system discussed in Section 6.1 where a uniform mesh was used and the database ended up to be in excess of 4 TB in size. If a reduction factor of 30 was found for the temperature axis and a factor of 3 for each composition axis, a total reduction factor of $30 \times 3^3 = 810$ was found. This meant that a 4 TB database would be reduced to about 5 GB (ten Cate et al. 2008).

Using a phase diagram to intelligently decide where to store thermodynamic data, rather than using a large uniform mesh, reduces the storage space requirement tremendously. This reduction makes the use of such a pre-calculated database more feasible but systems with more components would still require a large amount of storage space.

6.7 Polynomial Regression of Phase Region Boundaries

Regression has been used to fit polynomials to data points on phase region boundaries (Qiu et al. 2015). Using polynomials and linear interpolation is much less computationally intensive than Gibbs energy minimisation used in equilibrium calculations (Qiu et al. 2015).

Equilibrium calculations were performed at different concentrations of system components to obtain the liquidus and solidus temperatures of the phase region. Polynomial functions were fitted to these points to capture the phase region boundaries. The polynomial was fitted to the data points calculated from within the phase region, but the function could also be used with a set of independent variables outside the range of fitted data. The polynomial used to describe the phase region boundary therefore had to be bounded to capture the phase region boundary limits. Polynomials of a lower order were used to capture these limits.

This regression method was tested in the Al-Si-Mg-Fe system and compared against the equilibrium calculation software Thermo-Calc (Andersson et al. 2002). It was found that the liquidus and solidus temperatures determined by this method only differed by fractions of a degree Celsius to that calculated by Thermo-Calc (Qiu et al. 2015). Zhao et al. (2012) used a regression method in the Al-Cu-Si system. Compared to Thermo-Calc, the method had a maximum temperature error of 1.37 °C and less than a percentage error on any of the phase compositions. The direct integration with Themro-Calc took 3.66 hr compared to the regression method that only took 147.54 sec – an acceleration factor of almost 90.

6.8 Phase Diagram Discretisation

As discussed previously in Section 6.1, the number of nodes needed to represent the phase diagram become infeasible when uniform grid meshing is performed, especially for high-order systems. Meshing of a phase region, shown in Figure 6.3, was performed with a mesh generator developed by ten Cate et al. (2008). Non-uniform meshing, shown on the right, requires fewer nodes in comparison to a uniform mesh, shown on the left. A distance and size function were used to determine the desired edge length of a mesh cell depending on its distance to the nearest phase region boundary. This allowed for a non-uniform adaptive mesh to be applied to the region instead of a uniform mesh, reducing the number of cells needed to still accurately capture properties within the phase region. Equilibrium calculations were performed at each of the nodes and thermodynamic properties stored where interpolation could then be used when properties were requested by the model. This method can be used to include equilibrium calculations into models more efficiently and can even be implemented together with the Scheil-Gulliver solidification method (Sundman and Ansara 2008) to accelerate solidification models even more.

Fewer nodes are needed with non-uniform meshing to describe the phase region and will reduce the storage space requirement when thermodynamic properties are stored at every node, seeing that the storage space requirements are of order $O(n^2)$ where n is the number of discretisation cells (ten Cate et al. 2008).

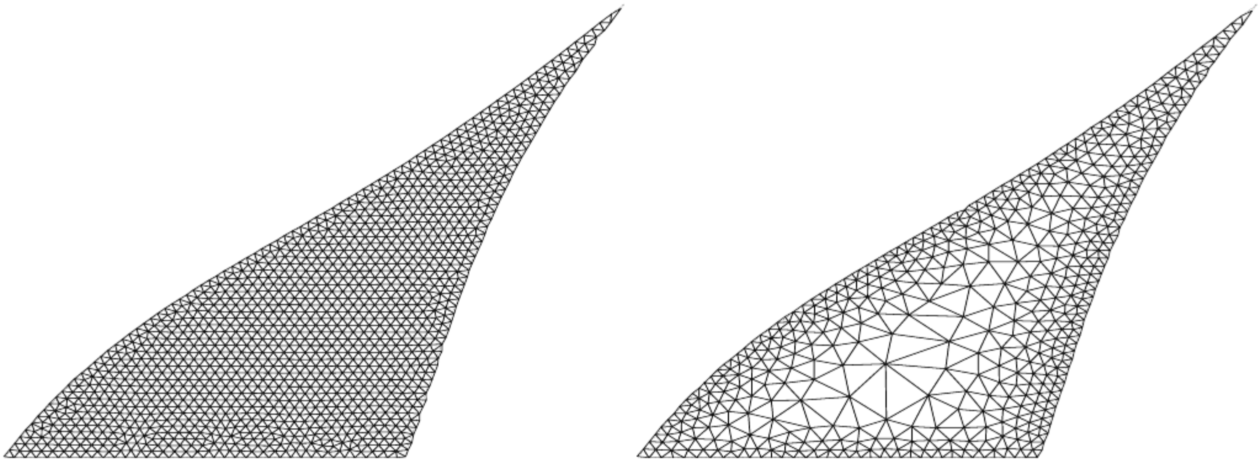


Figure 6.3: Meshing of the liquid-Pb phase region. Left: Uniform mesh size function. Right: Distance dependent mesh size function. As found in ten Cate et al. (2008).

The meshing of phase regions and the storage of thermochemical properties could be used effectively, but the meshing of all phase regions are not necessary. The phase fraction of all stable phases at equilibrium can be calculated with the lever rule (Smith and Hashemi 2006). Thermochemical properties of the system can be determined at a given system state within the phase region with the same chemical potential and non-compositional potentials (temperature and pressure) as the stable phases at equilibrium. This is done by taking the sum of the stable phases' thermochemical properties (properties at coordinates on the phase region boundaries) weighted by their calculated phase fractions (via the lever rule) towards the system state (properties at a coordinate within the phase region). For some phase regions, only the meshing of phase region boundaries would therefore suffice, and the lever rule can be used to interpolate within the phase region. This was the basis for a phase region discretisation accelerator proposed by Zietsman (2016).

It was shown that the Gibbs phase rule could be used to determine whether a phase region had to be meshed or only its boundaries. This would reduce the number of equilibrium calculations that had to be performed and the storage space requirement of the database immensely – an entire phase region can be described by its boundaries alone. For the liquid-Pb (two-phase) phase region in the Pb-Sn (two-component) system, seen in Figure 6.3, it would only be necessary to mesh the phase region boundaries according to the Gibbs phase rule. The lever rule could then been used to calculate thermodynamic properties within the phase region between the linearly interpolated phase region boundary nodes – no meshing would be required inside the phase region. The meshing of some phase regions are unavoidable, such as single-phase regions (Zietsman 2016).

6.9 Tie Simplices

Phase diagrams are geometrical representations of thermochemical systems (Gibbs 1873). There are many geometrical objects within a phase diagram, of which tie lines, tie triangles, etc. are used to represent behaviour within phase regions in isobaric and isothermal sections. A simplex is the generalization of a tetrahedral region of space in k-dimensions, such as a tetrahedron in three-dimensions, a triangle in two-dimensions or a line in one-dimension. Simplices can therefore be used to describe multi-dimensional tie-features in multi-phase regions. The vertices of a tie simplex are the compositions of the phases present at equilibrium – nodes on the phase region boundaries. A phase region with $\hat{\varphi}$ number of phases can be discretised by a series of tie simplices of order $\mathbb{R}^{\hat{\varphi}-1}$.

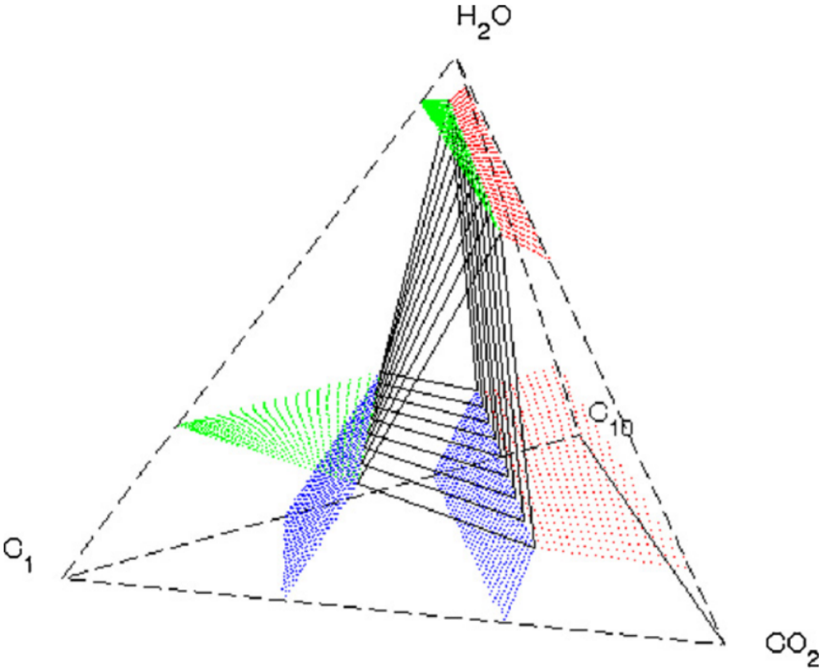


Figure 6.4: Phase diagram of four-component system with tie simplices capturing phase regions. As found in (Voskov and Tchelepi 2009).

Voskov and Tchelepi (2009) showed a method, illustrated in Figure 6.4, where, for a given phase region, the largest tie simplex was first identified; for a two-phase region this would be the longest tie line, for a three-phase region this would be the largest tie triangle, etc. From the initial tie simplex, an increment was made orthogonal to it and the next tie simplex calculated. This was repeated until the entire phase region was discretised by a set of tie simplices, as seen in Figure 6.4. Tie simplices were created for all the phase regions in the system. The phase compositions that are the tie simplex's vertices, and the thermochemical properties at each, were calculated by an equilibrium calculation for a given temperature and pressure within the phase region, and then stored. Interpolation could be performed between vertices of neighbouring tie simplices, and together with the lever rule, could calculate thermochemical properties to be used in models. The use of tie simplex database population had significant gains in computationally efficiency when compared to direct integration methods (Voskov and Tchelepi 2009).

An adaptive strategy was developed where only the necessary tie simplices were computed when a large number of these equilibrium calculations were required (Voskov and Tchelepi 2008). If sufficient tie simplex data was not available to provide the model with information regarding a query, equilibrium calculations would be performed and the database updated with corresponding tie simplices.

6.10 Support Vector Data Description

A support vector data description (SVDD), similar to a support vector machine, is a non-probabilistic machine-learning classifier that has been used to estimate the location of phase region boundaries of a thermochemical system (Kirk et al. 2018). Initial temperature and composition coordinates, either obtained from prior knowledge or an optimisation routine, are used to determine phase composition coordinates on phase region boundaries and create a crude support vector data description of the phase region boundaries. Thereafter, an adaptive sampling scheme is employed to determine more phase composition coordinates on the phase region boundaries where it is currently least defined, and the support vector data description is grown until a maximum number of coordinates are reached. In a case study performed by Kirk et al. (2018), a support vector data description, together with the adaptive sampling scheme, required just 1000 equilibrium calculations to represent phase region boundaries. A total of 505 000 direct equilibrium calculations were performed, as would be done in a traditional method, to verify the accuracy of the support vector data description and only a 5 % misclassification rate was observed – this is considered relatively accurate for its application (Kirk et al. 2018) considering the reduction in the number of equilibrium calculations that need to be performed.

6.11 Sensitivity Derivatives

An on-demand machine learning algorithm was developed that could quickly and accurately estimate equilibrium states based on stored results of previous equilibrium calculations (Leal et al. 2020). When an equilibrium calculation was performed, sensitivity derivatives with respect to temperature, pressure and system component concentrations where determined. These derivatives could be used, together with changes in input conditions such as temperature differences or change in the concentration of a system component, to estimate the equilibrium state.

Stored sensitivity derivatives can only be used to estimate an equilibrium state if the requested input system state is close to the reference state – the state at which the stored sensitivity derivatives were calculated. The most performance-critical step of this method is searching for an acceptable stored reference state (Leal et al. 2020). An acceptability test is first done on a reference state and only when the error was found to be within a specified tolerance could the reference state and its sensitivity derivatives be used in a first-order Taylor approximation to estimate the equilibrium state.

If no acceptable reference states were found, an equilibrium calculation would be performed and the accompanying sensitivity derivatives stored. No prior knowledge of the system or large pre-calculated databases were therefore required.

This method was tested and an acceleration factor of one to two orders of magnitude was achieved. Of the 1 000 000 equilibrium state requests from the model, only 258 equilibrium calculations had to be performed. The remaining equilibrium states were estimated by the on-demand machine learning algorithm. From these 258 equilibrium calculations, 91 % had been completed within the first 250 of the total 10 000 time steps.

6.12 Artificial Neural Networks

Artificial neural networks (ANNs) were trained on thermochemical properties obtained from equilibrium calculations performed for sets of compositions and temperatures. ANNs have been used together with empirical models (Christo et al. 1995) and integrated with probability density functions (Christo et al. 1996). The difficulty with selecting a composition range to be used as a training set is that the range is unknown prior to the simulation. A large training composition range can lead to unnecessary time-consuming equilibrium calculations to create the training set. If the training set composition range is too small then the ANN will not adequately represent the system.

To best capture the used compositional ranges of the model, statistical mapping was performed; the system components were selected and small-scale models were simulated by direct integration of equilibrium calculations (Christo et al. 1996). The small-scale models were used to generate a training set for the ANN that was representative of the larger system and its compositional ranges. The neural network was then trained on the automatically generated training set. The statistically mapped training sets were about 10 % of a populated database for the same system (Christo et al. 1996).

ANNs were used to accelerate equilibrium calculations by Guérillot and Bruyelle (2020) and an acceleration factor of 10 to 20 was achieved for most of the time steps. However, it was stated that ANNs do not conserve mass and the error increases with every time step and is highly dependent on how well the ANN is trained. To reduce the error a larger training set would be required. A suggestion was made to use the ANN's result as the initial guess for the equilibrium calculation software solver, potentially reducing the number of iterations required before convergence is achieved, reducing the computational cost.

ANNs were used to accelerate equilibrium calculations by Strandlund (2004) as well where an acceleration factor of 70 was observed with a 5 % maximum relative error allowed. Strandlund (2004) mentions that the number of hidden nodes should be reduced as much as possible in order to train ANNs as fast as possible, but the more hidden nodes are used, the more complicated functions the ANN can represent – thermochemical properties with more complex behaviour can be represented more accurately. Knowledge of the system is therefore important; knowledge of the behaviour of the thermochemical properties as temperature, pressure, and composition of the system varies can be used to intelligently select the number of hidden nodes required – reducing the amount of training data required and the time spent on training the ANN to accurately estimate thermochemical properties.

6.13 k-Nearest Neighbours

A k-NN (k-nearest neighbour) algorithm (Jha et al. 2018) was used to construct a thermochemical property model. Many equilibrium calculations were performed to create a training set of thermochemical properties as a function of the specified system states. After training was complete the k-NN algorithm could estimate thermochemical properties based on a specified system state by determining the nearest neighbours to the specified state and performing a weighted average between them. The error made by the k-NN model was within 1 % from the direct calculation results and showed an acceleration from 15.28 d to 4 min – a factor of about 5500. Other machine-learning methods, as referred to in (Jha et al. 2018), were considered when this method was developed.

6.14 Conclusion

The common denominator between the majority of these acceleration methods is based on a pre-calculated or in-situ calculated database of the specific thermochemical system and utilising the stored data in an interpolation algorithm to accelerate equilibrium calculations.

Pre-calculated databases have the drawback of possibly becoming too large for computers to use. A pre-calculated uniform grid database of two thermochemical properties in a four-component system was estimated to be in excess of 4 TB in size. By clustering nodes to important system component concentrations that correspond to discontinuities in thermodynamic properties this database could be reduced to 5 GB, which is significantly smaller but could still present recalling problems. For this reason, it would be better to have a database that is populated in-situ as the model is being solved – the model creates its own database of the accessed region for later recall. Initially this method is slower than direct integration, but as the database is populated the frequency of recalls and linear interpolation increase and the time spent on equilibrium calculations decrease, especially if many iterations and time steps are taken by the model. The database of an in-situ method only contains data of the system that the model has accessed before and no unnecessary data is generated and stored.

Using a system's phase diagram as a guide to discretise the temperature-compositional space allowed non-uniform discretisation of phase regions. Thermochemical property data could be stored at the nodes of the non-uniform mesh for later recall. The non-uniform discretisation allowed for less nodes to be used, compared to uniform discretisation, as the nodes could be concentrated at the phase region boundaries.

The discretisation of phase regions are however not always necessary. By utilising thermochemical theory such as the Gibbs phase rule it could be determined whether an entire phase region had to be discretised or if it was only necessary to discretise its boundaries and store the resulting tie simplices. The lever rule could be used with the stored tie simplices to determine the thermochemical properties within the entire phase region without there being a single data point stored inside the phase region. This leads to a remarkable decrease in the amount of data that needs to be stored. Because the lever rule is only another form of linear interpolation, the acceleration capability of this method shows great promise.

There are great advantages in using concepts from thermochemical theory such as phase diagrams, the Gibbs phase rule, and the lever rule in creating an accelerator. The thermochemical theory is a strong base that provides security for the decisions taken when the system is discretised and interpolation is performed with the stored data. Combining this with an in-situ method of discretisation would produce a sparse database that covers large temperature, pressure and composition ranges. An accelerator that utilises these thermochemical theories to build an in-situ database consisting of tie simplices created by equilibrium calculations could show great promise.

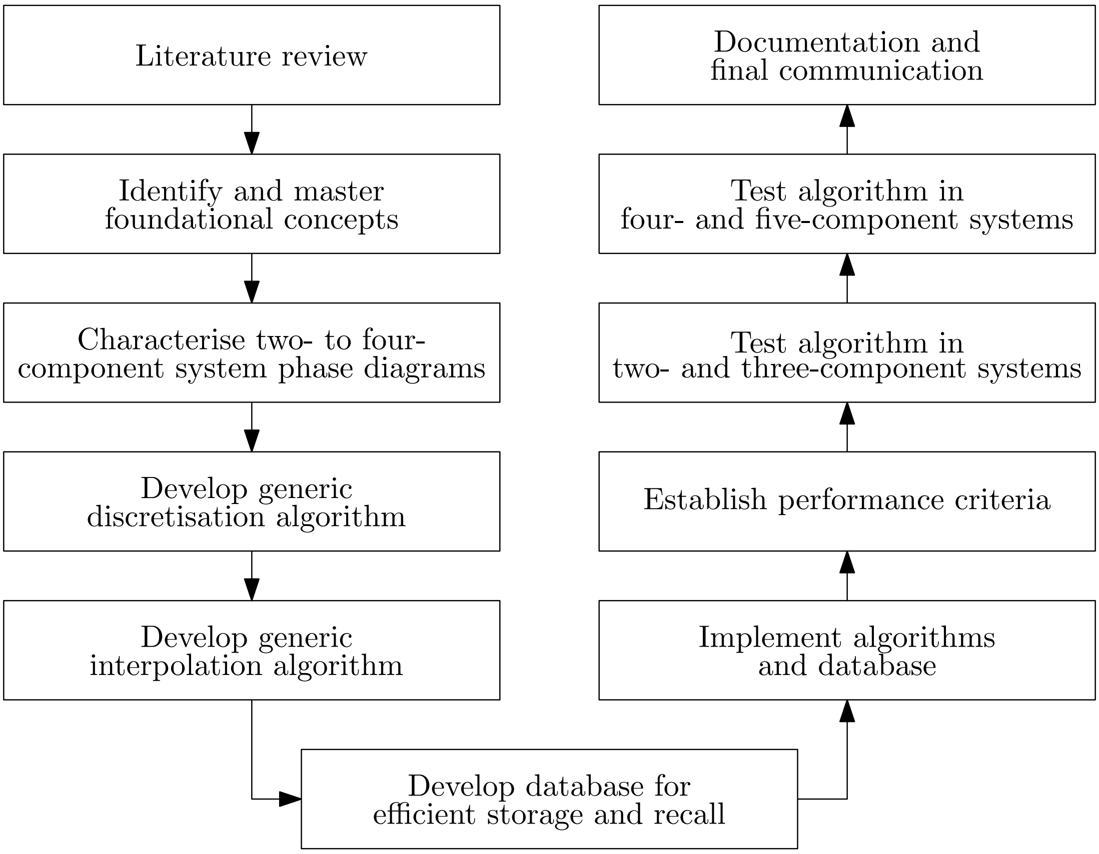
Part III

Research Approach and Methodology

Chapter 7

Approach

The high-level approach followed during this research is shown in Figure 7.1 and the steps are described below.



Literature review
Identify and master foundational concepts
Characterise two- to four-component system phase diagrams
Develop generic discretisation algorithm
Develop generic interpolation algorithm
Develop database for efficient storage and recall
Implement algorithms and database
Establish performance criteria
Test algorithm in two- and three-component systems
Test algorithm in four- and five-component systems
Documentation and final communication

Figure 7.1: Research approach flow diagram.

7.1 Literature Review

A review was done of the different methods that have been developed to accelerate equilibrium calculations. An algorithm was conceptualised that would utilise as many strengths as possible from existing methods while avoiding the identified pitfalls.

7.2 Identify and Master Foundational Concepts

From the literature study and conceptualised algorithm, foundational concepts were identified that would be needed to develop the accelerator algorithm. The most important identified concepts were phase diagram geometry, the Gibbs phase rule, the lever rule, and simplex mathematics. These concepts needed to be mastered so that the algorithm would be based on sound fundamental thermochemical theory, and it would utilise the generality of the foundational concepts to be generic.

7.3 Characterise Two- to Four-component System Phase Diagrams

Phase diagrams of various systems were studied to determine the relationship between the Gibbs phase rule and dimensionality of different phase region boundaries in these systems. The phase diagrams of systems with two to four components were visually inspected to verify the relationship. Having a method to verify the relationship in systems with small number of components that can still be visualised provided confidence that the relationship holds in systems with large number of components that cannot easily be visualised. This verified relationship is central to the discretisation algorithm and allows it to be applied to any system, regardless of the number of components.

7.4 Develop Generic Discretisation Algorithm

From the verified relationship between the Gibbs phase rule and the dimensionality of phase region boundaries, a generic accelerator algorithm was developed that can discretise any phase region, regardless of the number of stable phases, in a system with any number of components.

7.5 Develop Generic Interpolation Algorithm

A generic interpolation algorithm was developed based on the lever rule. It is known that the lever rule is generic and can be used in any phase region, regardless of the number of stable phases.

7.6 Develop Database for Efficient Storage and Recall

A phase diagram is a geometric representation of a thermochemical system. Because the phase regions, and therefore the phase diagram of a system is discretised and stored, a geometric database of the system is created. The geometric properties of the discretised phase regions were used to create an efficient uniform structured reference frame that improved the efficiency of database recalls.

7.7 Implement Algorithms and Database

The discretisation and interpolation algorithms, together with the geometric database and accompanying storage and recall algorithms, were implemented in the Python3 interpreted language. Although this language is not as computationally efficient as a compiled language, it was chosen because it has a large community with well-established libraries that allowed for easy and fast implementation and prototyping of these algorithms.

7.8 Establish Performance Criteria

Before any functionality and performance tests were performed, the way in which the performance and success of the accelerator algorithm are determined, were established.

7.9 Test Algorithm in Two- and Three-component Systems

Although testing the accelerator to determine its performance was important, at this stage of development it was more important to test the functionality of the accelerator. The accelerator was tested in two- and three-component systems to determine whether the algorithm functioned as intended; correctly discretised phase regions into discretisation cells, stored the discretisation cells at the correct indices in the database and recalled the correct discretisation cells when needed, as well as correctly and accurately interpolated within the recalled discretisation cells. These functionality tests were performed in two- and three-component systems because the created discretisation cells, and the interpolation within them, could still be visually inspected and verified that it was done correctly.

7.10 Test Algorithm in Four- and Five-component Systems

Once it was established that the accelerator algorithm functioned as intended and its performance was satisfactory, the accelerator's performance was tested in four- and five-component systems. The generality of the accelerator algorithm was tested with the four- and five-component systems as the discretisation cells could not easily be visualised and verified. The success of these tests gave confidence in the algorithm and that it would function as intended in any system, regardless of the number of components.

7.11 Documentation and Final Communication

The final step was to document all the findings from literature, describe the foundational concepts together with the developed nomenclature, present the developed algorithm and report on the functionality and performance test results.

Chapter 8

Methodology

8.1 Characterise Two- to Four-component System Phase Diagrams

The phase diagrams of several alloy and oxide systems were investigated to determine the relationship between the Gibbs phase rule and dimensionality of different phase region boundaries in these systems. The FactSage (Bale et al. 2016) software package was used to create phase diagrams, or rather sections of two- to four-component systems' phase diagrams. It was necessary to project sections of the systems' phase diagrams onto a 2D plane (that could be generated by FactSage) by taking one or more sections of constant pressure (isobar), temperature (isotherm), or concentrations (isopleth). The sections applied, based on the number of components, to perform 2D projections of a system's phase diagram are summarised in Table 8.1. The dimensionality of a phase region boundary was investigated by evaluating the boundary's shape on different sections and how it changed from one section to the next.

Table 8.1: Types of sections applied to visualise phase diagrams in 2D with FactSage (Bale et al. 2016).

Number of system components	Type of section		
	Isobar	Isotherm	Isopleth
2	Yes	No	No
3	Yes	Yes	No
4	Yes	Yes	Yes

8.2 Implement Algorithms and Database

The algorithm and database were implemented in the Python3 (Rossum and Drake 2009) interpreted language. There are compiled languages that would deliver faster solving times, but for development work it enables fast prototyping and investigations into different discretisation, storage and recall, and interpolation algorithms and sub-routines. The equilibrium calculation software used in the algorithm is ChemAppPy (Ex Mente Technologies 2019) – a package that makes the thermochemical data of ChemApp available in Python3. This allowed the algorithm to perform a large number of equilibrium calculations from within Python3.

Python3 also has a large number of powerful compiled libraries that can be used. The Numba library (Lam, Pitrou, and Seibert 2015) allows for just-in-time compilation of functions and classes and improve on the algorithm's performance. SciPy (Virtanen et al. 2020) is a powerful library that is used to create convex hulls of discretisation cells - even in high dimensions - and efficiently determine if a coordinate is within the convex hull. SciPy is also used to solve the system of non-linear equations to determine the phase compositions of an interpolated tie simplex.

8.3 Establish Performance Criteria

The interpolated equilibrium state $\tilde{\mathbf{Y}}^{\sigma}$ consists of the interpolated phase compositions $\tilde{\mathbf{x}}_{\epsilon}^{\varphi}$, phase fractions $\tilde{x}_{\varphi}^{\sigma}$, and physical $\tilde{\rho}^{\sigma}$ and thermochemical properties $\tilde{\tau}^{\sigma}$ of the system. The accelerator is not only developed with a reduction in computation time in mind but it has to maintain a degree of accuracy as well. Because linear interpolation is employed in the accelerator, some margin of error is expected, and when quantified, can be measured and compared to direct calculation.

8.3.1 Acceleration Factor

The acceleration factor (AF) indicates by what factor the computation time has been improved by the accelerator with respect to direct calculation, and is shown in Equation (8.1). When the accelerator's computation time $\Delta\mathrm{t}(\mathrm{accelerator})$ is less than direct calculation $\Delta\mathrm{t}(\mathrm{direct})$, the acceleration factor is larger than one – computation time has been accelerated.

$$\mathrm{AF} = \frac{\Delta\mathrm{t}(\mathrm{direct})}{\Delta\mathrm{t}(\mathrm{accelerator})} \tag{8.1}$$

8.3.2 Phase Composition

To calculate the error being made on the composition of phase i, the Euclidean distance is determined between the phase's composition as determined by the accelerator $\tilde{\mathbf{x}}_{\epsilon}^{\varphi_\mathrm{i}}$ and by direct calculation $\mathbf{x}_{\epsilon}^{\varphi_\mathrm{i}}$. The distance is calculated with Equation (8.2). This provides a positive scalar value that expresses the error being made.

$$\mathrm{d}x_{\epsilon}^{\varphi_\mathrm{i}} = \|\tilde{\mathbf{x}}_{\epsilon}^{\varphi_\mathrm{i}} - \mathbf{x}_{\epsilon}^{\varphi_\mathrm{i}}\| \tag{8.2}$$

In single-phase regions, no error is made because the phase composition always varies linearly in the single-phase region and no linear interpolation error can be made. The phase composition is simply the specified system state's composition.

8.3.3 Phase Fraction

The difference between the phase fraction of phase i calculated by the accelerator $\tilde{x}_{\varphi_\mathrm{i}}^{\sigma}$ and direct calculation $x_{\varphi_\mathrm{i}}^{\sigma}$ is determined with Equation (8.3).

$$\mathrm{d}x_{\varphi_\mathrm{i}}^{\sigma} = \tilde{x}_{\varphi_\mathrm{i}}^{\sigma} - x_{\varphi_\mathrm{i}}^{\sigma} \tag{8.3}$$

In single-phase region, no error can be made regarding the phase fraction because there simply is only one phase and its fraction is always one.

8.3.4 Properties

To measure the accuracy of physical and thermochemical properties, the percentage error between the property calculated by the accelerator and direct calculation, with respect to direct calculation, is determined. The property error calculation for a physical property is shown in Equation (8.4) and thermochemical property in Equation (8.5). This is done on the system physical and thermochemical properties only in this evaluation.

$$E\rho^{\sigma}(\%) = \frac{\tilde{\rho}^{\sigma} - \rho^{\sigma}}{\rho^{\sigma}} \times 100 \quad \text{where } \rho \in \{C_p\} \tag{8.4}$$

$$E\tau^{\sigma}(\%) = \frac{\tilde{\tau}^{\sigma} - \tau^{\sigma}}{\tau^{\sigma}} \times 100 \quad \text{where } \tau \in \{H, S, G\} \tag{8.5}$$

8.4 Test Algorithm in Two- and Three-component Systems

8.4.1 Test Conditions

The functionality and performance of the accelerator algorithm were tested in a number of two- and three-component alloy and oxide systems, listed in Table 8.2. In each system, a number of system states $\mathbf{X}^{\sigma}$ were randomly generated within the entire composition range of each system component and within a chosen temperature range. Each randomised system state was provided to the accelerator algorithm to obtain an interpolated equilibrium state $\tilde{\mathbf{Y}}^{\sigma}$ and then a direct calculation was performed for the same system state to obtain the calculated equilibrium state $\mathbf{Y}^{\sigma}$ which serves as control. The interpolated equilibrium states $\tilde{\mathbf{Y}}^{\sigma}$ were then compared to the calculated equilibrium states $\mathbf{Y}^{\sigma}$ to evaluate the accelerator algorithm's functionality and performance.

Table 8.2: Two- and three-component systems used to test the accelerator algorithm.

Alloy System	Oxide System
Al−Cu	$\mathrm{Al_2O_3{-}CaO}$
Al−Mn	$\mathrm{Al_2O_3{-}MgO}$
Al−Zn	$\mathrm{Al_2O_3{-}SiO_2}$
Fe−C	$\mathrm{CaO{-}SiO_2}$
Fe−Cr	$\mathrm{Al_2O_3{-}CaO{-}MgO}$
Fe−Si	$\mathrm{Al_2O_3{-}CaO{-}SiO_2}$
	$\mathrm{Al_2O_3{-}Fe_2O_3{-}MgO}$
	$\mathrm{Al_2O_3{-}Fe_2O_3{-}SiO_2}$

8.4.2 Result Presentation

Heat maps were generated on the system's phase diagram where the colour of each system state coordinate $\mathbf{X}^{\sigma}$ indicated the magnitude of acceleration factor AF, phase composition difference, phase fraction difference, and errors made on system physical $E\rho^{\sigma}$ and thermochemical properties $E\tau^{\sigma}$. Because each equilibrium state can have more than one phase present, the average phase composition difference $\mathrm{d}\bar{x}^{\varphi}_{\varepsilon}$ was determined between the present phases' composition difference $\mathrm{d}x^{\varphi_i}_{\varepsilon}$ and indicated on the heat map at the system state. The average phase fraction difference $\mathrm{d}\bar{x}^{\sigma}_{\varphi}$ was determined in the same manner and indicated on the heat map. Heat maps provide a clear indication on how the accelerator functions in different phase regions and were used to interrogate unexpected outcomes.

Distribution plots were created for each of the performance criteria but were separated based on the number of phases because the accelerator algorithm differs between these regions. Because no errors can be made regarding the phase composition and phase fraction in single-phase regions, as discussed in Section 8.3.2 and Section 8.3.3, the distribution plots of phase composition and phase fraction differences omit single-phase regions. Only the results from interpolated equilibrium states $\tilde{\mathbf{Y}}^{\sigma}$ were considered in the distribution plots. The calculated equilibrium states $\mathbf{Y}^{\sigma}$ were determined by direct calculations, and therefore no interpolation was performed and no errors made. Distribution plots provide a summary of the accelerator algorithm's performance in the different phase regions.

8.5 Test Algorithm in Four- and Five-component Systems

8.5.1 Test Conditions

The performance of the accelerator algorithm was tested in two systems related to real-life applications; a simplified ilmenite smelting system with four components, and a simplified iron- and steelmaking system with five components. In each system, several temperature and composition ranges were identified that correspond to regions that are encountered in the real-life process that would most-likely be accessed by a model when the accelerator is incorporated. For each system, a total of 1 000 000 system states $\mathbf{X}^{\sigma}$ were randomly generated from these identified regions. As with the tests performed in two- and three-component systems, the interpolated equilibrium states $\tilde{\mathbf{Y}}^{\sigma}$ were compared to the calculated equilibrium states $\mathbf{Y}^{\sigma}$ to evaluate the accelerator algorithm's performance.

8.5.2 Result Presentation

Distribution plots of the performance test results were presented in the same fashion as the tests performed in two- and three-component systems, discussed in Section 8.4.2. No heat maps were presented.

Part IV

Accelerator Algorithm

Chapter 9

Algorithm Development

9.1 Approach

The advantages and disadvantages of existing acceleration methods discussed in Chapter 6 were reviewed and compared. An accelerator algorithm could then be conceptualised that would utilise as many strengths as possible from existing acceleration methods while avoiding the identified pitfalls. The main conclusions from the review are presented below.

9.1.1 Geometrical Map of Thermochemical System

Based on the work of ten Cate et al. (2008), Qiu et al. (2015), Zhao et al. (2012), Voskov and Tchelepi (2008), and Voskov and Tchelepi (2009), it was concluded that (1) a phase diagram is a geometric representation of a thermochemical system, (2) phase regions and phase region boundaries are represented by geometric features, and (3) linear interpolation in the phase diagram is less computationally expensive than Gibbs energy minimisation. Equilibrium calculations can be performed and the phase compositions and thermochemical properties stored, geometrically mapping the thermochemical system to its phase diagram. Phase compositions are the primary information needed to be stored as they represent the locations on the phase region boundaries. Only the calculated thermochemical properties (or additional properties estimated from other material property models) that are required by the model have to be stored at the phase compositions and any unnecessary properties can be discarded. Once the system, or parts thereof, are represented by geometric features, can interpolation and the lever rule be employed to calculate the required thermochemical properties, within reasonable accuracy, at system temperatures and compositions where equilibrium calculations have not been performed yet. It was decided to develop an accelerator algorithm that maps the thermochemical system geometrically to its phase diagram.

9.1.2 Phase Region Boundary Discretisation

It was noticed in the work by ten Cate et al. (2008) that entire phase regions were discretised and thermochemical properties stored throughout the entire region. This is unnecessary because the Gibbs phase rule reduces dimensionality to the number of degrees of freedom. The lever rule then enables the calculation of thermochemical properties at any temperature, pressure and composition within a phase region when the phase compositions and thermochemical properties on the phase region boundaries are known. The lever rule can be used in any system regardless of the number of components and in any phase region; it does not reduce dimensionality in single-phase regions, though. Therefore, only the phase region boundaries have to be discretised, apart from single-phase regions, reducing the size of the database. The lever rule is applied in a tie simplex whose vertices are compositions of stable phases. By storing the tie simplices, as in (Voskov and Tchelepi 2009) and (Voskov and Tchelepi 2008), entire phase regions can be described based on a small number of direct equilibrium calculations. It was decided to employ the Gibbs phase rule to reduce the dimensionality of geometric features that need to be calculated and stored, and tie simplices, together with the lever rule, to reduce computation time.

9.1.3 In-situ Discretisation

Some acceleration methods make use of pre-calculated data, whether it is to populate a database or be used for machine learning. This requires detailed prior knowledge of the system because temperature and composition ranges have to be specified and the data generated within these ranges. If the ranges are set too widely, unnecessary data is generated, increasing the database storage space requirement and the computation time prior to solving the model. If the ranges are set too narrowly, the model may require data that is not contained within the database.

In the work done by Pope and Maas (1993), Liu and Pope (2005), and Lu and Pope (2009) an in-situ discretisation method was employed. Initially the database is empty. As the model requires data, the database is queried, and if no data is available for interpolation, it is generated by performing an equilibrium calculation and storing the results for future queries. In doing this, data is generated in parts of the system that the model accesses regularly and reduces the amount of unnecessary data present in the database. Another advantage of reducing database storage space is that searching time for usable data is reduced as well. Minimal prior knowledge of the system is required seeing that the accelerator creates its own database while the model is being solved. For these reasons, it was decided to populate the database in-situ.

9.2 Overview

Figure 9.1 provides a high-level overview of the accelerator algorithm. When a model requires an equilibrium state $\mathbf{Y}^{\sigma}$ at a specified system state $\mathbf{X}^{\sigma}$, the accelerator searches its database for a phase region cell $\delta^{\mathcal{R}}$ that contains $\mathbf{X}^{\sigma}$. If a phase region cell is found, it is used to create an interpolated tie simplex $\tilde{\Delta}^{\mathcal{R}}$ that describes the equilibrium states of the phases. The lever rule is employed to determine the phase fractions which are then used to calculate an interpolated system equilibrium state $\tilde{\mathbf{Y}}^{\sigma}$.

When a phase region cell is not found, a direct calculation is performed and a tie simplex $\Delta^{\mathcal{R}}$ is created and stored in the database. An attempt is then made to create a new phase region cell from all stored tie simplices. If successful, the new phase region cell is stored for later recall. Thereafter, the directly calculated system equilibrium state $\mathbf{Y}^{\sigma}$ is returned to the model.

The accelerator algorithm concept is generic, and is presented here for arbitrary numbers of independent system components $\hat{\epsilon}$ and non-compositional potentials $\hat{\psi}$. Its current implementation includes temperature as the only non-compositional potential, however. The algorithm is divided into three routines; (1) discretisation, (2) interpolation, and (3) storage and recall algorithms.

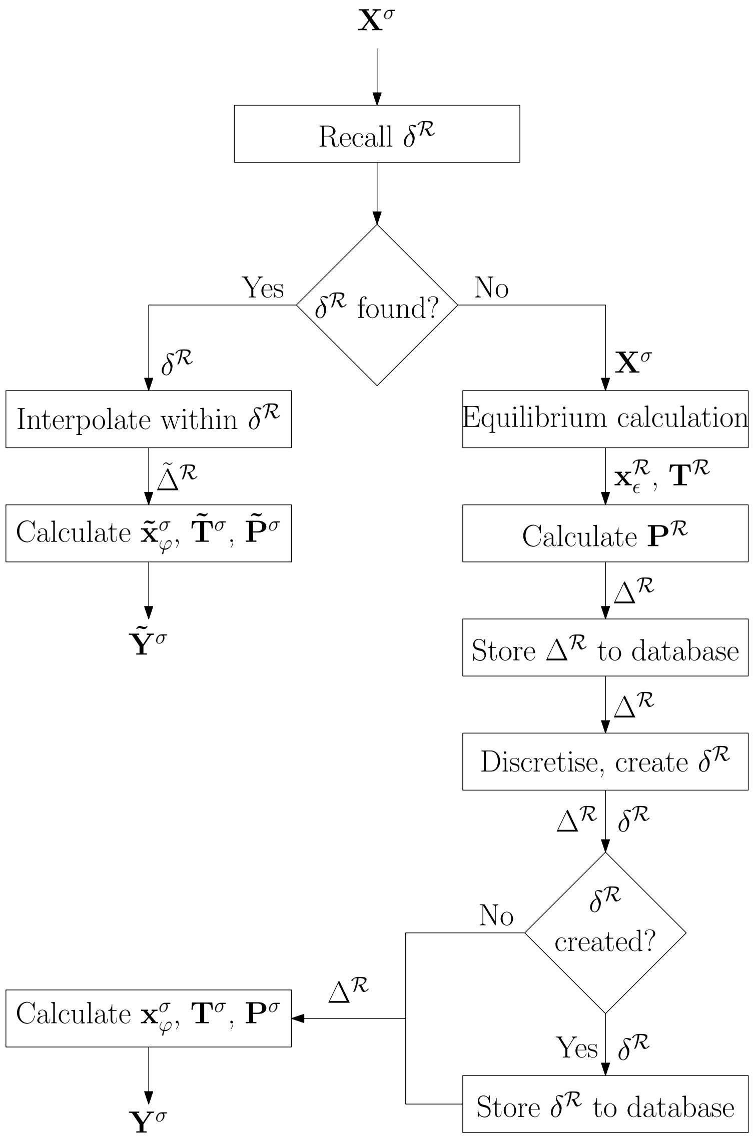

$\mathbf{X}^\sigma$
Recall $\delta^\mathcal{R}$
$\delta^\mathcal{R}$ found?
Yes
No
$\delta^\mathcal{R}$
$\mathbf{X}^\sigma$
Interpolate within $\delta^\mathcal{R}$
Equilibrium calculation
$\tilde{\Delta}^\mathcal{R}$
$\mathbf{x}^\mathcal{R}_\epsilon$, $\mathbf{T}^\mathcal{R}$
Calculate $\tilde{\mathbf{x}}^\sigma_\varphi$, $\tilde{\mathbf{T}}^\sigma$, $\tilde{\mathbf{P}}^\sigma$
Calculate $\mathbf{P}^\mathcal{R}$
$\Delta^\mathcal{R}$
$\tilde{\mathbf{Y}}^\sigma$
Store $\Delta^\mathcal{R}$ to database
$\Delta^\mathcal{R}$
Discretise, create $\delta^\mathcal{R}$
$\Delta^\mathcal{R}$
$\delta^\mathcal{R}$
No
$\delta^\mathcal{R}$ created?
$\Delta^\mathcal{R}$
Calculate $\mathbf{x}^\sigma_\varphi$, $\mathbf{T}^\sigma$, $\mathbf{P}^\sigma$
Yes
$\delta^\mathcal{R}$
Store $\delta^\mathcal{R}$ to database
$\mathbf{Y}^\sigma$

Figure 9.1: High-level flow diagram of the accelerator algorithm.

9.3 Discretisation

Figure 9.2 shows an overview of the algorithm's discretisation algorithm. The purpose of this algorithm is to discretise features of a system's phase diagram and create discrete finite regions of the phase diagram that can be used in the interpolation algorithm. Firstly, the features of a system's phase diagram that needs to be discretised, has to be identified. Thereafter can the discretisation of these features and the creation of discrete finite regions be discussed.

9.3.1 Features to Discretise

The lever rule allows for the calculation of phase fractions, thermochemical properties and some physical properties at any specified system state within a multi-phase region. Therefore, only OPF features $\mathcal{B}$, as shown in Figure 3.1, are discretised, rather than entire multi-phase regions. This reduces the number of direct calculations required to cover a phase region, and the amount of data that need to be stored, in comparison to previous methods that discretise entire phase regions (ten Cate et al. 2008). For some physical properties, such as viscosity, additional material property models are required to determine the property values from the calculated or interpolated equilibrium state.

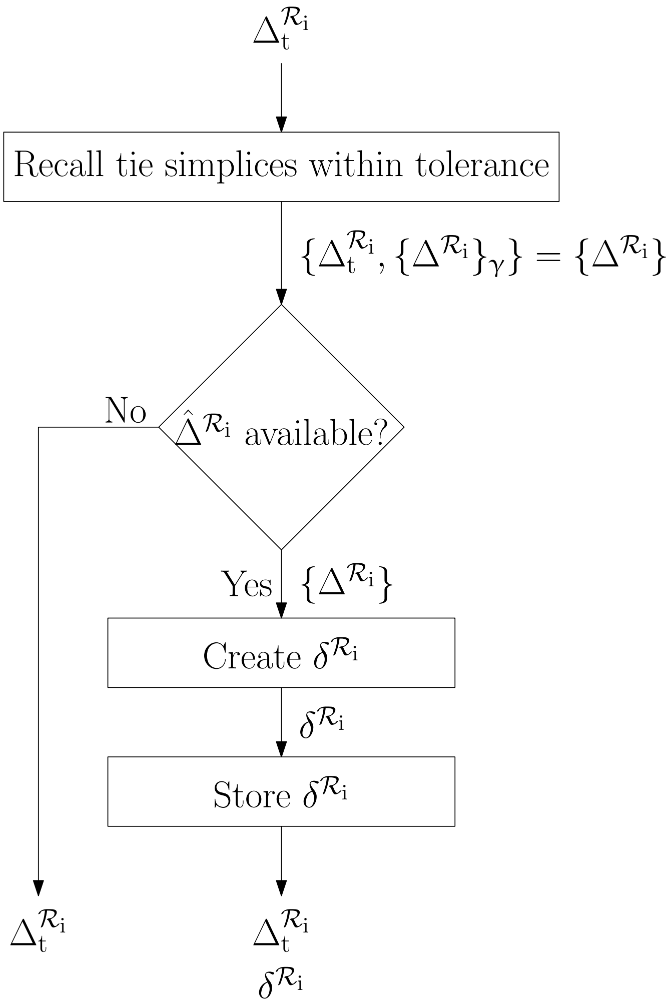

$\Delta_{\mathrm{t}}^{\mathcal{R}_{\mathrm{i}}}$
Recall tie simplices within tolerance
$\{\Delta_{\mathrm{t}}^{\mathcal{R}_{\mathrm{i}}}, \{\Delta^{\mathcal{R}_{\mathrm{i}}}\}_{\gamma}\} = \{\Delta^{\mathcal{R}_{\mathrm{i}}}\}$
$\hat{\Delta}^{\mathcal{R}_{\mathrm{i}}}$ available?
No
Yes
$\{\Delta^{\mathcal{R}_{\mathrm{i}}}\}$
Create $\delta^{\mathcal{R}_{\mathrm{i}}}$
$\delta^{\mathcal{R}_{\mathrm{i}}}$
Store $\delta^{\mathcal{R}_{\mathrm{i}}}$
$\Delta_{\mathrm{t}}^{\mathcal{R}_{\mathrm{i}}}$
$\Delta_{\mathrm{t}}^{\mathcal{R}_{\mathrm{i}}}$
$\delta^{\mathcal{R}_{\mathrm{i}}}$

Figure 9.2: Discretisation algorithm flow diagram.

9.3.2 Boundary Discretisation with Simplices

The algorithm uses k-simplices to discretise kD OPF boundaries. A single simplex is referred to as a discretisation segment $\delta^{\mathcal{B}}$ of OPF boundary $\mathcal{B}$. It has at least $\mathbb{D}_{\mathrm{g}}^{\mathcal{B}}+1$ vertices, and therefore requires this number of direct equilibrium calculations.

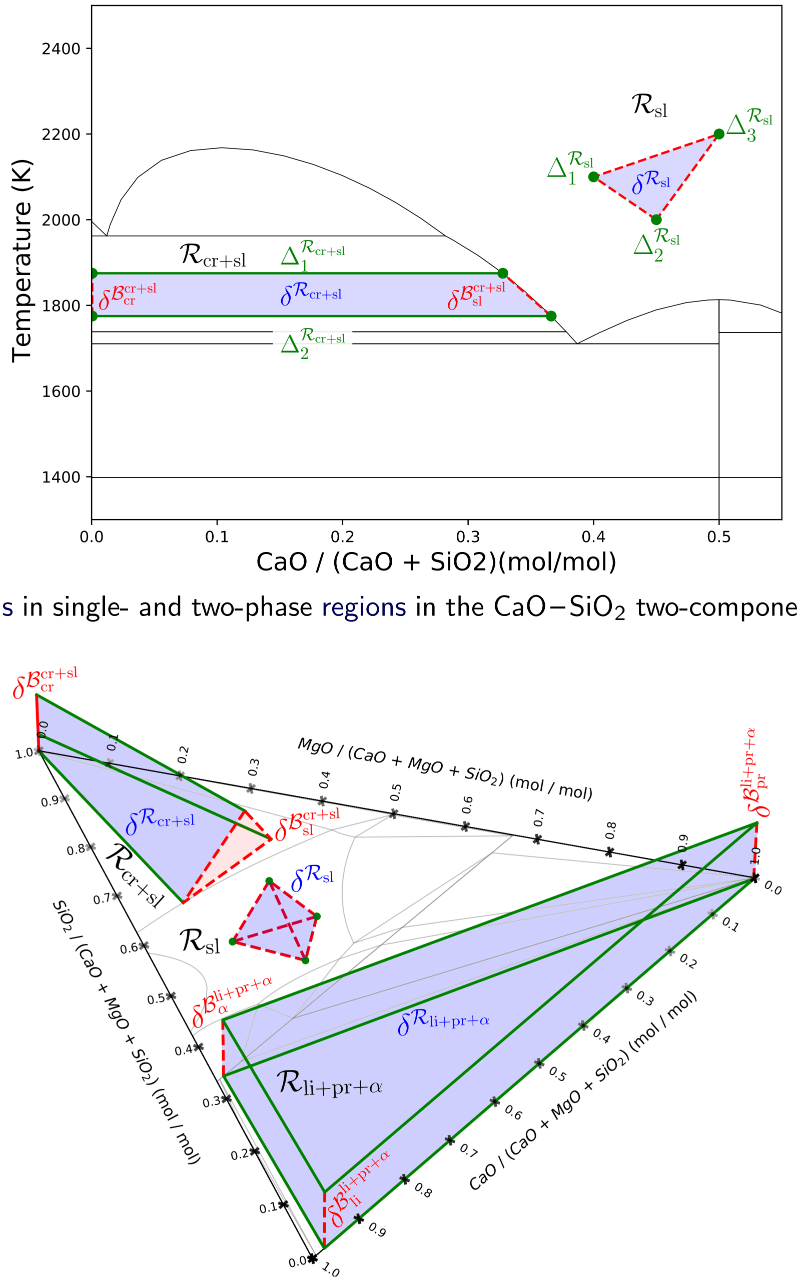
The OPF boundary $\mathcal{B}_{\mathrm{j}}^{\mathrm{i}}$ of phase j that has the highest geometrical dimensionality in phase region $\mathcal{R}_{\mathrm{i}}$ determines the number of tie simplices $\hat{\Delta}^{\mathcal{R}_{\mathrm{i}}}$ that must be directly calculated to create one complete discretisation segment $\delta^{\mathcal{B}_{\mathrm{j}}^{\mathrm{i}}}$ on each OPF boundary. This is calculated with Equation (9.1).

$$\hat{\Delta}^{\mathcal{R}_{\mathrm{i}}} = \max(\mathbb{D}_{\mathrm{g}}^{\mathcal{B}_{1}^{\mathrm{i}}}, \dots, \mathbb{D}_{\mathrm{g}}^{\mathcal{B}_{\varphi}^{\mathrm{i}}}) + 1 \tag{9.1}$$

For example, consider a two-phase region with a 1D OPF boundary and a 2D OPF boundary. The 1D boundary will require a line segment (2 vertices) and the 2D boundary will require a triangle (3 vertices) to generate one complete OPF boundary discretisation segment on each. Three equilibrium calculations are therefore required to generate three tie-lines, to generate a triangle on the 2D OPF boundary. This will yield three vertices on the 1D boundary, which is more than the minimum of two that is required to produce a line segment.

9.3.3 Phase Region Cell

A new tie simplex $\Delta_{\mathrm{t}}^{\mathcal{R}_{\mathrm{i}}}$ is created when a direct equilibrium calculation is performed. The database is then queried for all tie simplices $\{\Delta^{\mathcal{R}_{\mathrm{i}}}\}_{\gamma}$ that are within the specified temperature and composition tolerances ($\Delta T^{\mathcal{D}}$ and $\Delta x_{\epsilon}^{\mathcal{D}}$) from $\Delta_{\mathrm{t}}^{\mathcal{R}_{\mathrm{i}}}$ (discussed in more detail in Section 9.5). The tolerances allow control over the magnitude of interpolation errors. If $\hat{\Delta}^{\mathcal{R}_{\mathrm{i}}}$ tie simplices are available, they are combined to form a phase region cell $\delta^{\mathcal{R}_{\mathrm{i}}}$ that discretises a finite portion of $\mathcal{R}_{\mathrm{i}}$. Example phase region cells are shown in Figure 9.3 for two- and three-component systems. To cover further portions of $\mathcal{R}_{\mathrm{i}}$, more phase region cells are constructed by creating additional tie simplices through direct equilibrium calculations.



(a) Phase region cells in single- and two-phase regions in the CaO−SiO$_2$ two-component system. (sl: slag, cr: cristobalite)

(b) Phase region cells in single-, two-, and three-phase regions in the CaO−MgO−SiO$_2$ three-component system. (sl: slag, cr: cristobalite, li: lime, pr: periclase, α: α-Ca$_2$SiO$_4$). The vertical axis (not indicated) is the temperature axis.

Figure 9.3: Phase region cells in two- and three-component systems.

9.4 Interpolation

To calculate an interpolated equilibrium state $\tilde{\mathbf{Y}}^{\sigma}$ for a specified system state $\mathbf{X}^{\sigma}$ from a phase region cell $\delta^{\mathcal{R}_{\mathrm{i}}}$, an interpolated tie simplex $\tilde{\Delta}^{\mathcal{R}_{\mathrm{i}}}$ is created. This employs two interpolation steps, as shown in Figure 9.4; interpolation (1) to the non-compositional potentials $\mathbf{\Psi}^{\sigma}$ in $\mathbf{X}^{\sigma}$ and (2) to the chemical potentials $\boldsymbol{\mu}_{\epsilon}^{\sigma}$ corresponding to $\mathbf{X}^{\sigma}$.

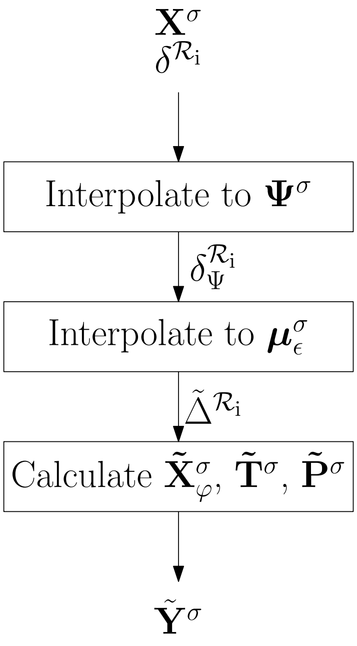

$\mathbf{X}^{\sigma}$
$\delta^{\mathcal{R}_{\mathrm{i}}}$
Interpolate to $\mathbf{\Psi}^{\sigma}$
$\delta_{\Psi}^{\mathcal{R}_{\mathrm{i}}}$
Interpolate to $\boldsymbol{\mu}_{\epsilon}^{\sigma}$
$\tilde{\Delta}^{\mathcal{R}_{\mathrm{i}}}$
Calculate $\tilde{\mathbf{X}}_{\varphi}^{\sigma}$, $\tilde{\mathbf{T}}^{\sigma}$, $\tilde{\mathbf{P}}^{\sigma}$
$\tilde{\mathbf{Y}}^{\sigma}$

Figure 9.4: Interpolation algorithm flow diagram.

9.4.1 Interpolation to Non-compositional Potentials

When interpolating to $\mathbf{\Psi}^{\sigma}$, the degrees of freedom, geometric dimensionality of the OPF boundaries, and consequently the number of tie simplices required to discretise the boundaries $\hat{\Delta}_{\psi}^{\mathcal{R}_{\mathrm{i}}}$, are reduced by $\hat{\psi}$, as shown in Equation (9.2). This yields an iso-non-compositional-potential (iso-ψ) section $\delta_{\psi}^{\mathcal{R}_{\mathrm{i}}}$ through $\delta^{\mathcal{R}_{\mathrm{i}}}$, consisting of $\hat{\Delta}_{\psi}^{\mathcal{R}_{\mathrm{i}}}$ interpolated iso-ψ tie simplices $\tilde{\Delta}_{\psi}^{\mathcal{R}_{\mathrm{i}}}$.

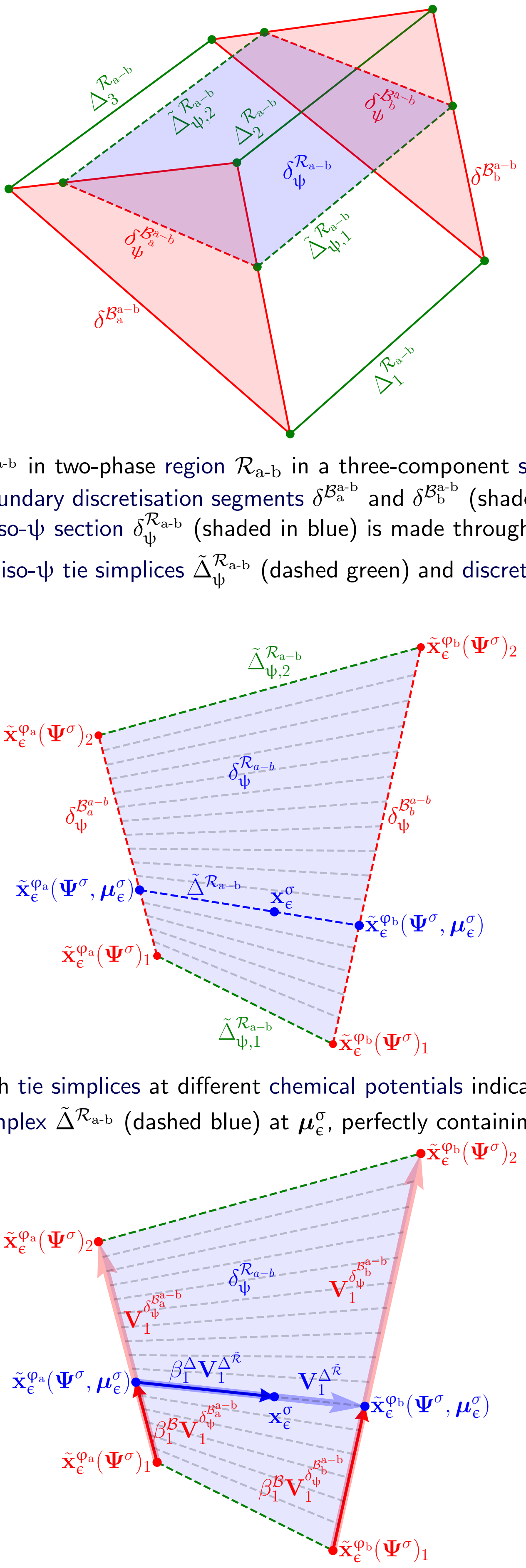
$$\hat{\Delta}_{\psi}^{\mathcal{R}} = \hat{\Delta}^{\mathcal{R}} - \hat{\psi} \tag{9.2}$$

Figure 9.5a shows an example phase region cell in a three-component system's two-phase region consisting of three tie simplices ($\hat{\Delta}^{\mathcal{R}} = 3$). When an iso-$\psi$ section $\delta_{\psi}^{\mathcal{R}_{\mathrm{i}}}$ is created through the phase region cell at the system temperature ($\hat{\psi} = 1$), two interpolated iso-ψ tie simplices ($\hat{\Delta}_{\psi}^{\mathcal{R}} = 2$) are produced. These tie simplices create an iso-ψ discretisation segment $\delta_{\psi}^{\mathcal{B}}$ on each OPF boundary.

Phase regions where $\hat{\epsilon} - \hat{\zeta}^{\sigma} = \hat{\varphi}$ have degrees of freedom $\mathrm{f} = \hat{\psi}$. In these cases, interpolating to $\mathbf{\Psi}^{\sigma}$ within a phase region cell yields only one iso-ψ tie simplex, which can be used as the final interpolated tie simplex $\tilde{\Delta}^{\mathcal{R}}$.

9.4.2 Interpolation to Chemical Potentials

Although the iso-ψ tie simplices $\tilde{\Delta}_{\psi}^{\mathcal{R}_{\mathrm{i}}}$ shown in Figure 9.5a, as well as all other tie simplices within $\delta_{\psi}^{\mathcal{R}_{\mathrm{i}}}$ have the same non-compositional potentials $\mathbf{\Psi}^{\sigma}$, they have different chemical potentials, as seen


$\Delta_3^{\mathcal{R}_{\text{a-b}}}$
$\tilde{\Delta}_{\psi,2}^{\mathcal{R}_{\text{a-b}}}$
$\Delta_2^{\mathcal{R}_{\text{a-b}}}$
$\delta_{\psi}^{\mathcal{B}_{b}^{a-b}}$
$\delta^{\mathcal{B}_{\text{b}}^{\text{a-b}}}$
$\delta_{\psi}^{\mathcal{R}_{\text{a-b}}}$
$\delta_{\psi}^{\mathcal{B}_{a}^{a-b}}$
$\tilde{\Delta}_{\psi,1}^{\mathcal{R}_{\text{a-b}}}$
$\delta^{\mathcal{B}_{\text{a}}^{\text{a-b}}}$
$\Delta_1^{\mathcal{R}_{\text{a-b}}}$

(a) A phase region cell $\delta^{\mathcal{R}_{\text{a-b}}}$ in two-phase region $\mathcal{R}_{\text{a-b}}$ in a three-component system where phases a and b are stable, with phase boundary discretisation segments $\delta^{\mathcal{B}_{\text{a}}^{\text{a-b}}}$ and $\delta^{\mathcal{B}_{\text{b}}^{\text{a-b}}}$ (shaded in red) and tie simplices $\Delta^{\mathcal{R}_{a-b}}$, (solid green). An iso-ψ section $\delta_{\psi}^{\mathcal{R}_{\text{a-b}}}$ (shaded in blue) is made through the phase region cell and is bounded by interpolated iso-ψ tie simplices $\tilde{\Delta}_{\psi}^{\mathcal{R}_{\text{a-b}}}$ (dashed green) and discretisation segments $\delta_{\psi}^{\mathcal{B}_{\text{a}}^{\text{a-b}}}$ and $\delta_{\psi}^{\mathcal{B}_{\text{b}}^{\text{a-b}}}$ (dashed red).

$\tilde{\Delta}_{\psi,2}^{\mathcal{R}_{\text{a-b}}}$
$\tilde{\mathbf{x}}_{\epsilon}^{\varphi_{\text{b}}}(\mathbf{\Psi}^{\sigma})_2$
$\tilde{\mathbf{x}}_{\epsilon}^{\varphi_{\text{a}}}(\mathbf{\Psi}^{\sigma})_2$
$\delta_{\psi}^{\mathcal{R}_{a-b}}$
$\delta_{\psi}^{\mathcal{B}_{a}^{a-b}}$
$\delta_{\psi}^{\mathcal{B}_{b}^{a-b}}$
$\tilde{\mathbf{x}}_{\epsilon}^{\varphi_{\text{a}}}(\mathbf{\Psi}^{\sigma},\boldsymbol{\mu}_{\epsilon}^{\sigma})$
$\tilde{\Delta}^{\mathcal{R}_{\text{a-b}}}$
$\mathbf{x}_{\epsilon}^{\sigma}$
$\tilde{\mathbf{x}}_{\epsilon}^{\varphi_{\text{b}}}(\mathbf{\Psi}^{\sigma},\boldsymbol{\mu}_{\epsilon}^{\sigma})$
$\tilde{\mathbf{x}}_{\epsilon}^{\varphi_{\text{a}}}(\mathbf{\Psi}^{\sigma})_1$
$\tilde{\Delta}_{\psi,1}^{\mathcal{R}_{\text{a-b}}}$
$\tilde{\mathbf{x}}_{\epsilon}^{\varphi_{\text{b}}}(\mathbf{\Psi}^{\sigma})_1$

(b) Iso-ψ section $\delta_{\psi}^{\mathcal{R}_{\text{a-b}}}$ with tie simplices at different chemical potentials indicated with grey dashed lines and the interpolated tie simplex $\tilde{\Delta}^{\mathcal{R}_{\text{a-b}}}$ (dashed blue) at $\boldsymbol{\mu}_{\epsilon}^{\sigma}$, perfectly containing $\mathbf{x}_{\epsilon}^{\sigma}$.

$\tilde{\mathbf{x}}_{\epsilon}^{\varphi_{\text{b}}}(\mathbf{\Psi}^{\sigma})_2$
$\tilde{\mathbf{x}}_{\epsilon}^{\varphi_{\text{a}}}(\mathbf{\Psi}^{\sigma})_2$
$\delta_{\psi}^{\mathcal{R}_{a-b}}$
$\mathbf{V}_1^{\delta_{\psi}^{\mathcal{B}_{b}^{a-b}}}$
$\mathbf{V}_1^{\delta_{\psi}^{\mathcal{B}_{a}^{a-b}}}$
$\beta_1^{\Delta}\mathbf{V}_1^{\Delta^{\tilde{\mathcal{R}}}}$
$\mathbf{V}_1^{\Delta^{\tilde{\mathcal{R}}}}$
$\tilde{\mathbf{x}}_{\epsilon}^{\varphi_{\text{a}}}(\mathbf{\Psi}^{\sigma},\boldsymbol{\mu}_{\epsilon}^{\sigma})$
$\tilde{\mathbf{x}}_{\epsilon}^{\varphi_{\text{b}}}(\mathbf{\Psi}^{\sigma},\boldsymbol{\mu}_{\epsilon}^{\sigma})$
$\mathbf{x}_{\epsilon}^{\sigma}$
$\beta_1^{\mathcal{B}}\mathbf{V}_1^{\delta_{\psi}^{\mathcal{B}_{a}^{a-b}}}$
$\tilde{\mathbf{x}}_{\epsilon}^{\varphi_{\text{a}}}(\mathbf{\Psi}^{\sigma})_1$
$\beta_1^{\mathcal{B}}\mathbf{V}_1^{\delta_{\psi}^{\mathcal{B}_{b}^{a-b}}}$
$\tilde{\mathbf{x}}_{\epsilon}^{\varphi_{\text{b}}}(\mathbf{\Psi}^{\sigma})_1$

(c) Vectors and barycentric coordinate weights used to describe $\mathbf{x}_{\epsilon}^{\sigma}$ as a function of all the $\tilde{\mathbf{x}}_{\epsilon}^{\varphi}(\mathbf{\Psi}^{\sigma})$.

Figure 9.5: Phase region cell interpolation steps in a two-phase region in a three-component system.

in Figure 9.5b. Somewhere within the iso-ψ section through the phase region cell $\delta_{\psi}^{\mathcal{R}_i}$ there exists a tie simplex, referred to as the interpolated tie simplex $\tilde{\Delta}^{\mathcal{R}}$, that contains the system composition $\mathbf{x}_{\epsilon}^{\sigma}$ and thus have the same chemical potentials $\boldsymbol{\mu}_{\epsilon}^{\sigma}$ as the specified system state, seen in Figure 9.5b. Therefore, the objective of this interpolation step is to find the phase compositions $\tilde{\mathbf{x}}_{\epsilon}^{\varphi}(\boldsymbol{\Psi}^{\sigma}, \boldsymbol{\mu}_{\epsilon}^{\sigma})$ that correspond to the vertices of the interpolated tie simplex $\tilde{\Delta}^{\mathcal{R}}$ that perfectly contains the system composition $\mathbf{x}_{\epsilon}^{\sigma}$.

As seen in Figure 9.5c, iso-ψ boundary discretisation segment barycentric coordinate weights $\boldsymbol{\beta}_{\mathrm{i}}^{\mathcal{B}}$ can be used to express the composition of phase i of the interpolated tie simplex $\tilde{\mathbf{x}}_{\epsilon}^{\varphi_i}(\boldsymbol{\Psi}^{\sigma}, \boldsymbol{\mu}_{\epsilon}^{\sigma})$ as a function of $\hat{\Delta}_{\psi}^{\mathcal{R}}$ iso-ψ boundary discretisation segment phase compositions $\tilde{\mathbf{x}}_{\epsilon}^{\varphi_i}(\boldsymbol{\Psi}^{\sigma})$. This expression can be seen in Equation (9.3). Because there are $\hat{\Delta}_{\psi}^{\mathcal{R}}$ iso-ψ boundary discretisation segment phase compositions $\tilde{\mathbf{x}}_{\epsilon}^{\varphi}(\boldsymbol{\Psi}^{\sigma})$, only $\hat{\Delta}_{\psi}^{\mathcal{R}} - 1$ iso-ψ boundary discretisation segment barycentric coordinate weights $\beta^{\mathcal{B}}$ are required in $\boldsymbol{\beta}_{\mathrm{i}}^{\mathcal{B}}$ to express the composition of phase i of the interpolated tie simplex $\tilde{\mathbf{x}}_{\epsilon}^{\varphi_i}(\boldsymbol{\Psi}^{\sigma}, \boldsymbol{\mu}_{\epsilon}^{\sigma})$.

$$\tilde{\mathbf{x}}_{\epsilon}^{\varphi_i}(\boldsymbol{\Psi}^{\sigma}, \boldsymbol{\mu}_{\epsilon}^{\sigma}) = \tilde{\mathbf{x}}_{\epsilon}^{\varphi_i}(\boldsymbol{\Psi}^{\sigma})_1 + \sum_{\mathrm{j}=1}^{\hat{\Delta}_{\psi}^{\mathcal{R}}-1} \beta_{\mathrm{j}}^{\mathcal{B}} \left(\tilde{\mathbf{x}}_{\epsilon}^{\varphi_i}(\boldsymbol{\Psi}^{\sigma})_{\mathrm{j}+1} - \tilde{\mathbf{x}}_{\epsilon}^{\varphi_i}(\boldsymbol{\Psi}^{\sigma})_1\right) \tag{9.3}$$

It is assumed that system component chemical potentials vary linearly between the phase composition coordinates of an iso-ψ boundary discretisation segment $\delta_{\psi}^{\mathcal{B}}$, as seen in Figures 9.5b and 9.5c. This assumption is reasonable when temperature and composition tolerances ($\Delta T^{\mathcal{D}}$ and $\Delta x_{\epsilon}^{\mathcal{D}}$) are sufficiently small. The assumption results in the set of iso-ψ boundary discretisation segment barycentric coordinate weights of all iso-ψ boundary discretisation segments being identical, as shown in Equation (9.4).

$$\boldsymbol{\beta}_1^{\mathcal{B}} = \cdots = \boldsymbol{\beta}_{\hat{\varphi}}^{\mathcal{B}} = \boldsymbol{\beta}^{\mathcal{B}} = [\beta_1^{\mathcal{B}}, \ldots, \beta_{\hat{\Delta}_{\psi}^{\mathcal{R}}-1}^{\mathcal{B}}] \tag{9.4}$$

Equation (9.3) only provides an expression of a phase composition contained within a iso-ψ boundary discretisation segment $\delta_{\psi}^{\mathcal{B}}$ – varying $\boldsymbol{\beta}^{\mathcal{B}}$ changes the location of these phase compositions on each iso-ψ boundary discretisation segment. These phase compositions have to be coupled to the system composition $\mathbf{x}_{\epsilon}^{\sigma}$ in order to determine the exact phase compositions that correspond to the chemical potential of the system composition.

Tie simplex barycentric coordinate weights $\boldsymbol{\beta}^{\Delta}$ can be used to express the specified system composition $\mathbf{x}_{\epsilon}^{\sigma}$ as a function of each of the $\hat{\varphi}$ phase compositions $\tilde{\mathbf{x}}_{\epsilon}^{\varphi}(\boldsymbol{\Psi}^{\sigma}, \boldsymbol{\mu}_{\epsilon}^{\sigma})$, as shown in Figure 9.5c. This expression is shown in Equation (9.5). An interpolated tie simplex that perfectly contains the system composition $\mathbf{x}_{\epsilon}^{\sigma}$ can be obtained if the conditions in Equation (3.16) are satisfied for all tie simplex barycentric coordinate weights in $\boldsymbol{\beta}^{\Delta}$. Because there are $\hat{\varphi}$ number of phases present in a tie simplex, $\hat{\varphi} - 1$ number of tie simplex barycentric coordinate weights β^{Δ} are required in $\boldsymbol{\beta}^{\Delta}$ to express the system composition $\mathbf{x}_{\epsilon}^{\sigma}$.

$$\mathbf{x}_{\epsilon}^{\sigma} = \tilde{\mathbf{x}}_{\epsilon}^{\varphi_1}(\boldsymbol{\Psi}^{\sigma}, \boldsymbol{\mu}_{\epsilon}^{\sigma}) + \sum_{\mathrm{i}=1}^{\hat{\varphi}-1} \beta_{\mathrm{i}}^{\Delta} \left(\tilde{\mathbf{x}}_{\epsilon}^{\varphi_{i+1}}(\boldsymbol{\Psi}^{\sigma}, \boldsymbol{\mu}_{\epsilon}^{\sigma}) - \tilde{\mathbf{x}}_{\epsilon}^{\varphi_1}(\boldsymbol{\Psi}^{\sigma}, \boldsymbol{\mu}_{\epsilon}^{\sigma})\right) \tag{9.5}$$

With the interpolated tie simplex's phase compositions $\tilde{\mathbf{x}}_{\epsilon}^{\varphi}(\boldsymbol{\Psi}^{\sigma}, \boldsymbol{\mu}_{\epsilon}^{\sigma})$ expressed in terms of iso-ψ boundary discretisation segment phase compositions $\tilde{\mathbf{x}}_{\epsilon}^{\varphi}(\boldsymbol{\Psi}^{\sigma})$ and barycentric coordinate weights

$\boldsymbol{\beta}^{\mathcal{B}}$ with Equation (9.3), the barycentric coordinate weights $\boldsymbol{\beta}^{\mathcal{B}}$ have to be solved that satisfy the conditions of Equation (3.16) for all tie simplex barycentric coordinate weights in $\boldsymbol{\beta}^{\Delta}$.

An expression for each of the interpolated tie simplex's $\hat{\varphi}$ phase compositions $\tilde{\mathbf{x}}_{\epsilon}^{\varphi}(\boldsymbol{\Psi}^{\sigma}, \boldsymbol{\mu}_{\epsilon}^{\sigma})$ is created with Equation (9.3) and substituted into Equation (9.5). This creates an expression of the system composition $\mathbf{x}_{\epsilon}^{\sigma}$ as a function of each iso-ψ boundary discretisation segment's phase compositions $\tilde{\mathbf{x}}_{\epsilon}^{\varphi}(\boldsymbol{\Psi}^{\sigma})$ and barycentric coordinate weights $\boldsymbol{\beta}^{\Delta}$ and $\boldsymbol{\beta}^{\mathcal{B}}$. This expression is factorised and written as a system of non-linear equations, shown in general form in Equation (9.6). A more detailed derivation of the generic system of non-linear equations can be found in Appendix A. The vectors and matrices have sizes as follow:

$\mathbf{x}_{\epsilon}^{\sigma}$ **:** $\hat{\epsilon} \times 1$ vector of the system composition.

$\mathbf{B}$ **:** $\hat{\epsilon} \times (\hat{\Delta}_{\psi}^{\mathcal{R}} \times \hat{\varphi})$ matrix of the iso-ψ boundary discretisation segment phase compositions.

$\mathbf{w}$ **:** $(\hat{\Delta}_{\psi}^{\mathcal{R}} \times \hat{\varphi}) \times 1$ vector of barycentric coordinate weight expressions.

$$\mathbf{x}_{\epsilon}^{\sigma} = \mathbf{B}\mathbf{w} \tag{9.6}$$

$$\mathbf{B} = \begin{bmatrix} \tilde{\mathbf{x}}_{\epsilon_{(\psi)}}^{\mathcal{B}_1} \\ \vdots \\ \tilde{\mathbf{x}}_{\epsilon_{(\psi)}}^{\mathcal{B}_{\hat{\varphi}}} \end{bmatrix}^{\mathsf{T}} \quad \text{where} \quad \tilde{\mathbf{x}}_{\epsilon_{(\psi)}}^{\mathcal{B}_i} = \begin{bmatrix} \tilde{\mathbf{x}}_{\epsilon}^{\varphi_i}(\boldsymbol{\Psi}^{\sigma})_1 \\ \vdots \\ \tilde{\mathbf{x}}_{\epsilon}^{\varphi_i}(\boldsymbol{\Psi}^{\sigma})_{\hat{\Delta}_{\psi}^{\mathcal{R}}} \end{bmatrix} \tag{9.7}$$

$$\mathbf{w} = \left[\begin{array}{c} (1-\Sigma\boldsymbol{\beta}^{\mathcal{B}})(1-\Sigma\boldsymbol{\beta}^{\Delta}) \\ (\beta_1^{\mathcal{B}})(1-\Sigma\boldsymbol{\beta}^{\Delta}) \\ \vdots \\ (\beta_{\hat{\Delta}_{\psi}^{\mathcal{R}}-1}^{\mathcal{B}})(1-\Sigma\boldsymbol{\beta}^{\Delta}) \\ \hdashline (1-\Sigma\boldsymbol{\beta}^{\mathcal{B}})(\beta_1^{\Delta}) \\ (\beta_1^{\mathcal{B}})(\beta_1^{\Delta}) \\ \vdots \\ (\beta_{\hat{\Delta}_{\psi}^{\mathcal{R}}-1}^{\mathcal{B}})(\beta_1^{\Delta}) \\ \hdashline \vdots \\ \hdashline (1-\Sigma\boldsymbol{\beta}^{\mathcal{B}})(\beta_{\hat{\varphi}-1}^{\Delta}) \\ (\beta_1^{\mathcal{B}})(\beta_{\hat{\varphi}-1}^{\Delta}) \\ \vdots \\ (\beta_{\hat{\Delta}_{\psi}^{\mathcal{R}}-1}^{\mathcal{B}})(\beta_{\hat{\varphi}-1}^{\Delta}) \end{array}\right] \tag{9.8}$$

$$\text{where} \quad \Sigma\boldsymbol{\beta}^{\mathcal{B}} = \sum_{j=1}^{\hat{\Delta}_{\psi}^{\mathcal{R}}-1} \beta_j^{\mathcal{B}} : \beta_j^{\mathcal{B}} \in \mathbb{R}[0,1]$$

$$\text{and} \quad \Sigma\boldsymbol{\beta}^{\Delta} = \sum_{i=1}^{\hat{\varphi}-1} \beta_i^{\Delta} : \beta_i^{\Delta} \in \mathbb{R}[0,1]$$

This system of non-linear equations is solved iteratively to obtain barycentric coordinate weight vector $\mathbf{w}$ and with it, the $\boldsymbol{\beta}^{\mathcal{B}}$ barycentric coordinate weights that relate to phase compositions of an interpolated tie simplex $\tilde{\mathbf{x}}_{\epsilon}^{\varphi}(\boldsymbol{\Psi}^{\sigma}, \boldsymbol{\mu}_{\epsilon}^{\sigma})$ through Equation (9.3) that perfectly contains the system's composition. The physical and thermochemical properties of each phase can then be calculated and the complete interpolated tie simplex $\tilde{\Delta}^{\mathcal{R}}$ has been obtained.

The iterative solving of a system of non-linear equations is not desired when attempting to accelerate equilibrium calculations. An explicit expression would be preferred, but no such generic explicit expression could yet be derived or found in literature.

9.4.3 Calculate the Interpolated Equilibrium State

With the interpolated tie simplex $\tilde{\Delta}^{\mathcal{R}}$ calculated, the lever rule is used to calculate phase fractions $\tilde{\mathbf{x}}_{\varphi}^{\sigma}$ at the specified system composition $\mathbf{x}_{\epsilon}^{\sigma}$ and in turn are used to calculate the physical and thermochemical properties of the system. This yields the interpolated equilibrium state $\tilde{\mathbf{Y}}^{\sigma}$.

9.5 Storage and Recall

The in-situ approach to populate the database with tie simplices and phase region cells is inherently unstructured; data is generated and stored at temperatures and phase compositions that have no fixed intervals. Searching through such unstructured data is inefficient, which leads to extended searching times. An efficient storage and recall algorithm is therefore needed.

9.5.1 A Structured Reference Frame

To configure the accelerator, limits and tolerances are specified for all non-compositional potentials, and for each system component. Taking temperature as an example, the limits are $T_{\text{min}}^{\mathcal{D}}$ and $T_{\text{max}}^{\mathcal{D}}$, and the tolerance is $\Delta T^{\mathcal{D}}$. In the case of composition, for all system components limits of zero and one are used and a single tolerance $\Delta x_{\epsilon}^{\mathcal{D}}$ is specified.

The specified limits and tolerances define a structured and uniform reference frame that covers the system's phase diagram, $\mathcal{D}$, as illustrated in Figure 9.6a. The size of each reference frame cell $\delta^{\mathcal{D}}$ is equal to the specified tolerances. Cells are identified with a non-compositional potential index array $\mathbf{I}^{\boldsymbol{\Psi}}$ and a compositional potential index array $\mathbf{I}^{\epsilon}$, with index values calculated according to Equations (9.9) and (9.10).

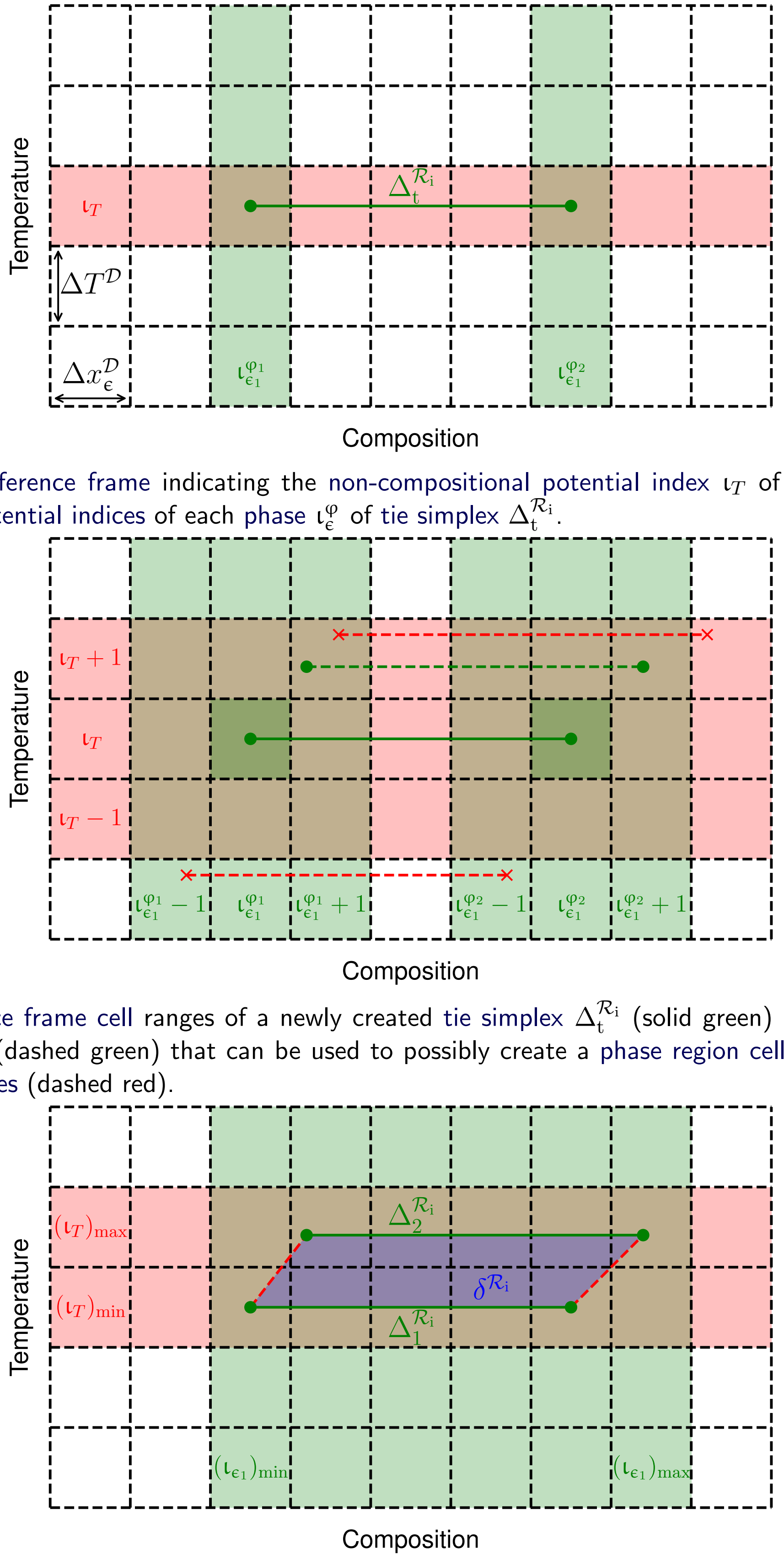
$$\iota_T = \text{floor}\left(\frac{T^{\sigma} - T_{\text{min}}^{\mathcal{D}}}{\Delta T^{\mathcal{D}}}\right) \tag{9.9}$$

$$\iota_{x_{\epsilon}} = \text{floor}\left(\frac{x_{\epsilon}^{\sigma}}{\Delta x_{\epsilon}^{\mathcal{D}}}\right) \tag{9.10}$$

9.5.2 Storing Tie Simplices

Figure 9.7 shows the flow diagram of the tie simplex storage algorithm. When a tie simplex is created, the ID of the phase region it is found within $\mathcal{R}_{\mathrm{i}}$ is determined. All tie simplices found in this phase region will have the same phase region ID. Thereafter, a unique tie simplex ID, t, is assigned to it.

The non-compositional potential index array $\mathbf{I}^{\boldsymbol{\Psi}}$ of the tie simplex is determined and then, for each of the tie simplex's $\hat{\varphi}$ phases, the compositional potential index array $\mathbf{I}^{\epsilon}$ in the structured reference


Temperature
ι_T
$\Delta_{\mathrm{t}}^{\mathcal{R}_{\mathrm{i}}}$
$\Delta T^{\mathcal{D}}$
$\Delta x_{\epsilon}^{\mathcal{D}}$
$\iota_{\epsilon_1}^{\varphi_1}$
$\iota_{\epsilon_1}^{\varphi_2}$
Composition
Temperature
$\iota_T + 1$
ι_T
$\iota_T - 1$
$\iota_{\epsilon_1}^{\varphi_1} - 1$
$\iota_{\epsilon_1}^{\varphi_1}$
$\iota_{\epsilon_1}^{\varphi_1} + 1$
$\iota_{\epsilon_1}^{\varphi_2} - 1$
$\iota_{\epsilon_1}^{\varphi_2}$
$\iota_{\epsilon_1}^{\varphi_2} + 1$
Composition
Temperature
$(\iota_T)_{\mathrm{max}}$
$(\iota_T)_{\mathrm{min}}$
$\Delta_2^{\mathcal{R}_{\mathrm{i}}}$
$\delta^{\mathcal{R}_{\mathrm{i}}}$
$\Delta_1^{\mathcal{R}_{\mathrm{i}}}$
$(\iota_{\epsilon_1})_{\mathrm{min}}$
$(\iota_{\epsilon_1})_{\mathrm{max}}$
Composition

(a) Structured reference frame indicating the non-compositional potential index ι_T of temperature and compositional potential indices of each phase $\iota_{\epsilon}^{\varphi}$ of tie simplex $\Delta_{\mathrm{t}}^{\mathcal{R}_{\mathrm{i}}}$.

(b) Valid reference frame cell ranges of a newly created tie simplex $\Delta_{\mathrm{t}}^{\mathcal{R}_{\mathrm{i}}}$ (solid green) and example of a valid tie simplex (dashed green) that can be used to possibly create a phase region cell and examples of invalid tie simplices (dashed red).

(c) Reference frame cell ranges of compositional and non-compositional potential indices that fully contain the phase region cell $\delta^{\mathcal{R}_{\mathrm{i}}}$.

Figure 9.6: Reference frame cells that are used in the storage and recall of tie simplices and phase region cells.

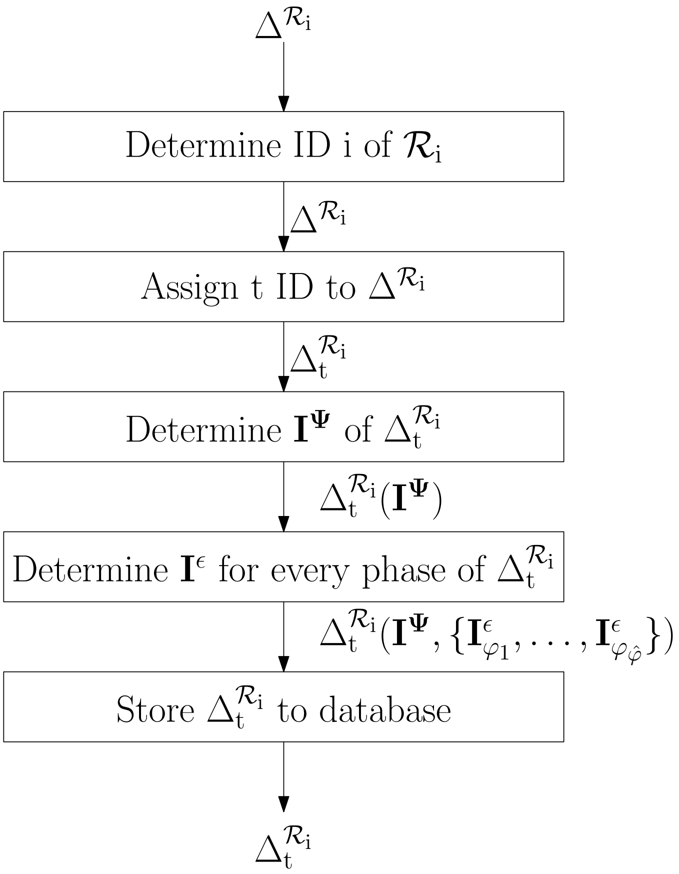

$\Delta^{\mathcal{R}_\mathrm{i}}$
Determine ID i of $\mathcal{R}_\mathrm{i}$
$\Delta^{\mathcal{R}_\mathrm{i}}$
Assign t ID to $\Delta^{\mathcal{R}_\mathrm{i}}$
$\Delta_\mathrm{t}^{\mathcal{R}_\mathrm{i}}$
Determine $\mathbf{I}^{\Psi}$ of $\Delta_\mathrm{t}^{\mathcal{R}_\mathrm{i}}$
$\Delta_\mathrm{t}^{\mathcal{R}_\mathrm{i}}(\mathbf{I}^{\Psi})$
Determine $\mathbf{I}^{\epsilon}$ for every phase of $\Delta_\mathrm{t}^{\mathcal{R}_\mathrm{i}}$
$\Delta_\mathrm{t}^{\mathcal{R}_\mathrm{i}}(\mathbf{I}^{\Psi}, \{\mathbf{I}^{\epsilon}_{\varphi_1}, \ldots, \mathbf{I}^{\epsilon}_{\varphi_{\hat{\varphi}}}\})$
Store $\Delta_\mathrm{t}^{\mathcal{R}_\mathrm{i}}$ to database
$\Delta_\mathrm{t}^{\mathcal{R}_\mathrm{i}}$

Figure 9.7: Tie simplex storage algorithm flow diagram.

frame is determined. Each of the phases' $\mathbf{I}^{\epsilon}$ in combination with the tie simplex's $\mathbf{I}^{\Psi}$ refers to a single cell in the structured reference frame $\delta^{\mathcal{D}}$, as shown in Figure 9.6a.

The complete description of the tie simplex is; (1) the tie simplex ID t, (2) the phase region ID $\mathcal{R}_\mathrm{i}$, (3) the non-compositional potential index array $\mathbf{I}^{\Psi}$ of the tie simplex, and (4) the compositional potential index array for each of the $\hat{\varphi}$ phases $\{\mathbf{I}^{\epsilon}_{\varphi_1}, \ldots, \mathbf{I}^{\epsilon}_{\varphi_{\hat{\varphi}}}\}$. The tie simplex, together with its complete description, is stored to the database.

9.5.3 Tie Simplices Within Tolerance

Because linear interpolation is performed between tie simplices to create an interpolated tie simplex, errors can be made with phase compositions, as shown in Figure 9.8, but also with physical and thermochemical properties of the phases.

To control the interpolation error magnitude, only tie simplices that are within the specified temperature tolerance $\Delta T^{\mathcal{D}}$ and composition tolerance $\Delta x_{\epsilon}^{\mathcal{D}}$ are used to create a phase region cell. The temperature tolerance is enforced by determining the difference between two tie simplices' temperatures, shown in Equation (9.11). The composition tolerance is enforced by determining the distance between the compositions of the same phase of two tie simplices, shown in Equation (9.12).

$$\left| T^{\Delta_\mathrm{i}^{\mathcal{R}}} - T^{\Delta_\mathrm{j}^{\mathcal{R}}} \right| <= \Delta T^{\mathcal{D}} \tag{9.11}$$

$$\| \mathbf{x}_{\epsilon}^{\varphi_\mathrm{k} \Delta_\mathrm{i}^{\mathcal{R}}} - \mathbf{x}_{\epsilon}^{\varphi_\mathrm{k} \Delta_\mathrm{j}^{\mathcal{R}}} \| <= \Delta x_{\epsilon}^{\mathcal{D}} \tag{9.12}$$

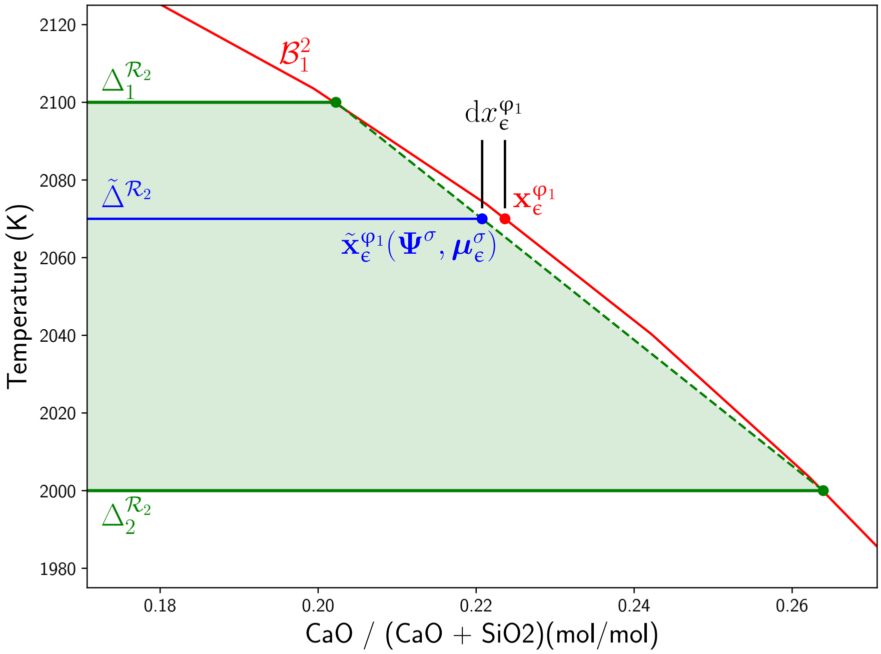


Figure 9.8: Illustration of error that can be made when linear interpolation is performed to obtain the phase composition $\tilde{\mathbf{x}}_\epsilon^{\varphi_1}(\mathbf{\Psi}^\sigma, \boldsymbol{\mu}_\epsilon^\sigma)$ of an interpolated tie simplex $\tilde{\Delta}^{\mathcal{R}_2}$ from two tie simplices $\Delta_1^{\mathcal{R}_2}$ and $\Delta_2^{\mathcal{R}_2}$. The temperature and composition tolerances depicted here are much larger than what would normally be specified, for illustration purposes.

When two tie simplices are within $\Delta T^{\mathcal{D}}$ from one another and each of their phases' compositions are within $\Delta x_\epsilon^{\mathcal{D}}$ from one another are they considered to be close enough to be included into a phase region cell.

9.5.4 Recalling Tie Simplices

Figure 9.9 shows the flow diagram of the tie simplex recall algorithm. An equilibrium calculation is performed and a new tie simplex $\Delta_t^{\mathcal{R}_i}$ created and stored, as discussed. A number of tie simplices surrounding the newly created tie simplex has to be found that are within tolerance, and then recalled to create a phase region cell.

The phase region to which the new tie simplex belong $\mathcal{R}_i$ is known, and only the stored tie simplices belonging to the same phase region has to be considered. This filter is applied and a list of eligible tie simplices returned $\{\Delta^{\mathcal{R}_i}\}_\gamma$.

Because the cell sizes of the reference frame is driven by the specified tolerances, only the set of reference frame cells surrounding the new tie simplex have to be searched. Anything further would automatically not be within tolerance, as shown in Figure 9.6b.

A tie simplex has $\hat{\psi}$ of non-compositional potential indices; $(\iota_\psi)_t$ refers to the new tie simplex and $(\iota_\psi)_j$ to a tie simplex in the recall set $\{\Delta^{\mathcal{R}_i}\}_\gamma$. Only when all of a tie simplex's non-compositional potential indices $(\iota_\psi)_j$ are within the same or neighbouring non-compositional potential indices as the new tie simplex $(\iota_\psi)_t$, as expressed in Equation (9.13) and illustrated in Figure 9.6b, is the tie simplex considered further. Otherwise it is eliminated from the recall set $\{\Delta^{\mathcal{R}_i}\}_\gamma$.

Each of the $\hat{\varphi}$ number of phases of a tie simplex have $\hat{\epsilon}$ of compositional potential indices; $(\iota_{x_\epsilon}^{\varphi_k})_t$ refers to phase k of the new tie simplex and $(\iota_{x_\epsilon}^{\varphi_k})_j$ of a tie simplex in the recall set $\{\Delta^{\mathcal{R}_i}\}_\gamma$. Only when each phase's compositional potential indices $(\iota_{x_\epsilon}^{\varphi_k})_j$ of a tie simplex are within the same or neighbouring compositional potential indices as the new tie simplex $(\iota_{x_\epsilon}^{\varphi_k})_t$, as expressed in Equation (9.14) and illustrated in Figure 9.6b, is the tie simplex considered further. Otherwise its eliminated from the recall set $\{\Delta^{\mathcal{R}_\mathrm{i}}\}_\gamma$.

$$(\iota_\psi)_\mathrm{j} \in [(\iota_\psi)_\mathrm{t} - 1, (\iota_\psi)_\mathrm{t} + 1] \tag{9.13}$$

$$(\iota_{x_\epsilon}^{\varphi_\mathrm{k}})_\mathrm{j} \in \left[(\iota_{x_\epsilon}^{\varphi_\mathrm{k}})_\mathrm{t} - 1, (\iota_{x_\epsilon}^{\varphi_\mathrm{k}})_\mathrm{t} + 1\right] \tag{9.14}$$

This is computationally cheap way of reducing the number of tie simplices that have to be investigated to confirm whether they are within tolerance. Each of the tie simplices in the recall set $\{\Delta^{\mathcal{R}_\mathrm{i}}\}_\gamma$ that passed the reference frame check are now investigated if they are in fact within the temperature tolerance with Equation (9.11) and eliminated from the recall set if they fail. The remaining tie simplices are investigated to confirm whether each of their phases are within compositional tolerance with the new tie simplex with Equation (9.12) and eliminated from the recall set if even one phase fails the compositional tolerance. The newly created tie simplex $\Delta_\mathrm{t}^{\mathcal{R}_\mathrm{i}}$, together with the recalled set of tie simplices that are within tolerance $\{\Delta^{\mathcal{R}_\mathrm{i}}\}_\gamma$, are passed to the discretisation algorithm.

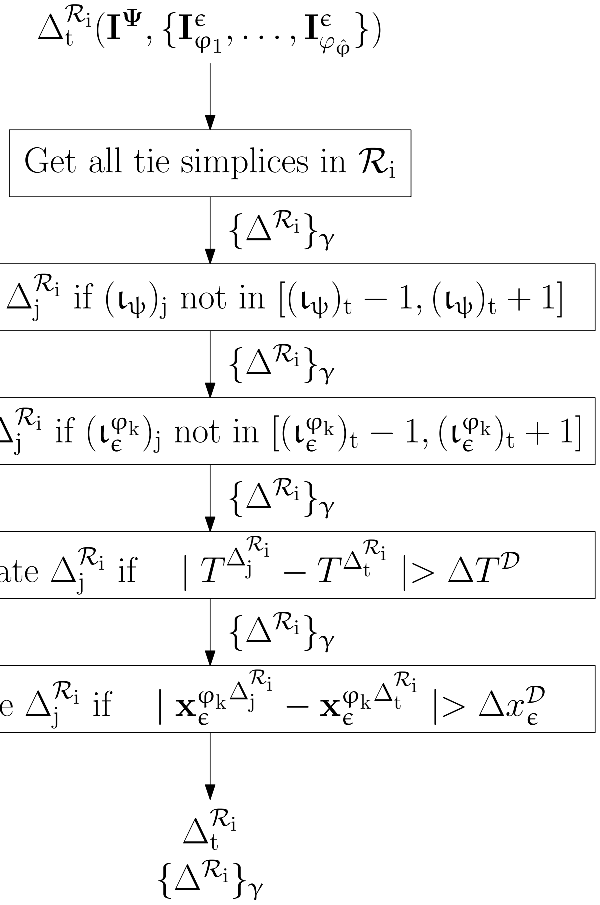

$\Delta_\mathrm{t}^{\mathcal{R}_\mathrm{i}}(\mathbf{I}^\Psi, \{\mathbf{I}^\epsilon_{\varphi_1}, \dots, \mathbf{I}^\epsilon_{\varphi_{\hat\varphi}}\})$
Get all tie simplices in $\mathcal{R}_\mathrm{i}$
$\{\Delta^{\mathcal{R}_\mathrm{i}}\}_\gamma$
Eliminate $\Delta_\mathrm{j}^{\mathcal{R}_\mathrm{i}}$ if $(\iota_\psi)_\mathrm{j}$ not in $[(\iota_\psi)_\mathrm{t} - 1, (\iota_\psi)_\mathrm{t} + 1]$
$\{\Delta^{\mathcal{R}_\mathrm{i}}\}_\gamma$
Eliminate $\Delta_\mathrm{j}^{\mathcal{R}_\mathrm{i}}$ if $(\iota_\epsilon^{\varphi_\mathrm{k}})_\mathrm{j}$ not in $[(\iota_\epsilon^{\varphi_\mathrm{k}})_\mathrm{t} - 1, (\iota_\epsilon^{\varphi_\mathrm{k}})_\mathrm{t} + 1]$
$\{\Delta^{\mathcal{R}_\mathrm{i}}\}_\gamma$
Eliminate $\Delta_\mathrm{j}^{\mathcal{R}_\mathrm{i}}$ if $\mid T^{\Delta_\mathrm{j}^{\mathcal{R}_\mathrm{i}}} - T^{\Delta_\mathrm{t}^{\mathcal{R}_\mathrm{i}}} \mid > \Delta T^{\mathcal{D}}$
$\{\Delta^{\mathcal{R}_\mathrm{i}}\}_\gamma$
Eliminate $\Delta_\mathrm{j}^{\mathcal{R}_\mathrm{i}}$ if $\mid \mathbf{x}_\epsilon^{\varphi_\mathrm{k}\Delta_\mathrm{j}^{\mathcal{R}_\mathrm{i}}} - \mathbf{x}_\epsilon^{\varphi_\mathrm{k}\Delta_\mathrm{t}^{\mathcal{R}_\mathrm{i}}} \mid > \Delta x_\epsilon^{\mathcal{D}}$
$\Delta_\mathrm{t}^{\mathcal{R}_\mathrm{i}}$
$\{\Delta^{\mathcal{R}_\mathrm{i}}\}_\gamma$

Figure 9.9: Tie simplex recall algorithm flow diagram.

9.5.5 Storing Phase Region Cells

Figure 9.10 shows the flow diagram of the phase region cell storage algorithm. When a phase region cell is created, a unique ID, r, is assigned to it $\delta_\mathrm{r}^{\mathcal{R}_\mathrm{i}}$. The temperature and composition of each phase of each tie simplex that the phase region cell is composed of, are used to create a convex hull of the phase region cell, $\Omega^{\delta_\mathrm{r}^{\mathcal{R}_\mathrm{i}}}$. The convex hull (Rockafellar 1997) creates the smallest convex set of temperature-phase composition coordinates wherein the phase region cell can be found.

The range of reference frame cell indices of each non-compositional potential that the phase region cell is found in $[(\iota_\psi)_{\min}, (\iota_\psi)_{\max}]$, as shown in Figure 9.6c, is determined. The same is done with the compositional potential indices $[(\iota_\epsilon)_{\min}, (\iota_\epsilon)_{\max}]$ for each system component.

The complete description of the phase region cell is; (1) the phase region cell ID r, (2) the convex hull of the phase region cell $\Omega^{\delta_{\mathrm{r}}^{\mathcal{R}_{\mathrm{i}}}}$, (3) the ranges of each non-compositional potential's indices $[(\iota_\psi)_{\min}, (\iota_\psi)_{\max}]$, and (4) the ranges of each system component's compositional potential indices $[(\iota_\epsilon)_{\min}, (\iota_\epsilon)_{\max}]$ that contain the phase region cell. The phase region cell, together with its complete description, is stored to the database.

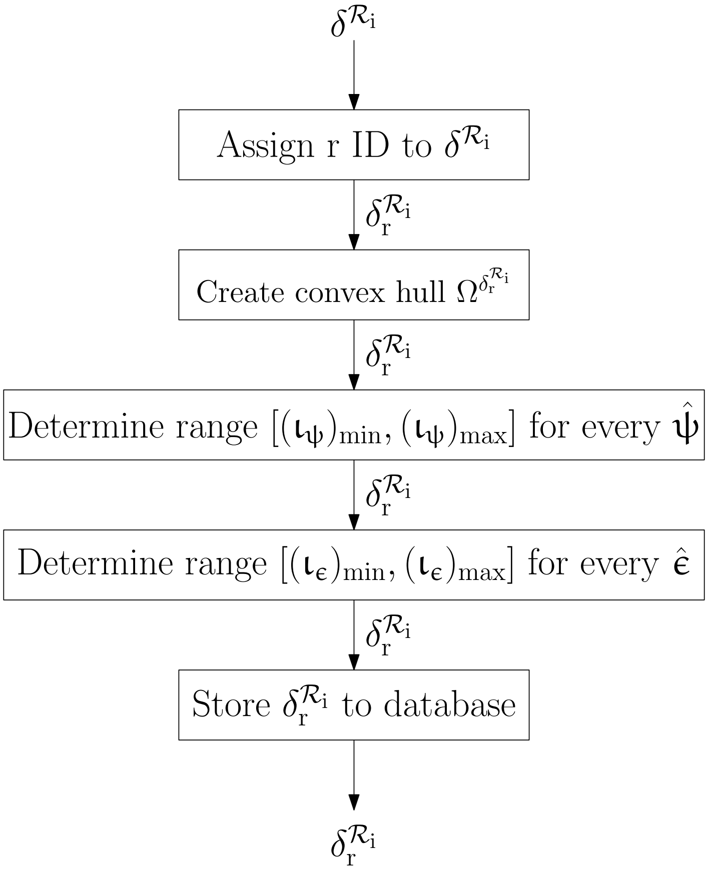

$\delta^{\mathcal{R}_{\mathrm{i}}}$
Assign r ID to $\delta^{\mathcal{R}_{\mathrm{i}}}$
$\delta_{\mathrm{r}}^{\mathcal{R}_{\mathrm{i}}}$
Create convex hull $\Omega^{\delta_{\mathrm{r}}^{\mathcal{R}_{\mathrm{i}}}}$
$\delta_{\mathrm{r}}^{\mathcal{R}_{\mathrm{i}}}$
Determine range $[(\iota_\psi)_{\min}, (\iota_\psi)_{\max}]$ for every $\hat{\psi}$
$\delta_{\mathrm{r}}^{\mathcal{R}_{\mathrm{i}}}$
Determine range $[(\iota_\epsilon)_{\min}, (\iota_\epsilon)_{\max}]$ for every $\hat{\epsilon}$
$\delta_{\mathrm{r}}^{\mathcal{R}_{\mathrm{i}}}$
Store $\delta_{\mathrm{r}}^{\mathcal{R}_{\mathrm{i}}}$ to database
$\delta_{\mathrm{r}}^{\mathcal{R}_{\mathrm{i}}}$

Figure 9.10: Phase region cell storage algorithm flow diagram.

9.5.6 Recalling Phase Region Cells

Figure 9.11 shows the flow diagram of the phase region cell recall algorithm. When an equilibrium calculation has to be performed at a specified system state $\mathbf{X}^\sigma$, the database is first queried to determine if a phase region cell is available to create an interpolated tie simplex with. The non-compositional $\mathbf{I}^\Psi$ and compositional potential index arrays $\mathbf{I}^\epsilon$ corresponding to the specified system state $\mathbf{X}^\sigma$ is first determined. This points to a single reference frame cell.

All phase region cells are stored with the range of each non-compositional potential's indices $[(\iota_\psi)_{\min}, (\iota_\psi)_{\max}]$ and compositional potential's indices $[(\iota_\epsilon)_{\min}, (\iota_\epsilon)_{\max}]$ it spans. All the phase region cells that contain every non-compositional potential index of the system ι_ψ^σ within its stored ranges are recalled and added to the recall set $\{\delta^{\mathcal{R}_{\mathrm{i}}}\}$. Each of the recalled phase region cells are then investigated to determine if it contains every compositional potential index of the system ι_ϵ^σ within its stored ranges; if it fails, it is removed from the recall set $\{\delta^{\mathcal{R}_{\mathrm{i}}}\}$. This is a computationally cheap method of reducing the number of phase region cells that have to be investigated to confirm whether one of them contains the specified system state.

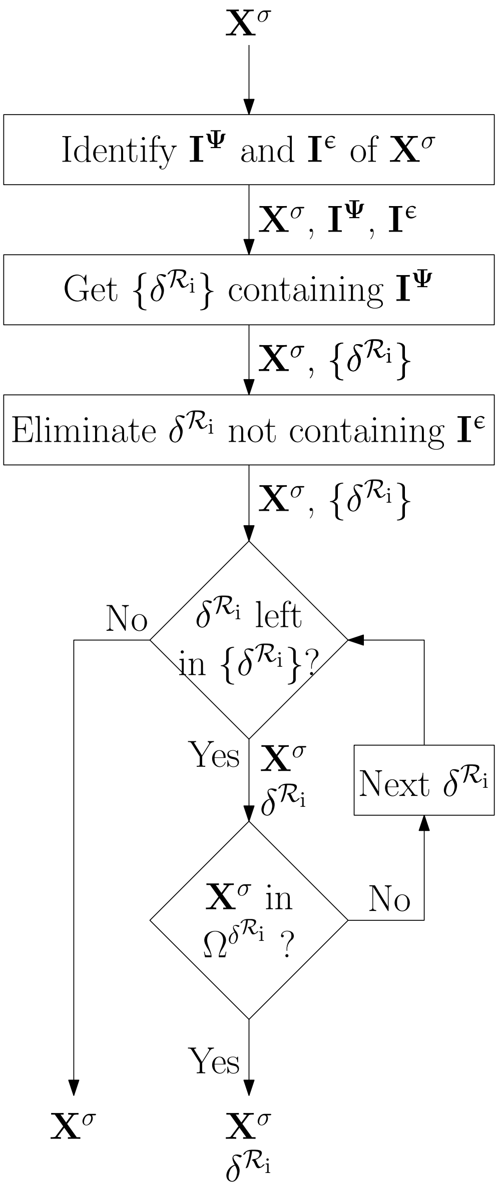

$\mathbf{X}^\sigma$
Identify $\mathbf{I}^\Psi$ and $\mathbf{I}^\epsilon$ of $\mathbf{X}^\sigma$
$\mathbf{X}^\sigma$, $\mathbf{I}^\Psi$, $\mathbf{I}^\epsilon$
Get $\{\delta^{\mathcal{R}_\mathrm{i}}\}$ containing $\mathbf{I}^\Psi$
$\mathbf{X}^\sigma$, $\{\delta^{\mathcal{R}_\mathrm{i}}\}$
Eliminate $\delta^{\mathcal{R}_\mathrm{i}}$ not containing $\mathbf{I}^\epsilon$
$\mathbf{X}^\sigma$, $\{\delta^{\mathcal{R}_\mathrm{i}}\}$
$\delta^{\mathcal{R}_\mathrm{i}}$ left in $\{\delta^{\mathcal{R}_\mathrm{i}}\}$?
No
Yes
$\mathbf{X}^\sigma$ $\delta^{\mathcal{R}_\mathrm{i}}$
Next $\delta^{\mathcal{R}_\mathrm{i}}$
$\mathbf{X}^\sigma$ in $\Omega^{\delta^{\mathcal{R}_\mathrm{i}}}$?
No
Yes
$\mathbf{X}^\sigma$
$\mathbf{X}^\sigma$ $\delta^{\mathcal{R}_\mathrm{i}}$

Figure 9.11: Phase region cell recall algorithm flow diagram.

The recall set of identified phase region cells $\{\delta^{\mathcal{R}_\mathrm{i}}\}$ are investigated one-by-one to determine whether the system state $\mathbf{X}^\sigma$ is contained within its convex hull $\Omega^{\delta^{\mathcal{R}_\mathrm{i}}}$. When a phase region cell is found to contain the specified system state, it is passed to the interpolation algorithm. Otherwise, when no phase region cell is found, the specified system state is passed to the equilibrium calculation software where a direct calculation is performed.

9.5.7 Conclusion

OPF boundaries $\mathcal{B}$ are discretised by simplices where the number of nodes per discretisation segment $\delta^{\mathcal{B}}$ are determined by the geometric dimensionality of the boundaries $\mathbb{D}_\mathrm{g}^{\mathcal{B}}$. A $\hat{\Delta}^{\mathcal{R}}$ number of tie simplices creates the discretisation nodes on each OPF boundary and are combined to create a phase region cell $\delta^{\mathcal{R}}$ if they are within a specified temperature and composition tolerance.

A two-step approach is used when creating an interpolated tie simplex $\tilde{\Delta}^{\mathcal{R}}$ from a phase region cell's tie simplices that perfectly contains the system state. Firstly, tie simplices are created through interpolation that are at the same non-compositional potentials as the system state $\tilde{\Delta}_\psi^{\mathcal{R}_\mathrm{i}}$. Secondly, these non-compositional potential tie simplices are used, together with barycentric coordinate weights, to determine the interpolated tie simplex $\tilde{\Delta}^{\mathcal{R}}$. The second step makes use of a system of non-linear equations that has to be solved iteratively, and is not desired when attempting to accelerationaccelerate equilibrium calculations. An explicit expression would be preferred, but no such generic explicit expression could yet be derived or found in literature. This is the most critical part of the accelerator algorithm that has to be addressed to improve on efficiency and performance.

The composition, physical and thermochemical properties of each phase present at equilibrium is known once the interpolated tie simplex is determined. The lever rule is then used to determine the phase fractions and physical and thermochemical properties of the system.

Even though the in-situ discretisation algorithm is unstructured, a uniform structured reference frame is used to assign compositional $\mathbf{I}^{\epsilon}$ and non-compositional potential indices $\mathbf{I}^{\Psi}$ to tie simplices and phase region cells. This resulted in more efficient recalls as its a computationally cheap method to filter the tie simplices and phase region cells, reducing the number of each to consider during their respective recall algorithms.

Chapter 10

Algorithm Functionality and Performance Tests

The accelerator algorithm was implemented, as discussed in Section 8.2. Performance criteria were established in Section 8.3 and used to assess the algorithm's functionality and performance in a number of two- and three-component systems, as outlined in Section 8.4.

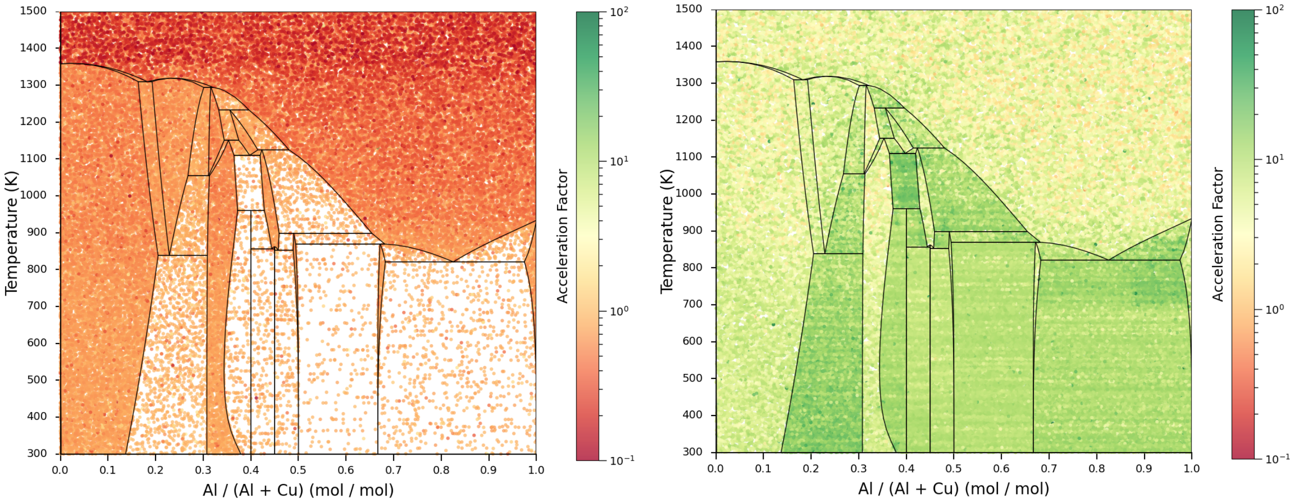
10.1 Two-component Systems

To investigate the functionality and performance of the accelerator algorithm in each two-component system, 200 000 system states $\mathbf{X}^\sigma$ were randomly generated within the entire composition range and within a temperature range from 300 K up to 100 K above the highest liquidus temperature in the system. A composition tolerance $\Delta x_\epsilon^{\mathcal{D}}$ of 0.01 mol mol^{-1} and temperature tolerance $\Delta T^{\mathcal{D}}$ of 10 K was chosen. The findings of the functionality and performance tests were very similar amongst all two-component systems and the Al−Cu system is used to illustrate these findings hereafter.

10.1.1 Acceleration Factor

Heat maps of acceleration factors, for calculated equilibrium states $\mathbf{Y}^\sigma$ obtained from the discretisation algorithm and interpolated equilibrium states $\tilde{\mathbf{Y}}^\sigma$ obtained from the interpolation algorithm, in reference to the equivalent direct equilibrium calculation, are shown in Figure 10.1. It can be seen that two-phase regions show mostly interpolations with large acceleration factors. Single-phase regions show a mix of interpolations with relatively large acceleration factors and discretisations with acceleration factors of 1 and less. This is due to the size of phase region cells in single-phase regions being much smaller than in two-phase regions, as illustrated in Figure 9.3a. Phase region cell size in single-phase regions are restricted to the specified temperature and composition tolerances, leading to relatively small cells. On the other hand, although phase region cells in two-phase regions are also restricted to the temperature tolerance, the phase region cells span the entire width of the phase region – from one phase region boundary to the other – and the composition tolerance only applies to compositions of the phases at the phase region boundaries. This means that fewer direct calculations have to be performed to discretise and cover large portions of two-phase regions, and a larger number of interpolated equilibrium calculations can be performed much sooner in the simulation.

Distribution plots of the acceleration factors for single- and two-phase regions are shown in Figure 10.2a. During the discretisation algorithm there is more computational overhead for the accelerator algorithm than the equivalent direct calculation; not only does a direct equilibrium calculation

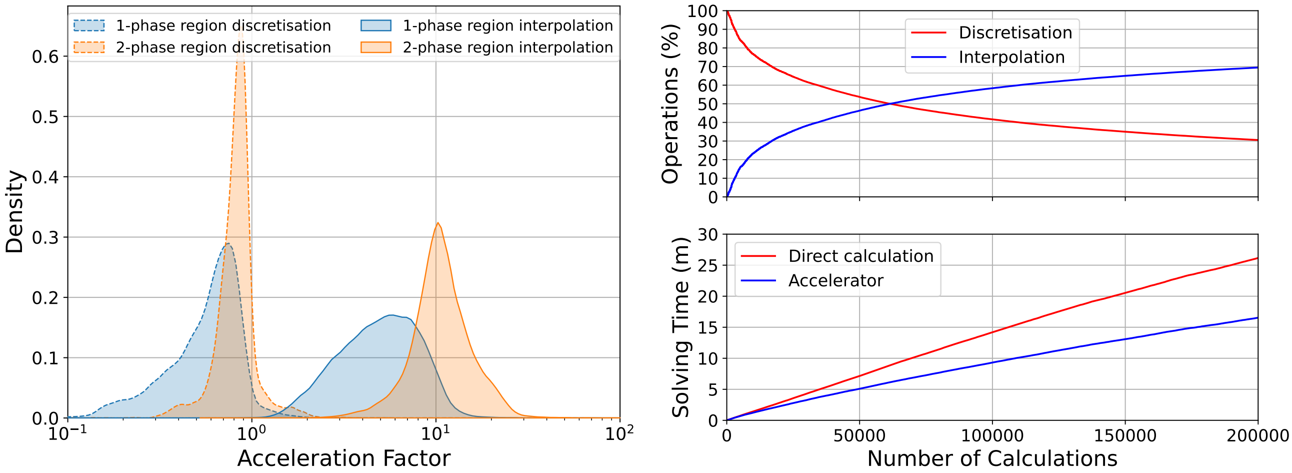


(a) Discretised states $\mathbf{Y}^\sigma$.

(b) Interpolated states $\tilde{\mathbf{Y}}^\sigma$.

Figure 10.1: Heat maps of acceleration factors for the Al−Cu system.

have to be performed, but the corresponding tie simplex created and stored, as well as the database interrogated for other tie simplices that can be combined to create a phase region cell. This results in acceleration factors smaller than one. Even though during the interpolation algorithm there are database searches as well, it can be seen that the acceleration factors in all phase regions are larger than one, with some interpolated equilibrium calculations showing acceleration factors in the order of 10 and more. Two-phase regions gain greater benefits compared to single-phase regions. Figure 10.2b shows that at the end of the test approximately 70 % of the equilibrium states returned by the accelerator algorithm were interpolated.



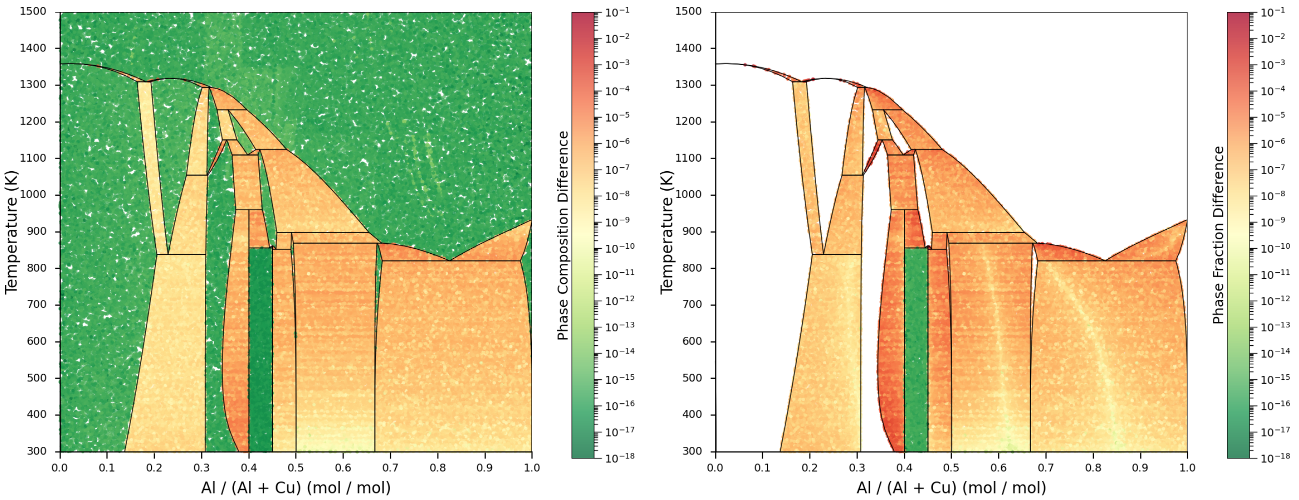
(a) Distribution of acceleration factors.

(b) Operation distribution and solving time.

Figure 10.2: Distribution of acceleration factors and operation distribution and cumulative solving time for the duration of the functionality and performance test for the Al−Cu system.

10.1.2 Phase Composition and Phase Fraction Difference

Heat maps of the difference in phase composition and phase fraction between the calculated and the interpolated equilibrium states are shown in Figure 10.3. It can be seen that the difference in phase composition for single-phase regions are in the order of 1×10^{-16}. Because linear interpolation is performed in these phase regions where temperature and composition vary linearly, the differences should be equal to zero. Due to machine precision when working with floating point values, an



(a) Phase composition difference.

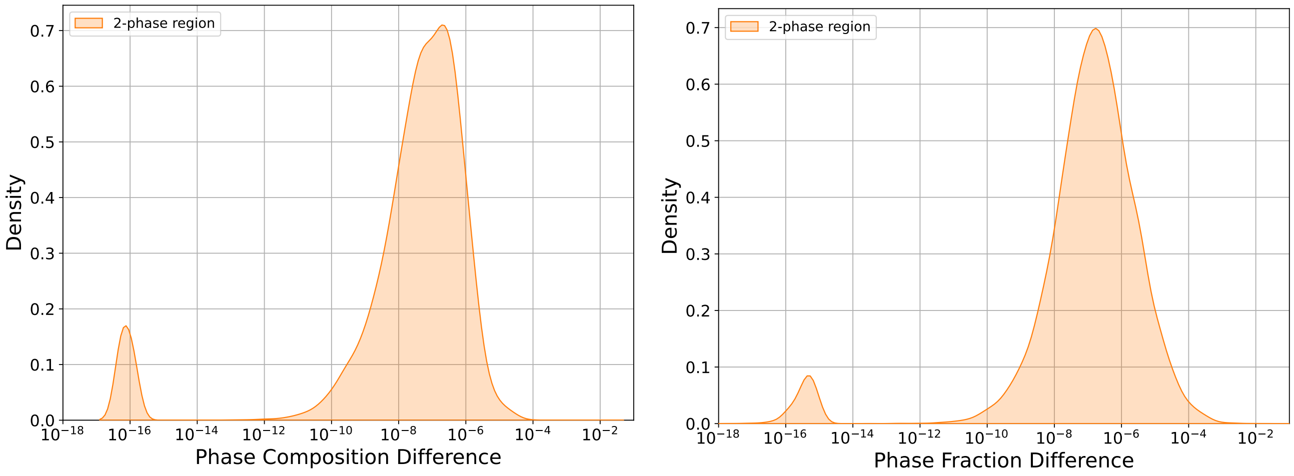
(b) Phase fraction difference.

Figure 10.3: Heat map of phase composition and phase fraction difference for the Al−Cu system.

observed value of 1×10^{-16} for the phase composition differences can be accepted to be exact. The same can be seen in two-phase regions where all present phases are pure substances. The OPF boundaries of pure substances are straight vertical lines with constant composition and no interpolation error is made. In two-phase regions where at least one solution phase is present, the difference in phase composition become more notable. OPF boundary of solution phases are often curved and varies non-linearly with composition and temperature. When performing linear interpolation on these curved OPF boundaries, errors are made, as shown in the heat maps.

Because phase fractions are calculated from phase compositions, the same trends observed with phase composition differences were observed with phase fraction differences. The one notable difference is in single-phase regions. Because only one phase is present, the phase fraction is always one and the difference between the calculated and interpolated equilibrium states is exactly zero. The zero value cannot be plotted on the log scale of the heat map.

Distribution plots of the phase composition and phase fraction differences of two-phase regions are shown in Figure 10.4. A narrow distribution can be seen in the phase composition difference at 1×10^{-16} associated with pure substances. There is also a wide distribution between 1×10^{-12} and



(a) Phase composition difference.

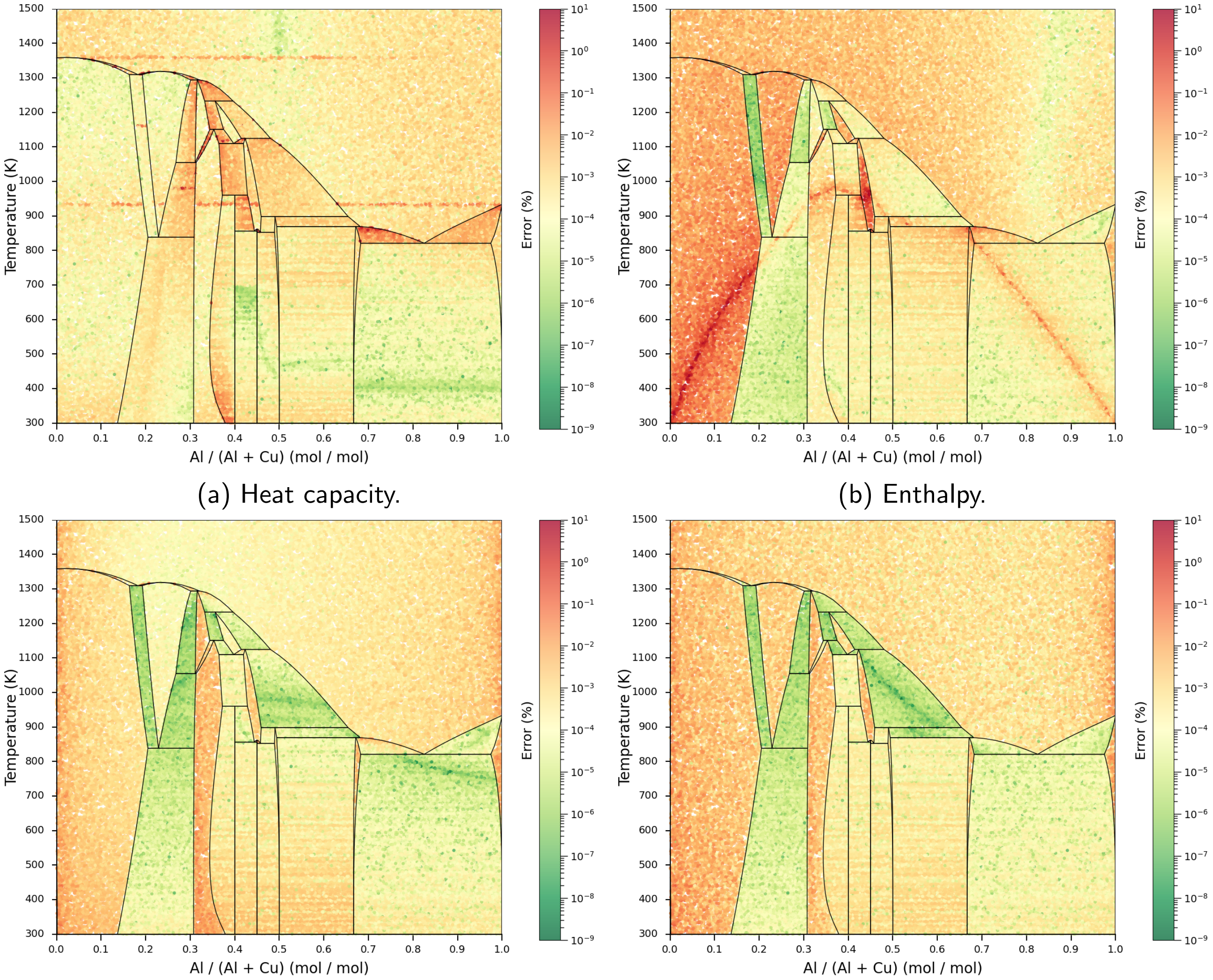
(b) Phase fraction difference.

Figure 10.4: Distribution of phase composition and phase fraction difference for the Al−Cu system.

1×10^{-4} associated with the solution phases. The more curved a OPF boundary, the larger the linear interpolation error made, and because there is a large variation in curvature of OPF boundaries in the system, the distribution of phase composition difference is wide. Because phase fractions are dependant on phase compositions, the same trends observed with differences in phase compositions were observed with differences in phase fractions.

10.1.3 Physical and Thermochemical Property Errors

Physical and thermochemical properties vary non-linearly with composition as well as temperature and linear interpolation errors are expected in all phase regions. Heat maps of the physical and thermochemical property errors are shown in Figure 10.5. System properties are determined by the phase fraction weighted sum of phase properties. Not only are interpolation errors made on the phase properties but also the phase fractions. The error made on the system properties can therefore become large as interpolation errors are made twice. Distribution plots of the errors made on physical and thermochemical properties are shown in Figure 10.6. A wide distribution of errors can be seen due to the non-linear property variations and interpolations. Errors for all the properties are in the order of 1 % or less.



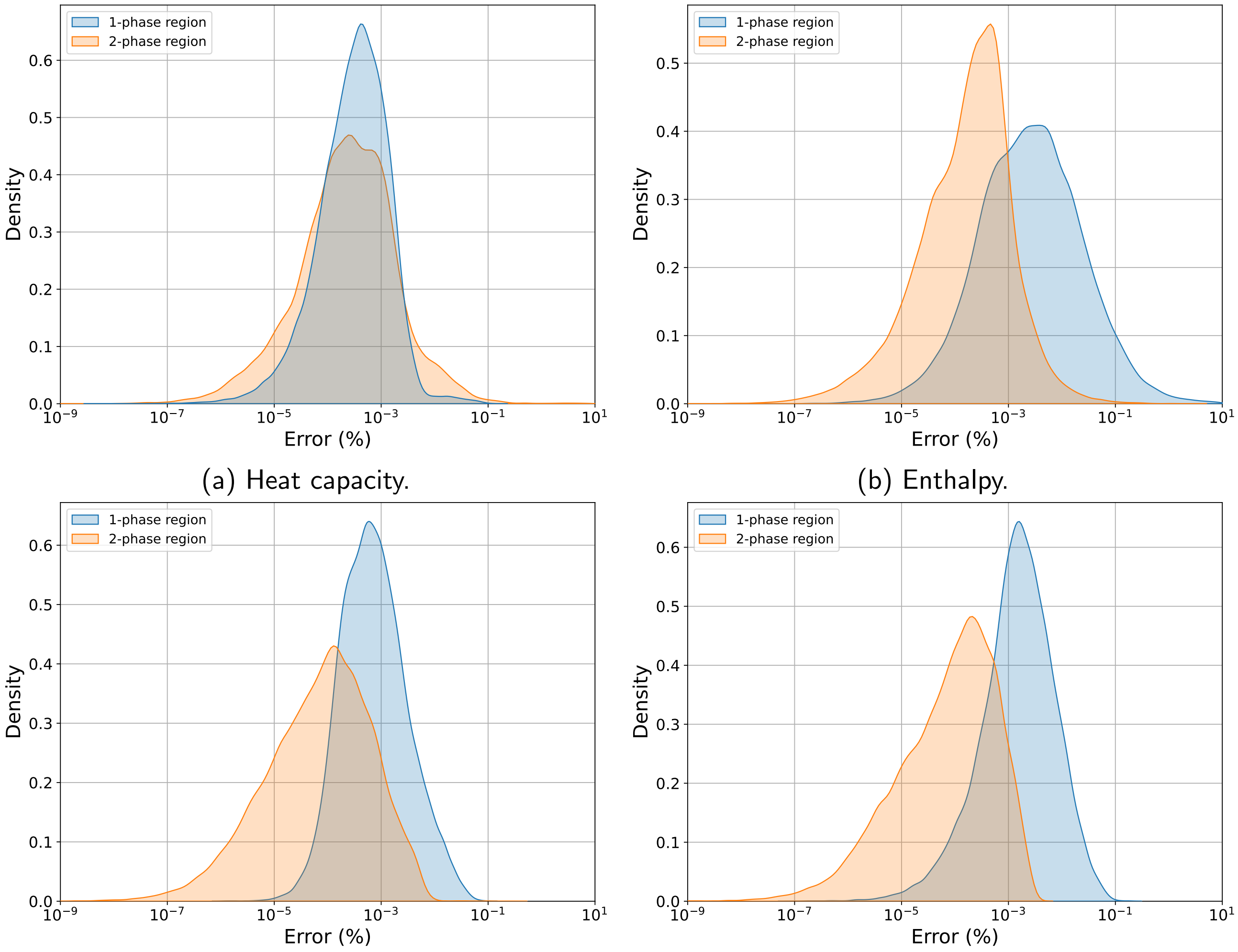
(a) Heat capacity.

(b) Enthalpy.

(c) Entropy.

(d) Gibbs energy.

Figure 10.5: Heat maps of physical and thermochemical property errors for the Al−Cu system.



(a) Heat capacity.

(b) Enthalpy.

(c) Entropy.

(d) Gibbs energy.

Figure 10.6: Distribution of physical and thermochemical property errors for the Al−Cu system.

10.1.4 Remaining Two-component Systems

Distribution plots of performance criterias for all the evaluated two-component systems can be found in Appendix B.1 with Table 10.1 summarising the average values for all interpolated equilibrium calculations based on the number of stable phases.

The average acceleration factor for all phase regions in the two-component systems are larger than one – acceleration was achieved across the set of system states that were evaluated. Some systems show more acceleration than others. This is due to the direct calculation computation time of some systems being longer than others, while the interpolation times remain fairly constant. It can be seen that two-phase regions' acceleration factors are higher because when non-compositional potential tie simplices are determined, only one can be created and it automatically becomes the interpolated tie simplex, and no additional interpolation steps are required. In single-phase regions, two non-compositional potential tie simplices are created and from there an additional interpolation step is required to determine the interpolated tie simplex.

In two-phase regions, the average differences in phase compositions and phase fractions are in the order of 1×10^{-5} or less. Very small interpolation errors are therefore made. The average errors made on physical and thermochemical properties are in the order of 1×10^{-2} % or less.

The functionality and performance of the accelerator algorithm is satisfactory in two-component systems. Phase composition and phase fraction differences are acceptable for the chosen temperature

Table 10.1: Summary of algorithm performance in two-component systems.

System	$\overline{\mathrm{AF}}$	$\overline{\mathrm{d}x_{\epsilon}^{\varphi_i}}$	$\overline{\mathrm{d}x_{\varphi_i}^{\sigma}}$	$\overline{E\rho^{\sigma}}$ (%)		$\overline{E\tau^{\sigma}}$ (%)	
				C_p	H	S	G
				Single-phase regions			
Al−Cu	5.5			8.9e-04	5.8e-02	2.1e-03	4.0e-03
Al−Mn	7.2			6.5e-04	4.9e-02	1.1e-03	2.8e-03
Al−Zn	5.1			1.2e-03	8.0e-04	2.0e-03	1.8e-03
Fe−C	2.6			1.5e-03	1.6e-03	1.7e-03	3.5e-03
Fe−Cr	4.1			8.6e-03	6.5e-04	1.3e-03	1.4e-03
Fe−Si	11.1			2.3e-02	9.7e-02	1.1e-03	4.2e-03
Al_2O_3−CaO	1.8			7.4e-02	5.7e-04	5.0e-04	2.2e-04
Al_2O_3−MgO	5.0			9.9e-02	3.7e-04	6.4e-04	2.4e-04
Al_2O_3−SiO_2	2.6			1.9e-01	2.0e-04	6.6e-04	1.5e-04
CaO−SiO_2	3.1			3.0e-02	8.1e-04	1.2e-03	2.7e-04
				Two-phase regions			
Al−Cu	11.5	5.0e-07	6.8e-06	2.9e-02	1.5e-03	6.6e-04	5.3e-04
Al−Mn	13.8	7.8e-07	1.6e-05	3.4e-02	5.5e-03	6.6e-04	4.8e-04
Al−Zn	9.9	1.2e-05	1.2e-05	5.9e-02	4.7e-03	6.4e-04	1.2e-03
Fe−C	4.8	7.6e-08	2.8e-07	1.7e-03	2.3e-03	2.6e-04	8.2e-04
Fe−Cr	10.5	7.2e-06	4.6e-06	3.8e-02	3.7e-03	4.3e-04	1.3e-03
Fe−Si	19.5	1.7e-07	9.2e-07	1.5e-02	1.2e-03	3.5e-04	2.0e-04
Al_2O_3−CaO	3.2	4.0e-08	3.0e-07	2.8e-03	5.5e-06	1.9e-04	2.3e-05
Al_2O_3−MgO	8.5	2.6e-07	7.2e-07	4.3e-02	1.4e-05	7.8e-05	8.3e-06
Al_2O_3−SiO_2	6.2	8.7e-07	1.6e-06	6.2e-03	1.5e-05	8.6e-05	9.6e-06
CaO−SiO_2	5.0	9.3e-08	3.4e-07	4.6e-03	6.4e-06	1.6e-04	2.2e-05

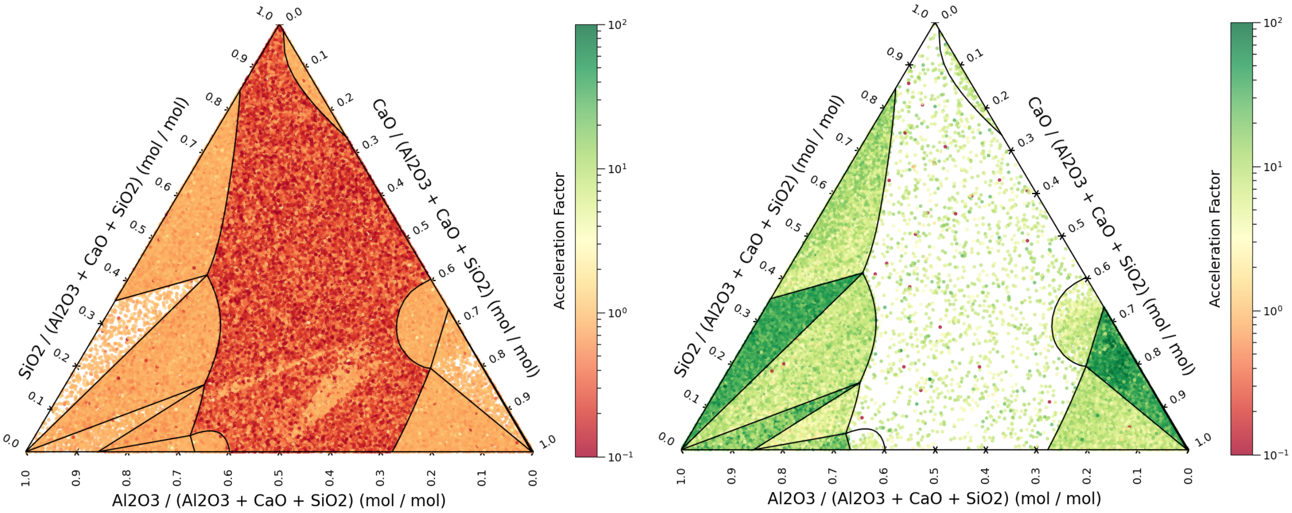
and composition tolerances, resulting in acceptable magnitudes of errors made on physical and thermochemical properties. Although the acceleration factor does not seem massive, it should be remembered that direct calculation computation times in two-component systems are in the order of 0.01 sec to 0.001 sec, which is already fast. In systems with more components and longer direct calculation computation times, which is what this accelerator is being developed for, it is expected to observe much larger acceleration factors.

10.2 Three-component Systems

To investigate the functionality and performance of the accelerator algorithm in each three-component system, 500 000 system states $\mathbf{X}^{\sigma}$ were randomly generated within the entire composition range of each system component and within a temperature range from 1300 K up to 100 K above the highest liquidus temperature in the system. The temperature range was reduced in comparison to the two-component system tests to maintain a comparable randomised system state density. A composition tolerance $\Delta x_{\epsilon}^{\mathcal{D}}$ of 0.01 mol mol^{-1} and temperature tolerance $\Delta T^{\mathcal{D}}$ of 10 K was chosen. The findings of the functionality and performance tests were very similar amongst all three-component systems and the Al_2O_3−CaO−SiO_2 system is used to illustrate these findings hereafter. It is difficult to interrogate heat maps in three-component systems when they are drawn on the entire 3D phase diagram. Instead, heat maps were drawn on isothermal sections and interrogated. The heat maps illustrated hereafter are shown on the 2000 K isotherm for all system states $\mathbf{X}^{\sigma}$ within ±5 K.

10.2.1 Acceleration Factor

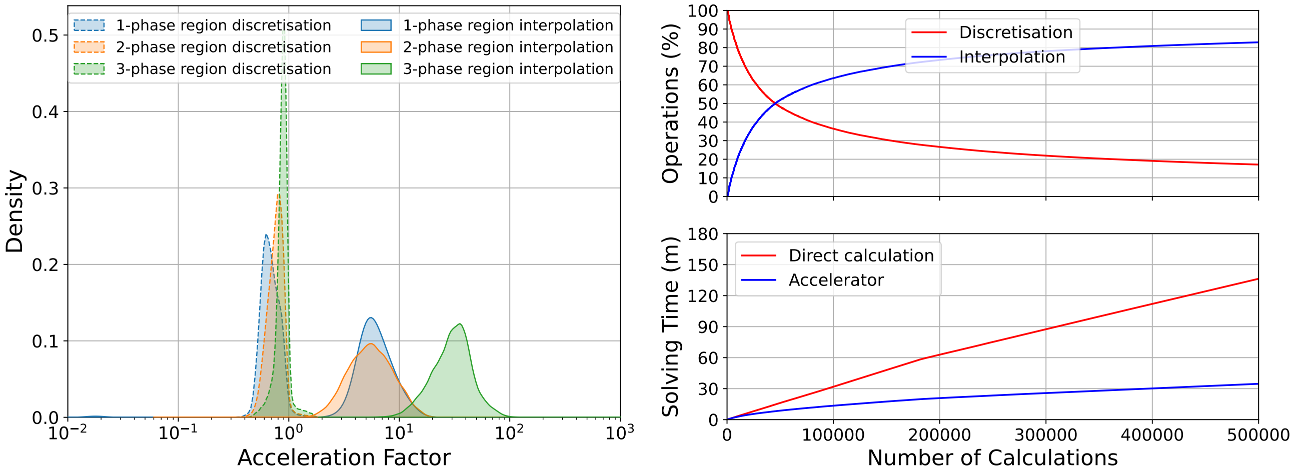
Heat maps of acceleration factors, for calculated equilibrium states $\mathbf{Y}^{\sigma}$ obtained from the discretisation algorithm and interpolated equilibrium states $\tilde{\mathbf{Y}}^{\sigma}$ obtained from the interpolation algorithm, are shown in Figure 10.7. It can be seen that three-phase regions have a small number of discretisations with mostly interpolations, while two-phase regions show more discretisations and less interpolations, and single-phase regions show the most discretisations and the fewest interpolations. This is, as in two-component systems, due to the size of phase region cells being the smallest in single-phase regions and increasing in size as the number of stable phases increase, as illustrated in Figure 9.3b for three-component systems.



(a) Discretised states $\mathbf{Y}^{\sigma}$.

(b) Interpolated states $\tilde{\mathbf{Y}}^{\sigma}$.

Figure 10.7: Heat maps of acceleration factors for the $Al_2O_3-CaO-SiO_2$ system on the 2000 K isotherm for system states $\mathbf{X}^{\sigma}$ within ± 5 K.



(a) Distribution of acceleration factors.

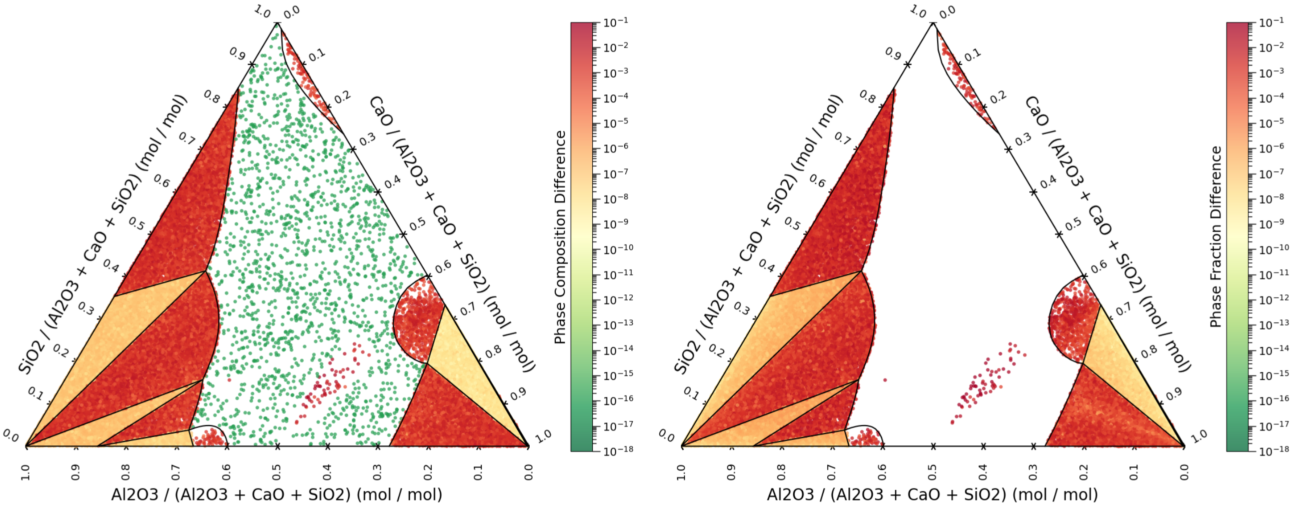
(b) Operation distribution and solving time.

Figure 10.8: Distribution of acceleration factors and operation distribution and cumulative solving time for the duration of the functionality and performance test for the $Al_2O_3-CaO-SiO_2$ system.

Distribution plots of the acceleration factors are shown in Figure 10.8a. As with two-component systems, the acceleration factors for discretisations are less than one and acceleration factors for interpolations are larger than one, with some interpolations showing acceleration factors of 10 to 100. Even though the additional interpolation step in single- and two-phase regions require iteratively solving a system of non-linear equations, interpolation computation times are shorter than that of the direct calculations and acceleration is still achieved. Figure 10.8b shows that at the end of the test approximately 83 % of the equilibrium states returned by the accelerator algorithm were interpolated.

10.2.2 Phase Composition and Phase Fraction Difference

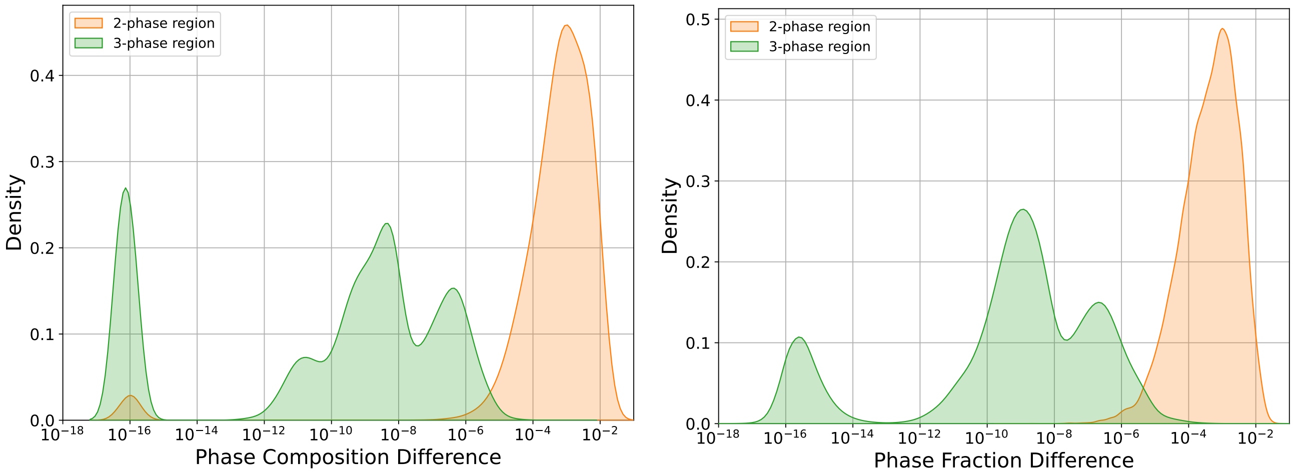
Heat maps of the difference in phase composition and phase fraction between the calculated and interpolated equilibrium states are shown in Figure 10.9. It can be seen that all the accessed three-phase regions have a solution phase (slag) present; hence the average phase composition difference at the system states not being as low as 1×10^{-16} as when only pure substances are present. The phase composition difference in two-phase regions are relatively large, but as in the two-component system tests, its is attributed to the curvature of the solution phase's OPF boundary and interpolation errors being made on the phase composition. Because phase fractions are dependant on phase compositions, the same trends observed with differences in phase compositions were observed with differences in phase fractions.



(a) Phase composition difference.

(b) Phase fraction difference.

Figure 10.9: Heat map of phase composition and phase fraction difference for the $Al_2O_3-CaO-SiO_2$ system on the 2000 K isotherm for interpolated states $\tilde{\mathbf{Y}}^{\sigma}$ within ± 5 K.



(a) Phase composition difference.

(b) Phase fraction difference.

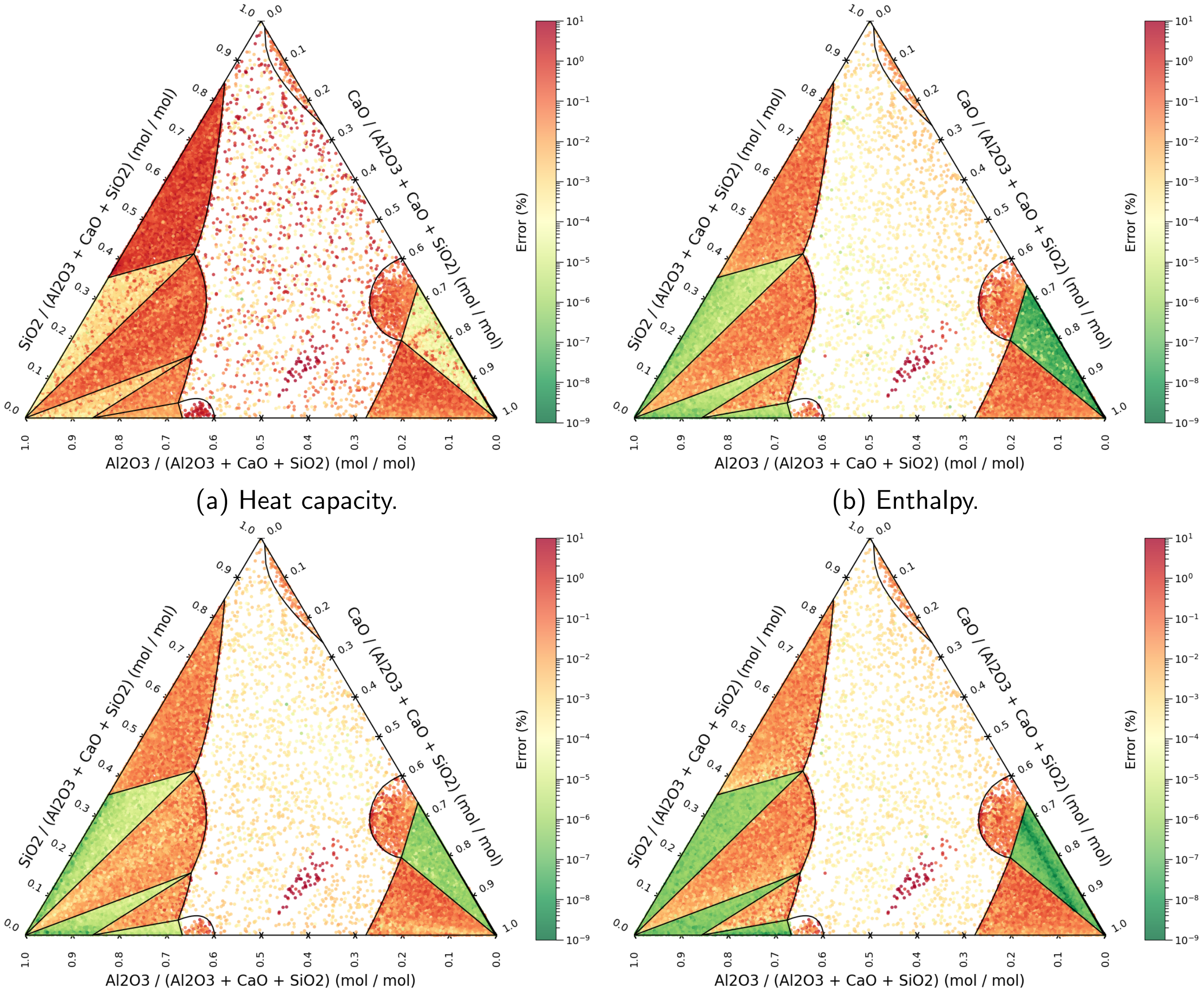
Figure 10.10: Distribution of phase composition and phase fraction difference for the $Al_2O_3-CaO-SiO_2$ system.

A patch of phase composition and phase fraction differences are seen in the single-phase region that seem out of place. These are system states that are found in a two-phase region below the 2000 K isotherm but is within the ±5 K range.

Distribution plots of the phase composition and phase fraction differences of two- and three-phase regions are shown in Figure 10.10. A narrow distribution can be seen in the phase composition difference at 1×10^{-16} associated with pure substances. There is also a wide distribution between 1×10^{-12} and 1×10^{-2} associated with the solution phases. Phase fractions are dependant on phase compositions, and therefore similar trends observed with differences in phase compositions were observed with differences in phase fractions.

10.2.3 Physical and Thermochemical Property Errors

Physical and thermochemical properties vary non-linearly with composition as well as temperature and linear interpolation errors are expected in all phase regions. Heat maps of the physical and thermochemical property errors are shown in Figure 10.11. Distribution plots of the errors made on physical and thermochemical properties are shown in Figure 10.12. Errors made on all the properties are in the order of 1 % or less with a small number of errors from 1 % to 10 % in two-phase regions.



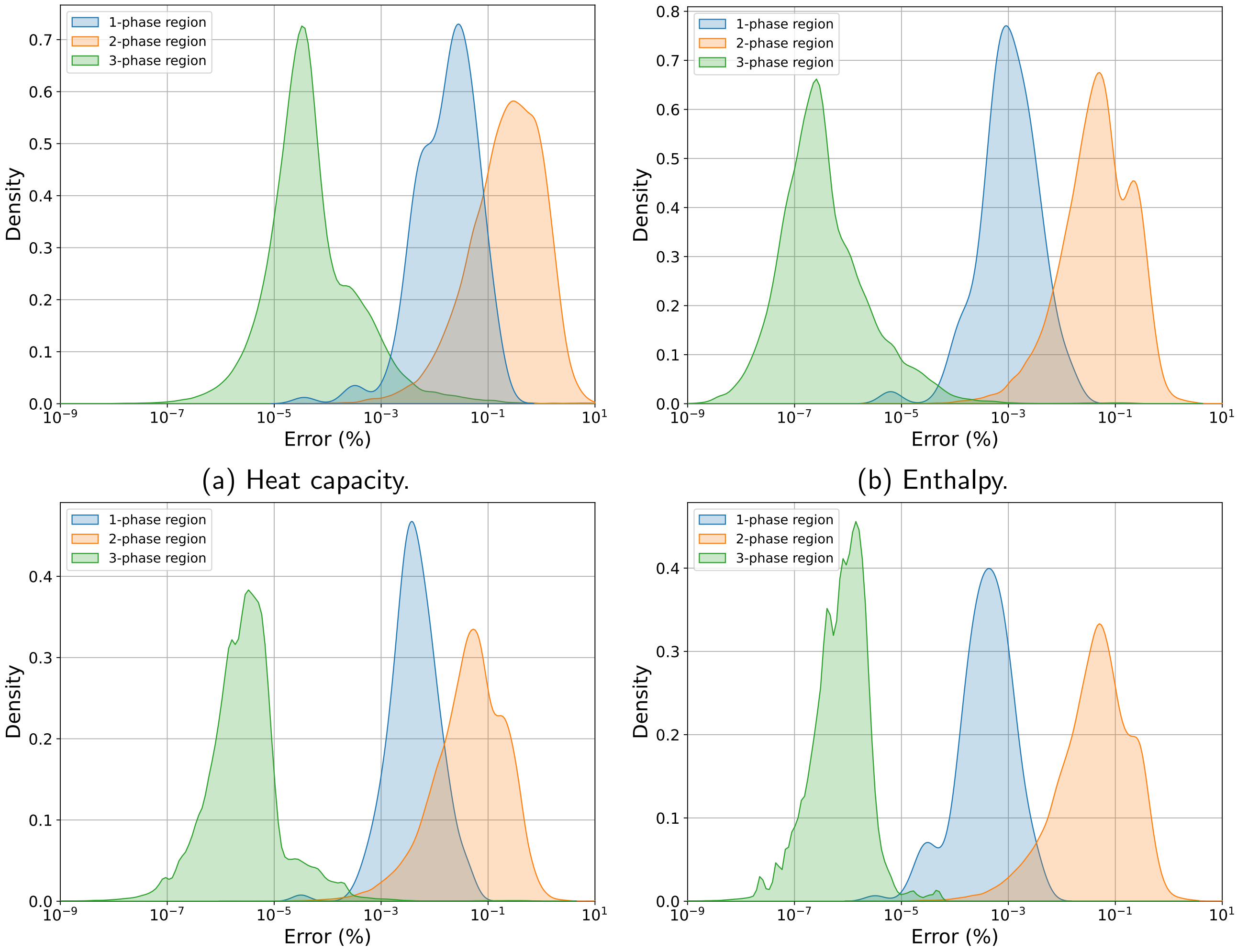
(a) Heat capacity.

(b) Enthalpy.

(c) Entropy.

(d) Gibbs energy.

Figure 10.11: Heat map of physical and thermochemical property errors for the $Al_2O_3-CaO-SiO_2$ system on the 2000 K isotherm for interpolated states $\tilde{\mathbf{Y}}^{\sigma}$ within ± 5 K.



(a) Heat capacity.

(b) Enthalpy.

(c) Entropy.

(d) Gibbs energy.

Figure 10.12: Distribution of physical and thermochemical property errors for the $Al_2O_3-CaO-SiO_2$ system.

10.2.4 Remaining Three-component Systems

Distribution plots of performance criterias for all the evaluated three-component systems can be found in Appendix B.2 with Table 10.2 summarising the average values for all interpolated equilibrium calculations based on the number of stable phases.

The average acceleration factor for all phase regions in the three-component systems are larger than one – acceleration was achieved across the set of system states that were evaluated. It can be seen that three-phase regions' acceleration factors are the highest because when non-compositional potential tie simplices are determined, only one can be created and it automatically becomes the interpolated tie simplex – no additional interpolation steps are required. This is always the case in phase regions where $\hat{\varepsilon} - \hat{\zeta}^{\sigma} = \hat{\varphi}$; two-phase regions in two-component systems, three-phase regions in three-component systems, etc. In single- and two-phase regions, multiple non-compositional potential tie simplices are created and from there an additional interpolation step is required to determine the interpolated tie simplex. Even though this additional step require iteratively solving a system of non-linear equations, the interpolated equilibrium state computation times were shorter than that of the calculated equilibrium state and acceleration was still achieved. As in two-component systems, some three-component systems show more acceleration than others seeing that the direct calculation computation time of some systems being longer than others.

Table 10.2: Summary of algorithm performance in three-component systems.

System	$\overline{\mathrm{AF}}$	$\overline{\mathrm{d}x_{\varepsilon}^{\varphi_i}}$	$\overline{\mathrm{d}x_{\varphi_i}^{\sigma}}$	$\overline{E\rho^{\sigma}}$ (%)	$\overline{E\tau^{\sigma}}$ (%)		
				C_p	H	S	G
Single-phase regions							
$Al_2O_3-CaO-MgO$	4.8			6.2e-03	1.4e-04	2.4e-03	7.3e-04
$Al_2O_3-CaO-SiO_2$	6.4			1.8e-02	2.0e-03	6.3e-03	6.4e-04
$Al_2O_3-Fe_2O_3-MgO$	6.3			2.1e-03	1.7e-04	2.2e-03	4.9e-04
$Al_2O_3-Fe_2O_3-SiO_2$	13.6			1.5e-04	4.7e-04	4.1e-03	1.3e-03
Two-phase regions							
$Al_2O_3-CaO-MgO$	3.9	5.4e-03	2.7e-03	2.9e-02	1.1e-01	2.7e-02	2.8e-02
$Al_2O_3-CaO-SiO_2$	6.3	2.0e-03	1.5e-03	2.8e-02	8.3e-02	2.4e-02	2.2e-02
$Al_2O_3-Fe_2O_3-MgO$	9.7	2.0e-03	1.2e-03	2.7e-02	1.1e-01	1.8e-02	2.1e-02
$Al_2O_3-Fe_2O_3-SiO_2$	63.6	1.9e-03	1.2e-03	1.8e-02	8.7e-02	1.6e-02	1.9e-02
Three-phase regions							
$Al_2O_3-CaO-MgO$	20.2	5.0e-02	1.2e-02	2.0e-04	9.4e-06	1.7e-05	5.5e-06
$Al_2O_3-CaO-SiO_2$	34.0	1.7e-07	1.1e-06	4.6e-04	1.6e-04	3.0e-05	1.5e-06
$Al_2O_3-Fe_2O_3-MgO$	89.6	3.2e-01	1.1e-01	1.7e-03	1.4e-05	3.0e-05	6.8e-06
$Al_2O_3-Fe_2O_3-SiO_2$	86.4	4.4e-02	1.4e-02	2.8e-05	5.6e-07	1.8e-05	5.8e-06

In two-phase regions, phase composition differences are in the order of 1×10^{-3} mol mol^{-1} or less and phase fraction differences are in the order of 1×10^{-3} or less. In the three-phase regions, phase composition differences are in the order of 1×10^{-2} mol mol^{-1} and phase fraction differences are in the order of 1×10^{-2} or less. The average error made on physical and thermochemical properties are in the order of 1×10^{-2} % or less.

The functionality and performance of the accelerator algorithm is satisfactory in three-component systems. The phase composition and phase fraction differences are acceptable for the chosen temperature and composition tolerances, resulting in acceptable magnitudes of errors made on physical and thermochemical properties. A notable acceleration in equilibrium calculation computation time is already observed in three-component systems and even larger acceleration factors are expected in systems with more components.

Chapter 11

Accelerator Applications

Although incorporating the accelerator algorithm into a model is not part of this project's scope, the performance of the accelerator algorithm was tested in two systems that are related to industrial processes where the inclusion of a large number of equilibrium calculations into models could provide valuable insight. These are a simplified four-component ilmenite smelting system, C−Fe−O−Ti, and a simplified five-component iron- and steelmaking system, C−Ca−Fe−O−Si. This provides insight into how the accelerator would perform if it was incorporated into models of these processes.

Exact temperature and composition ranges that will be encountered when the accelerator is included into the models are not known, and may also not be constant during an entire set of simulations, but wide expected ranges were estimated. The purpose of these tests are not to replicate or emulate the inclusion of the accelerator into these models, but rather test the accelerator algorithm's performance in multiple phase regions within temperature and compositional ranges that could be expected in the processes.

11.1 Simplified Ilmenite Smelting System

11.1.1 Background

The ilmenite smelting process is a step in the upgrading of TiO_2-containing minerals to TiO_2 pigment and Ti metal (Zietsman 2004). Minerals such as rutile (nominally TiO_2), ilmenite (nominally $FeTiO_3$), leucoxene (naturally upgraded ilmenite to a higher TiO_2 content), and zircon (nominally $ZrSiO_4$) are often found together in sand-type deposits, collectively referred to as heavy minerals. Ilmenite is also found in hard rock deposits with much lower concentrations of other heavy minerals.

The purpose of the smelting process is to upgrade the ilmenite, whether separated from the mineral sands or as ore from the hard rock deposits, to a slag (Zietsman 2004) with high TiO_2 concentration. The tapped slag is either granulated, or cast into large blocks and then crushed and milled. The high-TiO_2 slag in particulate form is used as a feedstock to produce pigment or titanium metal through additional processes.

The alloy product from the smelting furnace is typically iron with 2 % C (Zietsman 2004). Once the alloy is tapped from the smelting furnace, it is refined to increase the C content and decrease the S content. The high-C alloy is then cast into pigs and distributed to other industries.

A key feature of ilmenite smelting is that the corrosive high-TiO_2 slag cannot be contained through direct contact with any available refractory materials. A layer of slag must be frozen onto the furnace sidewall for safe slag bath containment; this solid layer is known as a slag freeze lining. Growth and melting of the freeze lining involves an intricate thermochemical process influenced by fluid flow and heat transfer in the slag bath, heat transfer through the sidewall, and changes in liquid slag composition.

Comprehensive multiphysics models of the ilmenite smelting process, including its thermochemical interactions between slag bath and freeze lining, are needed to improve the understanding of this challenging high-temperature process. This will allow the design of more stable, robust, and efficient smelting furnaces, and the development of improved operating strategies and control systems, all of which will yield both economic and sustainability benefits. Such models are currently infeasible due to the computational expense of direct equilibrium calculations.

11.1.2 Test Conditions

Because of the high operating temperatures of the alloy and slag baths, the accelerator algorithm was tested from 1500 K to 2100 K. Four regions of interest in the ilmenite smelting system were investigated with ranges of average compositions identified from literature; (1) the alloy bath (Zietsman 2004; Gous 2006), (2) the slag bath (Zietsman 2004; Gous 2006; Murty, Upadhyay, and Asokan 2007; Pistorius 2007), (3) the reduction zone where ilmenite feedstock (Zietsman 2004; Murty, Upadhyay, and Asokan 2007) and reducing agents are introduced into the furnace in ratios identified from (Pistorius 2007), and (4) the interface between the slag and alloy bath as a 1:1 mass-based ratio between the alloy and slag bath. The mass fraction and corresponding system component mole fraction ranges for the four regions are shown in Table 11.1. A total of 1 000 000 system states $\mathbf{X}^{\sigma}$

Table 11.1: Mass fraction ranges of phase constituents and corresponding system component mole fraction ranges for regions of interest in the simplified ilmenite smelting system tests.

Phase Constituent	Mass fraction ($\mathrm{g\,g^{-1}}$) Min.	Mass fraction ($\mathrm{g\,g^{-1}}$) Max.	System Component	Mole fraction ($\mathrm{mol\,mol^{-1}}$) Min.	Mole fraction ($\mathrm{mol\,mol^{-1}}$) Max.
Alloy bath					
C	0.00	0.04	C	0.00	0.16
Fe	0.94	1.00	Fe	0.80	1.00
O	0.00	0.01	O	0.00	0.03
Ti	0.00	0.01	Ti	0.00	0.01
Slag bath					
FeO	0.15	0.20	C	0.00	0.00
TiO_2	0.55	0.60	Fe	0.06	0.08
Ti_2O_3	0.20	0.30	O	0.62	0.64
			Ti	0.29	0.32
Reduction zone					
C	0.08	0.12	C	0.18	0.26
$FeTiO_2$	0.65	0.90	Fe	0.12	0.19
Fe_2O_3	0.00	0.14	O	0.44	0.50
TiO_2	0.00	0.15	Ti	0.13	0.18
Alloy and slag bath interface					
C	0.00	0.02	C	0.00	0.08
Fe	0.46	0.50	Fe	0.41	0.55
FeO	0.07	0.10	O	0.30	0.34
TiO_2	0.27	0.31	Ti	0.14	0.18
Ti_2O_3	0.09	0.15			

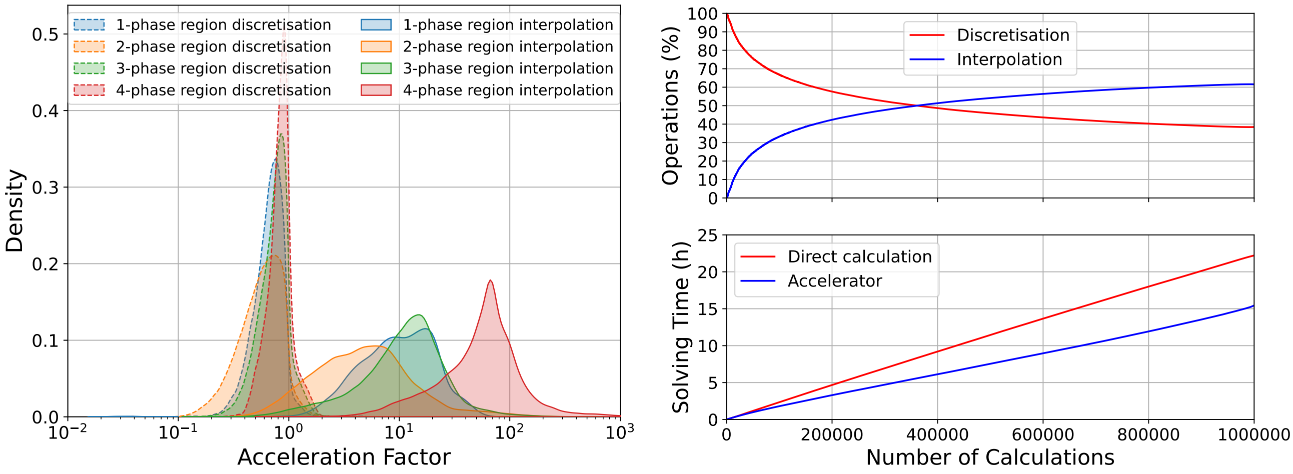
were randomly generated in these identified temperature and composition ranges. A composition tolerance $\Delta x_{\epsilon}^{\mathcal{D}}$ of 0.01 mol mol^{-1} and temperature tolerance $\Delta T^{\mathcal{D}}$ of 10 K was chosen.

11.1.3 Results

The distribution plots of acceleration factors for all the phase regions are shown in Figure 11.1a. As in the functionality and performance tests in two- and three-component systems, it can be seen that the acceleration factor was smaller than one during the discretisation algorithm for the majority of the cases. In a small number of cases, the acceleration factor during the discretisation algorithm was found to be larger than one. This was simply attributed to the control direct calculation taking notably more iterations to determine the equilibrium state in comparison to the equilibrium calculation performed by the discretisation algorithm.

Even though database searches and frequent iterative calculations are performed during the interpolation algorithm, the majority of acceleration factors for all phase regions are larger than one. Acceleration factors in the order of 100 and as high as 1000 are achieved in four-phase regions. In a small number of cases, the acceleration factor for an interpolation was smaller than one, which is attributed to a slow searching time for a suitable phase region cell from the database.

Figure 11.1b shows that at the end of the test approximately 62 % of the equilibrium states returned by the accelerator algorithm were interpolated. This is less than the two- and three-component system tests because the 1 000 000 system states $\mathbf{X}^{\sigma}$ were divided between four regions of interest.



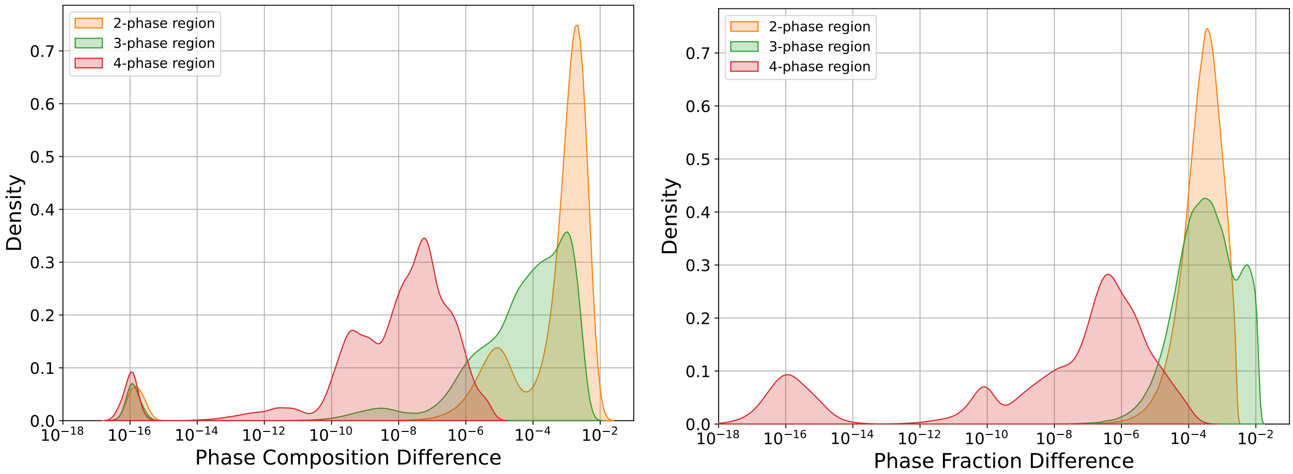
(a) Distribution of acceleration factors.

(b) Operation distribution and solving time.

Figure 11.1: Distribution of acceleration factors and operation distribution and cumulative solving time for the duration of the functionality and performance test for the ilmenite smelting system.

The distribution of phase composition and phase fraction differences are shown in Figure 11.2. It can be seen that the difference in phase composition of two- and three-phase regions are in the order of 1×10^{-2} mol mol^{-1} and less whereas with four-phase regions the difference is in the order of 1×10^{-5} mol mol^{-1} and less. Differences in the order of 1×10^{-16} mol mol^{-1} are associated with errors made on the compositions of pure substances. The same trend was observed with the phase fraction differences seeing that it is dependent on the interpolated phase compositions.

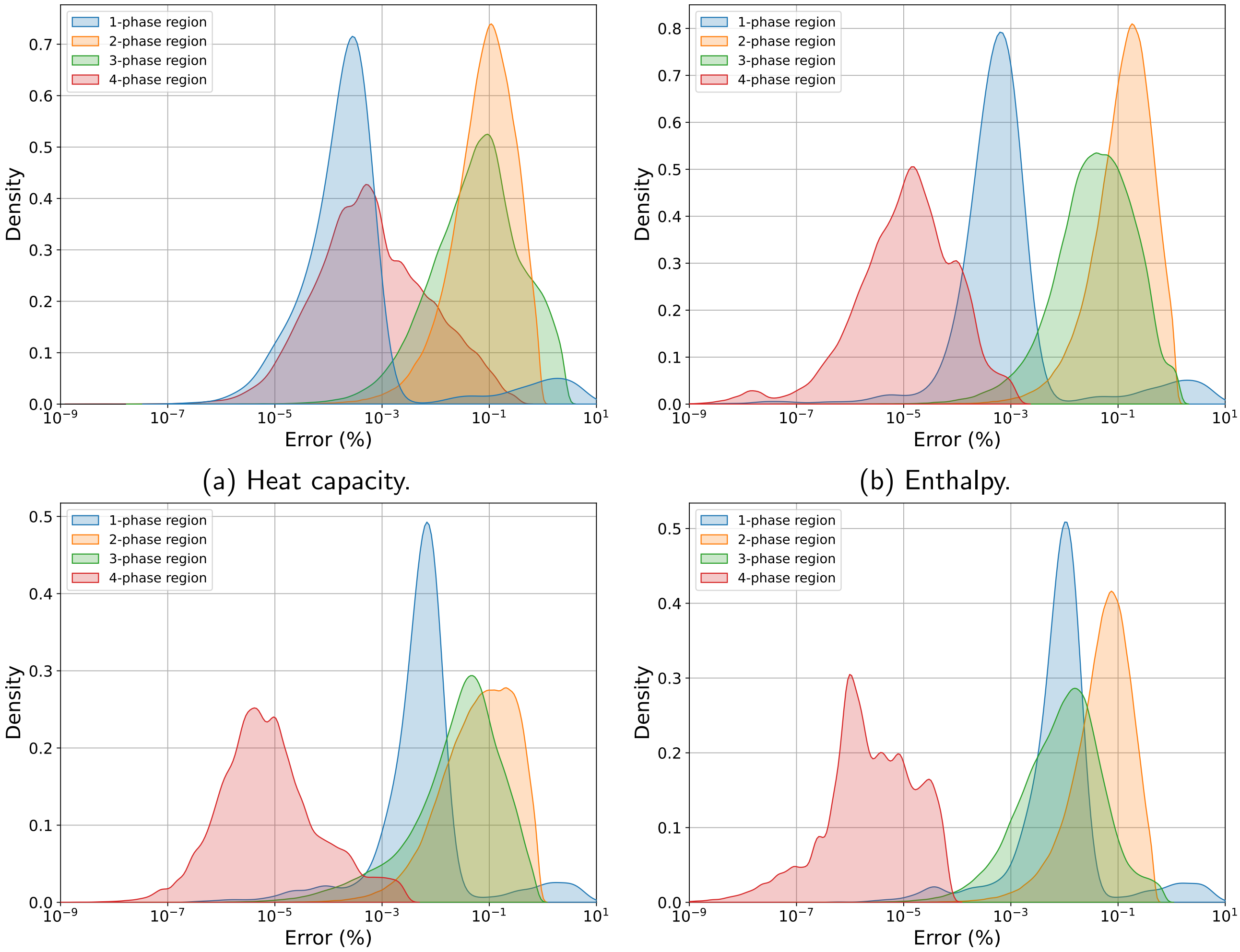
The distribution of errors made on physical and thermochemical properties are shown in Figure 11.3. As mentioned, errors made on the system properties can become large as interpolation errors are made twice. The majority of the errors made on all the properties are in the order of 1 % and less for all phase regions, apart from single-phase regions that show a small number of errors made in the order of 1 % to 10 %.



(a) Phase composition difference.

(b) Phase fraction difference.

Figure 11.2: Distribution of phase composition and phase fraction difference for the ilmenite smelting system test.



(a) Heat capacity.

(b) Enthalpy.

(c) Entropy.

(d) Gibbs energy.

Figure 11.3: Distribution of physical and thermochemical property errors for the ilmenite smelting system test.

11.2 Simplified Iron- and Steelmaking System

11.2.1 Background

Iron- and steelmaking is one of the most CO_2-intensive industries (Zhang et al. 2021) and accounts for 6 % to 9 % of global CO_2 emissions (Pandit, Watson, and Qader 2020). Regulations have been imposed to reduce emissions, which is leading to the development of new and alternative processes and technologies, some of which substitute carbon with natural gas and hydrogen as reducing agents and utilise renewable energy sources (Zhang et al. 2021).

With stringent deadlines by when emissions have to be reduced and new processes be implemented, there is limited time to build pilot plants to test newly developed processes and even less time for demonstration plants. There is, however, time to build models and use them to rapidly investigate multiple new process concepts, weed out concepts that do not meet requirements, and optimise feasible concepts before new industrial plants are built.

Iron- and steelmaking processes can consist of many system components – in excess of 15 – and this is too many to test the accelerator algorithm with at this stage. However, some preliminary models of the processes can be modelled with a simplified iron- and steelmaking system, C−Ca−Fe−O−Si, and is fitting to perform performance tests with. Efficient inclusion of equilibrium calculations into multiphysics models of simplified iron- and steelmaking processes can enhance our ability to investigate and develop new process technologies. The slag and alloy baths are of interest as this is where reduction and oxidation processes occur and gas is formed.

11.2.2 Test Conditions

Because of the high temperatures associated with the alloy and slag baths, it was decided to test the accelerator algorithm in the simplified iron- and steelmaking system over a temperature range from 1500 K to 2100 K. Three regions of interest were investigated in the ironmaking stage of the process associated with the blast furnace with ranges of average compositions identified from literature; (1) the alloy bath (Ghosh and Chatterjee 2008), (2) the slag bath (Ghosh and Chatterjee 2008; Geiseler 2000; USDoT 2016), and (3) the reduction zone where iron ore (Muwanguzi et al. 2012; Taylor, Page, and Geldenhuys 1988) are mixed with reducing agents in ratios identified from (Taylor, Page, and Geldenhuys 1988). The mass fraction and corresponding system component mole fraction ranges for these three regions are shown in Table 11.2.

Three additional regions of interest were investigated in the steelmaking stage of the process associated with the basic oxygen furnace with ranges of average compositions identified from literature; (1) the alloy bath (Fruehan 1998), (2) the slag bath (Fruehan 1998; Belhadj, Diliberto, and Lecomte 2012; Naidu, Sheridan, and van Dyk 2020; Ruth 2004), and (3) the oxidation zone where hot metal from the blast furnace, scrap iron and steel, and fluxing agents are introduced into the furnace in close proximity to an oxygen lance, with average compositions and mass fractions identified from (Fruehan 1998). The mass fraction and corresponding system component mole fraction ranges for these three regions are shown in Table 11.3.

A total of 1 000 000 system states $\mathbf{X}^{\sigma}$ were randomly generated in these identified temperature and composition ranges. A composition tolerance $\Delta x_{\epsilon}^{\mathcal{D}}$ of 0.01 mol mol^{-1} and temperature tolerance $\Delta T^{\mathcal{D}}$ of 10 K was chosen.

Table 11.2: Mass fraction ranges of phase constituents and corresponding system component mole fraction ranges for regions of interest in the simplified ironmaking system tests.

Phase	Mass fraction ($g\,g^{-1}$)		System	Mole fraction ($mol\,mol^{-1}$)	
Constituent	Min.	Max.	Component	Min.	Max.
		Alloy bath			
C	0.03	0.05	C	0.12	0.19
Ca	0.00	0.01	Ca	0.00	0.01
Fe	0.91	0.97	Fe	0.74	0.87
O	0.00	0.01	O	0.00	0.03
Si	0.00	0.02	Si	0.00	0.04
		Slag bath			
CaO	0.45	0.65	C	0.00	0.00
FeO	0.00	0.02	Ca	0.19	0.28
SiO_2	0.35	0.54	Fe	0.00	0.01
			O	0.57	0.60
			Si	0.14	0.21
		Reduction zone			
C	0.19	0.25	C	0.19	0.43
CaO	0.07	0.10	Ca	0.02	0.05
FeO	0.00	0.01	Fe	0.16	0.22
Fe_2O_3	0.64	0.67	O	0.34	0.48
SiO_2	0.09	0.13	Si	0.03	0.06

Table 11.3: Mass fraction ranges of phase constituents and corresponding system component mole fraction ranges for regions of interest in the simplified steelmaking system tests.

Phase	Mass fraction ($g\,g^{-1}$)		System	Mole fraction ($mol\,mol^{-1}$)	
Constituent	Min.	Max.	Component	Min.	Max.
		Alloy bath			
C	0.00	0.03	C	0.00	0.12
Ca	0.00	0.01	Ca	0.00	0.01
Fe	0.97	1.00	Fe	0.88	1.00
O	0.00	0.01	O	0.00	0.03
Si	0.00	0.01	Si	0.00	0.02
		Slag bath			
CaO	0.40	0.55	C	0.00	0.00
FeO	0.05	0.10	Ca	0.20	0.28
Fe_2O_3	0.15	0.30	Fe	0.09	0.15
SiO_2	0.10	0.20	O	0.54	0.57
			Si	0.05	0.09
		Oxidation zone			
C	0.00	0.04	C	0.00	0.16
Fe	0.91	1.00	Ca	0.00	0.04
Si	0.00	0.01	Fe	0.76	1.00
CaO	0.00	0.04	O	0.00	0.04
			Si	0.00	0.02

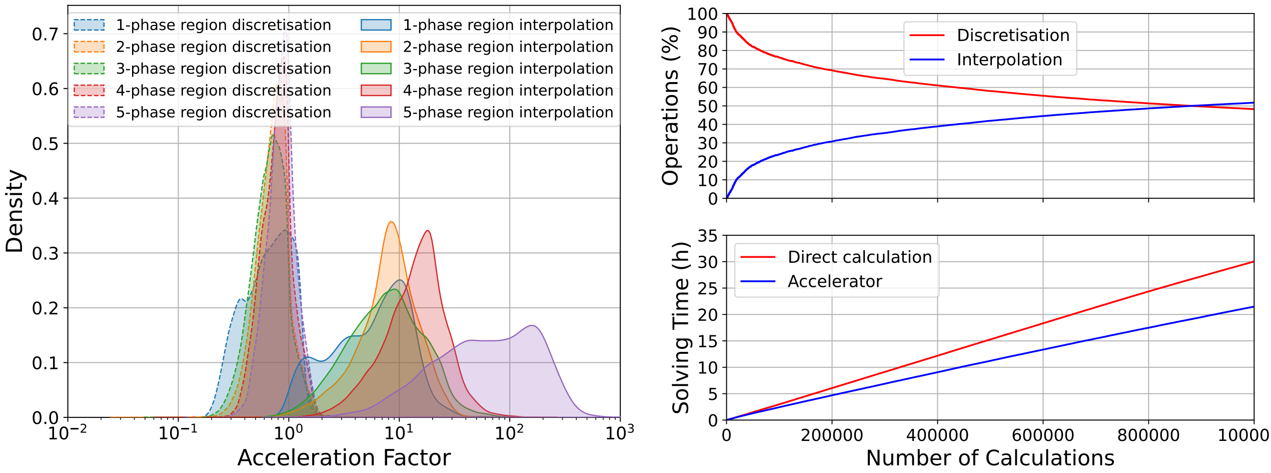
11.2.3 Results

The distribution plots of acceleration factors for all the phase regions are shown in Figure 11.4a. As with the ilmenite smelting system, the acceleration factor is mostly smaller than one during the discretisation algorithm and mostly larger than one during the interpolation algorithm.

It can be seen that during the discretisation of single-phase regions, some acceleration factors are as small as 0.1. This is due to database searches for applicable tie simplices to construct new phase region cells. As mentioned before, phase region cells in single-phase regions are the smallest and requires the most tie simplices to construct a phase region cell, and therefore need much more tie simplices to be generated, and searched through. As the number of stored tie simplices increase, so does the cost of the recall algorithm. This becomes less problematic as the number of phases increase.

During the interpolation algorithm, acceleration factors in the order of 100 and as high as 1000 are achieved with the accelerator in five-phase regions. In a small number of cases, the acceleration factor during the interpolation algorithm was slightly smaller than one and was attributed to a slow searching time for a suitable phase region cell from the database.

Figure 11.4b shows that at the end of the test approximately 54 % of the equilibrium states returned by the accelerator algorithm were interpolated. As in the ilmenite smelting system, this is less than the two- and three-component system tests because the 1 000 000 system states $\mathbf{X}^\sigma$ were divided between the six regions of interest.



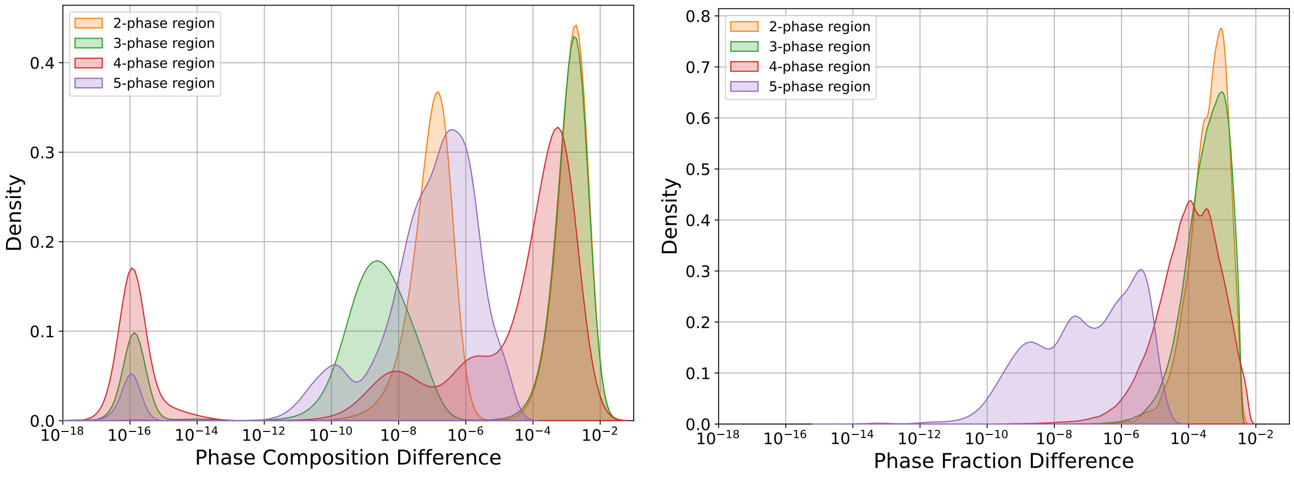
(a) Distribution of acceleration factors.

(b) Operation distribution and solving time.

Figure 11.4: Distribution of acceleration factors and operation distribution and cumulative solving time for the duration of the functionality and performance test for the iron- and steelmaking system.

The distribution of phase composition and phase fraction differences are shown in Figure 11.5. It can be seen that the difference in phase composition of two- to five-phase regions are in the order of 1×10^{-2} mol mol^{-1} and less. Differences in the order of 1×10^{-16} mol mol^{-1} are associated with errors made on the compositions of pure substances. The same trend was observed with the phase fraction differences seeing that it is dependent on the interpolated phase compositions.

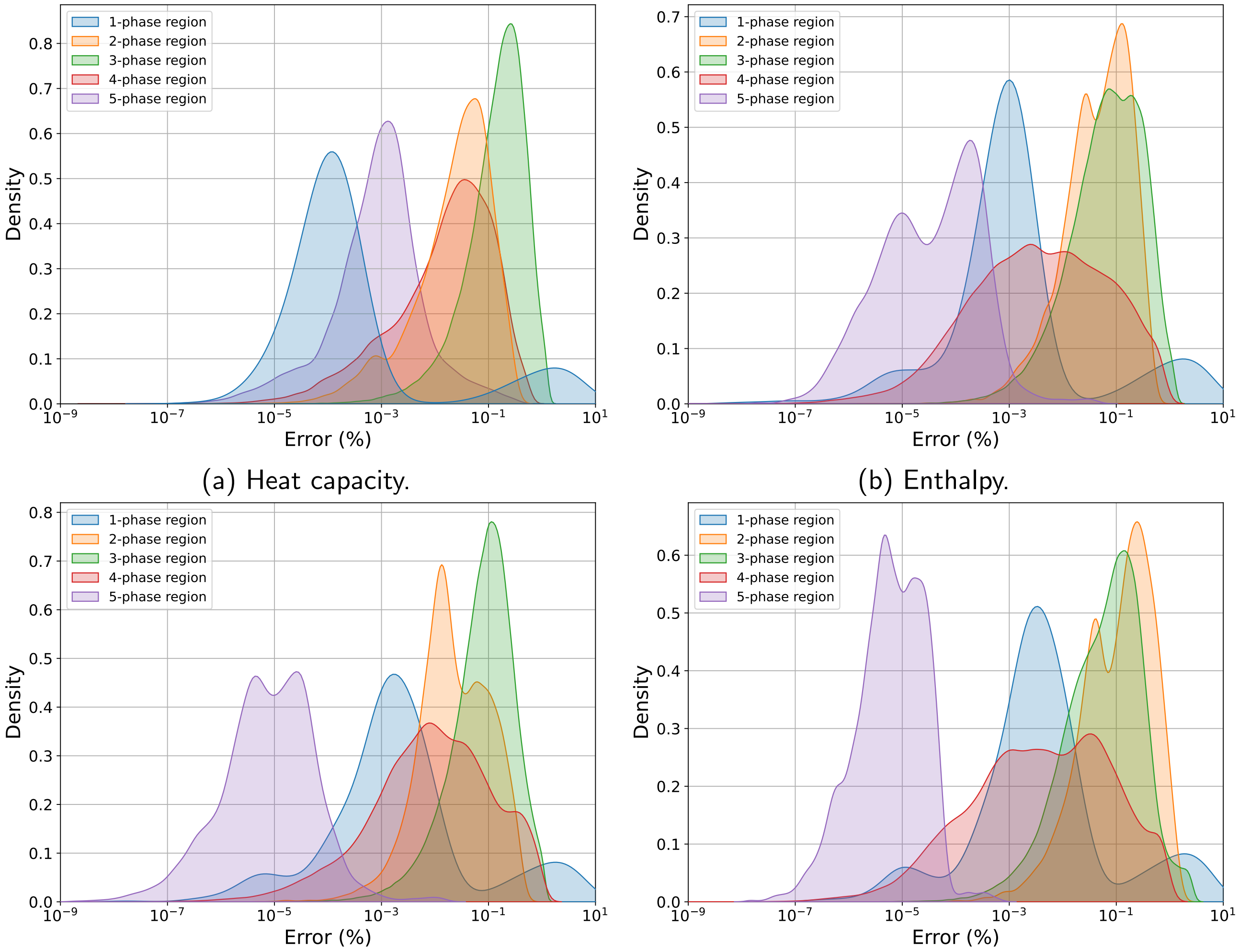
The distribution of errors made on physical and thermochemical properties are shown in Figure 11.6. As mentioned, errors made on the system properties can become large as interpolation errors are made twice. Majority of the errors made on all the properties are in the order of 1 % and less for all phase regions. However, a small number of interpolations resulted in errors made on properties in the order of 1 % to 10 %.



(a) Phase composition difference.

(b) Phase fraction difference.

Figure 11.5: Distribution of phase composition and phase fraction difference for the simplified iron- and steelmaking system test.



(a) Heat capacity.

(b) Enthalpy.

(c) Entropy.

(d) Gibbs energy.

Figure 11.6: Distribution of physical and thermochemical property errors for the simplified iron- and steelmaking system test.

11.3 Conclusion

As expected, the acceleration factors associated with the discretisation algorithm were smaller than one seeing that there are additional computational overhead for the accelerator compared to the equivalent direct calculation. Even though database searches and frequent iterative calculations are performed during the interpolation algorithm, the majority of acceleration factors for all phase regions are larger than one. The algorithm showed a significant acceleration of equilibrium calculations with a notable fraction of interpolations reporting acceleration factors in the order of 100 with some interpolations showing acceleration factors as high as 1000.

During the discretisation algorithm, some acceleration factors were as small as 0.1 and were attributed to slow database searches for applicable tie simplices to construct new phase region cells. It was also observed in a small number of cases that acceleration factors during the interpolation algorithm were smaller than one and were also attributed to slow database searches for suitable phase region cells. The storage and recall algorithms were developed to provide a structured framework to orderly store tie simplices and phase region cells and efficiently recall them when needed, however, these algorithms were implemented in Python3 and it appears that database searches may become the rate limiting step as larger amounts of data are being generated as the number of system components increase, and have to be searched through. For further testing of the accelerator algorithm, it is advised to transfer the accelerator algorithm to a more computationally efficient compiled programming language and utilise a more performant database system to improve the efficiency of the storage and recall algorithm. For this stage of development the acceleration portion of the performance requirement is satisfied.

The magnitude of errors made on phase composition were in the order of 1×10^{-2} mol mol^{-1} and less and phase fraction were in the order of 1×10^{-2} and less. This would translate to an interpolated phase composition being accurate to within 99 % of the calculated phase composition, which is acceptable for the chosen discretisation tolerances and satisfies the accuracy portion of the performance requirement.

Although property errors as high as 10 % are not desired, the accuracy of the majority of physical and thermochemical property errors were in the order of 1 % and less, which, for this stage of development, satisfies the accuracy portion of the performance requirement. Accuracy can be improved by reducing temperature and compositional tolerance; smaller phase region cells would lead to smaller discretisation steps of OPF boundaries and reduce interpolation errors. However, this would also lead to tie simplices having to be much closer to one-another before a smaller phase region cell can be constructed. More of the smaller phase region cells would need to be created to cover the same space as a larger phase region cell, resulting in more computationally expensive direct calculations to be performed.

The accelerator algorithm showed noteworthy acceleration of equilibrium calculations when tested on the two industry-related processes while maintaining acceptable levels of accuracy. Large models of these processes could certainly benefit from the accelerator. However, the accelerator should first be transferred to a more computationally efficient compiled programming language and utilise a more performant database system. This would improve the performance of the accelerator even more while making database searches more efficient.

Part V

Closure

Chapter 12

Conclusion

In this chapter the findings from the literature review on existing methods developed to accelerate equilibrium calculations are concluded and the identified advantages of some methods that were combined to develop the new equilibrium calculation accelerator algorithm summarised. The newly developed equilibrium calculation accelerator algorithm is also summarised and results from the functionality and performance tests discussed.

12.1 Existing Equilibrium Acceleration Algorithms

Several methods have been developed by other researchers in an attempt to accelerate equilibrium calculations and make it more feasible to include into models. The common denominator between the majority of these acceleration methods is based on a pre-calculated or in-situ populated databases of the specific thermodynamic system and utilising the stored data in an interpolation algorithm to accelerate equilibrium calculations.

Pre-populated databases tend to become very large and require a large amount of storage space, even when unstructured grids are used, because the entire thermochemical system space is mapped to the database. In most cases prior knowledge of the system is required to determine the limits of the database mapping and can become unnecessarily large with parts of the database not being accessed by the models.

In-situ or on-demand methods of creating a database showed great promise because only the regions of the thermochemical system that are accessed by the model are captured in the database, reducing the storage space requirement and the amount of data to search through. In this case, no prior knowledge of the system is required to create the database.

Using a system's phase diagram as a guide to discretise the temperature-compositional space allow sparse non-uniform discretisation of phase regions. By utilising thermochemical tools such as the Gibbs phase rule it can be determined whether an entire phase region has to be discretised or if it is only necessary to discretise its boundaries and store the resulting tie simplices. The lever rule can be used with the stored tie simplices to determine the thermochemical properties within the entire phase region without there being a single data point stored inside the phase region. This leads to a remarkable decrease in the amount of data that has to be stored.

There are great advantages in using concepts from fundamental thermochemical theory such as phase diagram geometry, the Gibbs phase rule, and the lever rule in creating an accelerator. The thermochemical theory is a strong base that provides security for the decisions taken when the system is discretised and interpolation is performed with the stored data. Combining this with an in-situ method of discretisation would produce a sparse database that covers large temperature, pressure and compositional ranges. It was decided to develop an accelerator algorithm that utilises these thermochemical tools to build an in-situ populated database of a system.

12.2 Developed Equilibrium Acceleration Algorithm

A new accelerator algorithm was developed based on phase diagram geometry where a system's phase diagram is used to map the thermochemical system to geometric space. An in-situ discretisation method was employed where the database was initially empty. As the model required data, the database was queried, and if no data was available for interpolation, it was generated by performing an equilibrium calculation and storing the results for future queries. In doing this, data was only generated in parts of the system that the model accesses regularly, keeping the amount of data in the database to a minimum.

The lever rule was employed to calculate thermochemical properties at any temperature, pressure and composition within a phase region when the phase compositions and thermochemical properties on the OPF boundaries were known. Therefore, only the OPF boundaries had to be discretised, apart from single-phase regions, reducing the storage space requirement of the database.

The Gibbs phase rule was employed to determine the geometric dimensionalities of the OPF boundaries, which were used to determine the smallest number of direct equilibrium calculations that needed to be performed to discretise the boundaries. By storing the tie simplices produced from equilibrium calculations and creating phase region cells, entire phase regions could be described by a small number of direct equilibrium calculations.

Both the Gibbs phase rule and the lever rule enabled the accelerator algorithm to determine and perform the minimum number of direct equilibrium calculations to discretise phase regions and interpolate within, minimising the storage space requirement. The ability to determine the minimum number of equilibrium calculations required to discretise OPF boundaries, together with the in-situ discretisation method that only stores data in parts of the system that the model accesses regularly satisfies the storage space requirement of the accelerator algorithm.

The Gibbs phase rule, phase diagram and simplex geometry, and the lever rule provides a general basis based on sound principles for the discretisation and interpolation algorithms. Employing these well known, established, thermochemical principles satisfies the requirement for the accelerator algorithm to be thermodynamically consistent. All the employed principles are generic, and the developed discretisation and interpolation algorithms were based and developed on this generality. This results in an accelerator algorithm that is also generic and can be incorporated into a system with any number of components, satisfying the requirement for the accelerator algorithm to be generic.

Because linear interpolation is performed between tie simplices to create an interpolated tie simplex, interpolation errors can be made with phase compositions as well as with physical and thermochemical properties of the phases. To control the interpolation error magnitude, only tie simplices that are within a specified temperature and composition tolerances are used to create a phase region cell. By reducing the tolerance, tie simplices have to be closer to each other before a phase region cell can be created from them, reducing the interpolation errors that are made. This however means that the phase region cells are smaller in size and more phase region cells are required to discretise the phase region. The database would require more storage space to accommodate the larger number of phase region cells, potentially leading to recall times from the database becoming longer, reducing the acceleration factor. At the beginning of the simulation, the frequency of interpolations would remain low for longer periods as the size of phase region cells are small and would not be accessed as regularly compared to larger phase region cells. There is a balance between the acceleration and accuracy of the interpolations and the tolerances provide the user control over the balance, satisfying the requirement.

12.3 Functionality and Performance Tests

The functionality and performance of the accelerator algorithm was tested in 10 two-component and 4 three-component systems. The algorithm's performance was also tested in two industry-related processes; a simplified four-component ilmenite smelting system and a simplified five-component iron- and steelmaking system.

It was observed that single-phase regions had the largest ratio between number of discretisations and number of interpolations. As the number of stable phases increased, so did the number of interpolations and a decrease in the number of discretisations was observed. This is because the size of phase region cells produced by the discretisation algorithm are the smallest in single-phase regions and increase in size as the number of stable phases increase. These larger cells in regions with more stable phases are accessed more regularly for interpolation, leading to a higher frequency of interpolations earlier in the simulation process.

As the number of system components increase, so does the computational expense of direct equilibrium calculations. This translates to larger acceleration factors being achieved as the number of system components increase – from acceleration factors as high as 20 in two-component systems to 1000 in the four- and five-component systems.

During the discretisation algorithm, some acceleration factors were as small as 0.1 and were attributed to slow database searches for applicable tie simplices to construct new phase region cells. It was also observed in a small number of cases that acceleration factors during the interpolation algorithm were smaller than one and were also attributed to slow database searches for suitable phase region cells. The storage and recall algorithms were developed to provide a structured framework to orderly store tie simplices and phase region cells and efficiently recall them when needed. However, these algorithms were implemented in Python3 and it appears that database searches may become the rate limiting step as larger amounts of data are being generated as the number of system components increase, and have to be searched through. The algorithm's acceleration performance can certainly be improved by migrating the implementation to a compiled programming language and using a more performant database system, but for this stage of development the acceleration portion of the performance requirement is satisfied.

Interpolation errors made on phase compositions are in the order of 10^{-2} mol mol^{-1} and less and on phase fractions are in the order of 10^{-2} and less. This would translate to an interpolated phase composition being accurate to within 99 % of the calculated phase composition. This level of accuracy satisfies the accuracy portion of the performance requirement.

Because system properties are determined by the phase fraction weighted sum of the phases' properties, the error made on the system properties can become large as interpolation errors are made twice. Although property errors as high as 10 % are not desired, the accuracy of the majority of physical and thermochemical property errors were in the order of 1 % and less, which, for this stage of development, satisfies the accuracy portion of the performance requirement. Accuracy can be improved by reducing temperature and composition tolerances, resulting in smaller phase region cells that would lead to smaller discretisation segments of OPF boundaries and reduce interpolation errors.

The accelerator algorithm showed noteworthy acceleration of equilibrium calculations when tested on the two industry-related processes while maintaining acceptable levels of accuracy. There is great potential for the accelerator algorithm to make the inclusion of equilibrium calculations in models with many system components feasible. However, the algorithm should first be transferred to a more computationally efficient compiled programming language and utilise a more performant database system. This would make database searches more efficient and improve the performance of the accelerator even more.

Chapter 13

Recommendations

The following recommendations are made regarding improvements to the algorithm implementation as well as alterations and additions that can be made to the algorithm to improve the functionality and performance of the accelerator.

13.1 Implementation Improvements

The accelerator algorithm implementation is by no means perfect and there are much more performance that can be gained by improving it.

13.1.1 Compiled Language

The algorithm and database were implemented in the Python3 interpreted language. An interpreted language was chosen for the development phase because it enabled fast investigations into different discretisation, storage and recall, and interpolation algorithms and sub-routines. Now that the development phase has been completed, migrating the accelerator algorithm to a compiled language would certainly improve the computational efficiency.

13.1.2 Database Design

It was observed that recalls from the database (when it contains a large amount of data) can become the rate limiting step in the accelerator algorithm to the point where acceleration is no longer achieved. Storage and recall algorithms were developed, but are by no means optimised and there is room for improving these algorithms. The database was also implemented in Python3 and the use of a more performant database package, that is designed and properly set up for the accelerator algorithm, will see an improvement in recall times and alleviate the problems of this potential rate limiting step.

13.2 Algorithm Alterations

Some sub-routines of the accelerator algorithm could benefit from a redesign or improvement. Some accelerator algorithm concepts were not pursued further, but showed promise and could have some advantage over the current implementation. Hereafter are a few alterations to the accelerator algorithm that could be investigated.

13.2.1 Iterative Interpolation Step

The iterative step in the interpolation algorithm that solves the system of non-linear equations to obtain an interpolated tie simplex needs to be addressed to improve the performance of the accelerator algorithm further. An explicit solution is desired, however, no explicit solution to this problem or a similar problem has been found in literature or could be derived yet. An alternative is to improve the efficiency of the iterative solution by determining the generic Jacobian matrix for the generic system of non-linear equations.

13.2.2 Structured Grid Approach

Although a uniform structured reference frame was used to construct the database to assist with storing and recalling tie simplices and phase region cells, the accelerator algorithm followed an unstructured approach – tie simplices were created at arbitrary system states and added to the database. One accelerator algorithm concept that was not pursued further utilised the uniform structured reference frame and performed equilibrium calculations at the nodes nearest to the arbitrary system state, and combining them to create a phase region cell.

The advantage is that the phase region cells would be governed by the specified tolerances (which governs the uniform structured reference frame) and could, in most cases, be as large as the tolerances allow. It would also improve the recall algorithm efficiency as the indices of tie simplices and phase region cells could be determined and retrieved efficiently. The disadvantage is that some phase regions are smaller than the specified tolerances and could not possibly be captured without performing some local refinement of the uniform structured reference frames to a level where discretisation could be performed and the phase region captured. This was the main reason why this concept was not pursued further, but there may be advantages that can be utilised when combined with a local refinement of the uniform structured reference frame.

13.2.3 Adaptive Phase Region Cells

The algorithm follows an unstructured approach where tie simplices were created at arbitrary system states and added to the database. From these tie simplices, phase region cells were created when the tie simplices were within a specified tolerance. This was a continuous process and often lead to phase region cells being much smaller than the theoretical maximum size and the temperature and compositional space was not optimally discretised.

When a new tie simplex is created near an existing phase region cell, and is within tolerance with the tie simplices of the phase region cell, the cell could be enlarged by substituting a currently used tie simplex for the new tie simplex. Phase region cells could be enlarged until they reach the tolerance limit or until they come into contact with another phase region cell. With this method the same temperature and composition space could be covered with less phase region cells.

One problem that presented itself in early testing of adaptive phase region cells, especially in single-phase regions, was that as phase region cells were enlarged towards new tie simplices, some would start to overlap, which is not a problem, but small portions of the temperature and compositional space were left uncovered. This occurred because the enlargement was done with no knowledge or awareness of neighbouring phase region cells.

Adaptive phase region cells were not implemented because of the additional complexity that it would bring to the algorithm at this early stage of development. However, an improved discretisation algorithm with a more intelligent tie simplex selecting sub-routine to enlarge phase region cells could make this a powerful approach. This would reduce the number of phase region cells that need to be stored in the database and searched through, reducing database storage spaces size and recall time. Adding knowledge of neighbouring phase region cells could reduce the chances of enlarged phase region cells overlapping others. Another approach could be to periodically apply a relaxation (with multiphysics model mesh relaxation as an example) to the database, or a portion thereof, to optimally discretise the temperature and compositional space with the available tie simplices.

13.3 Algorithm Additions

Additions can be made to the accelerator algorithm to add functionality or improve performance.

13.3.1 Pressure as Non-compositional Intensive Variable

The developed accelerator algorithm is generic and system pressure can be included as a non-compositional potential variable, however, the accelerator algorithm is currently implemented with pressure assumed to be constant. A valuable addition to the implementation would be to include pressure as a non-compositional potential variable.

13.3.2 Surrogate Material Property Models

Currently the physical and thermochemical properties that are stored by the accelerator algorithm are the properties obtained from the equilibrium calculation software. There are many more material properties that are of importance to models that are not available from equilibrium calculations such as thermal and electrical conductivity. However, surrogate material property models do exist that can take the system state or equilibrium state and estimate these additional properties. These estimated properties can then be added to the database together with the existing properties and the accelerator algorithm can provide interpolated estimates of all the properties.

13.3.3 Phase Region Cell Skewness Check

In some cases in single-phase regions, interpolation errors on physical and thermochemical properties could be as high as 10 %, which is undesired. One possible reason for the large errors could be that the phase region cell was highly skewed. Before the phase region cell is constructed a check could be done to determine its skewness and, if it is outside a specified tolerance, could be discarded and different tie simplices considered to construct a less skew phase region cell instead.

13.3.4 Verification of Conservation of Mass and Energy

During interpolation, errors are made on the phase compositions and phase fractions. This leads to inaccuracies regarding the conservation of mass of system components. Verification should be done to ensure that the amounts of system components specified at the system state is distributed between the stable phases at the equilibrium state within an acceptable accuracy.

Enthalpy is regularly used in models to relate the amount of energy absorbed by a substance to the change in its temperature. In the tests performed in this project, the system state is specified in terms of temperature and composition and the interpolated system enthalpy is determined. The specified system state can also consist of the composition and system enthalpy and would result in an interpolated temperature. If a specified temperature is used to determine the interpolated system enthalpy, which in turn is used as input, the same specified temperature should be obtained. Verification should be done on this cycle to ensure the returned temperature is within reasonable accuracy to the specified temperature. The same should also be done to enthalpy. Large errors in these cyclic verification calculations may lead to large model inaccuracies regarding the conservation of energy and can cause erroneous results or model instability.

13.3.5 Quadratic Interpolation

The accelerator algorithm currently employs linear interpolation between stored tie simplices to obtain an interpolated tie simplex. This can lead to large interpolation errors being made on phase compositions and phase fractions, and in turn physical and thermochemical properties. To reduce the errors made, quadratic interpolation can be employed instead. This require more tie simplices to be combined to create a single phase region cell in comparison to when linear interpolation is employed. The dimensionality of the phase region boundaries are described by the Gibbs phase rule and, just as with linear interpolation, the number of tie simplices required to create a phase region cell for quadratic interpolation can be determined.

13.3.6 Parallelise Direct Calculations and Interpolations

The current accelerator algorithm implementation perform direct calculations in series. Parallelising these calculations can reduce the total time spent on equilibrium calculations. Equilibrium calculation software that can parallelise single direct calculations exist and can easily be integrated into the current accelerator algorithm. Another possible method of parallelising equilibrium calculations is to allow the algorithm to distribute single equilibrium calculations over a number of processors – use single-processor equilibrium calculation software, but deploy multiple instances across multiple processors. In either case the database would be populated much faster by parallelising equilibrium calculations.

Interpolations are also performed in series in the current accelerator algorithm implementation. These interpolations can also be parallelised across a number of processors to reduce the total time spent on them.

13.3.7 GPU Implementation

Modern workstation GPUs can have onboard memory of up to 48 GB and in excess of 10 000 cores. When mentioning GPU parallelisation, the first thing that comes to mind is using the large number of GPU cores to perform a vast number of interpolations in parallel. However, a known potential problem is the data transfer between the computer memory or storage and the GPU memory. Nowadays, it is possible to host databases on a GPU and query the database efficiently, alleviating the need to transfer data to and from the GPU. GPU based equilibrium calculation software have been developed and utilising it, together with the GPU hosted database and parallelised interpolation algorithm, the need for large amounts of data transfer to and from the GPU would not exist. The only data that needs to be transferred to the GPU would be the system state at which the equilibrium state is required and the only data transferred from the GPU would be the calculated or interpolated equilibrium state. A GPU implementation of the algorithm may provide an efficient and greatly parallelised method to accelerate equilibrium calculations.

Bibliography

Agelet de Saracibar, C., M. Chiumenti, and M. Cervera (2006). “Current Developments on the Coupled Thermomechanical Computational Modeling of Metal Casting Processes”. In: *Enhancement and Promotion of Computational Methods in Engineering and Science*, pp. 247–260. DOI: 10.1007/978-3-540-48260-4_93.

Alexandria, Heron of (100AD). *Metrica*.

Andersson, J.O. et al. (2002). “Thermo-Calc and DICTRA, Computational Tools for Materials Science”. In: *CALPHAD: Computer Coupling of Phase Diagrams and Thermochemistry* 26, pp. 273–312. DOI: 10.1016/S0364-5916(02)00037-8.

Bale, C.W. et al. (2016). “FactSage Thermochemical Software and Databases - 2010 – 2016”. In: *CALPHAD: Computer Coupling of Phase Diagrams and Thermochemistry* 54, pp. 35–53. DOI: 10.1016/j.calphad.2016.05.002. URL: www.factsage.com.

Belhadj, E., C. Diliberto, and A. Lecomte (2012). “Characterization and Activation of Basic Oxygen Furnace Slag”. In: *Cement and Concrete Composites* 34, pp. 34–40. DOI: 10.1016/j.cemconcomp.2011.08.012.

Blond, E. et al. (2014). “Multiphysics Modelling Applied to Refractory Behaviour in Severe Environments”. In: *Advances in Science and Technology* 92, pp. 301–309. ISSN: 1662-0356. DOI: 10.4028/www.scientific.net/AST.92.301. URL: http://www.scientific.net/AST.92.301.

Callister, W.D. and D.G. Rethwisch (2011). 8th ed. John Wiley & Sons. ISBN: 978-0-470-50586-1.

Chen, J.-Y. (2004). “Analysis of In Situ Adaptive Tabulation Performance for Combustion Chemistry and Improvement With a Modified Search Algorithm”. In: *Combustion Science and Technology* 176.7, pp. 1153–1169. DOI: 10.1080/00102200490426488.

Chen, S.-L. et al. (2002). “The PANDAT Software Package and Its Applications”. In: *CALPHAD: Computer Coupling of Phase Diagrams and Thermochemistry* 26, pp. 175–188. DOI: 10.1016/S0364-5916(02)00034-2.

Christo, F.C. et al. (1995). “Utilising Artificial Neural Network and Repro-modelling in Turbulent Combustion”. In: *Proceedings of ICNN’95 - International Conference on Neural Networks*, pp. 911–916. DOI: 10.1109/ICNN.1995.487540.

Christo, F.C. et al. (1996). “An Integrated PDF/Neural Network Approach for Simulating Turbulent Reacting Systems”. In: *Symposium (International) on Combustion/The Combustion Institute* 26, pp. 43–48. DOI: 10.1016/S0082-0784(96)80198-6.

Corcoran, E.C., M.H. Kaye, and M.H.A. Piro (2016). “An Overview of Thermochemical Modelling of CANDU Fuel and Applications to the Nuclear Industry”. In: *CALPHAD: Computer Coupling of Phase Diagrams and Thermochemistry* 55, pp. 52–62. ISSN: 03645916. DOI: 10.1016/j.calphad.2016.04.010. URL: 10.1016/j.calphad.2016.04.010.

Davies, R.H. et al. (2002). “MTDATA - Thermodynamic and Phase Equilibrium Software From the National Physical Laboratory”. In: *CALPHAD: Computer Coupling of Phase Diagrams and Thermochemistry* 26, pp. 229–241. DOI: 10.1016/S0364-5916(02)00036-6.

Doré, X, H. Combeau, and M. Rappaz (2000). "Modelling of Microsegregation in Ternary Alloys: Application to the Solidification of Al-Mg-Si". In: *Acta Materialia* 48, pp. 3951–3962. ISSN: 13596454. DOI: 10.1016/S1359-6454(00)00177-4.

Eriksson, G. (1971). "Thermodynamic Studies of High Temperature Equilibria. III. SOLGAS, a Computer Program for Calculating the Composition and Heat Condition of an Equilibrium Mixture". In: *Acta Chemica Scandinavica* 25, pp. 2651–2658023. DOI: 10.3891/acta.chem.scand.25-2651.

Eriksson, G. and K. Hack (1990). "ChemSage - A Computer Program for the Calculation of Complex Chemical Equilibria". In: *Metallurgical Transactions B* 21, pp. 1013–1023.

Ex Mente Technologies (2019). *ChemAppPy*. 'ChemAppPy' was a package developed by Ex Mente Technologies that made the ChemApp library developed by GTT Technologies available in Python. A commercial version is now developed and maintained by GTT Technologies under the name 'ChemApp for Python'. URL: https://gtt-technologies.de/chemapp-for-python/ (visited on 07/18/2023).

Fruehan, R.J. (1998). *The Making, Shaping and Treating of Steel: Steelmaking and Refining Volume*. 11th. Pittsburgh: The AISE Steel Foundation. ISBN: 978-0-930767-02-0.

Gandham, R. et al. (2016). "GPU Acceleration of Equation of State Calculations in Compositional Reservoir Simulation". In: *15th European Conference on the Mathematics of Oil Recovery*. Amsterdam, Netherlands.

Properties of Iron and Steel Slags Regarding Their Use (2000). Molten Slags, Fluxes and Salts. Stockholm, Sweden and Helsinki, Finland. URL: https://www.pyrometallurgy.co.za/MoltenSlags2000/pdfs/207.pdf.

Ghosh, A. and A. Chatterjee (2008). *Ironmaking and Steelmaking: Theory and Practice*. 2nd. New Delhi: PHI Learning Private Limited. ISBN: 978-81-203-3289-8.

Gibbs, J.W. (1873). "A Method of Geometrical Representation of the Thermodynamic Properties of Substances by means of Surfaces". In: *Transactions of the Connecticut Academy of Arts and Sciences*, pp. 382–404.

Gous, M. (2006). "An Overview of the Namakwa Sands Ilmenite Smelting Operations". In: *The South African Institute of Mining and Metallurgy* 106, pp. 379–384. DOI: 10520/AJA0038223X_3100.

Guérillot, D. and J. Bruyelle (2020). "Geochemical Equilibrium Determination Using an Artificial Neural Network in Compositional Reservoir Flow Simulation". In: *Computational Geosciences* 24, pp. 697–707. DOI: 10.1007/s10596-019-09861-4.

Hack, K. (2008). "I.3 - Phase Diagrams". In: *The SGTE Casebook*. Ed. by K. Hack. Second Edition. Woodhead Publishing Series in Metals and Surface Engineering. Woodhead Publishing, pp. 43–72. ISBN: 978-1-84569-215-5. DOI: 10.1533/9781845693954.1.43. URL: https://www.sciencedirect.com/science/article/pii/B9781845692155500024.

Hay, T., T. Echterhof, and V.-V. Visuri (2019). "Development of an Electric Arc Furnace Simulator Based on a Comprehensive Dynamic Process Model". In: *Processes* 7, p. 852. DOI: 10.3390/pr7110852.

Hay, T., A. Reimann, and T. Echterhof (2019). "Improving the Modeling of Slag and Steel Bath Chemistry in an Electric Arc Furnace Process Model". In: *Metallurgical and Materials Transactions B* 50, pp. 2377–2388. DOI: 10.1007/s11663-019-01632-x.

Jha, R. et al. (2018). "Combined Machine Learning and CALPHAD Approach for Discovering Process-structure Relationships in Soft Magnetic Alloys". In: *Computational Materials Science* 150, pp. 202–211. DOI: 10.1016/j.commatsci.2018.04.008.

Kirk, T. et al. (2018). "Computational Design of Gradient Paths in Additively Manufactured Functional Graded Materials". In: *Journal of Mechanical Design* 140, pp. 1111410–1–1111410–9. DOI: 10.1115/1.4040816.

Koukkari, P. and R. Pajarre (2011). "A Gibbs Energy Minimization Method for Constrained and Partial Equilibria". In: *Pure and Applied Chemistry* 83, pp. 1243–1254. DOI: 10.1351/PAC-CON-10-09-36.

Lam, S.K., A. Pitrou, and S. Seibert (2015). "Numba: A LLVM-based Python JIT Compiler". In: *Proceedings of the Second Workshop on the LLVM Compiler Infrastructure in HPC*, pp. 1–6.

Larsson, H. and L. Höglund (2015). "A Scheme for More Efficient Usage of CALPHAD Data in Simulations". In: *CALPHAD: Computer Coupling of Phase Diagrams and Thermochemistry* 50, pp. 1–5. DOI: 10.1016/j.calphad.2015.04.007.

Leal, A.M.M. et al. (2020). "Accelerating Reactive Transport Modeling: On-Demand Machine Learning Algorithm for Chemical Equilibrium Calculations". In: *Transport in Porous Media* 133, pp. 161–204. DOI: 10.1007/s11242-020-01412-1.

Liu, B.J.D. and S.B. Pope (2005). "The Performance of In-situ Adaptive Tabulation in Computations of Turbulent Flames". In: *Combustion Theory and Modelling* 9.4, pp. 549–568. DOI: 10.1080/13647830500307436.

Lu, L. and S.B. Pope (2009). "An Improved Algorithm For In-situ Adaptive Tabulation". In: *Journal of Computational Physics* 228.2, pp. 361 –386. DOI: 10.1016/j.jcp.2008.09.015.

Marin-Alvarado, T.L. (2015). "Combining Multiphysics Modelling and Solution Thermodynamics Using M4Dlib, an External Library". In: *Proceedings of the 2015 COMSOL Conference*. Boston, USA.

Murty, C.V.G.K., R. Upadhyay, and S. Asokan, eds. (2007). *Electro Smelting of Ilmenite for Production of TiO_2 Slag – Potential of India as a Global Partner*. Infacon XI.

Muwanguzi, A.J.B. et al. (2012). "Characterization of Chemical Composition and Microstructure of Natural Iron Ore from Muko Deposits". In: *International Scholarly Research Network Materials Science*. DOI: doi:10.5402/2012/174803.

Naidu, T.S., C.M. Sheridan, and L.D. van Dyk (2020). "Basic Oxygen Furnace Slag: Review of Current and Potential Uses". In: *Minerals Engineering* 149. DOI: 10.1016/j.mineng.2020.106234.

Nichita, Dan Vladimir, Susana Gomez, and Eduardo Luna (2002). "Multiphase Equilibria Calculation by Direct Minimization of Gibbs Free Energy With a Global Optimization Method". In: *Computers & Chemical Engineering* 26.12, pp. 1703 –1724. ISSN: 0098-1354. DOI: 10.1016/S0098-1354(02)00144-8. URL: http://www.sciencedirect.com/science/article/pii/S0098135402001448.

Otis, R. and Z.-K. Liu (2017). "pycalphad: CALPHAD-based Computational Thermodynamics in Python". In: *Journal of Open Research Software* 5, pp. 1 –11. ISSN: 2049-9647. DOI: 10.5334/jors.140.

Pandit, J.K., M. Watson, and A. Qader (2020). *Reduction of Greenhouse Gas Emission in Steel Production Final Report. Publication Number RPT20-6205.* URL: https://www.resourcesregulator.nsw.gov.au/sites/default/files/2022-11/report-reduction-of-ghg-emissions-in-steel-industries.pdf.

Pannala, S., S. Simunovic, and G. Frantziskonis (2010). *Computational Modeling in Lignocellulosic Biofuel Production*. Chap. 11: Multiscale/Multiphysics Modelling of Biomass Thermochemical Processes, pp. 245–271. ISBN: 9780841225718. DOI: 10.1021/bk-2010-1052.ch011.

Pillai, R. et al. (2016). "Methods to Increase Computational Efficiency of CALPHAD-based Thermodynamic and Kinetic Models Employed in Describing High Temperature Material Degradation". In: *CALPHAD: Computer Coupling of Phase Diagrams and Thermochemistry* 53. DOI: 10.1016/j.calphad.2016.03.004.

Piro, M.H.A. et al. (2013). "The Thermochemistry Library Thermochimica". In: *Computational Materials Science* 67, pp. 266 –272. ISSN: 0927-0256. DOI: 10.1016/j.commatsci.2012.09.011. URL: http://www.sciencedirect.com/science/article/pii/S0927025612005502.

Pistorius, P.C. (2007). "Ilmenite Smelting: The Basics". In: *The Southern African Institute of Mining and Metallurgy* 108, pp. 35–43. ISSN: 2411-9717.

Pope, S.B. (1997). "Computationally Efficient Implementation of Combustion Chemistry Using In-situ Adaptive Tabulation". In: *Combustion Theory and Modelling* 1.1, pp. 41–63. DOI: 10.1080/713665229.

Pope, S.B. and U. Maas (1993). "Mechanical and Aerospace Engineering Report FDA 93-11". In: *Cornell University*.

Poschmann, M., M.H.A. Piro, and S. Simunovic (2020). *Acceleration of Thermochimica Calculations in Bison*. type ORNL/TM-2020/1473. Oak Ridge National Laboratory. DOI: 10.2172/1615992.

Poschmann, M. et al. (2021). "Recent developments for molten salt systems in Thermochimica". In: *CALPHAD: Computer Coupling of Phase Diagrams and Thermochemistry* 75. DOI: 10.1016/j.calphad.2021.102341.

Qiu, K. et al. (2015). "Polynomial Regression and Interpolation of Thermodynamic Data in Al-Si-Mg-Fe System". In: *CALPHAD: Computer Coupling of Phase Diagrams and Thermochemistry* 48, pp. 175–183. DOI: 10.1016/j.calphad.2015.01.005.

Rockafellar, R.T. (1997). "Convex Analysis : (PMS-28)." In: Princeton University Press. Chap. 1, pp. 3–40. ISBN: 9780691015866.

Roos, W.A., A.E.J. Bogaers, and J.H. Zietsman (2023). "Geometric Acceleration of Complex Chemical Equilibrium Calculations – Performance in Two- to Five-component Systems". In: *CALPHAD: Computer Coupling of Phase Diagrams and Thermochemistry* 82. DOI: 10.1016/j.calphad.2023.102584.

Roos, W.A. and J.H. Zietsman (2022a). "Accelerating Complex Chemical Equilibrium Calculations – A Review". In: *CALPHAD: Computer Coupling of Phase Diagrams and Thermochemistry* 77. DOI: 10.1016/j.calphad.2021.102380.

— (2022b). "Geometric Acceleration of Complex Chemical Equilibrium Calculations – Algorithm and Application to Two- and Three-component Systems". In: *CALPHAD: Computer Coupling of Phase Diagrams and Thermochemistry* 77. DOI: 10.1016/j.calphad.2022.102420.

Rossum, G. van and F.L. Drake (2009). *Python 3 Reference Manual*. Scotts Valley, CA: CreateSpace. ISBN: 1441412697.

Ruth, M. (2004). "Steel Production and Energy". In: *Encyclopedia of Energy* 5, pp. 695–706. DOI: 10.1016/B0-12-176480-X/00371-5.

Saad, A., C.-A. Gandin, and M. Bellet (2015). "Temperature-based Energy Solver Coupled With Tabulated Thermodynamic Properties – Application to the Prediction of Macrosegregation in Multicomponent Alloys". In: *Computational Materials Science* 99, pp. 221–231. DOI: 10.1016/j.commatsci.2014.12.009.

Samuelsson, K. et al. (2020). "Simulation of the Chemical State of High Burnup (U, Pu)O2 Fuel in Fast Reactors Based on Thermodynamic Calculations". In: *Journal of Nuclear Materials* 532, pp. 151969–1–151969–12. DOI: 10.1016/j.jnucmat.2019.151969.

Shobu, K. (2009). "CaTCalc: New Thermodynamic Equilibrium Calculation Software". In: *CALPHAD: Computer Coupling of Phase Diagrams and Thermochemistry* 33, pp. 279–287. DOI: 10.1016/j.calphad.2008.09.015.

Smith, W.F. and J. Hashemi (2006). *Foundations of Materials Science and Engineering*. 4th. McGraw-Hill.

Smith, W.R. (1980). "The Computation of Chemical Equilibria in Complex Systems". In: *Industrial and Engineering Chemistry Fundamentals* 19.1, pp. 1–10. DOI: 10.1021/i160073a001.

Sommerville, D.M.Y. (1929). "An Introduction to the Geometry of N Dimensions". In: vol. 1. 36 Essex Street W.C. London: Methuen and Co. Ltd. Chap. 8, pp. 124–126.

Strandlund, H. (2004). "High-speed Thermodynamic Calculations for Kinetic Simulations". In: *Computational Materials Science* 29.2, pp. 187–194. DOI: 10.1016/j.commatsci.2003.09.001.

Sundman, B. and I. Ansara (2008). "The Gulliver–Scheil Method for the Calculation of Solidification Paths". In: *The SGTE Casebook*. Ed. by K. Hack. Second Edition. Woodhead Publishing Series in Metals and Surface Engineering. Woodhead Publishing, pp. 343–346. ISBN: 978-1-84569-215-5. DOI: 10.1533/9781845693954.3.343.

Sundman, B., X.-G. Lu, and H. Ohtani (2015). "The Implementation of an Algorithm to Calculate Thermodynamic Equilibria for Multi-component Systems With Non-ideal Phases in a Free Software". In: *Computational Material Science* 101, pp. 127–137. DOI: 10.1016/j.commatsci.2015.01.029.

Sundman, B. et al. (2015). "OpenCalphad - A Free Thermodynamic Software". In: *Ingegrating Materials and Manufacturing Innovation* 4, pp. 1 –15. ISSN: 2193-9772. DOI: 10.1186/s40192-014-0029-1.

Tabiei, A. and S. Sockalingam (2012). "Multiphysics Coupled Fluid/Thermal/Structural Simulation for Hypersonic Reentry Vehicles". In: *Journal of Aerospace Engineering* 25.2, pp. 273–281. ISSN: 0893-1321. DOI: 10.1061/(ASCE)AS.1943-5525.0000113.

Taylor, D.J.C., D.C. Page, and P. Geldenhuys (1988). "Iron and Steel in South Africa". In: *The Southern African Institute of Mining and Metallurgy* 88.3, pp. 73–95. ISSN: 0038-223X.

Teh, Y.S. and G.P. Rangaiah (2002). "A Study of Equation-solving and Gibbs Free Energy Minimization Methods for Phase Equilibrium Calculations". In: *Chemical Enginnering Research and Design* 80.7, pp. 745–759. DOI: 10.1205/026387602320776821.

ten Cate, A. et al. (2008). "Modeling and Simulation of Phase-transitions in Multicomponent Aluminium Alloy Casting". In: *Proceedings of the sixty-third European Study Group Mathematics With Industry 2008*. Enschede, The Netherlances, pp. 117–139.

Transportation, Federal Highway Administration U.S. Department of (2016). *User Guidelines for Waste and Byproduct Materials in Pavement Construction*. U.S. Department of Transportation, Federal Highway Administration. URL: https://www.fhwa.dot.gov/publications/research/infrastructure/structures/97148/bfs1.cfm#.

Virtanen, P. et al. (2020). "SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python". In: *Nature Methods* 17, pp. 261–272. DOI: 10.1038/s41592-019-0686-2.

Voskov, A.L., A.V. Dzuban, and A.I. Maksimov (2015). "TernAPI Program for the Calculation of Ternary Phase Diagrams With Isolated Miscibility Gaps by the Convex Hull Method". In: *Fluid Phase Equilibria* 388, pp. 50–58. ISSN: 0378-3812. DOI: 10.1016/j.fluid.2014.12.028.

Voskov, D.V. and H.A. Tchelepi (2008). "Compositional Space Parametrization for Miscible Displacement Simulation". In: *Transport in Porous Media* 75, pp. 111–128. DOI: 10.1007/s11242-008-9212-1.

— (2009). "Tie-simplex Based Mathematical Framework for Thermodynamical Equilibrium Computation of Mixtures With an Arbitrary Number of Phases". In: *Fluid Phase Equilibria* 283, pp. 1–11. DOI: 10.1016/j.fluid.2009.04.018.

Wu, M. et al. (2013). "Modeling Diffusion-governed Solidification of Ternary Alloys – Part 1: Coupling Solidification Kinetics With Thermodynamics". In: *Computational Materials Science* 79, pp. 830–840. DOI: 10.1016/j.commatsci.2013.05.015.

Zhang, X. et al. (2021). "A Review on Low Carbon Emissions Projects of Steel Industry in the World". In: *Journal of Cleaner Production* 306. DOI: doi.org/10.1016/j.jclepro.2021.127259.

Zhao, G. et al. (2012). "Numerical Computations for Temperature, Fraction of Solid Phase and Composition Couplings in Ternary Alloy Solidification With Three Different Thermodynamic Data-acquisition Methods". In: *CALPHAD: Computer Coupling of Phase Diagrams and Thermochemistry* 36, pp. 155–162. DOI: 10.1016/j.calphad.2011.07.002.

Zietsman, J.H. (2004). "Interactions Between Freeze Lining and Slag Bath in Ilmenite Smelting". PhD thesis. Pretoria, South Africa: University of Pretoria.

— (2016). "Efficient Storage and Recall of Slag Thermochemical Properties for Use in Multiphysics Models". In: *The 10th International Conference on Molten Slags, Fluxes and Salts*. Ed. by R.G.

Reddy et al. The Minerals, Metals and Materials Society. Seattle, USA, pp. 635–644. DOI: 10.1002/9781119333197.ch68.

Zietsman, J.H. and P.C. Pistorius (2005). "Ilmenite Smelter Freeze Lining Monitoring by Thermocouple Measurements: Model Results". In: *Heavy Minerals 2005*. Society for Mining, Metallurgy, and Exploration, pp. 221–228.

— (2006). "Modelling of an Ilmenite-smelting DC Arc Furnace Process". In: *Minerals Engineering* 19, pp. 262–279. DOI: 10.1016/j.mineng.2005.06.016.

Appendices

Appendix A

Derivation of Equations for Interpolating to Chemical Potentials

Although iso-non-compositional-potential (iso-ψ) tie simplices $\tilde{\Delta}^{\mathcal{R}_i}_{\psi}$ as well as all other tie simplices within an iso-ψ phase region section $\delta^{\mathcal{R}_i}_{\psi}$ have the same non-compositional potentials $\boldsymbol{\Psi}^{\sigma}$, they have different chemical potentials. As shown in Figures A.1, A.3 and A.5, somewhere within the iso-ψ section through the phase region cell $\delta^{\mathcal{R}_i}_{\psi}$ there exists a tie simplex, referred to as the interpolated tie simplex $\tilde{\Delta}^{\mathcal{R}}$, that contains the system composition and thus have the same chemical potentials $\boldsymbol{\mu}^{\sigma}_{\epsilon}$ as the specified system state. The objective of this interpolation step is to find the phase compositions $\tilde{\mathbf{x}}^{\varphi}_{\epsilon}(\boldsymbol{\Psi}^{\sigma}, \boldsymbol{\mu}^{\sigma}_{\epsilon})$ that correspond to the vertices of the interpolated tie simplex $\tilde{\Delta}^{\mathcal{R}}$ that perfectly contains the system composition $\mathbf{x}^{\sigma}_{\epsilon}$.

The iso-ψ phase region section $\delta^{\mathcal{R}}_{\psi}$ of three- and four-component systems can still be visualised in 2D and 3D respectively; systems with more components are beyond the author's ability to visualise. The method of using barycentric coordinate weights to determine the phase compositions $\tilde{\mathbf{x}}^{\varphi}_{\epsilon}(\boldsymbol{\Psi}^{\sigma}, \boldsymbol{\mu}^{\sigma}_{\epsilon})$ of the interpolated tie simplex $\tilde{\Delta}^{\mathcal{R}}$ for the two-phase region in a three-component system as well as the two- and three-phase regions in a four-component system is discussed hereafter as they can still be visually verified. These three cases are then used to determine a generic method that can be used in any applicable iso-ψ phase region section $\delta^{\mathcal{R}}_{\psi}$, regardless of the number of system components.

A.1 Iso-non-compositional-potential Boundary Discretisation Segment Barycentric Coordinate Weights

As seen in Figures A.2, A.4 and A.6, iso-ψ boundary discretisation segment barycentric coordinate weights $\beta^{\mathcal{B}}$ can be used to express the composition of phase i of the interpolated tie simplex $\tilde{\mathbf{x}}^{\varphi_i}_{\epsilon}(\boldsymbol{\Psi}^{\sigma}, \boldsymbol{\mu}^{\sigma}_{\epsilon})$ as a function of $\hat{\Delta}^{\mathcal{R}}_{\psi}$ number of iso-ψ boundary discretisation segment phase compositions $\tilde{\mathbf{x}}^{\varphi_i}_{\epsilon}(\boldsymbol{\Psi}^{\sigma})$. This expression can be seen in Equation (A.1). Because there are $\hat{\Delta}^{\mathcal{R}}_{\psi}$ iso-ψ boundary discretisation segment phase compositions $\tilde{\mathbf{x}}^{\varphi}_{\epsilon}(\boldsymbol{\Psi}^{\sigma})$, only $\hat{\Delta}^{\mathcal{R}}_{\psi} - 1$ iso-ψ boundary discretisation segment barycentric coordinate weights $\beta^{\mathcal{B}}$ are required in $\boldsymbol{\beta}^{\mathcal{B}}$ to express the composition of phase i of the interpolated tie simplex $\tilde{\mathbf{x}}^{\varphi_i}_{\epsilon}(\boldsymbol{\Psi}^{\sigma}, \boldsymbol{\mu}^{\sigma}_{\epsilon})$.

$$\tilde{\mathbf{x}}^{\varphi_i}_{\epsilon}(\boldsymbol{\Psi}^{\sigma}, \boldsymbol{\mu}^{\sigma}_{\epsilon}) = \tilde{\mathbf{x}}^{\varphi_i}_{\epsilon}(\boldsymbol{\Psi}^{\sigma})_1 + \sum_{j=1}^{\hat{\Delta}^{\mathcal{R}}_{\psi}-1} \beta^{\mathcal{B}}_{j} \left(\tilde{\mathbf{x}}^{\varphi_i}_{\epsilon}(\boldsymbol{\Psi}^{\sigma})_{j+1} - \tilde{\mathbf{x}}^{\varphi_i}_{\epsilon}(\boldsymbol{\Psi}^{\sigma})_1\right) \tag{A.1}$$

It is assumed that system component chemical potentials vary linearly between the phase composition coordinates of an iso-ψ boundary discretisation segment $\delta^{\mathcal{B}}_{\psi}$. This assumption is reasonable when temperature and composition tolerances ($\Delta T^{\mathcal{D}}$ and $\Delta x^{\mathcal{D}}_{\epsilon}$) are sufficiently small. The assumption results in the set of iso-ψ boundary discretisation segment barycentric coordinate weights of all iso-ψ boundary discretisation segments being identical, as shown in Equation (A.2).

$$\boldsymbol{\beta}^{\mathcal{B}}_{1} = \cdots = \boldsymbol{\beta}^{\mathcal{B}}_{\hat{\varphi}} = \boldsymbol{\beta}^{\mathcal{B}} = [\beta^{\mathcal{B}}_{1}, \ldots, \beta^{\mathcal{B}}_{\hat{\Delta}^{\mathcal{R}}_{\psi}-1}] \tag{A.2}$$

Equation (A.1) only provides an expression of a phase composition contained within a iso-ψ boundary discretisation segment $\delta^{\mathcal{B}}_{\psi}$ – varying $\boldsymbol{\beta}^{\mathcal{B}}$ changes the location of these phase compositions on each iso-ψ boundary discretisation segment. These phase compositions have to be coupled to the system composition $\mathbf{x}^{\sigma}_{\epsilon}$ in order to determine the exact phase compositions that correspond to the chemical potential of the system composition.

A.2 Interpolated Tie Simplex Barycentric Coordinate Weights

As seen in Figures A.2, A.4 and A.6, tie simplex barycentric coordinate weights $\boldsymbol{\beta}^{\Delta}$ can be used to express the specified system composition $\mathbf{x}^{\sigma}_{\epsilon}$ as a function of each of the $\hat{\varphi}$ phase compositions $\tilde{\mathbf{x}}^{\varphi}_{\epsilon}(\boldsymbol{\Psi}^{\sigma}, \boldsymbol{\mu}^{\sigma}_{\epsilon})$. This expression is shown in Equation (A.3). An interpolated tie simplex that perfectly contains the system composition $\mathbf{x}^{\sigma}_{\epsilon}$ can be obtained if the conditions in Equation (3.16) are satisfied for all tie simplex barycentric coordinate weights in $\boldsymbol{\beta}^{\Delta}$. Because there are $\hat{\varphi}$ number of phases present in a tie simplex, $\hat{\varphi} - 1$ number of tie simplex barycentric coordinate weights β^{Δ} are required in $\boldsymbol{\beta}^{\Delta}$ to express the system composition $\mathbf{x}^{\sigma}_{\epsilon}$.

$$\mathbf{x}^{\sigma}_{\epsilon} = \tilde{\mathbf{x}}^{\varphi_1}_{\epsilon}(\boldsymbol{\Psi}^{\sigma}, \boldsymbol{\mu}^{\sigma}_{\epsilon}) + \sum_{\mathrm{i}=1}^{\hat{\varphi}-1} \beta^{\Delta}_{\mathrm{i}} \left(\tilde{\mathbf{x}}^{\varphi_{\mathrm{i}+1}}_{\epsilon}(\boldsymbol{\Psi}^{\sigma}, \boldsymbol{\mu}^{\sigma}_{\epsilon}) - \tilde{\mathbf{x}}^{\varphi_1}_{\epsilon}(\boldsymbol{\Psi}^{\sigma}, \boldsymbol{\mu}^{\sigma}_{\epsilon})\right) \tag{A.3}$$

A.3 Creating a System of Equations

With the interpolated tie simplex's phase compositions $\tilde{\mathbf{x}}^{\varphi}_{\epsilon}(\boldsymbol{\Psi}^{\sigma}, \boldsymbol{\mu}^{\sigma}_{\epsilon})$ expressed in terms of iso-ψ boundary discretisation segment phase compositions $\tilde{\mathbf{x}}^{\varphi}_{\epsilon}(\boldsymbol{\Psi}^{\sigma})$ and barycentric coordinate weights $\boldsymbol{\beta}^{\mathcal{B}}$ with Equation (A.1), the barycentric coordinate weights $\boldsymbol{\beta}^{\mathcal{B}}$ have to be solved that satisfy the conditions of Equation (3.16) for all tie simplex barycentric coordinate weights in $\boldsymbol{\beta}^{\Delta}$.

An expression for each of the interpolated tie simplex's $\hat{\varphi}$ phase compositions $\tilde{\mathbf{x}}^{\varphi}_{\epsilon}(\boldsymbol{\Psi}^{\sigma}, \boldsymbol{\mu}^{\sigma}_{\epsilon})$ is created with Equation (A.1) and substituted into Equation (A.3). This creates an expression of the system composition $\mathbf{x}^{\sigma}_{\epsilon}$ as a function of each iso-ψ boundary discretisation segment's phase compositions $\tilde{\mathbf{x}}^{\varphi}_{\epsilon}(\boldsymbol{\Psi}^{\sigma})$ and barycentric coordinate weights $\boldsymbol{\beta}^{\Delta}$ and $\boldsymbol{\beta}^{\mathcal{B}}$.

These expressions are applied hereafter to a two-phase region of a three-component system as well as a two- and three-phase region of a four-component system whereafter they are factorised and written as a system of non-linear equations. The patterns from these three cases are used to create a generic system of non-linear equations that can be used in any iso-ψ phase region section $\delta^{\mathcal{R}_{\mathrm{i}}}_{\psi}$, regardless of the number of system components.

A.3.1 Three-component System

Two-phase Region

Figure A.1 shows a general iso-ψ section through a two-phase region cell of a three-component system. Figure A.2 shows the barycentric coordinate weights used to express the system composition $\mathbf{x}^{\sigma}_{\epsilon}$ in terms of the iso-ψ boundary discretisation segment phase compositions $\tilde{\mathbf{x}}^{\varphi}_{\epsilon}(\mathbf{\Psi}^{\sigma})$. Equations (A.4) and (A.5) expresses each of the interpolated tie simplex's phase compositions $\tilde{\mathbf{x}}^{\varphi}_{\epsilon}(\mathbf{\Psi}^{\sigma}, \boldsymbol{\mu}^{\sigma}_{\epsilon})$ with iso-ψ boundary discretisation segment barycentric coordinate weights $\boldsymbol{\beta}^{\mathcal{B}}$.

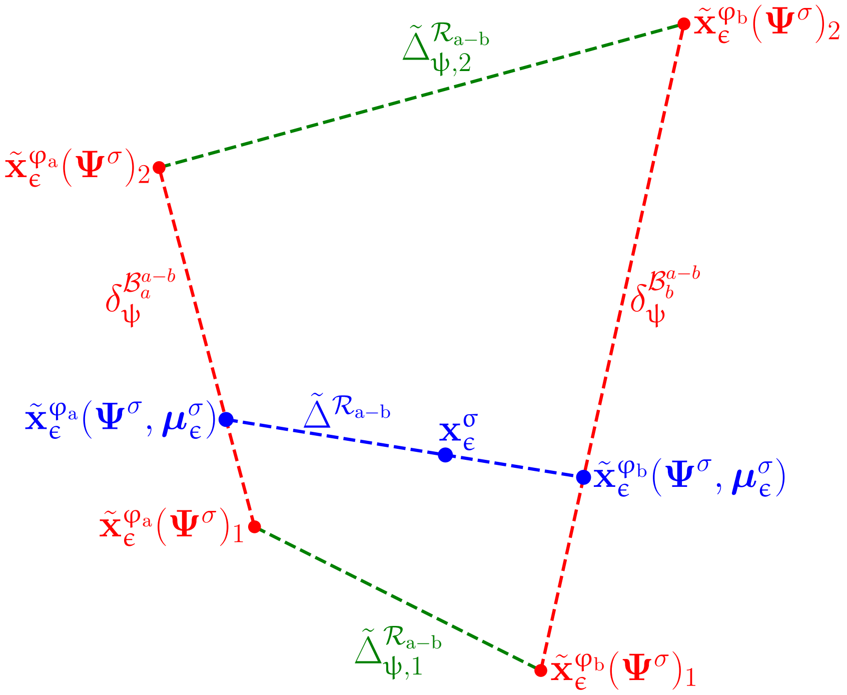

$\tilde{\Delta}^{\mathcal{R}_{a-b}}_{\psi,2}$
$\tilde{\mathbf{x}}^{\varphi_b}_{\epsilon}(\mathbf{\Psi}^{\sigma})_2$
$\tilde{\mathbf{x}}^{\varphi_a}_{\epsilon}(\mathbf{\Psi}^{\sigma})_2$
$\delta^{\mathcal{B}^{a-b}_{a}}_{\psi}$
$\delta^{\mathcal{B}^{a-b}_{b}}_{\psi}$
$\tilde{\mathbf{x}}^{\varphi_a}_{\epsilon}(\mathbf{\Psi}^{\sigma}, \boldsymbol{\mu}^{\sigma}_{\epsilon})$
$\tilde{\Delta}^{\mathcal{R}_{a-b}}$
$\mathbf{x}^{\sigma}_{\epsilon}$
$\tilde{\mathbf{x}}^{\varphi_b}_{\epsilon}(\mathbf{\Psi}^{\sigma}, \boldsymbol{\mu}^{\sigma}_{\epsilon})$
$\tilde{\mathbf{x}}^{\varphi_a}_{\epsilon}(\mathbf{\Psi}^{\sigma})_1$
$\tilde{\Delta}^{\mathcal{R}_{a-b}}_{\psi,1}$
$\tilde{\mathbf{x}}^{\varphi_b}_{\epsilon}(\mathbf{\Psi}^{\sigma})_1$

Figure A.1: Iso-ψ section $\delta^{\mathcal{R}_{a\text{-}b}}_{\psi}$ through a two-phase region cell of a three-component system with iso-ψ tie simplices $\tilde{\Delta}^{\mathcal{R}_{a\text{-}b}}_{\psi}$ and interpolated tie simplex $\tilde{\Delta}^{\mathcal{R}_{a\text{-}b}}$ at the same non-compositional potentials $\mathbf{\Psi}^{\sigma}$. Interpolated tie simplex $\tilde{\Delta}^{\mathcal{R}_{a\text{-}b}}$ is at chemical potential $\boldsymbol{\mu}^{\sigma}_{\epsilon}$, perfectly containing $\mathbf{x}^{\sigma}_{\epsilon}$. The three-component system exist in $\mathbb{D}^{\sigma}_{g} = \hat{\epsilon} - \hat{\zeta}^{\sigma} - 1 + \hat{\psi} = 2 + \hat{\psi}$, meaning an iso-$\psi$ section is perceived in $\mathbb{D}^{\sigma}_{g} = 2$.

$$\begin{aligned}\tilde{\mathbf{x}}^{\varphi_a}_{\epsilon}(\mathbf{\Psi}^{\sigma}, \boldsymbol{\mu}^{\sigma}_{\epsilon}) &= \tilde{\mathbf{x}}^{\varphi_a}_{\epsilon}(\mathbf{\Psi}^{\sigma})_1 + \beta^{\mathcal{B}}_{1}\left(\tilde{\mathbf{x}}^{\varphi_a}_{\epsilon}(\mathbf{\Psi}^{\sigma})_2 - \tilde{\mathbf{x}}^{\varphi_a}_{\epsilon}(\mathbf{\Psi}^{\sigma})_1\right) \\ &= \tilde{\mathbf{x}}^{\varphi_a}_{\epsilon}(\mathbf{\Psi}^{\sigma})_1\left(1 - \beta^{\mathcal{B}}_{1}\right) + \tilde{\mathbf{x}}^{\varphi_a}_{\epsilon}(\mathbf{\Psi}^{\sigma})_2\left(\beta^{\mathcal{B}}_{1}\right)\end{aligned} \tag{A.4}$$

$$\begin{aligned}\tilde{\mathbf{x}}^{\varphi_b}_{\epsilon}(\mathbf{\Psi}^{\sigma}, \boldsymbol{\mu}^{\sigma}_{\epsilon}) &= \tilde{\mathbf{x}}^{\varphi_b}_{\epsilon}(\mathbf{\Psi}^{\sigma})_1 + \beta^{\mathcal{B}}_{1}\left(\tilde{\mathbf{x}}^{\varphi_b}_{\epsilon}(\mathbf{\Psi}^{\sigma})_2 - \tilde{\mathbf{x}}^{\varphi_b}_{\epsilon}(\mathbf{\Psi}^{\sigma})_1\right) \\ &= \tilde{\mathbf{x}}^{\varphi_b}_{\epsilon}(\mathbf{\Psi}^{\sigma})_1\left(1 - \beta^{\mathcal{B}}_{1}\right) + \tilde{\mathbf{x}}^{\varphi_b}_{\epsilon}(\mathbf{\Psi}^{\sigma})_2\left(\beta^{\mathcal{B}}_{1}\right)\end{aligned} \tag{A.5}$$

Equation (A.6) expresses the system composition $\mathbf{x}^{\sigma}_{\epsilon}$ in terms of the interpolated tie simplex's phase compositions $\tilde{\mathbf{x}}^{\varphi}_{\epsilon}(\mathbf{\Psi}^{\sigma}, \boldsymbol{\mu}^{\sigma}_{\epsilon})$ with the tie simplex barycentric coordinate weights $\boldsymbol{\beta}^{\Delta}$.

$$\begin{aligned}\mathbf{x}^{\sigma}_{\epsilon} =& \tilde{\mathbf{x}}^{\varphi_a}_{\epsilon}(\mathbf{\Psi}^{\sigma}, \boldsymbol{\mu}^{\sigma}_{\epsilon}) + \beta^{\Delta}_{1}\left(\tilde{\mathbf{x}}^{\varphi_b}_{\epsilon}(\mathbf{\Psi}^{\sigma}, \boldsymbol{\mu}^{\sigma}_{\epsilon}) - \tilde{\mathbf{x}}^{\varphi_a}_{\epsilon}(\mathbf{\Psi}^{\sigma}, \boldsymbol{\mu}^{\sigma}_{\epsilon})\right) \\ =& \tilde{\mathbf{x}}^{\varphi_a}_{\epsilon}(\mathbf{\Psi}^{\sigma}, \boldsymbol{\mu}^{\sigma}_{\epsilon})\left(1 - \beta^{\Delta}_{1}\right) + \tilde{\mathbf{x}}^{\varphi_b}_{\epsilon}(\mathbf{\Psi}^{\sigma}, \boldsymbol{\mu}^{\sigma}_{\epsilon})\left(\beta^{\Delta}_{1}\right)\end{aligned} \tag{A.6}$$

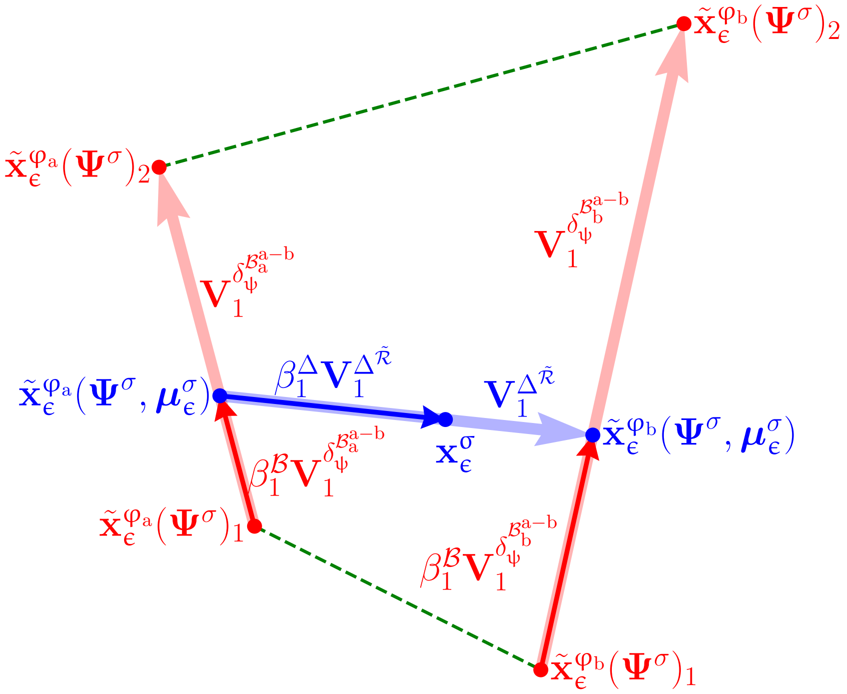

$\tilde{\mathbf{x}}_\epsilon^{\varphi_b}(\mathbf{\Psi}^\sigma)_2$
$\tilde{\mathbf{x}}_\epsilon^{\varphi_a}(\mathbf{\Psi}^\sigma)_2$
$\mathbf{V}_1^{\delta_\psi^{\mathcal{B}_b^{a-b}}}$
$\mathbf{V}_1^{\delta_\psi^{\mathcal{B}_a^{a-b}}}$
$\beta_1^\Delta \mathbf{V}_1^{\Delta^{\tilde{\mathcal{R}}}}$
$\mathbf{V}_1^{\Delta^{\tilde{\mathcal{R}}}}$
$\tilde{\mathbf{x}}_\epsilon^{\varphi_a}(\mathbf{\Psi}^\sigma, \boldsymbol{\mu}_\epsilon^\sigma)$
$\tilde{\mathbf{x}}_\epsilon^{\varphi_b}(\mathbf{\Psi}^\sigma, \boldsymbol{\mu}_\epsilon^\sigma)$
$\beta_1^{\mathcal{B}}\mathbf{V}_1^{\delta_\psi^{\mathcal{B}_a^{a-b}}}$
$\mathbf{x}_\epsilon^\sigma$
$\tilde{\mathbf{x}}_\epsilon^{\varphi_a}(\mathbf{\Psi}^\sigma)_1$
$\beta_1^{\mathcal{B}}\mathbf{V}_1^{\delta_\psi^{\mathcal{B}_b^{a-b}}}$
$\tilde{\mathbf{x}}_\epsilon^{\varphi_b}(\mathbf{\Psi}^\sigma)_1$

Figure A.2: Vectors and barycentric coordinate weights used in an iso-ψ section through a two-phase region cell of a three-component system to describe the system composition $\mathbf{x}_\epsilon^\sigma$ as a function of all the phase compositions $\tilde{\mathbf{x}}_\epsilon^\varphi(\mathbf{\Psi}^\sigma)$ of the iso-ψ tie simplices.

By substituting Equations (A.4) and (A.5) into Equation (A.6) and factorising, the system composition $\mathbf{x}_\epsilon^\sigma$ is expressed as a function of each iso-ψ boundary discretisation segment's phase compositions $\tilde{\mathbf{x}}_\epsilon^\varphi(\mathbf{\Psi}^\sigma)$ and barycentric coordinate weights $\boldsymbol{\beta}^\Delta$ and $\boldsymbol{\beta}^{\mathcal{B}}$, shown in Equation (A.7).

$$\begin{aligned}
\mathbf{x}_\epsilon^\sigma = & \left[\tilde{\mathbf{x}}_\epsilon^{\varphi_a}(\mathbf{\Psi}^\sigma)_1\left(1-\beta_1^{\mathcal{B}}\right)+\tilde{\mathbf{x}}_\epsilon^{\varphi_a}(\mathbf{\Psi}^\sigma)_2\left(\beta_1^{\mathcal{B}}\right)\right]\left(1-\beta_1^\Delta\right)\\
& +\left[\tilde{\mathbf{x}}_\epsilon^{\varphi_b}(\mathbf{\Psi}^\sigma)_1\left(1-\beta_1^{\mathcal{B}}\right)+\tilde{\mathbf{x}}_\epsilon^{\varphi_b}(\mathbf{\Psi}^\sigma)_2\left(\beta_1^{\mathcal{B}}\right)\right]\left(\beta_1^\Delta\right)\\
= & \ \tilde{\mathbf{x}}_\epsilon^{\varphi_a}(\mathbf{\Psi}^\sigma)_1\left(1-\beta_1^{\mathcal{B}}\right)\left(1-\beta_1^\Delta\right)\\
& +\tilde{\mathbf{x}}_\epsilon^{\varphi_a}(\mathbf{\Psi}^\sigma)_2\left(\beta_1^{\mathcal{B}}\right)\left(1-\beta_1^\Delta\right)\\
& +\tilde{\mathbf{x}}_\epsilon^{\varphi_b}(\mathbf{\Psi}^\sigma)_1\left(1-\beta_1^{\mathcal{B}}\right)\left(\beta_1^\Delta\right)\\
& +\tilde{\mathbf{x}}_\epsilon^{\varphi_b}(\mathbf{\Psi}^\sigma)_2\left(\beta_1^{\mathcal{B}}\right)\left(\beta_1^\Delta\right)
\end{aligned} \qquad \text{(A.7)}$$

The factorised expression can be written as a system of non-linear equations, shown in Equation (A.8), with vectors and matrices with sizes as follow:

$\mathbf{x}_\epsilon^\sigma$ **:** $\hat{\epsilon}\times 1 = 3\times 1$ vector of the system composition.

$\mathbf{B}$ **:** $\hat{\epsilon}\times(\hat{\Delta}_\psi^{\mathcal{R}}\times\hat{\varphi}) = 3\times 4$ matrix of the iso-ψ boundary discretisation segment phase compositions.

$\mathbf{w}$ **:** $(\hat{\Delta}_\psi^{\mathcal{R}}\times\hat{\varphi})\times 1 = 4\times 1$ vector of barycentric coordinate weight expressions.

$$\begin{aligned}
\mathbf{x}_\epsilon^\sigma &= \mathbf{B}\mathbf{w}\\
\mathbf{x}_\epsilon^\sigma &= \begin{bmatrix}\tilde{\mathbf{x}}_\epsilon^{\varphi_a}(\mathbf{\Psi}^\sigma)_1\\ \tilde{\mathbf{x}}_\epsilon^{\varphi_a}(\mathbf{\Psi}^\sigma)_2\\ \tilde{\mathbf{x}}_\epsilon^{\varphi_b}(\mathbf{\Psi}^\sigma)_1\\ \tilde{\mathbf{x}}_\epsilon^{\varphi_b}(\mathbf{\Psi}^\sigma)_2\end{bmatrix}^\intercal \begin{bmatrix}\left(1-\beta_1^{\mathcal{B}}\right)\left(1-\beta_1^\Delta\right)\\ \left(\beta_1^{\mathcal{B}}\right)\left(1-\beta_1^\Delta\right)\\ \left(1-\beta_1^{\mathcal{B}}\right)\left(\beta_1^\Delta\right)\\ \left(\beta_1^{\mathcal{B}}\right)\left(\beta_1^\Delta\right)\end{bmatrix}
\end{aligned} \qquad \text{(A.8)}$$

A.3.2 Four-component System

Two-phase Region

Figure A.3 shows a general iso-ψ section through a two-phase region cell of a four-component system. Figure A.4 shows the barycentric coordinate weights used to express the system composition $\mathbf{x}^{\sigma}_{\epsilon}$ in terms of the iso-ψ boundary discretisation segment phase compositions $\tilde{\mathbf{x}}^{\varphi}_{\epsilon}(\boldsymbol{\Psi}^{\sigma})$. Equations (A.9) and (A.10) expresses each of the interpolated tie simplex's phase compositions $\tilde{\mathbf{x}}^{\varphi}_{\epsilon}(\boldsymbol{\Psi}^{\sigma}, \boldsymbol{\mu}^{\sigma}_{\epsilon})$ with iso-ψ boundary discretisation segment barycentric coordinate weights $\boldsymbol{\beta}^{\mathcal{B}}$.

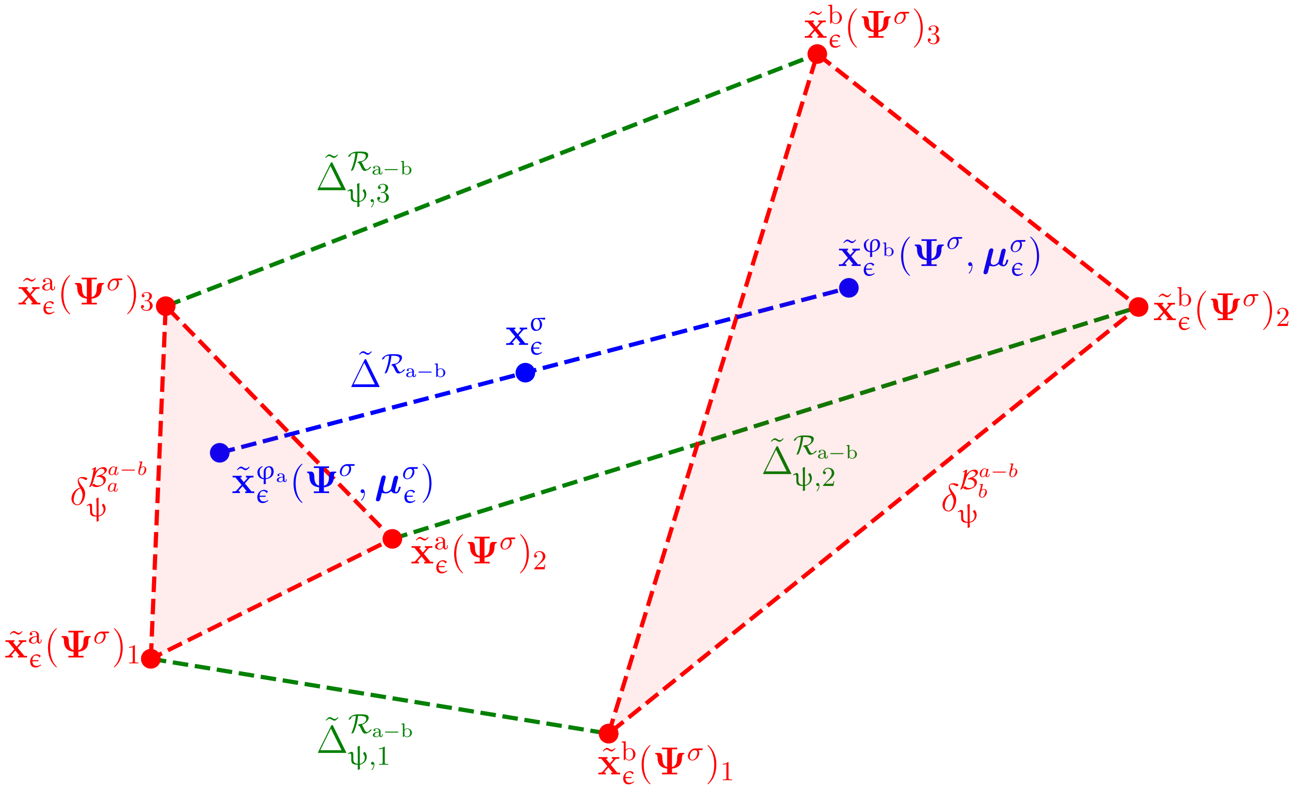

$\tilde{\mathbf{x}}^{\mathrm{b}}_{\epsilon}(\boldsymbol{\Psi}^{\sigma})_3$
$\tilde{\Delta}^{\mathcal{R}_{\mathrm{a-b}}}_{\psi,3}$
$\tilde{\mathbf{x}}^{\varphi_{\mathrm{b}}}_{\epsilon}(\boldsymbol{\Psi}^{\sigma}, \boldsymbol{\mu}^{\sigma}_{\epsilon})$
$\tilde{\mathbf{x}}^{\mathrm{a}}_{\epsilon}(\boldsymbol{\Psi}^{\sigma})_3$
$\mathbf{x}^{\sigma}_{\epsilon}$
$\tilde{\mathbf{x}}^{\mathrm{b}}_{\epsilon}(\boldsymbol{\Psi}^{\sigma})_2$
$\tilde{\Delta}^{\mathcal{R}_{\mathrm{a-b}}}$
$\delta^{\mathcal{B}^{a-b}_{a}}_{\psi}$
$\tilde{\mathbf{x}}^{\varphi_{\mathrm{a}}}_{\epsilon}(\boldsymbol{\Psi}^{\sigma}, \boldsymbol{\mu}^{\sigma}_{\epsilon})$
$\tilde{\Delta}^{\mathcal{R}_{\mathrm{a-b}}}_{\psi,2}$
$\delta^{\mathcal{B}^{a-b}_{b}}_{\psi}$
$\tilde{\mathbf{x}}^{\mathrm{a}}_{\epsilon}(\boldsymbol{\Psi}^{\sigma})_2$
$\tilde{\mathbf{x}}^{\mathrm{a}}_{\epsilon}(\boldsymbol{\Psi}^{\sigma})_1$
$\tilde{\Delta}^{\mathcal{R}_{\mathrm{a-b}}}_{\psi,1}$
$\tilde{\mathbf{x}}^{\mathrm{b}}_{\epsilon}(\boldsymbol{\Psi}^{\sigma})_1$

Figure A.3: Iso-ψ section $\delta^{\mathcal{R}_{\text{a-b}}}_{\psi}$ through a two-phase region cell of a four-component system with iso-ψ tie simplices $\tilde{\Delta}^{\mathcal{R}_{\text{a-b}}}_{\psi}$ and interpolated tie simplex $\tilde{\Delta}^{\mathcal{R}_{\text{a-b}}}$ at the same non-compositional potentials $\boldsymbol{\Psi}^{\sigma}$. Interpolated tie simplex $\tilde{\Delta}^{\mathcal{R}_{\text{a-b}}}$ is at chemical potential $\boldsymbol{\mu}^{\sigma}_{\epsilon}$, perfectly containing $\mathbf{x}^{\sigma}_{\epsilon}$. The four-component system exist in $\mathbb{D}^{\sigma}_{\mathrm{g}} = \hat{\varepsilon} - \hat{\zeta}^{\sigma} - 1 + \hat{\psi} = 3 + \hat{\psi}$, meaning an iso-ψ section is perceived in $\mathbb{D}^{\sigma}_{\mathrm{g}} = 3$.

$$\begin{aligned}\tilde{\mathbf{x}}^{\varphi_{\mathrm{a}}}_{\epsilon}(\boldsymbol{\Psi}^{\sigma}, \boldsymbol{\mu}^{\sigma}_{\epsilon}) &= \tilde{\mathbf{x}}^{\varphi_{\mathrm{a}}}_{\epsilon}(\boldsymbol{\Psi}^{\sigma})_1 + \beta^{\mathcal{B}}_1 \left(\tilde{\mathbf{x}}^{\varphi_{\mathrm{a}}}_{\epsilon}(\boldsymbol{\Psi}^{\sigma})_2 - \tilde{\mathbf{x}}^{\varphi_{\mathrm{a}}}_{\epsilon}(\boldsymbol{\Psi}^{\sigma})_1\right) + \beta^{\mathcal{B}}_2 \left(\tilde{\mathbf{x}}^{\varphi_{\mathrm{a}}}_{\epsilon}(\boldsymbol{\Psi}^{\sigma})_3 - \tilde{\mathbf{x}}^{\varphi_{\mathrm{a}}}_{\epsilon}(\boldsymbol{\Psi}^{\sigma})_1\right) \\ &= \tilde{\mathbf{x}}^{\varphi_{\mathrm{a}}}_{\epsilon}(\boldsymbol{\Psi}^{\sigma})_1 \left(1 - \beta^{\mathcal{B}}_1 - \beta^{\mathcal{B}}_2\right) + \tilde{\mathbf{x}}^{\varphi_{\mathrm{a}}}_{\epsilon}(\boldsymbol{\Psi}^{\sigma})_2 \left(\beta^{\mathcal{B}}_1\right) + \tilde{\mathbf{x}}^{\varphi_{\mathrm{a}}}_{\epsilon}(\boldsymbol{\Psi}^{\sigma})_3 \left(\beta^{\mathcal{B}}_2\right)\end{aligned} \tag{A.9}$$

$$\begin{aligned}\tilde{\mathbf{x}}^{\varphi_{\mathrm{b}}}_{\epsilon}(\boldsymbol{\Psi}^{\sigma}, \boldsymbol{\mu}^{\sigma}_{\epsilon}) &= \tilde{\mathbf{x}}^{\varphi_{\mathrm{b}}}_{\epsilon}(\boldsymbol{\Psi}^{\sigma})_1 + \beta^{\mathcal{B}}_1 \left(\tilde{\mathbf{x}}^{\varphi_{\mathrm{b}}}_{\epsilon}(\boldsymbol{\Psi}^{\sigma})_2 - \tilde{\mathbf{x}}^{\varphi_{\mathrm{b}}}_{\epsilon}(\boldsymbol{\Psi}^{\sigma})_1\right) + \beta^{\mathcal{B}}_2 \left(\tilde{\mathbf{x}}^{\varphi_{\mathrm{b}}}_{\epsilon}(\boldsymbol{\Psi}^{\sigma})_3 - \tilde{\mathbf{x}}^{\varphi_{\mathrm{b}}}_{\epsilon}(\boldsymbol{\Psi}^{\sigma})_1\right) \\ &= \tilde{\mathbf{x}}^{\varphi_{\mathrm{b}}}_{\epsilon}(\boldsymbol{\Psi}^{\sigma})_1 \left(1 - \beta^{\mathcal{B}}_1 - \beta^{\mathcal{B}}_2\right) + \tilde{\mathbf{x}}^{\varphi_{\mathrm{b}}}_{\epsilon}(\boldsymbol{\Psi}^{\sigma})_2 \left(\beta^{\mathcal{B}}_1\right) + \tilde{\mathbf{x}}^{\varphi_{\mathrm{b}}}_{\epsilon}(\boldsymbol{\Psi}^{\sigma})_3 \left(\beta^{\mathcal{B}}_2\right)\end{aligned} \tag{A.10}$$

Equation (A.11) expresses the system composition $\mathbf{x}^{\sigma}_{\epsilon}$ in terms of the interpolated tie simplex's phase compositions $\tilde{\mathbf{x}}^{\varphi}_{\epsilon}(\boldsymbol{\Psi}^{\sigma}, \boldsymbol{\mu}^{\sigma}_{\epsilon})$ with the tie simplex barycentric coordinate weights $\boldsymbol{\beta}^{\Delta}$.

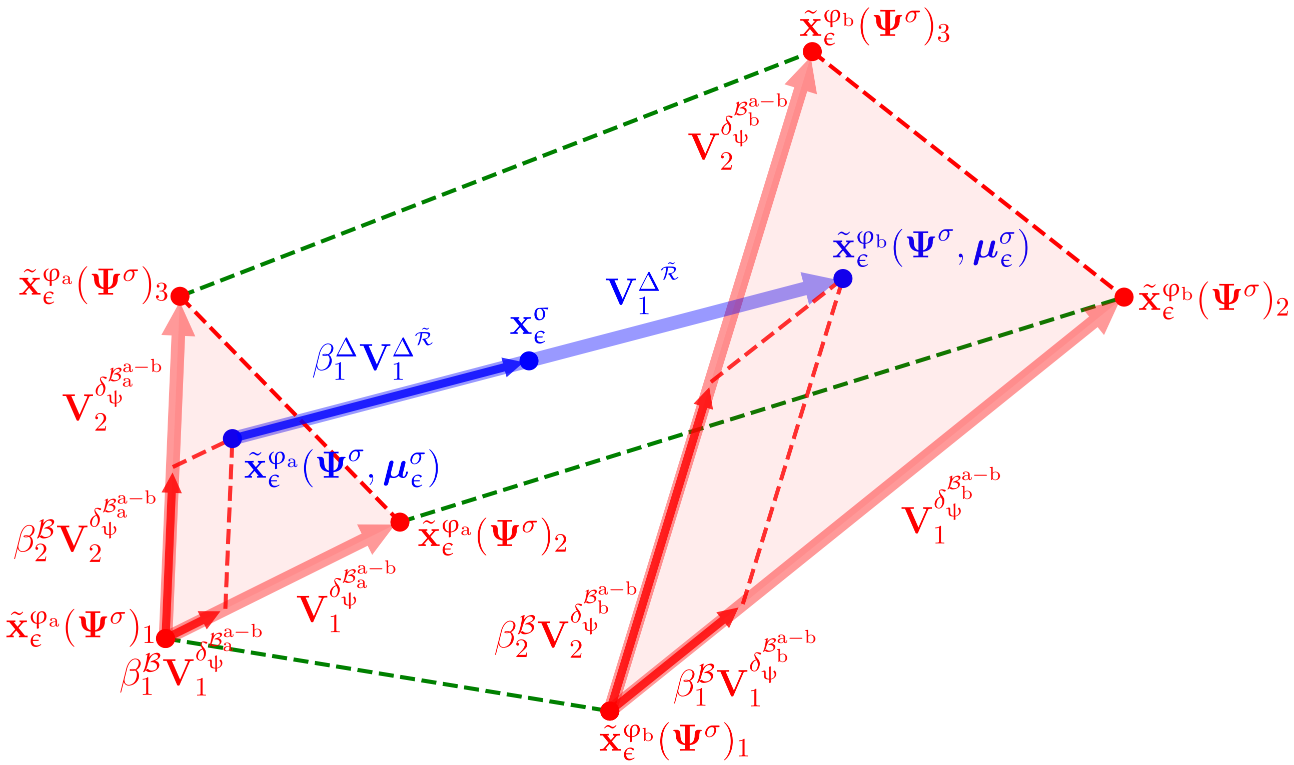

$\tilde{\mathbf{x}}^{\varphi_{\mathrm{b}}}_{\epsilon}(\mathbf{\Psi}^{\sigma})_3$
$\mathbf{V}_2^{\delta_{\psi}^{\mathcal{B}_{\mathrm{b}}^{\mathrm{a-b}}}}$
$\tilde{\mathbf{x}}^{\varphi_{\mathrm{b}}}_{\epsilon}(\mathbf{\Psi}^{\sigma},\boldsymbol{\mu}^{\sigma}_{\epsilon})$
$\tilde{\mathbf{x}}^{\varphi_{\mathrm{a}}}_{\epsilon}(\mathbf{\Psi}^{\sigma})_3$
$\mathbf{V}_1^{\Delta^{\tilde{\mathcal{R}}}}$
$\tilde{\mathbf{x}}^{\varphi_{\mathrm{b}}}_{\epsilon}(\mathbf{\Psi}^{\sigma})_2$
$\mathbf{x}^{\sigma}_{\epsilon}$
$\beta_1^{\Delta}\mathbf{V}_1^{\Delta^{\tilde{\mathcal{R}}}}$
$\mathbf{V}_2^{\delta_{\psi}^{\mathcal{B}_{\mathrm{a}}^{\mathrm{a-b}}}}$
$\tilde{\mathbf{x}}^{\varphi_{\mathrm{a}}}_{\epsilon}(\mathbf{\Psi}^{\sigma},\boldsymbol{\mu}^{\sigma}_{\epsilon})$
$\beta_2^{\mathcal{B}}\mathbf{V}_2^{\delta_{\psi}^{\mathcal{B}_{\mathrm{a}}^{\mathrm{a-b}}}}$
$\tilde{\mathbf{x}}^{\varphi_{\mathrm{a}}}_{\epsilon}(\mathbf{\Psi}^{\sigma})_2$
$\mathbf{V}_1^{\delta_{\psi}^{\mathcal{B}_{\mathrm{b}}^{\mathrm{a-b}}}}$
$\mathbf{V}_1^{\delta_{\psi}^{\mathcal{B}_{\mathrm{a}}^{\mathrm{a-b}}}}$
$\tilde{\mathbf{x}}^{\varphi_{\mathrm{a}}}_{\epsilon}(\mathbf{\Psi}^{\sigma})_1$
$\beta_2^{\mathcal{B}}\mathbf{V}_2^{\delta_{\psi}^{\mathcal{B}_{\mathrm{b}}^{\mathrm{a-b}}}}$
$\beta_1^{\mathcal{B}}\mathbf{V}_1^{\delta_{\psi}^{\mathcal{B}_{\mathrm{a}}^{\mathrm{a-b}}}}$
$\beta_1^{\mathcal{B}}\mathbf{V}_1^{\delta_{\psi}^{\mathcal{B}_{\mathrm{b}}^{\mathrm{a-b}}}}$
$\tilde{\mathbf{x}}^{\varphi_{\mathrm{b}}}_{\epsilon}(\mathbf{\Psi}^{\sigma})_1$

Figure A.4: Vectors and barycentric coordinate weights used in an iso-ψ section through a two-phase region cell of a four-component system to describe the system composition $\mathrm{x}^{\sigma}_{\epsilon}$ as a function of all the phase compositions $\tilde{\mathrm{x}}^{\varphi}_{\epsilon}(\mathbf{\Psi}^{\sigma})$ of the iso-ψ tie simplices.

$$\begin{aligned}\mathrm{x}^{\sigma}_{\epsilon} =& \tilde{\mathrm{x}}^{\varphi_{\mathrm{a}}}_{\epsilon}(\mathbf{\Psi}^{\sigma},\boldsymbol{\mu}^{\sigma}_{\epsilon}) + \beta_1^{\Delta}\left(\tilde{\mathrm{x}}^{\varphi_{\mathrm{b}}}_{\epsilon}(\mathbf{\Psi}^{\sigma},\boldsymbol{\mu}^{\sigma}_{\epsilon}) - \tilde{\mathrm{x}}^{\varphi_{\mathrm{a}}}_{\epsilon}(\mathbf{\Psi}^{\sigma},\boldsymbol{\mu}^{\sigma}_{\epsilon})\right)\\ =& \tilde{\mathrm{x}}^{\varphi_{\mathrm{a}}}_{\epsilon}(\mathbf{\Psi}^{\sigma},\boldsymbol{\mu}^{\sigma}_{\epsilon})\left(1-\beta_1^{\Delta}\right) + \tilde{\mathrm{x}}^{\varphi_{\mathrm{b}}}_{\epsilon}(\mathbf{\Psi}^{\sigma},\boldsymbol{\mu}^{\sigma}_{\epsilon})\left(\beta_1^{\Delta}\right)\end{aligned} \tag{A.11}$$

By substituting Equations (A.9) and (A.10) into Equation (A.11) and factorising, the system composition $\mathrm{x}^{\sigma}_{\epsilon}$ is expressed as a function of each iso-ψ boundary discretisation segment's phase compositions $\tilde{\mathrm{x}}^{\varphi}_{\epsilon}(\mathbf{\Psi}^{\sigma})$ and barycentric coordinate weights $\boldsymbol{\beta}^{\Delta}$ and $\boldsymbol{\beta}^{\mathcal{B}}$, shown in Equation (A.12).

$$\begin{aligned}\mathrm{x}^{\sigma}_{\epsilon} =& \left[\tilde{\mathrm{x}}^{\varphi_{\mathrm{a}}}_{\epsilon}(\mathbf{\Psi}^{\sigma})_1\left(1-\beta_1^{\mathcal{B}}-\beta_2^{\mathcal{B}}\right) + \tilde{\mathrm{x}}^{\varphi_{\mathrm{a}}}_{\epsilon}(\mathbf{\Psi}^{\sigma})_2\left(\beta_1^{\mathcal{B}}\right) + \tilde{\mathrm{x}}^{\varphi_{\mathrm{a}}}_{\epsilon}(\mathbf{\Psi}^{\sigma})_3\left(\beta_2^{\mathcal{B}}\right)\right]\left(1-\beta_1^{\Delta}\right)\\ &+\left[\tilde{\mathrm{x}}^{\varphi_{\mathrm{b}}}_{\epsilon}(\mathbf{\Psi}^{\sigma})_1\left(1-\beta_1^{\mathcal{B}}-\beta_2^{\mathcal{B}}\right) + \tilde{\mathrm{x}}^{\varphi_{\mathrm{b}}}_{\epsilon}(\mathbf{\Psi}^{\sigma})_2\left(\beta_1^{\mathcal{B}}\right) + \tilde{\mathrm{x}}^{\varphi_{\mathrm{b}}}_{\epsilon}(\mathbf{\Psi}^{\sigma})_3\left(\beta_2^{\mathcal{B}}\right)\right]\left(\beta_1^{\Delta}\right)\\ =& \tilde{\mathrm{x}}^{\varphi_{\mathrm{a}}}_{\epsilon}(\mathbf{\Psi}^{\sigma})_1\left(1-\beta_1^{\mathcal{B}}-\beta_2^{\mathcal{B}}\right)\left(1-\beta_1^{\Delta}\right)\\ &+\tilde{\mathrm{x}}^{\varphi_{\mathrm{a}}}_{\epsilon}(\mathbf{\Psi}^{\sigma})_2\left(\beta_1^{\mathcal{B}}\right)\left(1-\beta_1^{\Delta}\right)\\ &+\tilde{\mathrm{x}}^{\varphi_{\mathrm{a}}}_{\epsilon}(\mathbf{\Psi}^{\sigma})_3\left(\beta_2^{\mathcal{B}}\right)\left(1-\beta_1^{\Delta}\right)\\ &+\tilde{\mathrm{x}}^{\varphi_{\mathrm{b}}}_{\epsilon}(\mathbf{\Psi}^{\sigma})_1\left(1-\beta_1^{\mathcal{B}}-\beta_2^{\mathcal{B}}\right)\left(\beta_1^{\Delta}\right)\\ &+\tilde{\mathrm{x}}^{\varphi_{\mathrm{b}}}_{\epsilon}(\mathbf{\Psi}^{\sigma})_2\left(\beta_1^{\mathcal{B}}\right)\left(\beta_1^{\Delta}\right)\\ &+\tilde{\mathrm{x}}^{\varphi_{\mathrm{b}}}_{\epsilon}(\mathbf{\Psi}^{\sigma})_3\left(\beta_2^{\mathcal{B}}\right)\left(\beta_1^{\Delta}\right)\end{aligned} \tag{A.12}$$

The factorised expression can be written as a system of non-linear equations, shown in Equation (A.13), with vectors and matrices with sizes as follow:

$\mathbf{x}_\epsilon^\sigma$ **:** $\hat{\epsilon} \times 1 = 4 \times 1$ vector of the system composition.

$\mathbf{B}$ **:** $\hat{\epsilon} \times (\hat{\Delta}_\psi^{\mathcal{R}} \times \hat{\varphi}) = 4 \times 6$ matrix of the iso-ψ boundary discretisation segment phase compositions.

$\mathbf{w}$ **:** $(\hat{\Delta}_\psi^{\mathcal{R}} \times \hat{\varphi}) \times 1 = 6 \times 1$ vector of barycentric coordinate weight expressions.

$$\mathbf{x}_\epsilon^\sigma = \mathbf{B}\mathbf{w}$$

$$\mathbf{x}_\epsilon^\sigma = \begin{bmatrix} \tilde{\mathbf{x}}_\epsilon^{\varphi_a}(\boldsymbol{\Psi}^\sigma)_1 \\ \tilde{\mathbf{x}}_\epsilon^{\varphi_a}(\boldsymbol{\Psi}^\sigma)_2 \\ \tilde{\mathbf{x}}_\epsilon^{\varphi_a}(\boldsymbol{\Psi}^\sigma)_3 \\ \hdashline \tilde{\mathbf{x}}_\epsilon^{\varphi_b}(\boldsymbol{\Psi}^\sigma)_1 \\ \tilde{\mathbf{x}}_\epsilon^{\varphi_b}(\boldsymbol{\Psi}^\sigma)_2 \\ \tilde{\mathbf{x}}_\epsilon^{\varphi_b}(\boldsymbol{\Psi}^\sigma)_3 \end{bmatrix}^\intercal \begin{bmatrix} \left(1 - \beta_1^{\mathcal{B}} - \beta_2^{\mathcal{B}}\right) \left(1 - \beta_1^{\Delta}\right) \\ \left(\beta_1^{\mathcal{B}}\right) \left(1 - \beta_1^{\Delta}\right) \\ \left(\beta_2^{\mathcal{B}}\right) \left(1 - \beta_1^{\Delta}\right) \\ \hdashline \left(1 - \beta_1^{\mathcal{B}} - \beta_2^{\mathcal{B}}\right) \left(\beta_1^{\Delta}\right) \\ \left(\beta_1^{\mathcal{B}}\right) \left(\beta_1^{\Delta}\right) \\ \left(\beta_2^{\mathcal{B}}\right) \left(\beta_1^{\Delta}\right) \end{bmatrix} \tag{A.13}$$

Three-phase Region

Figure A.5 shows a general iso-ψ section through a three-phase region cell of a four-component system. Figure A.6 shows the barycentric coordinate weights used to express the system composition $\mathbf{x}_\epsilon^\sigma$ in terms of the iso-ψ boundary discretisation segment phase compositions $\tilde{\mathbf{x}}_\epsilon^{\varphi}(\boldsymbol{\Psi}^\sigma)$. Equations (A.14) to (A.16) expresses each of the interpolated tie simplex's phase compositions $\tilde{\mathbf{x}}_\epsilon^{\varphi}(\boldsymbol{\Psi}^\sigma, \boldsymbol{\mu}_\epsilon^\sigma)$ with iso-ψ boundary discretisation segment barycentric coordinate weights $\boldsymbol{\beta}^{\mathcal{B}}$.

$$\begin{aligned} \tilde{\mathbf{x}}_\epsilon^{\varphi_a}(\boldsymbol{\Psi}^\sigma, \boldsymbol{\mu}_\epsilon^\sigma) &= \tilde{\mathbf{x}}_\epsilon^{\varphi_a}(\boldsymbol{\Psi}^\sigma)_1 + \beta_1^{\mathcal{B}} \left(\tilde{\mathbf{x}}_\epsilon^{\varphi_a}(\boldsymbol{\Psi}^\sigma)_2 - \tilde{\mathbf{x}}_\epsilon^{\varphi_a}(\boldsymbol{\Psi}^\sigma)_1\right) \\ &= \tilde{\mathbf{x}}_\epsilon^{\varphi_a}(\boldsymbol{\Psi}^\sigma)_1 \left(1 - \beta_1^{\mathcal{B}}\right) + \tilde{\mathbf{x}}_\epsilon^{\varphi_a}(\boldsymbol{\Psi}^\sigma)_2 \left(\beta_1^{\mathcal{B}}\right) \end{aligned} \tag{A.14}$$

$$\begin{aligned} \tilde{\mathbf{x}}_\epsilon^{\varphi_b}(\boldsymbol{\Psi}^\sigma, \boldsymbol{\mu}_\epsilon^\sigma) &= \tilde{\mathbf{x}}_\epsilon^{\varphi_b}(\boldsymbol{\Psi}^\sigma)_1 + \beta_1^{\mathcal{B}} \left(\tilde{\mathbf{x}}_\epsilon^{\varphi_b}(\boldsymbol{\Psi}^\sigma)_2 - \tilde{\mathbf{x}}_\epsilon^{\varphi_b}(\boldsymbol{\Psi}^\sigma)_1\right) \\ &= \tilde{\mathbf{x}}_\epsilon^{\varphi_b}(\boldsymbol{\Psi}^\sigma)_1 \left(1 - \beta_1^{\mathcal{B}}\right) + \tilde{\mathbf{x}}_\epsilon^{\varphi_b}(\boldsymbol{\Psi}^\sigma)_2 \left(\beta_1^{\mathcal{B}}\right) \end{aligned} \tag{A.15}$$

$$\begin{aligned} \tilde{\mathbf{x}}_\epsilon^{\varphi_c}(\boldsymbol{\Psi}^\sigma, \boldsymbol{\mu}_\epsilon^\sigma) &= \tilde{\mathbf{x}}_\epsilon^{\varphi_c}(\boldsymbol{\Psi}^\sigma)_1 + \beta_1^{\mathcal{B}} \left(\tilde{\mathbf{x}}_\epsilon^{\varphi_c}(\boldsymbol{\Psi}^\sigma)_2 - \tilde{\mathbf{x}}_\epsilon^{\varphi_c}(\boldsymbol{\Psi}^\sigma)_1\right) \\ &= \tilde{\mathbf{x}}_\epsilon^{\varphi_c}(\boldsymbol{\Psi}^\sigma)_1 \left(1 - \beta_1^{\mathcal{B}}\right) + \tilde{\mathbf{x}}_\epsilon^{\varphi_c}(\boldsymbol{\Psi}^\sigma)_2 \left(\beta_1^{\mathcal{B}}\right) \end{aligned} \tag{A.16}$$

Equation (A.17) expresses the system composition $\mathbf{x}_\epsilon^\sigma$ in terms of the interpolated tie simplex's phase compositions $\tilde{\mathbf{x}}_\epsilon^{\varphi}(\boldsymbol{\Psi}^\sigma, \boldsymbol{\mu}_\epsilon^\sigma)$ with the tie simplex barycentric coordinate weights $\boldsymbol{\beta}^{\Delta}$.

$$\begin{aligned} \mathbf{x}_\epsilon^\sigma =& \tilde{\mathbf{x}}_\epsilon^{\varphi_a}(\boldsymbol{\Psi}^\sigma, \boldsymbol{\mu}_\epsilon^\sigma) + \beta_1^{\Delta} \left(\tilde{\mathbf{x}}_\epsilon^{\varphi_b}(\boldsymbol{\Psi}^\sigma, \boldsymbol{\mu}_\epsilon^\sigma) - \tilde{\mathbf{x}}_\epsilon^{\varphi_a}(\boldsymbol{\Psi}^\sigma, \boldsymbol{\mu}_\epsilon^\sigma)\right) + \beta_2^{\Delta} \left(\tilde{\mathbf{x}}_\epsilon^{\varphi_c}(\boldsymbol{\Psi}^\sigma, \boldsymbol{\mu}_\epsilon^\sigma) - \tilde{\mathbf{x}}_\epsilon^{\varphi_a}(\boldsymbol{\Psi}^\sigma, \boldsymbol{\mu}_\epsilon^\sigma)\right) \\ =& \tilde{\mathbf{x}}_\epsilon^{\varphi_a}(\boldsymbol{\Psi}^\sigma, \boldsymbol{\mu}_\epsilon^\sigma) \left(1 - \beta_1^{\Delta} - \beta_2^{\Delta}\right) + \tilde{\mathbf{x}}_\epsilon^{\varphi_b}(\boldsymbol{\Psi}^\sigma, \boldsymbol{\mu}_\epsilon^\sigma) \left(\beta_1^{\Delta}\right) + \tilde{\mathbf{x}}_\epsilon^{\varphi_c}(\boldsymbol{\Psi}^\sigma, \boldsymbol{\mu}_\epsilon^\sigma) \left(\beta_2^{\Delta}\right) \end{aligned} \tag{A.17}$$

By substituting Equations (A.14) to (A.16) into Equation (A.17) and factorising, the system composition $\mathbf{x}_\epsilon^\sigma$ is expressed as a function of each iso-ψ boundary discretisation segment's phase compositions $\tilde{\mathbf{x}}_\epsilon^{\varphi}(\boldsymbol{\Psi}^\sigma)$ and barycentric coordinate weights $\boldsymbol{\beta}^{\Delta}$ and $\boldsymbol{\beta}^{\mathcal{B}}$, shown in Equation (A.18).

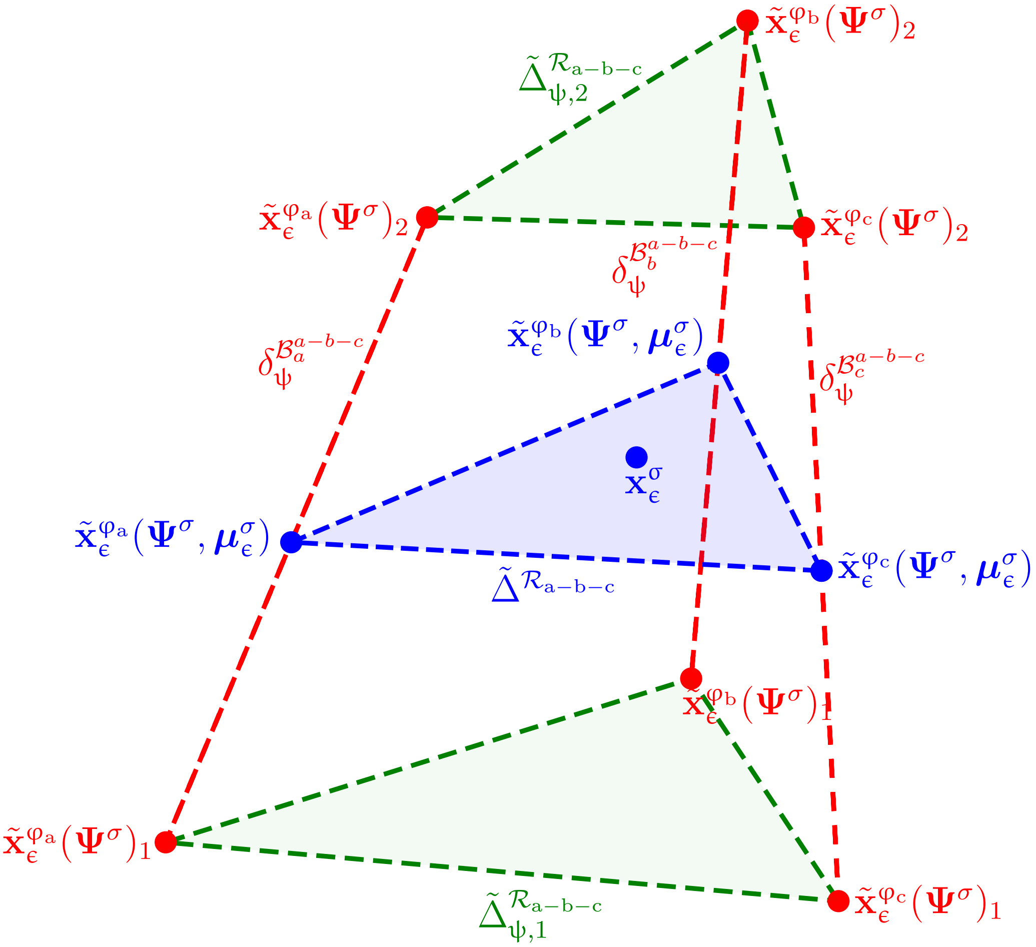

$\tilde{\mathbf{x}}_{\epsilon}^{\varphi_{\mathrm{b}}}(\mathbf{\Psi}^{\sigma})_2$
$\tilde{\Delta}_{\psi,2}^{\mathcal{R}_{\mathrm{a-b-c}}}$
$\tilde{\mathbf{x}}_{\epsilon}^{\varphi_{\mathrm{a}}}(\mathbf{\Psi}^{\sigma})_2$
$\tilde{\mathbf{x}}_{\epsilon}^{\varphi_{\mathrm{c}}}(\mathbf{\Psi}^{\sigma})_2$
$\delta_{\psi}^{\mathcal{B}_b^{a-b-c}}$
$\tilde{\mathbf{x}}_{\epsilon}^{\varphi_{\mathrm{b}}}(\mathbf{\Psi}^{\sigma}, \boldsymbol{\mu}_{\epsilon}^{\sigma})$
$\delta_{\psi}^{\mathcal{B}_a^{a-b-c}}$
$\delta_{\psi}^{\mathcal{B}_c^{a-b-c}}$
$\mathbf{x}_{\epsilon}^{\sigma}$
$\tilde{\mathbf{x}}_{\epsilon}^{\varphi_{\mathrm{a}}}(\mathbf{\Psi}^{\sigma}, \boldsymbol{\mu}_{\epsilon}^{\sigma})$
$\tilde{\mathbf{x}}_{\epsilon}^{\varphi_{\mathrm{c}}}(\mathbf{\Psi}^{\sigma}, \boldsymbol{\mu}_{\epsilon}^{\sigma})$
$\tilde{\Delta}^{\mathcal{R}_{\mathrm{a-b-c}}}$
$\tilde{\mathbf{x}}_{\epsilon}^{\varphi_{\mathrm{b}}}(\mathbf{\Psi}^{\sigma})_1$
$\tilde{\mathbf{x}}_{\epsilon}^{\varphi_{\mathrm{a}}}(\mathbf{\Psi}^{\sigma})_1$
$\tilde{\mathbf{x}}_{\epsilon}^{\varphi_{\mathrm{c}}}(\mathbf{\Psi}^{\sigma})_1$
$\tilde{\Delta}_{\psi,1}^{\mathcal{R}_{\mathrm{a-b-c}}}$

Figure A.5: Iso-ψ section $\delta_{\psi}^{\mathcal{R}_{\text{a-b-c}}}$ through a three-phase region cell of a four-component system with iso-ψ tie simplices $\tilde{\Delta}_{\psi}^{\mathcal{R}_{\text{a-b-c}}}$ and interpolated tie simplex $\tilde{\Delta}^{\mathcal{R}_{\text{a-b-c}}}$ at the same non-compositional potentials $\mathbf{\Psi}^{\sigma}$. Interpolated tie simplex $\tilde{\Delta}^{\mathcal{R}_{\text{a-b-c}}}$ is at chemical potential $\boldsymbol{\mu}_{\epsilon}^{\sigma}$, perfectly containing $\mathbf{x}_{\epsilon}^{\sigma}$. The four-component system exist in $\mathbb{D}_{\mathrm{g}}^{\sigma} = \hat{\varepsilon} - \hat{\zeta}^{\sigma} - 1 + \hat{\psi} = 3 + \hat{\psi}$, meaning an iso-ψ section is perceived in $\mathbb{D}_{\mathrm{g}}^{\sigma} = 3$.

$$\begin{aligned}
\mathbf{x}_{\epsilon}^{\sigma} = \; & \left[\tilde{\mathbf{x}}_{\epsilon}^{\varphi_{\mathrm{a}}}(\mathbf{\Psi}^{\sigma})_1 \left(1-\beta_1^{\mathcal{B}}\right) + \tilde{\mathbf{x}}_{\epsilon}^{\varphi_{\mathrm{a}}}(\mathbf{\Psi}^{\sigma})_2 \left(\beta_1^{\mathcal{B}}\right)\right] \left(1-\beta_1^{\Delta}-\beta_2^{\Delta}\right) \\
& + \left[\tilde{\mathbf{x}}_{\epsilon}^{\varphi_{\mathrm{b}}}(\mathbf{\Psi}^{\sigma})_1 \left(1-\beta_1^{\mathcal{B}}\right) + \tilde{\mathbf{x}}_{\epsilon}^{\varphi_{\mathrm{b}}}(\mathbf{\Psi}^{\sigma})_2 \left(\beta_1^{\mathcal{B}}\right)\right] \left(\beta_1^{\Delta}\right) \\
& + \left[\tilde{\mathbf{x}}_{\epsilon}^{\varphi_{\mathrm{c}}}(\mathbf{\Psi}^{\sigma})_1 \left(1-\beta_1^{\mathcal{B}}\right) + \tilde{\mathbf{x}}_{\epsilon}^{\varphi_{\mathrm{c}}}(\mathbf{\Psi}^{\sigma})_2 \left(\beta_1^{\mathcal{B}}\right)\right] \left(\beta_2^{\Delta}\right) \\
= \; & \tilde{\mathbf{x}}_{\epsilon}^{\varphi_{\mathrm{a}}}(\mathbf{\Psi}^{\sigma})_1 \left(1-\beta_1^{\mathcal{B}}\right) \left(1-\beta_1^{\Delta}-\beta_2^{\Delta}\right) \\
& + \tilde{\mathbf{x}}_{\epsilon}^{\varphi_{\mathrm{a}}}(\mathbf{\Psi}^{\sigma})_2 \left(\beta_1^{\mathcal{B}}\right) \left(1-\beta_1^{\Delta}-\beta_2^{\Delta}\right) \\
& + \tilde{\mathbf{x}}_{\epsilon}^{\varphi_{\mathrm{b}}}(\mathbf{\Psi}^{\sigma})_1 \left(1-\beta_1^{\mathcal{B}}\right) \left(\beta_1^{\Delta}\right) \\
& + \tilde{\mathbf{x}}_{\epsilon}^{\varphi_{\mathrm{b}}}(\mathbf{\Psi}^{\sigma})_2 \left(\beta_1^{\mathcal{B}}\right) \left(\beta_1^{\Delta}\right) \\
& + \tilde{\mathbf{x}}_{\epsilon}^{\varphi_{\mathrm{c}}}(\mathbf{\Psi}^{\sigma})_1 \left(1-\beta_1^{\mathcal{B}}\right) \left(\beta_2^{\Delta}\right) \\
& + \tilde{\mathbf{x}}_{\epsilon}^{\varphi_{\mathrm{c}}}(\mathbf{\Psi}^{\sigma})_2 \left(\beta_1^{\mathcal{B}}\right) \left(\beta_2^{\Delta}\right)
\end{aligned} \tag{A.18}$$

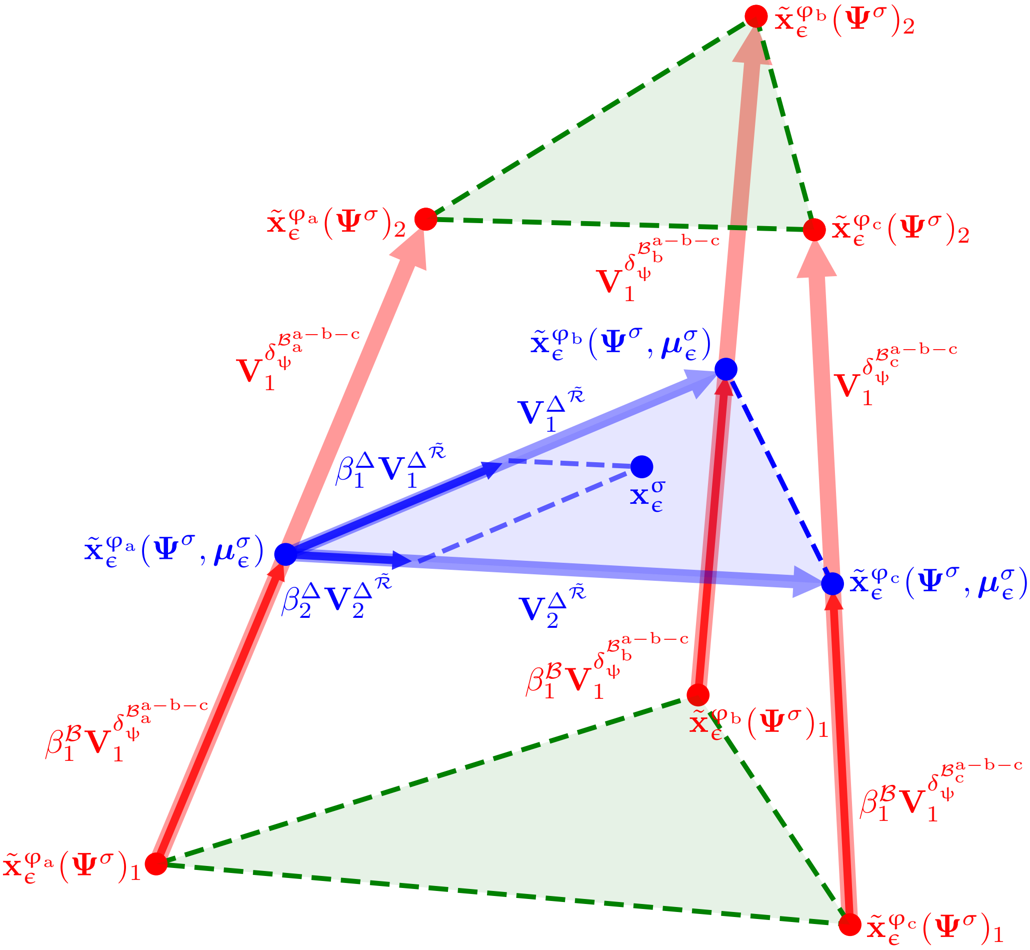

$\tilde{\mathbf{x}}_\epsilon^{\varphi_b}(\boldsymbol{\Psi}^\sigma)_2$
$\tilde{\mathbf{x}}_\epsilon^{\varphi_a}(\boldsymbol{\Psi}^\sigma)_2$
$\tilde{\mathbf{x}}_\epsilon^{\varphi_c}(\boldsymbol{\Psi}^\sigma)_2$
$\mathbf{V}_1^{\delta_\psi^{\mathcal{B}_b^{a-b-c}}}$
$\mathbf{V}_1^{\delta_\psi^{\mathcal{B}_a^{a-b-c}}}$
$\tilde{\mathbf{x}}_\epsilon^{\varphi_b}(\boldsymbol{\Psi}^\sigma, \boldsymbol{\mu}_\epsilon^\sigma)$
$\mathbf{V}_1^{\delta_\psi^{\mathcal{B}_c^{a-b-c}}}$
$\mathbf{V}_1^{\Delta^{\tilde{\mathcal{R}}}}$
$\beta_1^\Delta \mathbf{V}_1^{\Delta^{\tilde{\mathcal{R}}}}$
$\mathbf{x}_\epsilon^\sigma$
$\tilde{\mathbf{x}}_\epsilon^{\varphi_a}(\boldsymbol{\Psi}^\sigma, \boldsymbol{\mu}_\epsilon^\sigma)$
$\tilde{\mathbf{x}}_\epsilon^{\varphi_c}(\boldsymbol{\Psi}^\sigma, \boldsymbol{\mu}_\epsilon^\sigma)$
$\beta_2^\Delta \mathbf{V}_2^{\Delta^{\tilde{\mathcal{R}}}}$
$\mathbf{V}_2^{\Delta^{\tilde{\mathcal{R}}}}$
$\beta_1^{\mathcal{B}} \mathbf{V}_1^{\delta_\psi^{\mathcal{B}_b^{a-b-c}}}$
$\tilde{\mathbf{x}}_\epsilon^{\varphi_b}(\boldsymbol{\Psi}^\sigma)_1$
$\beta_1^{\mathcal{B}} \mathbf{V}_1^{\delta_\psi^{\mathcal{B}_a^{a-b-c}}}$
$\beta_1^{\mathcal{B}} \mathbf{V}_1^{\delta_\psi^{\mathcal{B}_c^{a-b-c}}}$
$\tilde{\mathbf{x}}_\epsilon^{\varphi_a}(\boldsymbol{\Psi}^\sigma)_1$
$\tilde{\mathbf{x}}_\epsilon^{\varphi_c}(\boldsymbol{\Psi}^\sigma)_1$

Figure A.6: Vectors and barycentric coordinate weights used in an iso-ψ section through a three-phase region cell of a four-component system to describe the system composition $\mathbf{x}_\epsilon^\sigma$ as a function of all the phase compositions $\tilde{\mathbf{x}}_\epsilon^\varphi(\boldsymbol{\Psi}^\sigma)$ of the iso-ψ tie simplices.

The factorised expression can be written as a system of non-linear equations, shown in Equation (A.19), with vectors and matrices with sizes as follow:

$\mathbf{x}_\epsilon^\sigma$ **:** $\hat{\epsilon} \times 1 = 4 \times 1$ vector of the system composition.

$\mathbf{B}$ **:** $\hat{\epsilon} \times (\hat{\Delta}_\psi^{\mathcal{R}} \times \hat{\varphi}) = 4 \times 6$ matrix of the iso-ψ boundary discretisation segment phase compositions.

$\mathbf{w}$ **:** $(\hat{\Delta}_\psi^{\mathcal{R}} \times \hat{\varphi}) \times 1 = 6 \times 1$ vector of barycentric coordinate weight expressions.

$$\mathbf{x}_\epsilon^\sigma = \mathbf{B}\mathbf{w}$$

$$\mathbf{x}_\epsilon^\sigma = \begin{bmatrix} \tilde{\mathbf{x}}_\epsilon^{\varphi_a}(\boldsymbol{\Psi}^\sigma)_1 \\ \tilde{\mathbf{x}}_\epsilon^{\varphi_a}(\boldsymbol{\Psi}^\sigma)_2 \\ \hdashline \tilde{\mathbf{x}}_\epsilon^{\varphi_b}(\boldsymbol{\Psi}^\sigma)_1 \\ \tilde{\mathbf{x}}_\epsilon^{\varphi_b}(\boldsymbol{\Psi}^\sigma)_2 \\ \hdashline \tilde{\mathbf{x}}_\epsilon^{\varphi_c}(\boldsymbol{\Psi}^\sigma)_1 \\ \tilde{\mathbf{x}}_\epsilon^{\varphi_c}(\boldsymbol{\Psi}^\sigma)_2 \end{bmatrix}^{\mathsf{T}} \begin{bmatrix} \left(1-\beta_1^{\mathcal{B}}\right)\left(1-\beta_1^\Delta-\beta_2^\Delta\right) \\ \left(\beta_1^{\mathcal{B}}\right)\left(1-\beta_1^\Delta-\beta_2^\Delta\right) \\ \hdashline \left(1-\beta_1^{\mathcal{B}}\right)\left(\beta_1^\Delta\right) \\ \left(\beta_1^{\mathcal{B}}\right)\left(\beta_1^\Delta\right) \\ \hdashline \left(1-\beta_1^{\mathcal{B}}\right)\left(\beta_2^\Delta\right) \\ \left(\beta_1^{\mathcal{B}}\right)\left(\beta_2^\Delta\right) \end{bmatrix} \tag{A.19}$$

A.3.3 Generic System

From the three investigated cases, it can be seen that an expression of the system composition $\mathbf{x}_{\epsilon}^{\sigma}$ as a function of each iso-ψ boundary discretisation segment's phase compositions $\tilde{\mathbf{x}}_{\epsilon}^{\varphi}(\boldsymbol{\Psi}^{\sigma})$ and barycentric coordinate weights $\boldsymbol{\beta}^{\Delta}$ and $\boldsymbol{\beta}^{\mathcal{B}}$ can be written as a system of non-linear equations, as shown in Equation (A.20).

Matrix $\mathbf{B}$ is created by listing the $\hat{\Delta}_{\psi}^{\mathcal{R}}$ number of phase compositions on an iso-ψ boundary discretisation segment $\delta_{\psi}^{\mathcal{B}_{\varphi}}$ of each of the $\hat{\varphi}$ number of phases, one phase at a time, as described by Equation (A.21). Matrix $\mathbf{B}$ therefore has $\hat{\Delta}_{\psi}^{\mathcal{R}} \times \hat{\varphi}$ columns and depending on the number of system components, $\hat{\epsilon}$ number of rows.

After factorising an expression where barycentric coordinate weights are used, the origin vertex always has a product with a weight equal to the difference between unity and the sum of all barycentric coordinate weights; $1 - \Sigma\boldsymbol{\beta}$. The remaining vertices only has a product with a single barycentric coordinate weight; the weight between itself and the origin vertex. This can be seen in the way vector $\mathbf{w}$ is created with Equation (A.22). The first portion of $\mathbf{w}$ is associated with the first phase composition of the interpolated tie simplex $\tilde{\mathbf{x}}_{\epsilon}^{\varphi_1}(\boldsymbol{\Psi}^{\sigma}, \boldsymbol{\mu}_{\epsilon}^{\sigma})$, therefore all the entries in this portion contain a tie simplex barycentric coordinate weight equal to $(1 - \Sigma\boldsymbol{\beta}^{\Delta})$. The portions that follow contain one of the $\hat{\varphi} - 1$ number of tie simplex barycentric coordinate weights β_1^{Δ} to $\beta_{\hat{\varphi}-1}^{\Delta}$. Each of the interpolated tie simplex's phase compositions $\tilde{\mathbf{x}}_{\epsilon}^{\varphi}(\boldsymbol{\Psi}^{\sigma}, \boldsymbol{\mu}_{\epsilon}^{\sigma})$ are described by the same iso-ψ boundary discretisation segment barycentric coordinate weights $\boldsymbol{\beta}^{\mathcal{B}}$. Therefore, in each portion of vector $\mathbf{w}$ associated with a phase φ, the first entry contains an iso-ψ boundary discretisation segment barycentric coordinate weight equal to $(1 - \Sigma\boldsymbol{\beta}^{\mathcal{B}})$. The entries that follow contain one of the $\hat{\Delta}_{\psi}^{\mathcal{R}} - 1$ number of iso-ψ boundary discretisation segment barycentric coordinate weights $\beta_1^{\mathcal{B}}$ to $\beta_{\hat{\Delta}_{\psi}^{\mathcal{R}}-1}^{\mathcal{B}}$. Vector $\mathbf{w}$ therefore has $\hat{\Delta}_{\psi}^{\mathcal{R}} \times \hat{\varphi}$ rows.

$\mathbf{x}_{\epsilon}^{\sigma}$ **:** $\hat{\epsilon} \times 1$ vector of the system composition.

$\mathbf{B}$ **:** $\hat{\epsilon} \times (\hat{\Delta}_{\psi}^{\mathcal{R}} \times \hat{\varphi})$ matrix of the iso-ψ boundary discretisation segment phase compositions.

$\mathbf{w}$ **:** $(\hat{\Delta}_{\psi}^{\mathcal{R}} \times \hat{\varphi}) \times 1$ vector of barycentric coordinate weight expressions.

$$\mathbf{x}_{\epsilon}^{\sigma} = \mathbf{B}\mathbf{w} \tag{A.20}$$

$$\mathbf{B} = \begin{bmatrix} \tilde{\mathbf{x}}_{\epsilon_{(\psi)}}^{\mathcal{B}_1} \\ \vdots \\ \tilde{\mathbf{x}}_{\epsilon_{(\psi)}}^{\mathcal{B}_{\hat{\varphi}}} \end{bmatrix}^{\mathsf{T}} \quad \text{where} \quad \tilde{\mathbf{x}}_{\epsilon_{(\psi)}}^{\mathcal{B}_{i}} = \begin{bmatrix} \tilde{\mathbf{x}}_{\epsilon}^{\varphi_i}(\boldsymbol{\Psi}^{\sigma})_1 \\ \vdots \\ \tilde{\mathbf{x}}_{\epsilon}^{\varphi_i}(\boldsymbol{\Psi}^{\sigma})_{\hat{\Delta}_{\psi}^{\mathcal{R}}} \end{bmatrix} \tag{A.21}$$

$$\mathbf{w} = \left[\begin{array}{c} (1-\Sigma\boldsymbol{\beta}^{\mathcal{B}})(1-\Sigma\boldsymbol{\beta}^{\Delta}) \\ (\beta^{\mathcal{B}}_{1})(1-\Sigma\boldsymbol{\beta}^{\Delta}) \\ \vdots \\ (\beta^{\mathcal{B}}_{\hat{\Delta}^{\mathcal{R}}_{\psi}-1})(1-\Sigma\boldsymbol{\beta}^{\Delta}) \\ \hdashline (1-\Sigma\boldsymbol{\beta}^{\mathcal{B}})(\beta^{\Delta}_{1}) \\ (\beta^{\mathcal{B}}_{1})(\beta^{\Delta}_{1}) \\ \vdots \\ (\beta^{\mathcal{B}}_{\hat{\Delta}^{\mathcal{R}}_{\psi}-1})(\beta^{\Delta}_{1}) \\ \hdashline \vdots \\ \hdashline (1-\Sigma\boldsymbol{\beta}^{\mathcal{B}})(\beta^{\Delta}_{\hat{\varphi}-1}) \\ (\beta^{\mathcal{B}}_{1})(\beta^{\Delta}_{\hat{\varphi}-1}) \\ \vdots \\ (\beta^{\mathcal{B}}_{\hat{\Delta}^{\mathcal{R}}_{\psi}-1})(\beta^{\Delta}_{\hat{\varphi}-1}) \end{array}\right] \tag{A.22}$$

$$\text{where} \quad \Sigma\boldsymbol{\beta}^{\mathcal{B}} = \sum_{\mathrm{j}=1}^{\hat{\Delta}^{\mathcal{R}}_{\psi}-1} \beta^{\mathcal{B}}_{\mathrm{j}} : \beta^{\mathcal{B}}_{\mathrm{j}} \in \mathbb{R}[0,1]$$

$$\text{and} \quad \Sigma\boldsymbol{\beta}^{\Delta} = \sum_{\mathrm{i}=1}^{\hat{\varphi}-1} \beta^{\Delta}_{\mathrm{i}} : \beta^{\Delta}_{\mathrm{i}} \in \mathbb{R}[0,1]$$

Appendix B

Algorithm Performance Test Result Distributions

B.1 Two-component Systems

Distribution plots of the performance test results of all tested two-component systems are presented hereafter.

B.1.1 Al−Cu

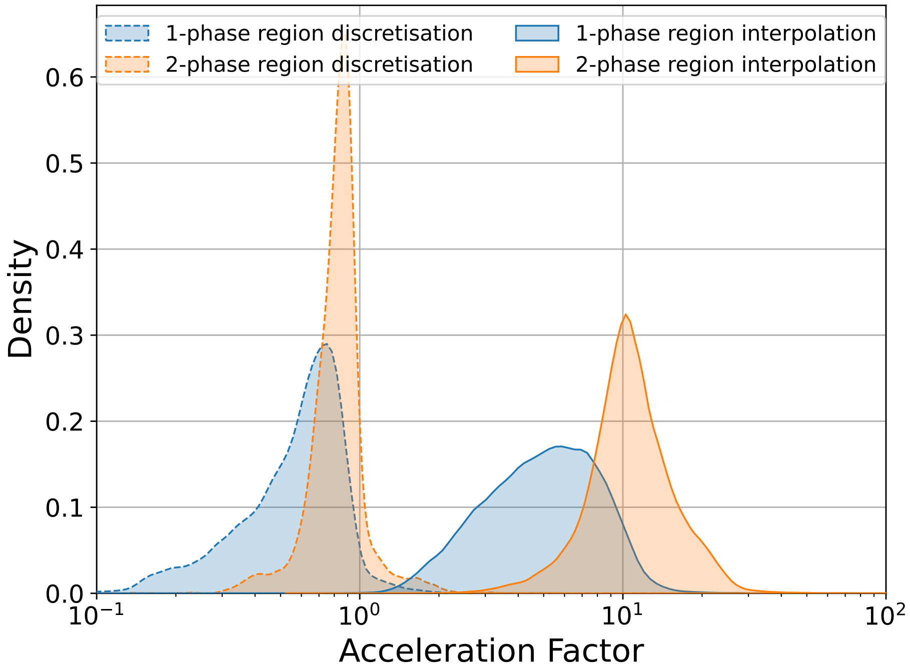
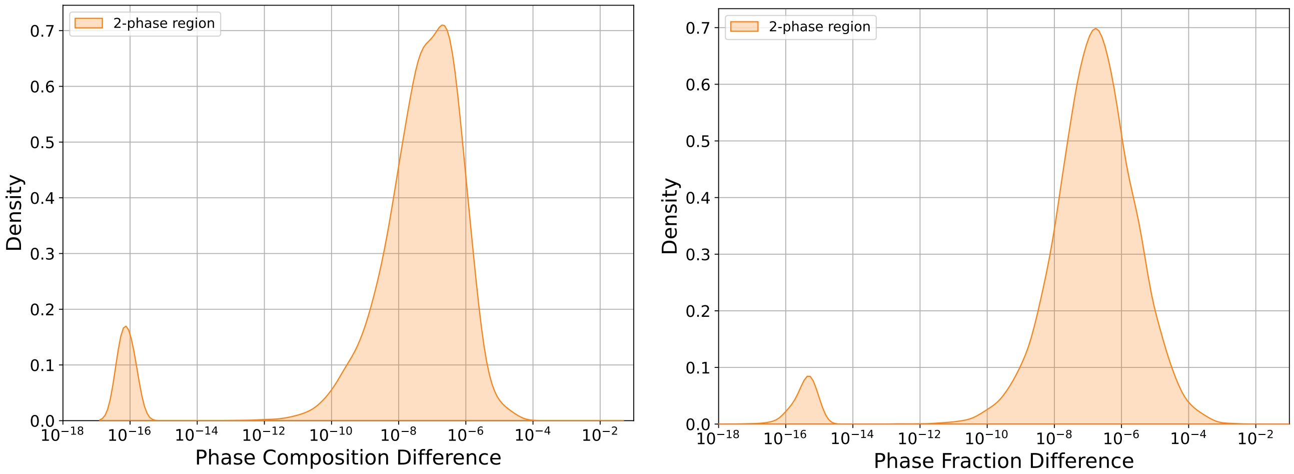


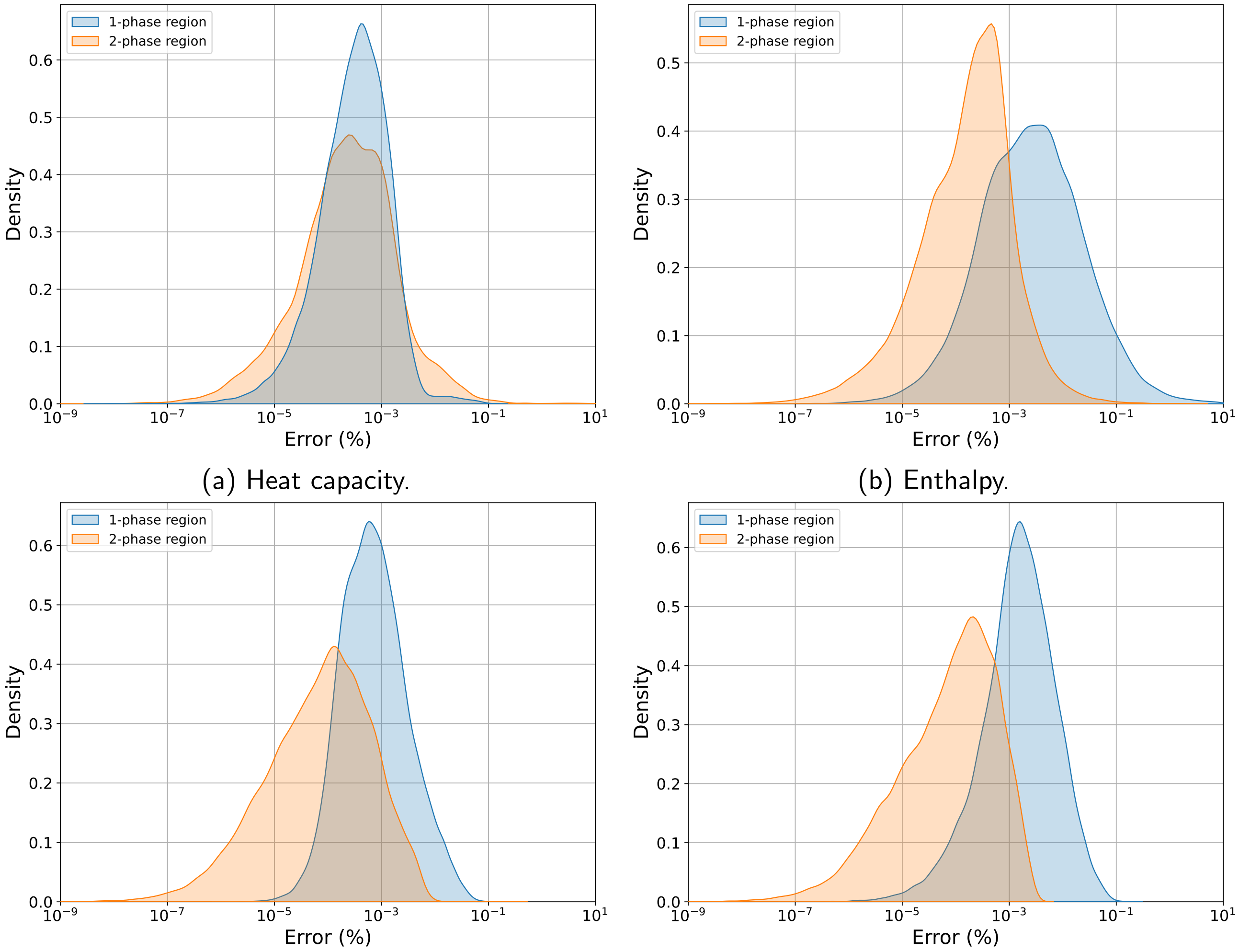
Figure B.1: Distribution of acceleration factors for the Al−Cu system.



(a) Phase composition.

(b) Phase fraction.

Figure B.2: Distribution of phase composition and phase fraction difference for the Al−Cu system.



(a) Heat capacity.

(b) Enthalpy.

(c) Entropy.

(d) Gibbs energy.

Figure B.3: Distribution of physical and thermochemical property errors for the Al−Cu system.

B.1.2 Al−Mn

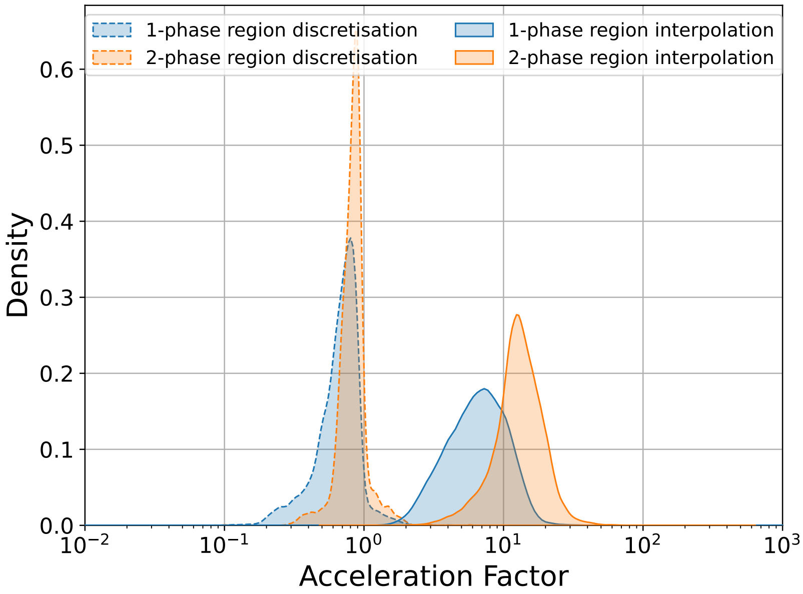
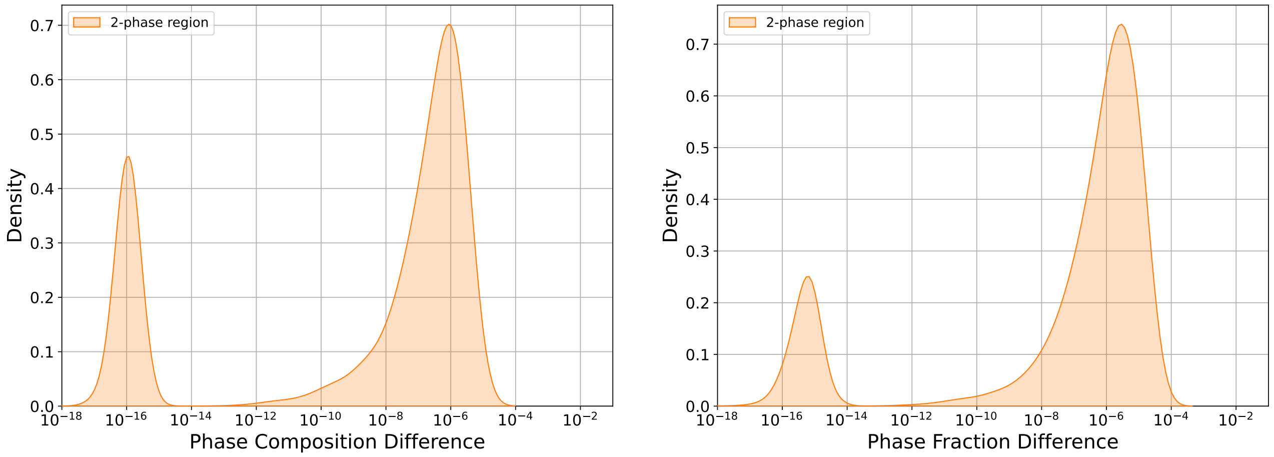


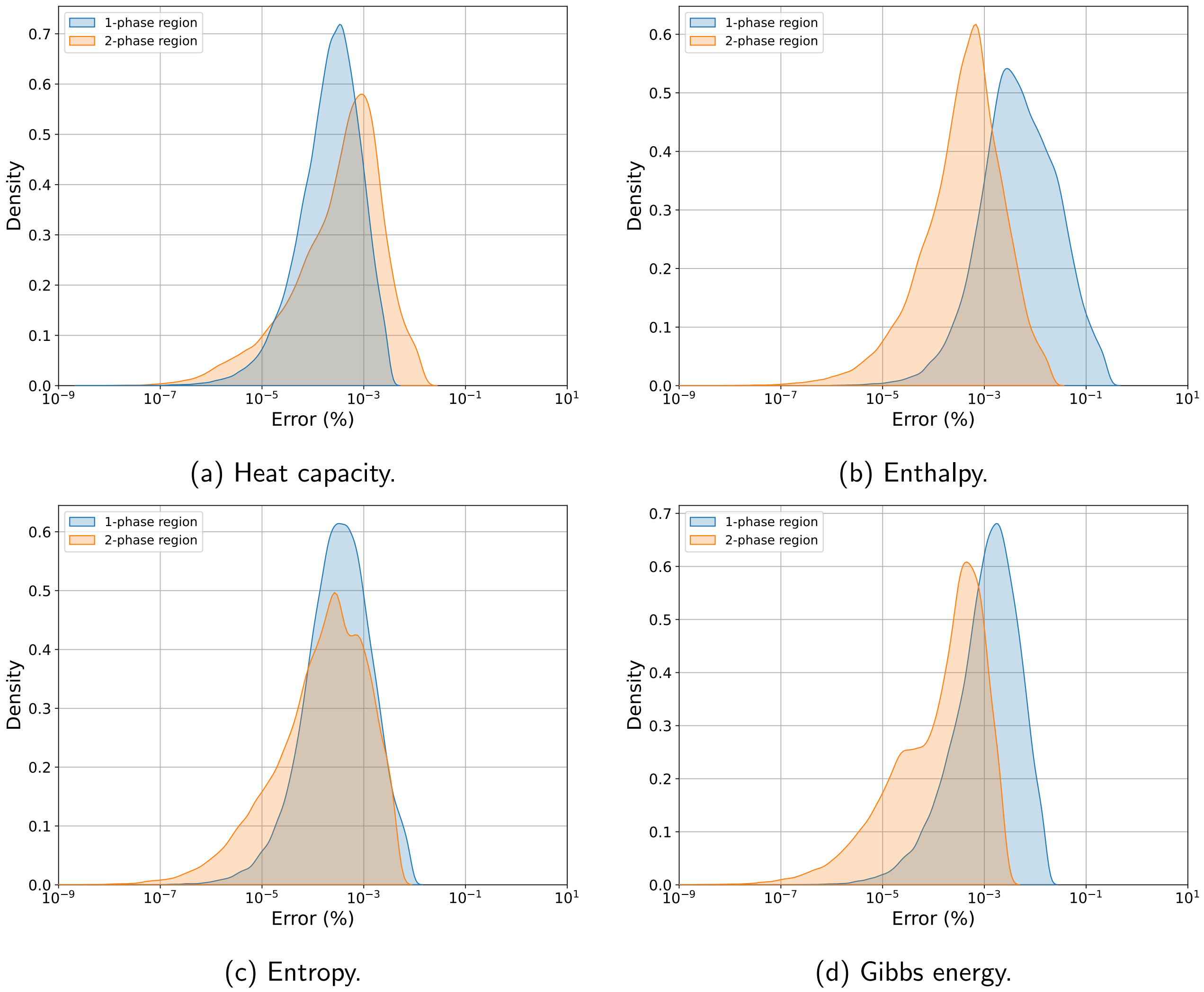
Figure B.4: Distribution of acceleration factors for the Al−Mn system.



(a) Phase composition.

(b) Phase fraction.

Figure B.5: Distribution of phase composition and phase fraction difference for the Al−Mn system.



(a) Heat capacity.

(b) Enthalpy.

(c) Entropy.

(d) Gibbs energy.

Figure B.6: Distribution of physical and thermochemical property errors for the Al−Mn system.

B.1.3 Al−Zn

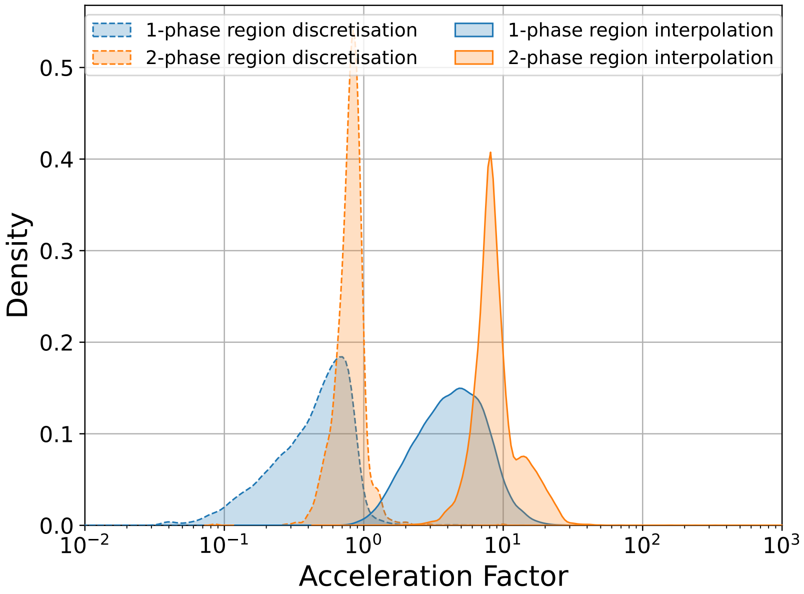
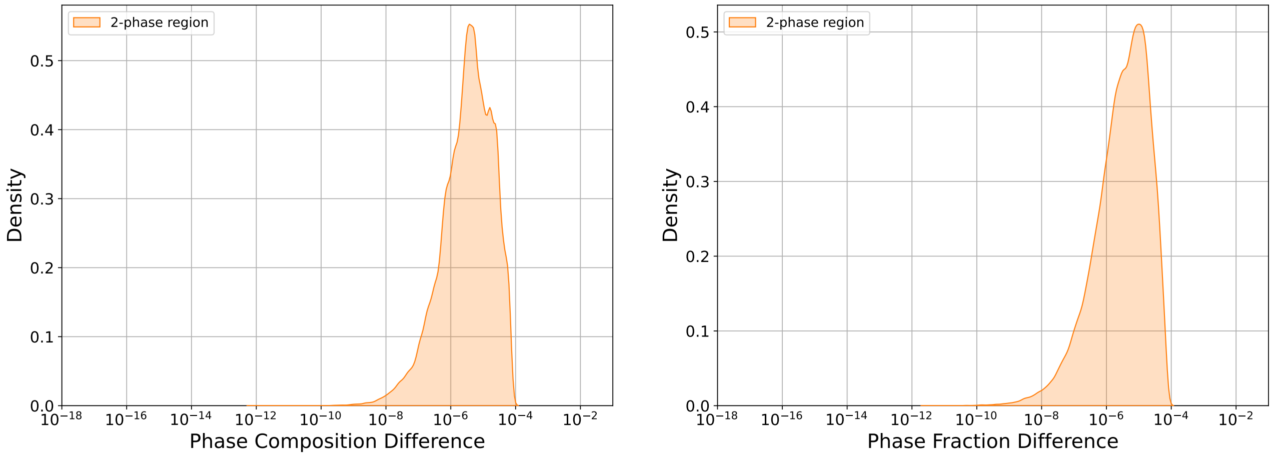


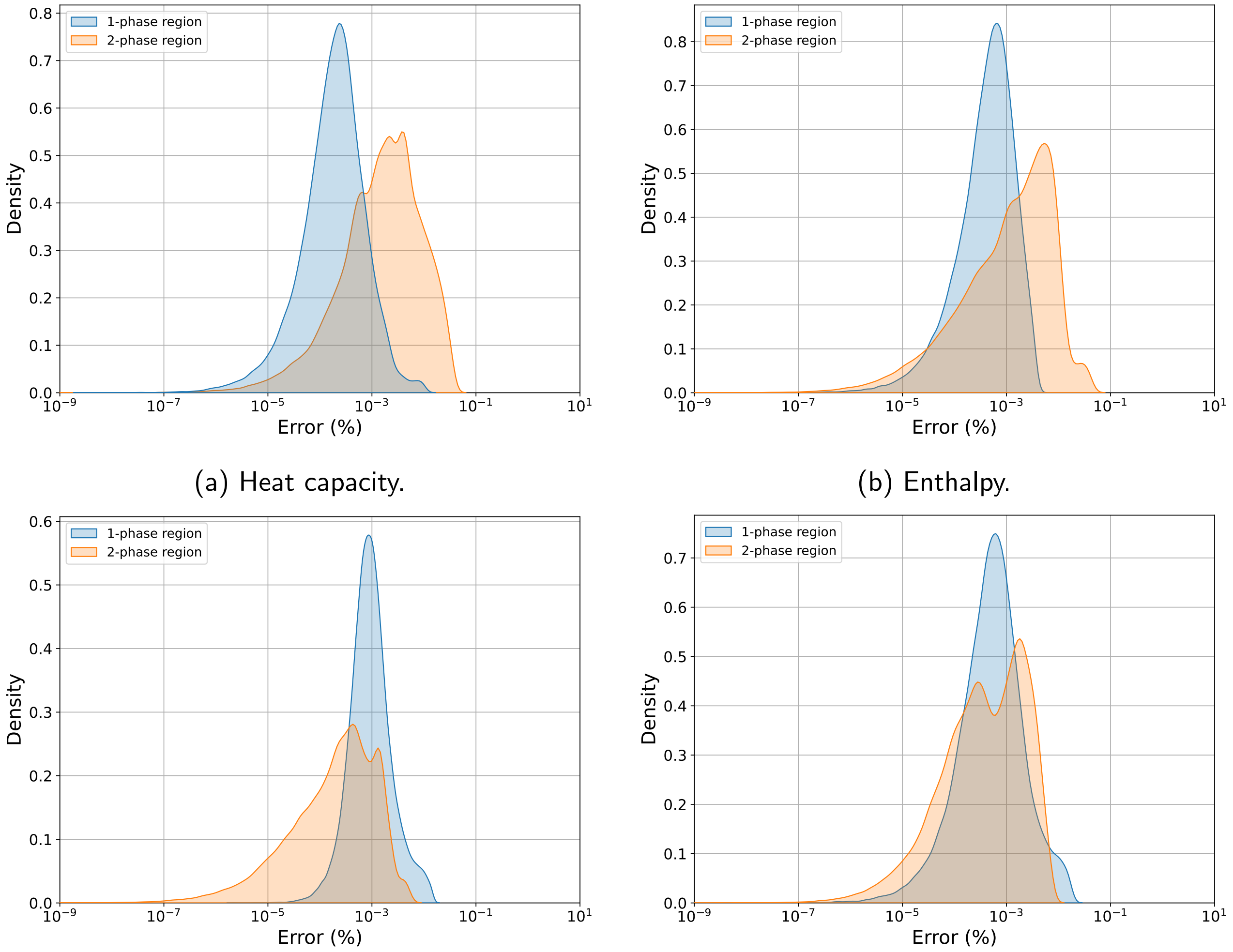
Figure B.7: Distribution of acceleration factors for the Al−Zn system.



(a) Phase composition.

(b) Phase fraction.

Figure B.8: Distribution of phase composition and phase fraction difference for the Al−Zn system.



(a) Heat capacity.

(b) Enthalpy.

(c) Entropy.

(d) Gibbs energy.

Figure B.9: Distribution of physical and thermochemical property errors for the Al$-$Zn system.

B.1.4 Fe−C

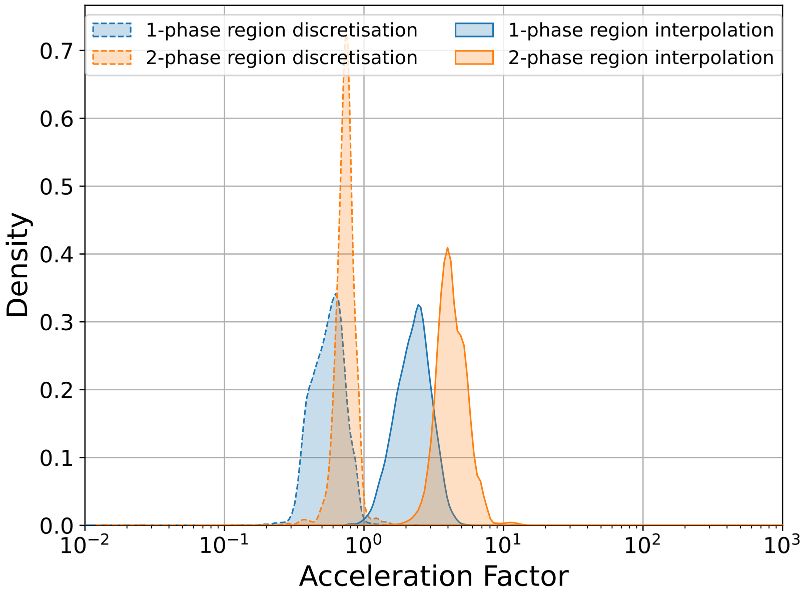
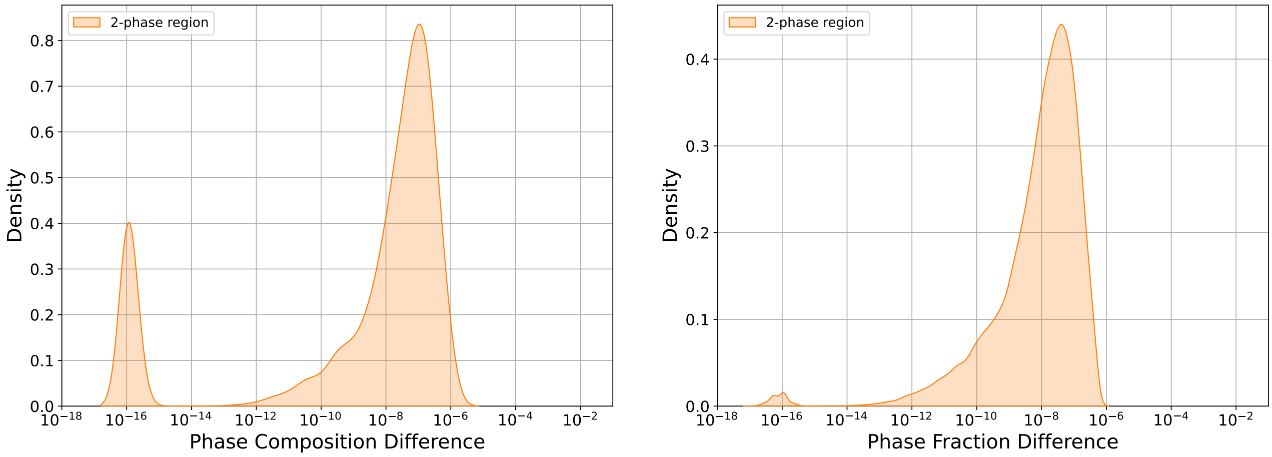


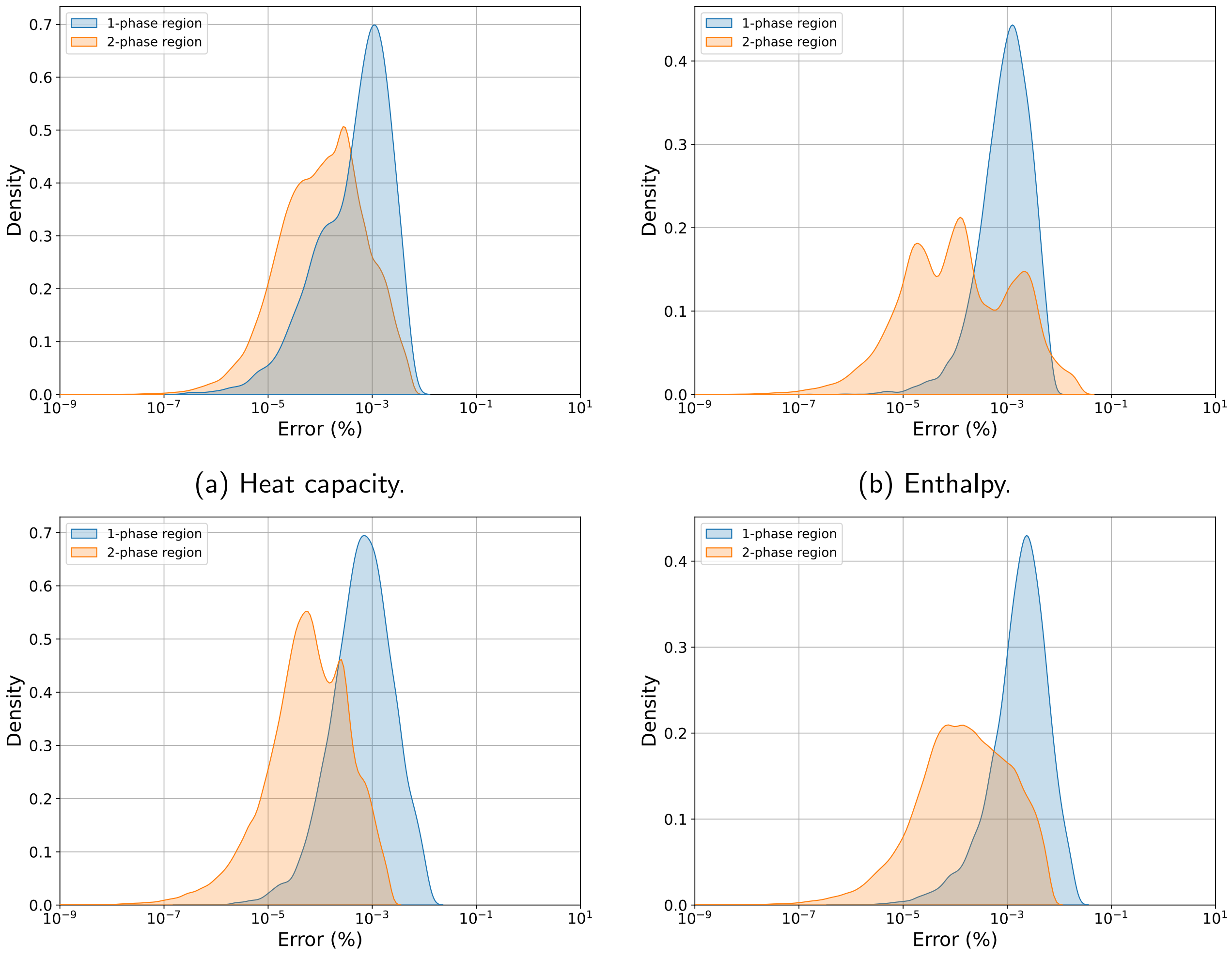
Figure B.10: Distribution of acceleration factors for the Fe−C system.



(a) Phase composition.

(b) Phase fraction.

Figure B.11: Distribution of phase composition and phase fraction difference for the Fe−C system.



(a) Heat capacity.

(b) Enthalpy.

(c) Entropy.

(d) Gibbs energy.

Figure B.12: Distribution of physical and thermochemical property errors for the Fe−C system.

B.1.5 Fe−Cr

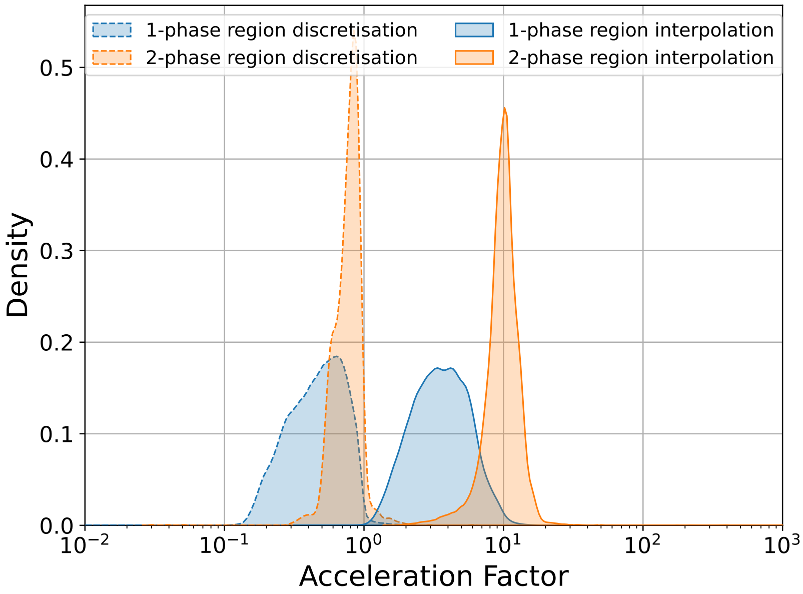
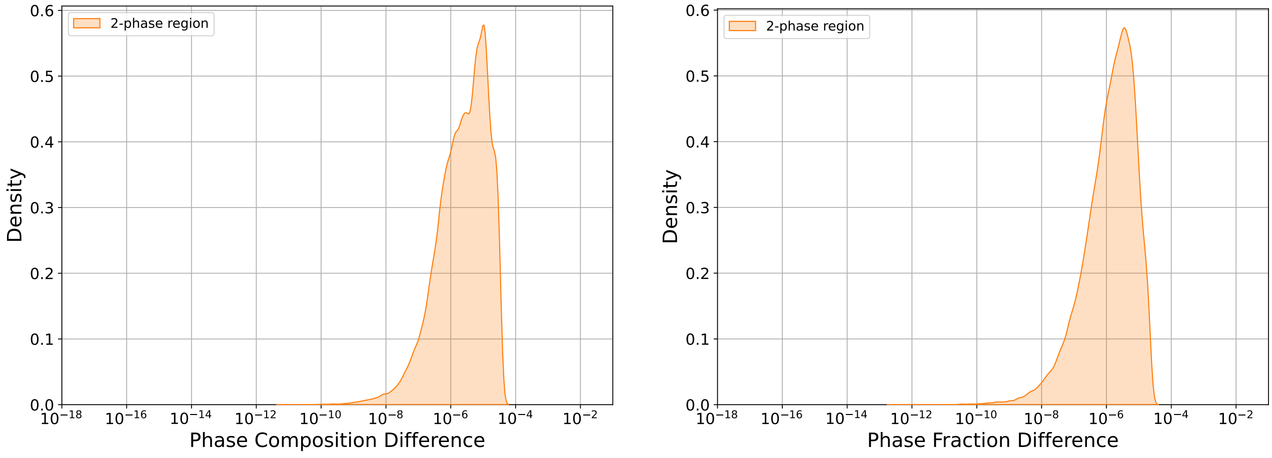


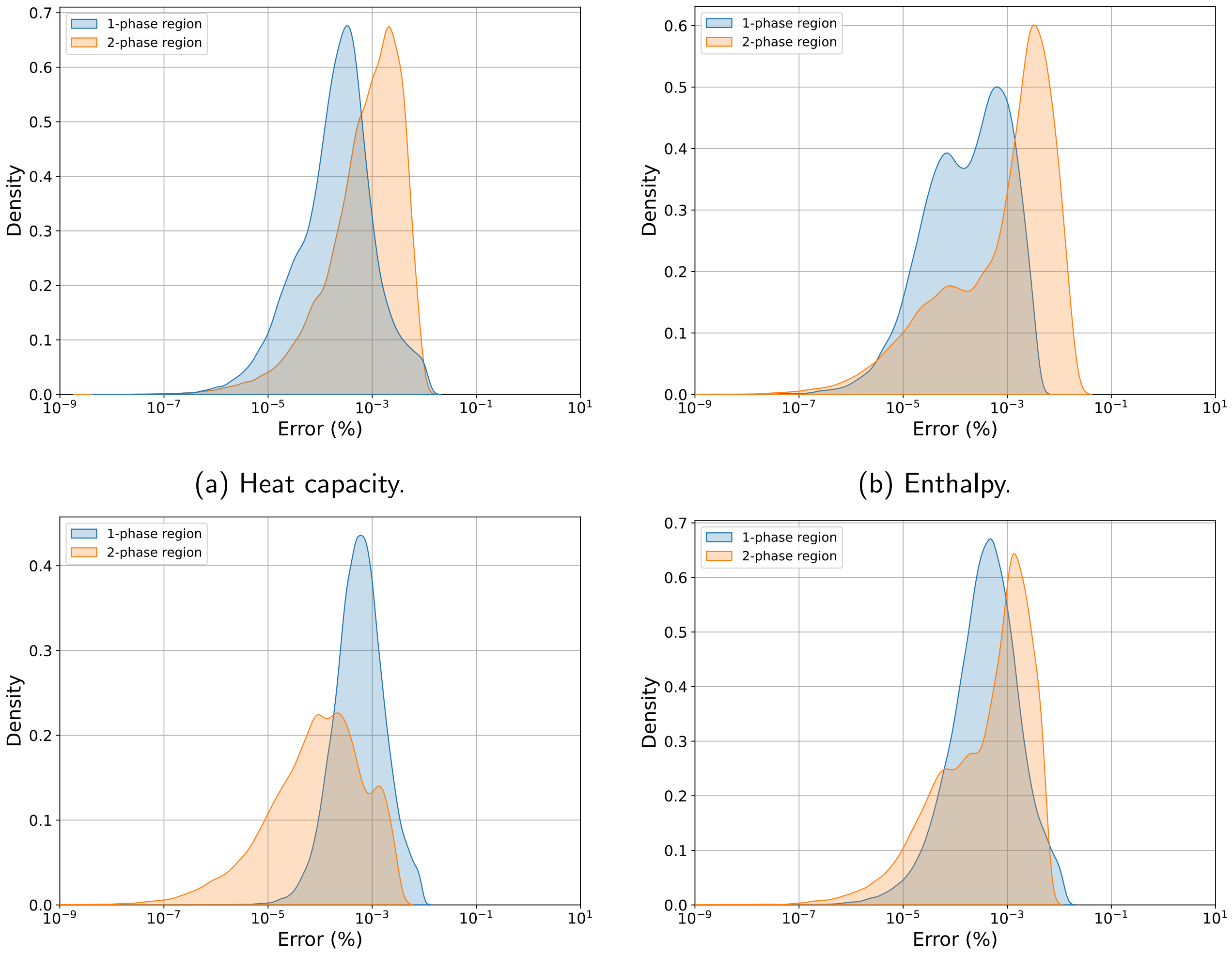
Figure B.13: Distribution of acceleration factors for the Fe−Cr system.



(a) Phase composition.

(b) Phase fraction.

Figure B.14: Distribution of phase composition and phase fraction difference for the Fe−Cr system.



(a) Heat capacity.

(b) Enthalpy.

(c) Entropy.

(d) Gibbs energy.

Figure B.15: Distribution of physical and thermochemical property errors for the Fe−Cr system.

B.1.6 Fe−Si

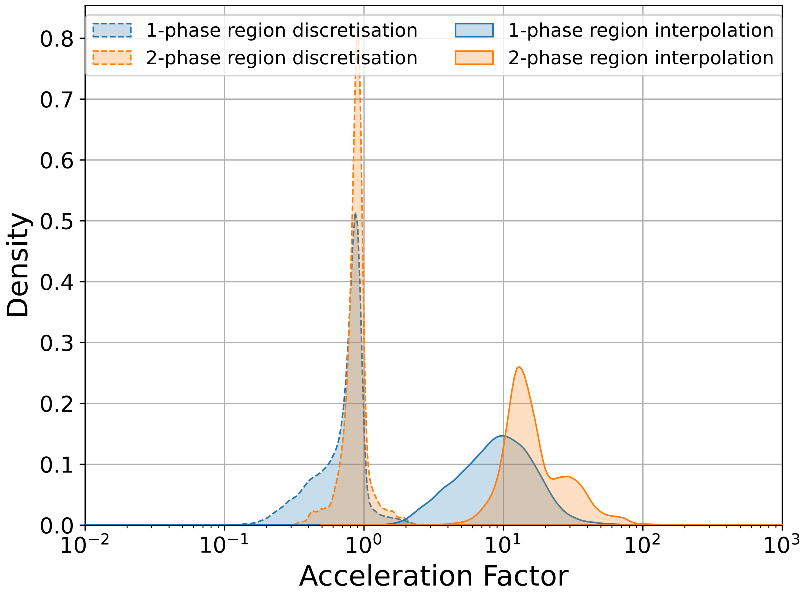
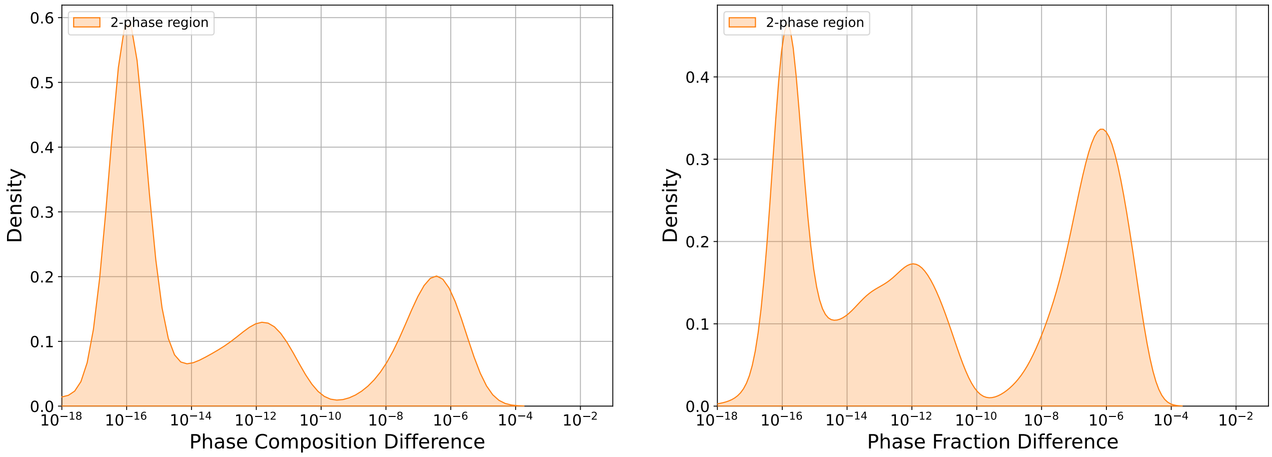


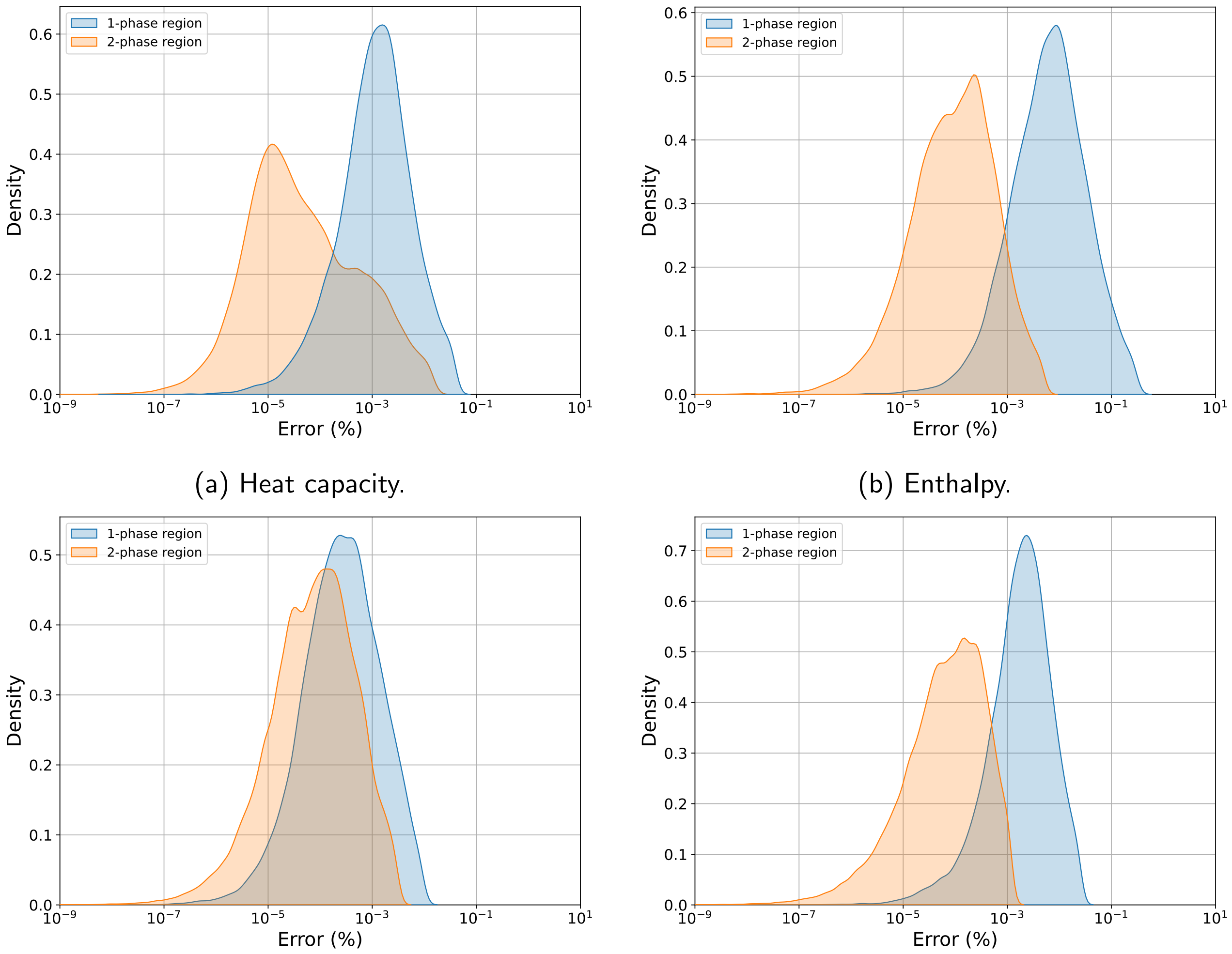
Figure B.16: Distribution of acceleration factors for the Fe−Si system.



(a) Phase composition.

(b) Phase fraction.

Figure B.17: Distribution of phase composition and phase fraction difference for the Fe−Si system.



(a) Heat capacity.

(b) Enthalpy.

(c) Entropy.

(d) Gibbs energy.

Figure B.18: Distribution of physical and thermochemical property errors for the Fe−Si system.

B.1.7 $CaO-SiO_2$

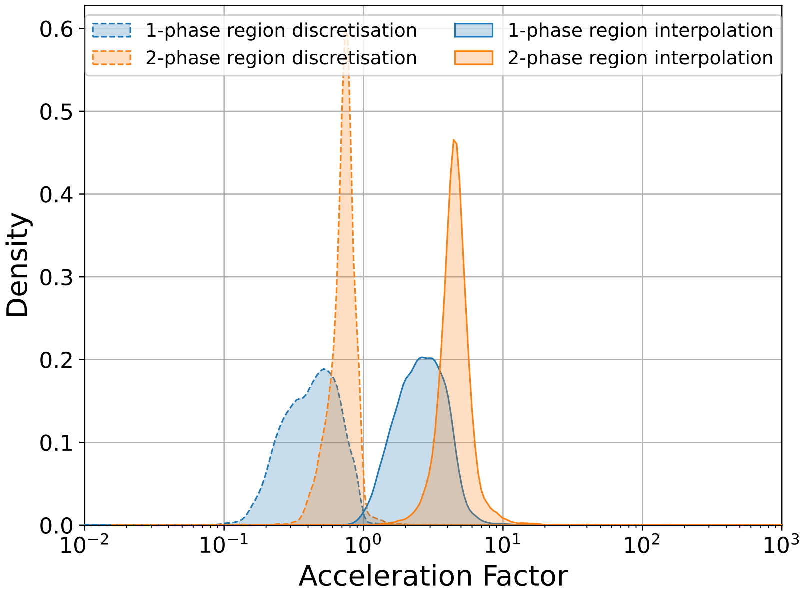
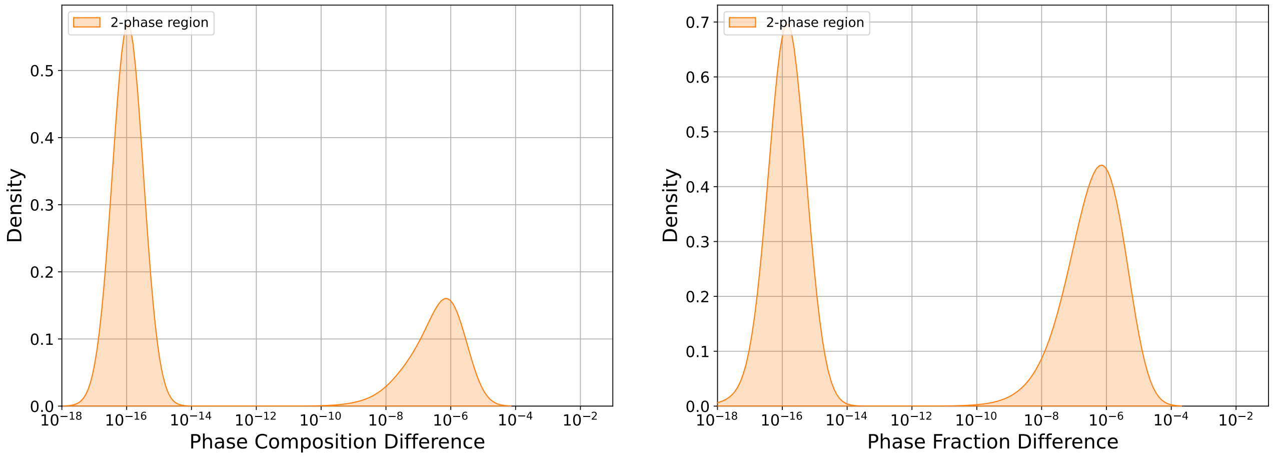


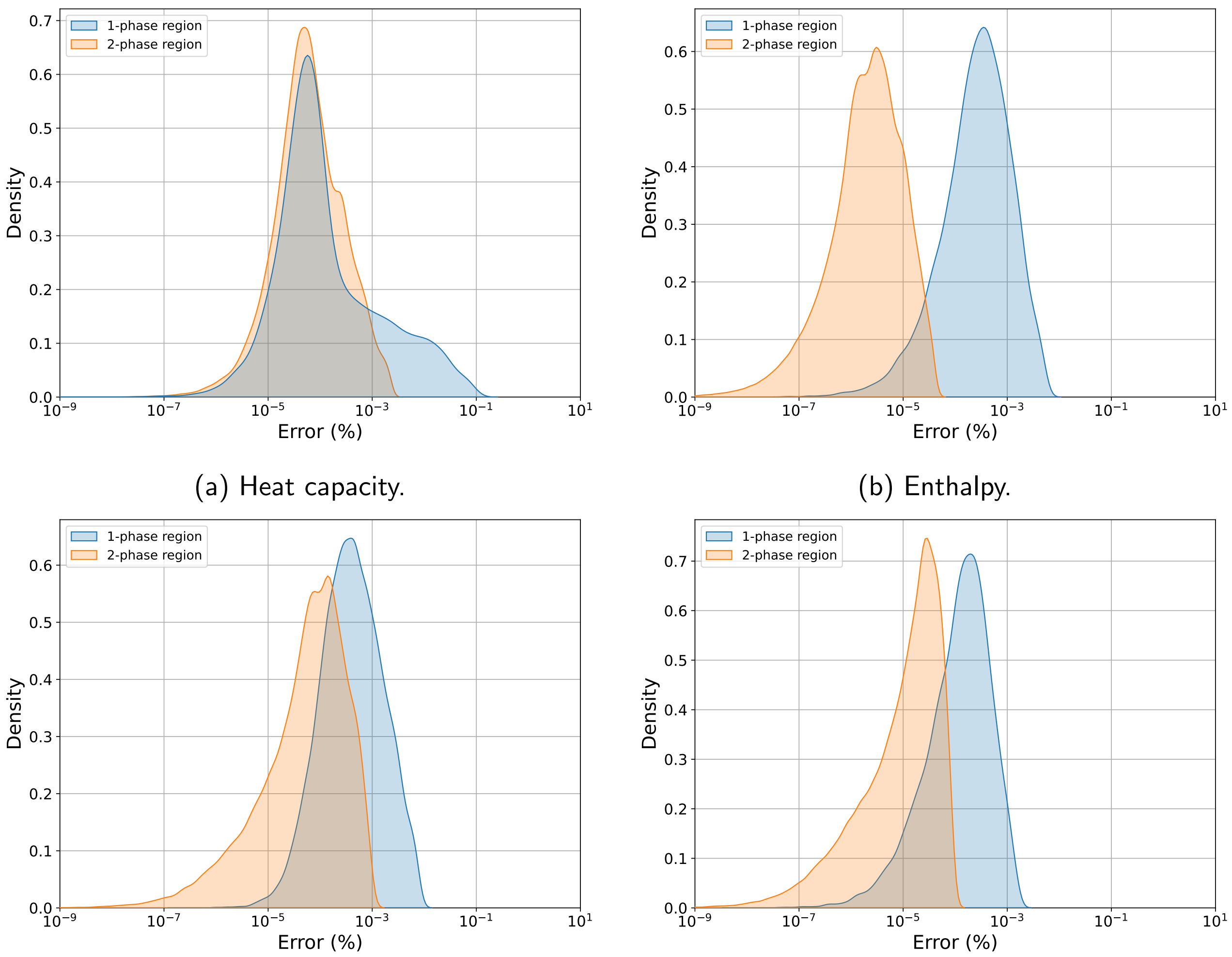
Figure B.19: Distribution of acceleration factors for the $CaO-SiO_2$ system.



(a) Phase composition.

(b) Phase fraction.

Figure B.20: Distribution of phase composition and phase fraction difference for the $CaO-SiO_2$ system.



(a) Heat capacity.

(b) Enthalpy.

(c) Entropy.

(d) Gibbs energy.

Figure B.21: Distribution of physical and thermochemical property errors for the $CaO-SiO_2$ system.

B.1.8 Al_2O_3-CaO

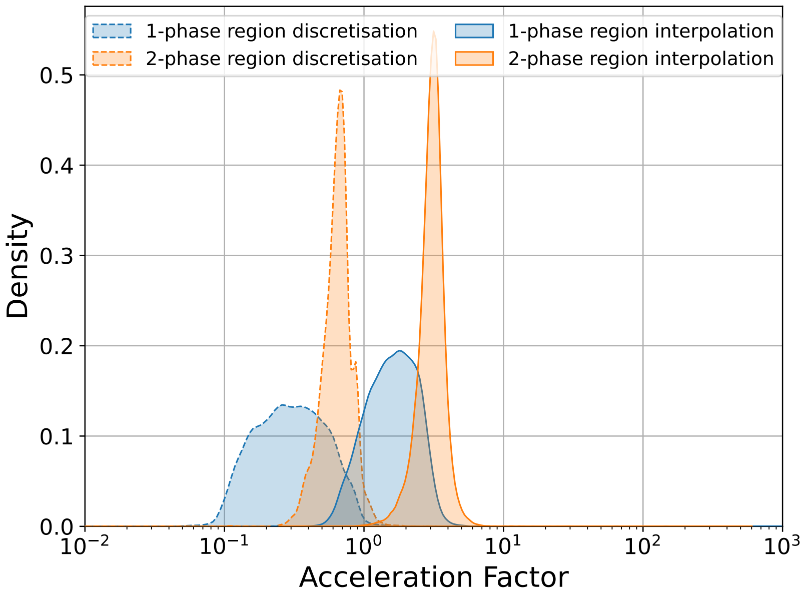
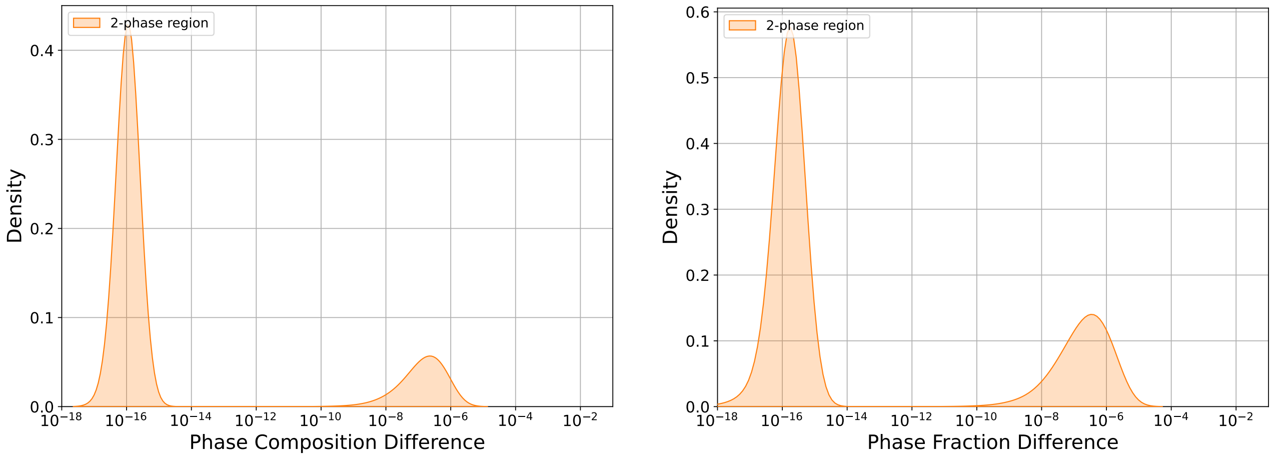


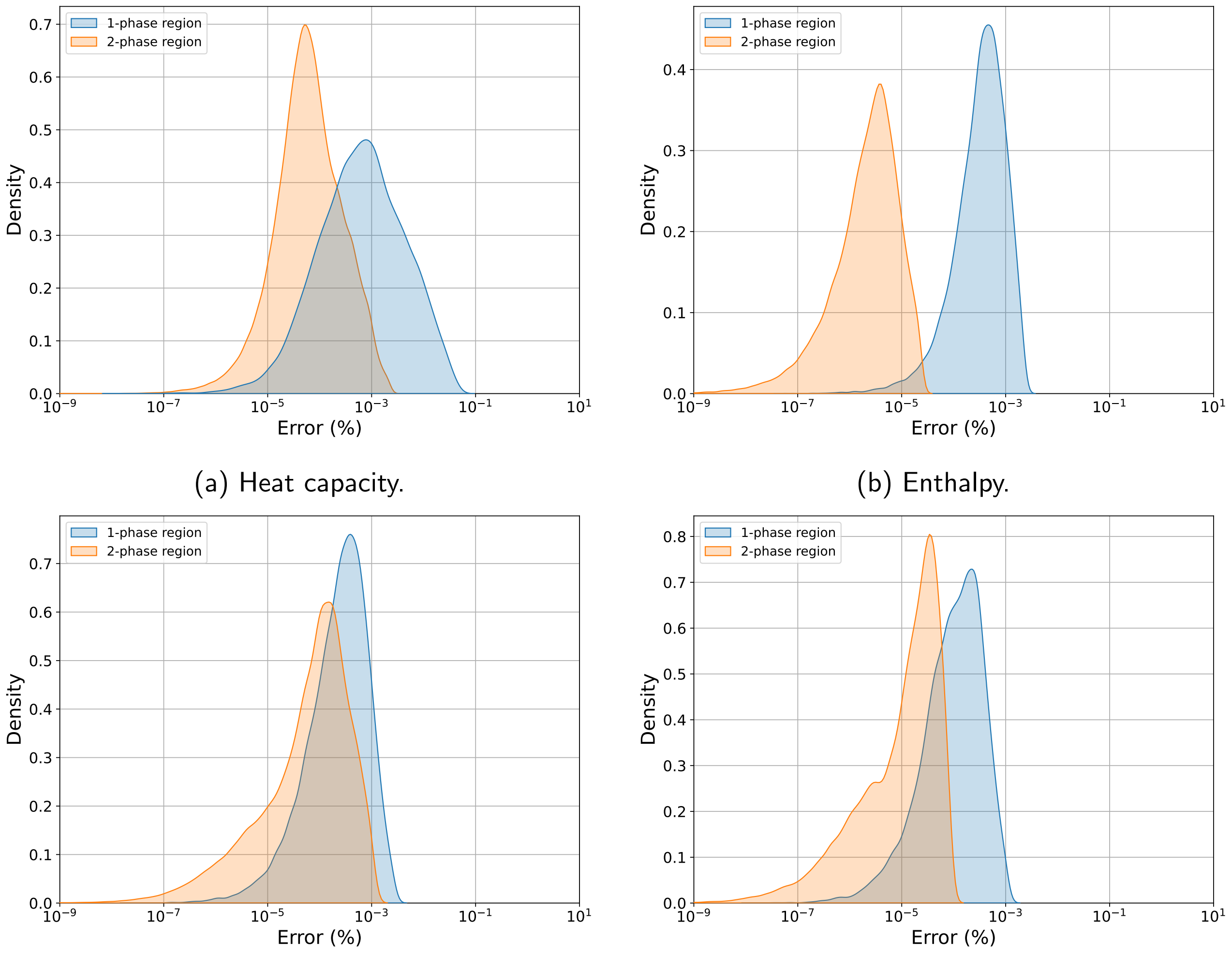
Figure B.22: Distribution of acceleration factors for the Al_2O_3-CaO system.



(a) Phase composition.

(b) Phase fraction.

Figure B.23: Distribution of phase composition and phase fraction difference for the Al_2O_3-CaO system.



(a) Heat capacity.

(b) Enthalpy.

(c) Entropy.

(d) Gibbs energy.

Figure B.24: Distribution of physical and thermochemical property errors for the Al_2O_3-CaO system.

B.1.9 Al_2O_3-MgO

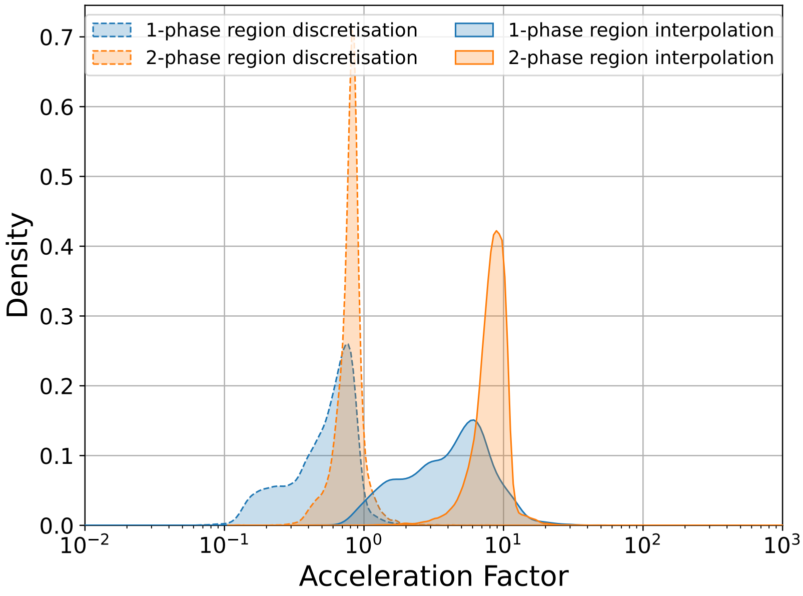
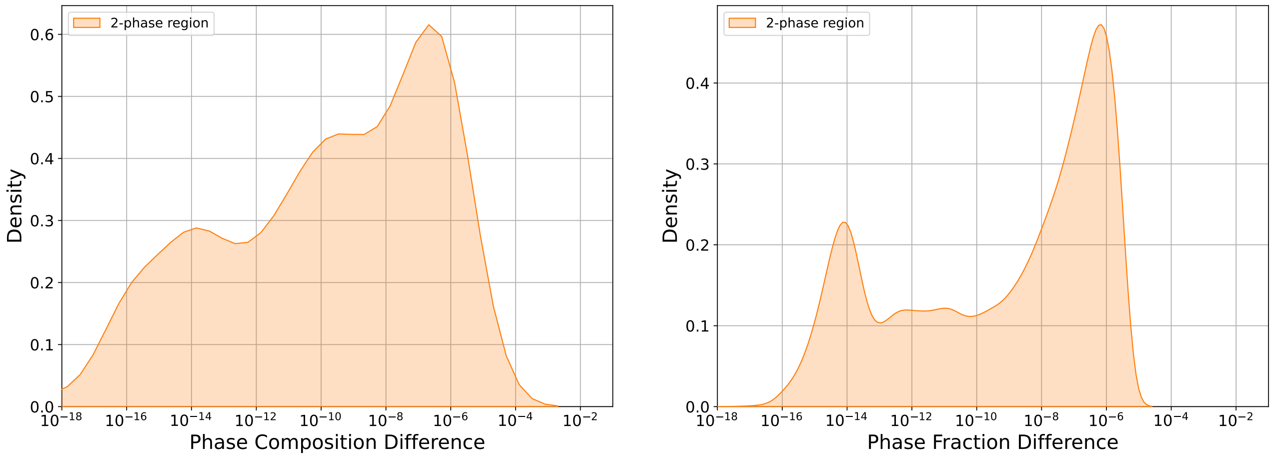


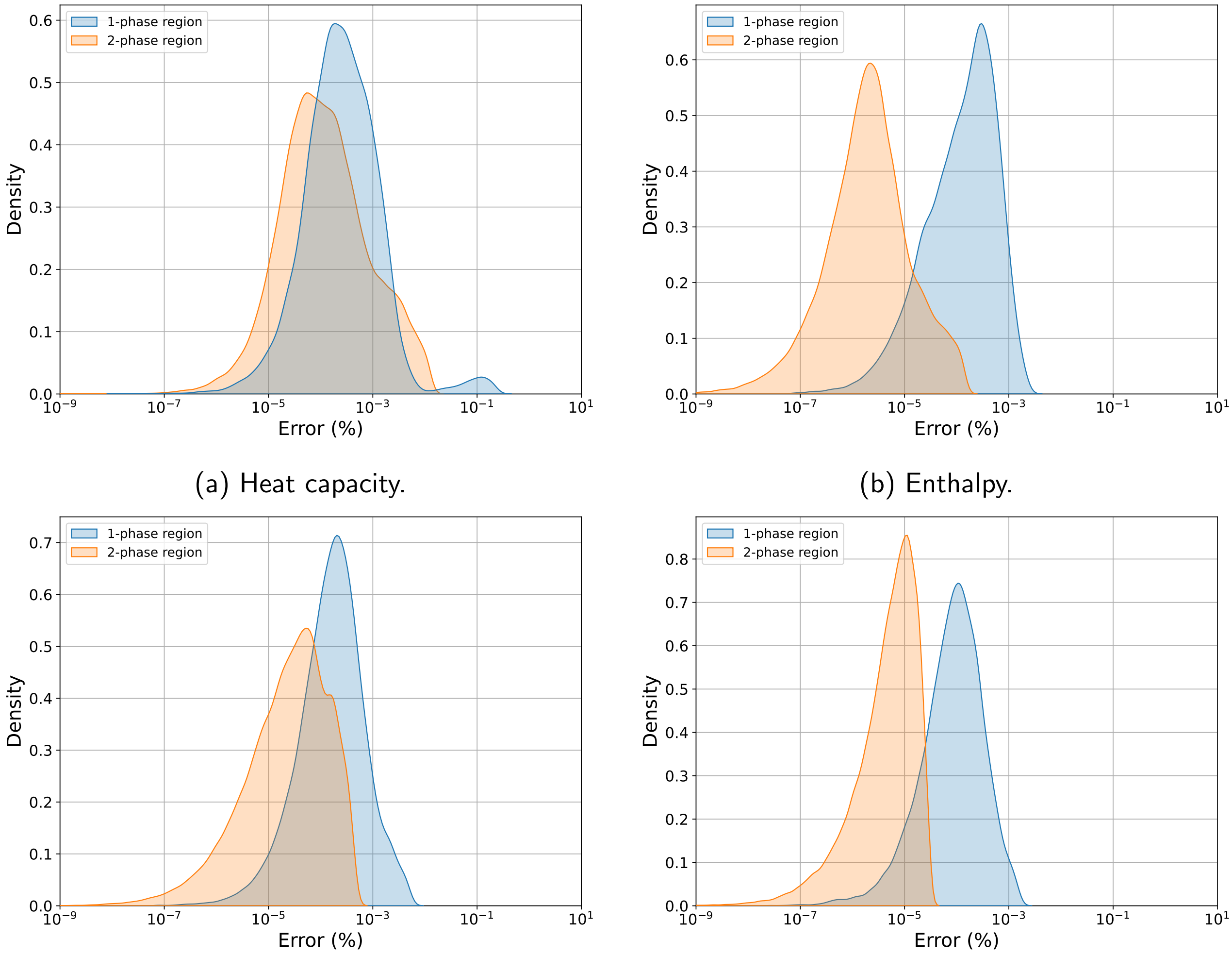
Figure B.25: Distribution of acceleration factors for the Al_2O_3-MgO system.



(a) Phase composition.

(b) Phase fraction.

Figure B.26: Distribution of phase composition and phase fraction difference for the Al_2O_3-MgO system.



(a) Heat capacity.

(b) Enthalpy.

(c) Entropy.

(d) Gibbs energy.

Figure B.27: Distribution of physical and thermochemical property errors for the Al_2O_3-MgO system.

B.1.10 $Al_2O_3-SiO_2$

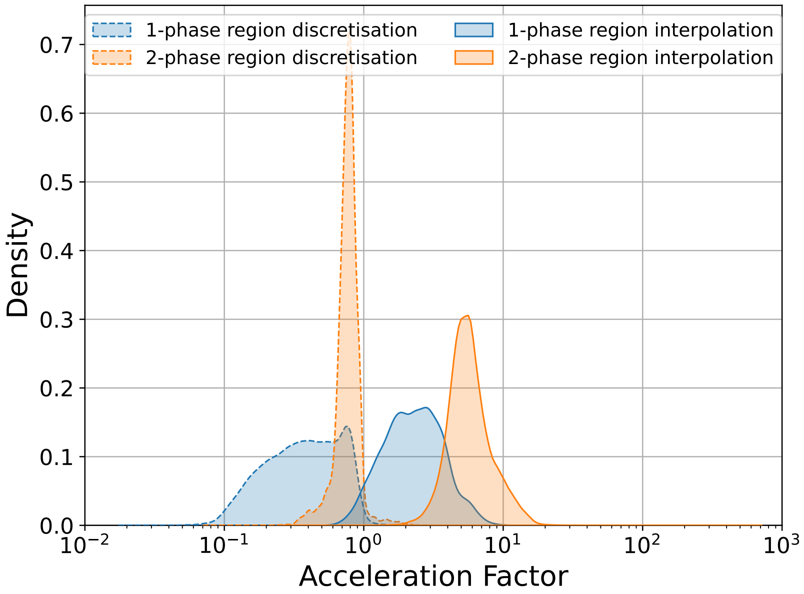
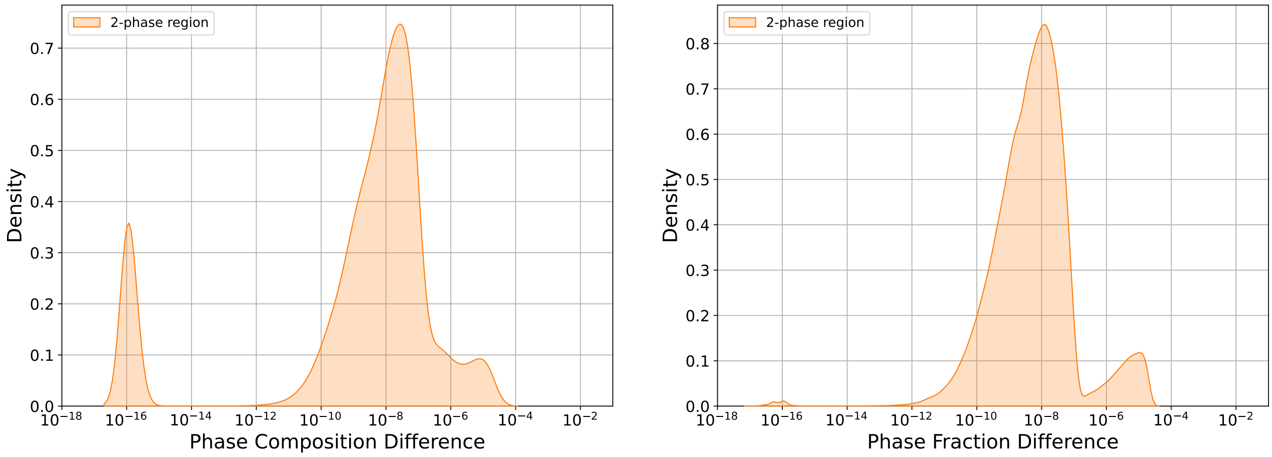


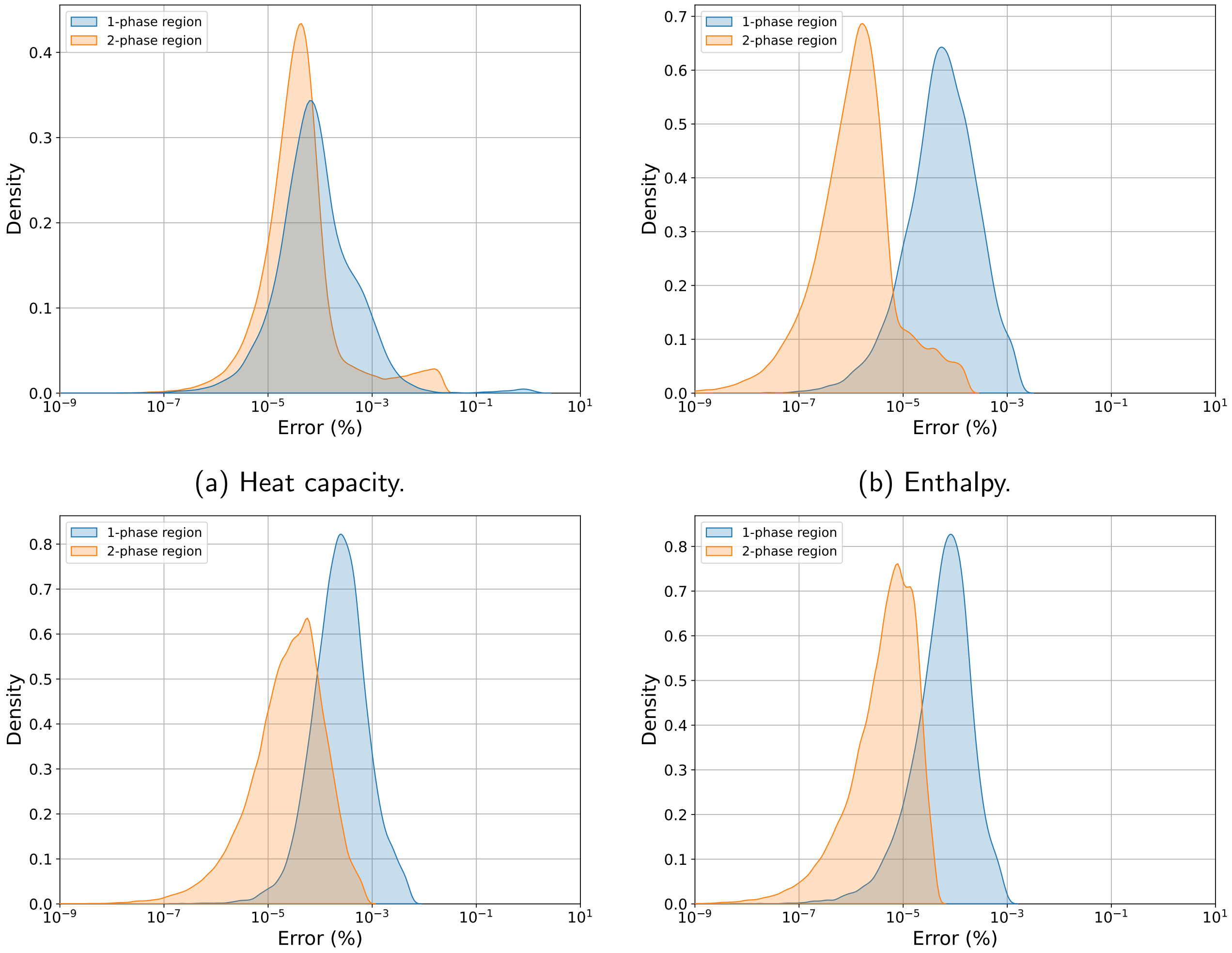
Figure B.28: Distribution of acceleration factors for the $Al_2O_3-SiO_2$ system.



(a) Phase composition.

(b) Phase fraction.

Figure B.29: Distribution of phase composition and phase fraction difference for the $Al_2O_3-SiO_2$ system.



(a) Heat capacity.

(b) Enthalpy.

(c) Entropy.

(d) Gibbs energy.

Figure B.30: Distribution of physical and thermochemical property errors for the $Al_2O_3-SiO_2$ system.

B.2 Three-component Systems

Distribution plots of the performance test results of all tested three-component systems are presented hereafter.

B.2.1 $Al_2O_3-CaO-MgO$

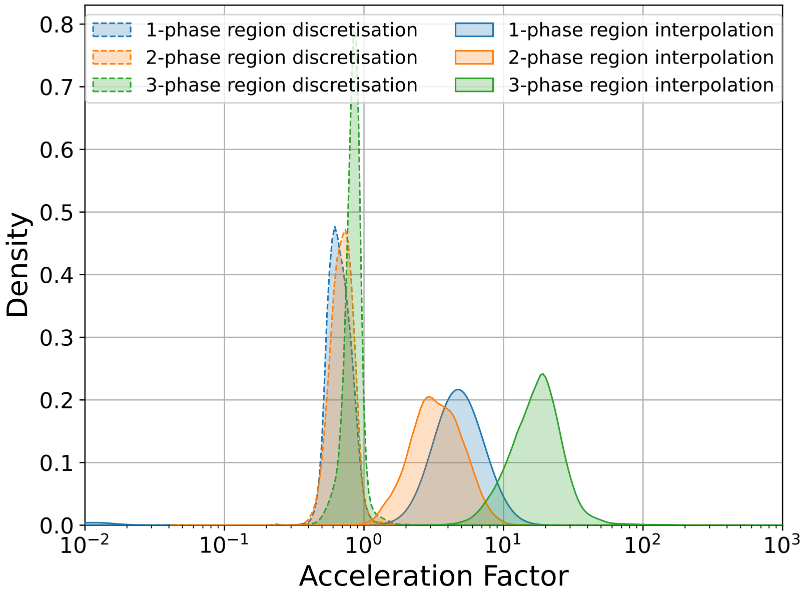
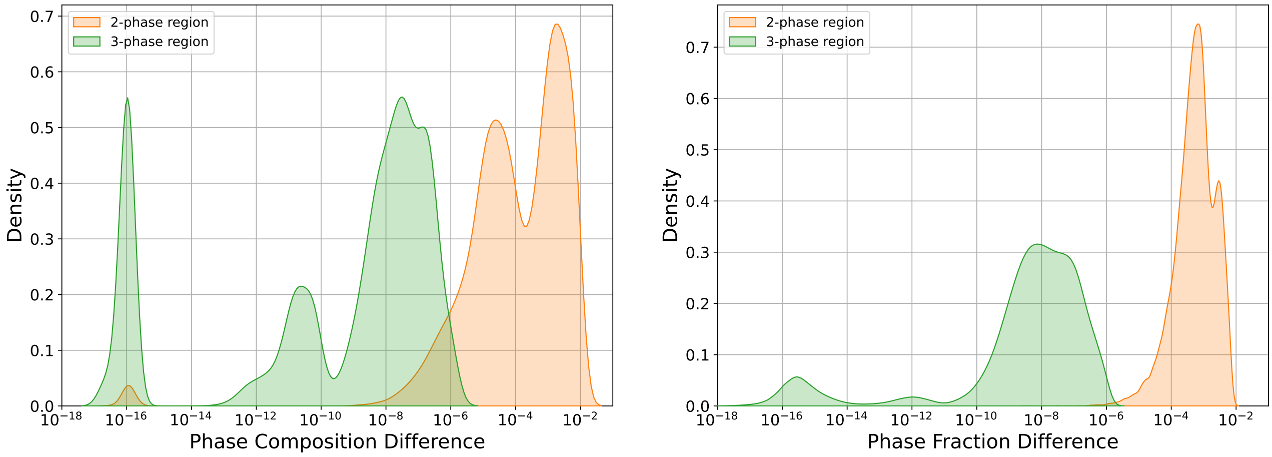


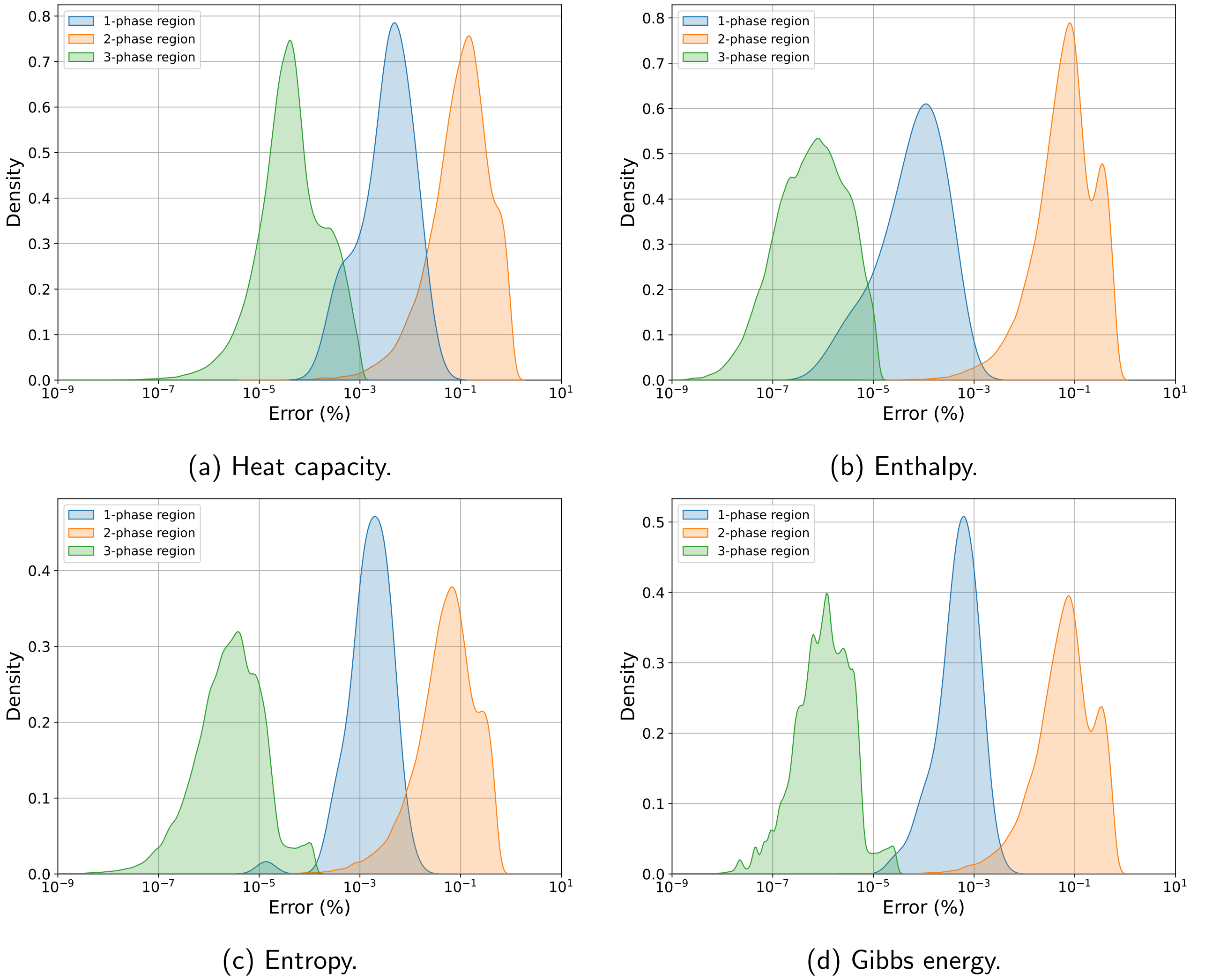
Figure B.31: Distribution of acceleration factors for the $Al_2O_3-CaO-MgO$ system.



(a) Phase composition.

(b) Phase fraction.

Figure B.32: Distribution of phase composition and phase fraction difference for the $Al_2O_3-CaO-MgO$ system.



(a) Heat capacity.

(b) Enthalpy.

(c) Entropy.

(d) Gibbs energy.

Figure B.33: Distribution of physical and thermochemical property errors for the $Al_2O_3-CaO-MgO$ system.

B.2.2 $Al_2O_3-CaO-SiO_2$

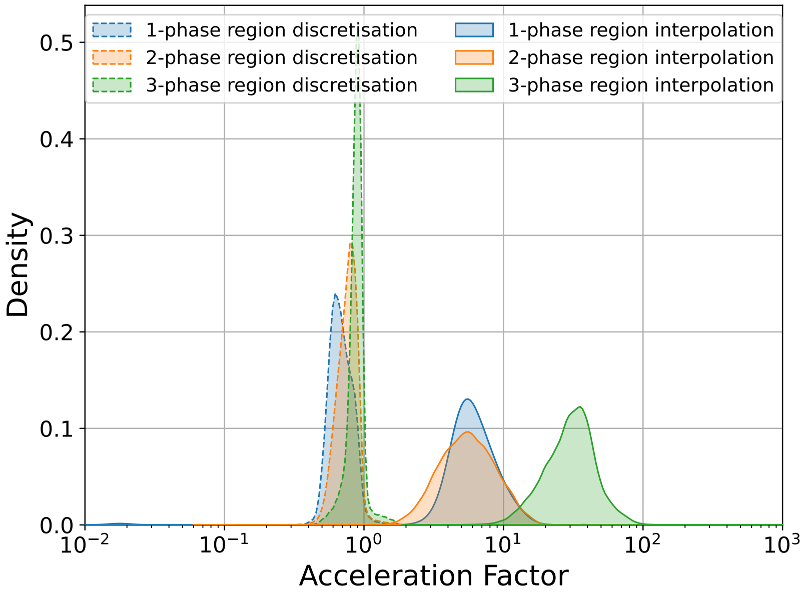
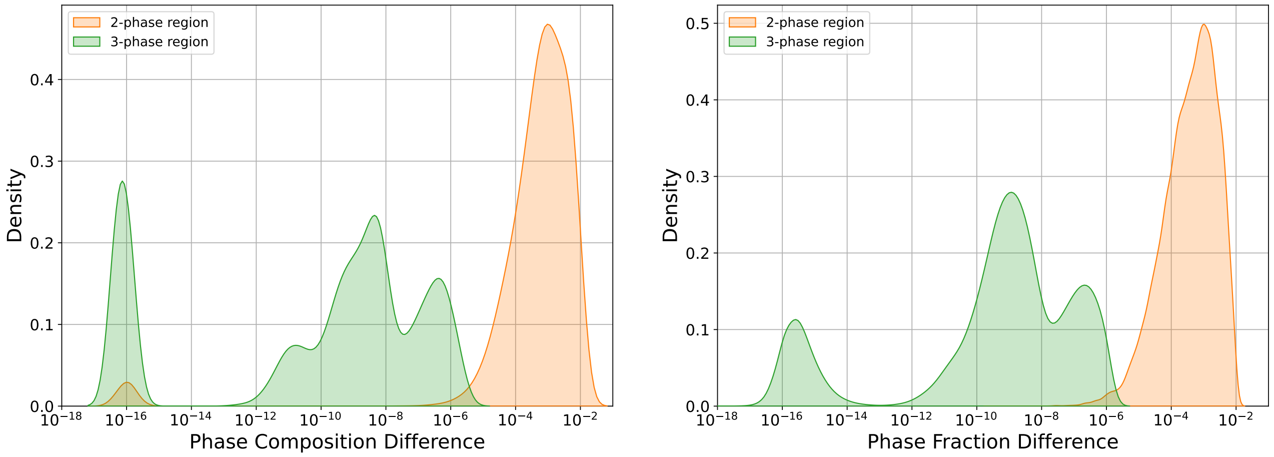


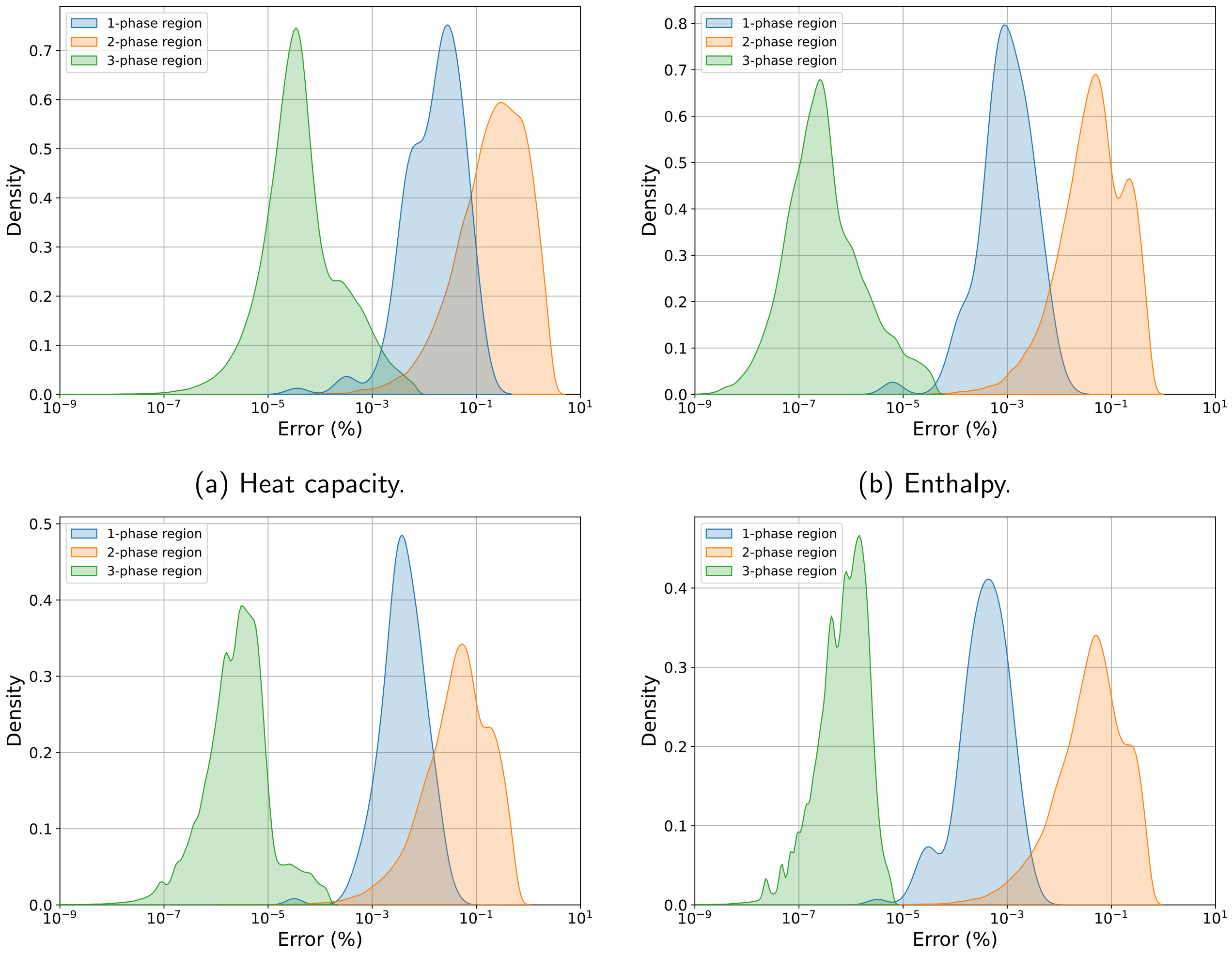
Figure B.34: Distribution of acceleration factors for the $Al_2O_3-CaO-SiO_2$ system.



(a) Phase composition.

(b) Phase fraction.

Figure B.35: Distribution of phase composition and phase fraction difference for the $Al_2O_3-CaO-SiO_2$ system.



(a) Heat capacity.

(b) Enthalpy.

(c) Entropy.

(d) Gibbs energy.

Figure B.36: Distribution of physical and thermochemical property errors for the $Al_2O_3-CaO-SiO_2$ system.

B.2.3 $Al_2O_3-Fe_2O_3-MgO$

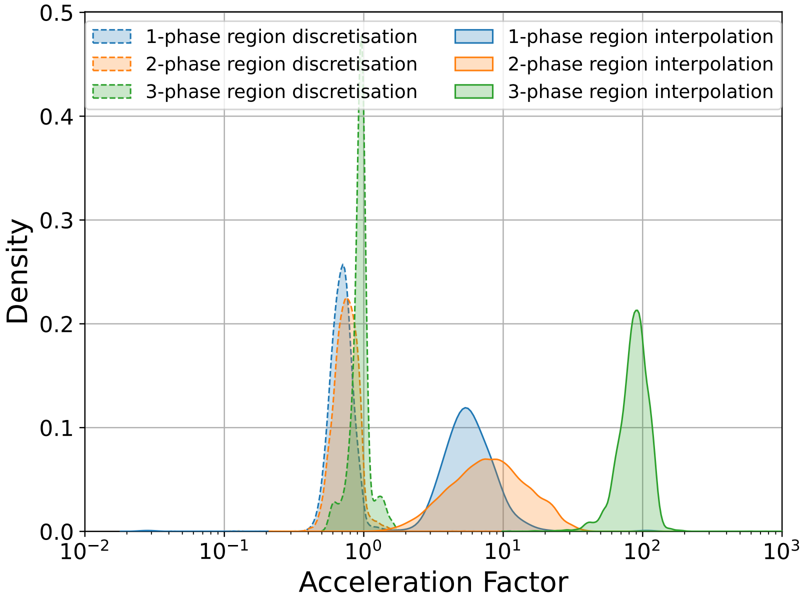
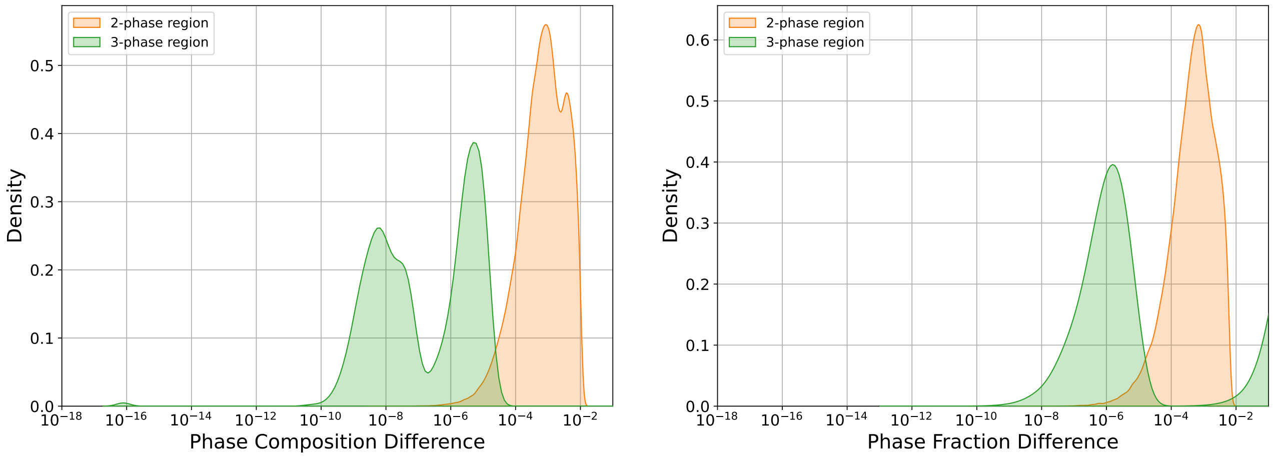


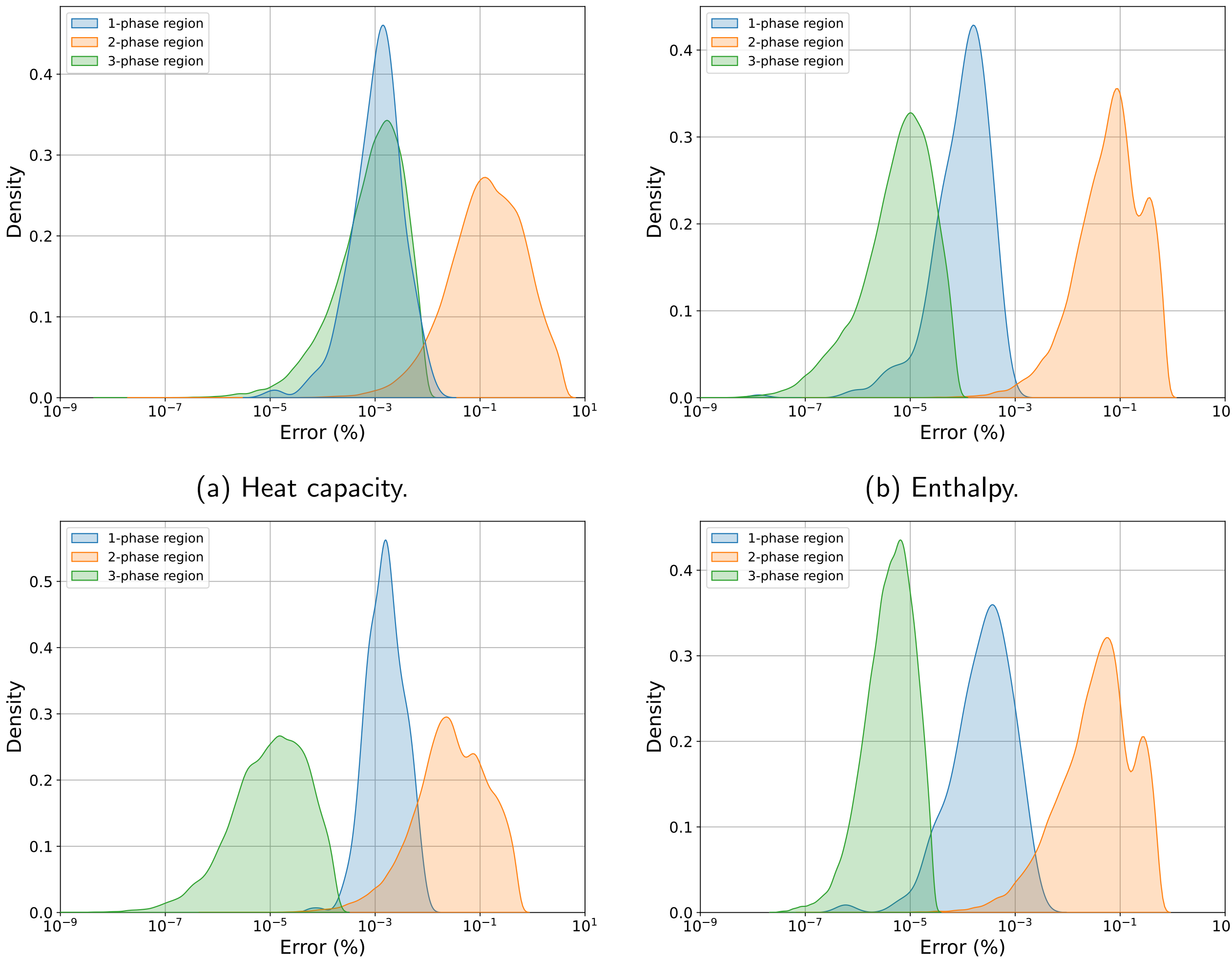
Figure B.37: Distribution of acceleration factors for the $Al_2O_3-Fe_2O_3-MgO$ system.



(a) Phase composition.

(b) Phase fraction.

Figure B.38: Distribution of phase composition and phase fraction difference for the $Al_2O_3-Fe_2O_3-MgO$ system.



(a) Heat capacity.

(b) Enthalpy.

(c) Entropy.

(d) Gibbs energy.

Figure B.39: Distribution of physical and thermochemical property errors for the $Al_2O_3-Fe_2O_3-MgO$ system.

B.2.4 $Al_2O_3-Fe_2O_3-SiO_2$

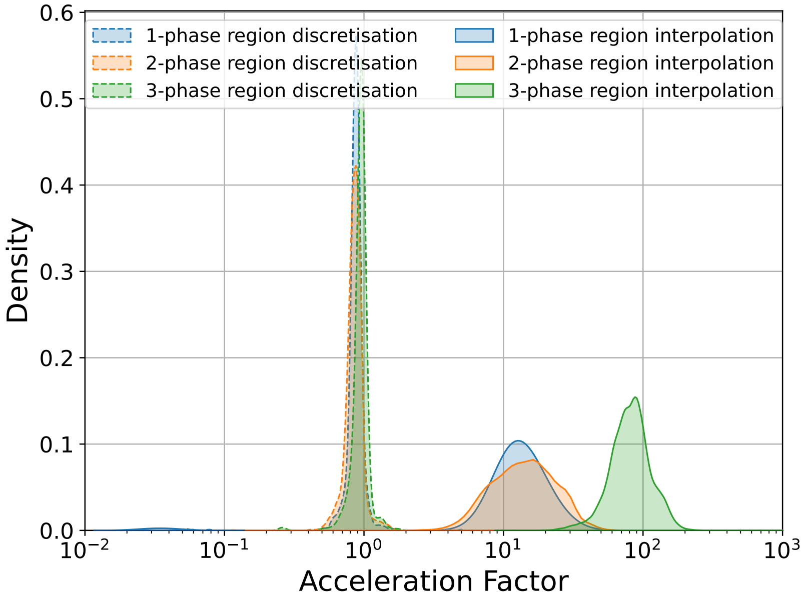
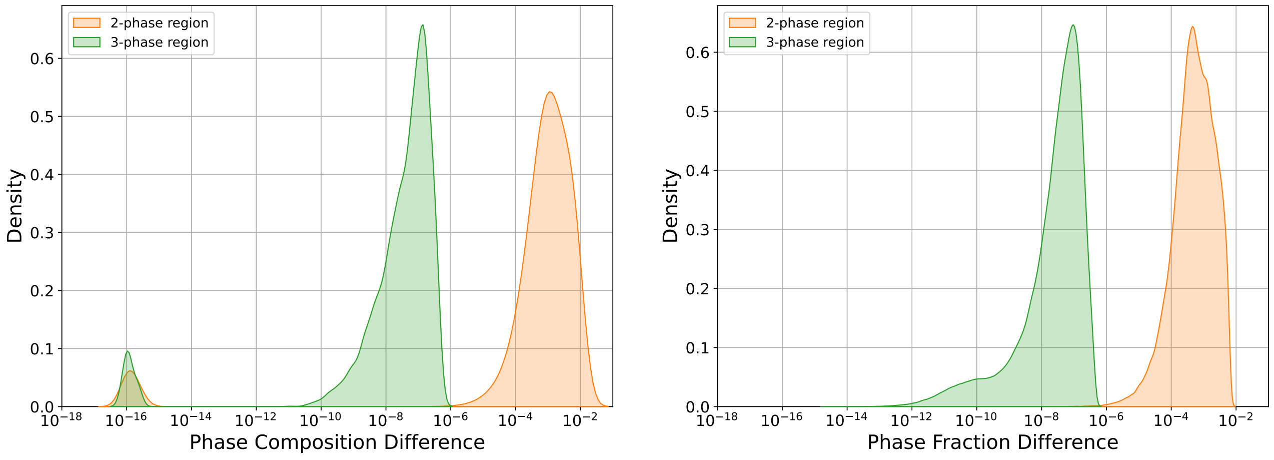


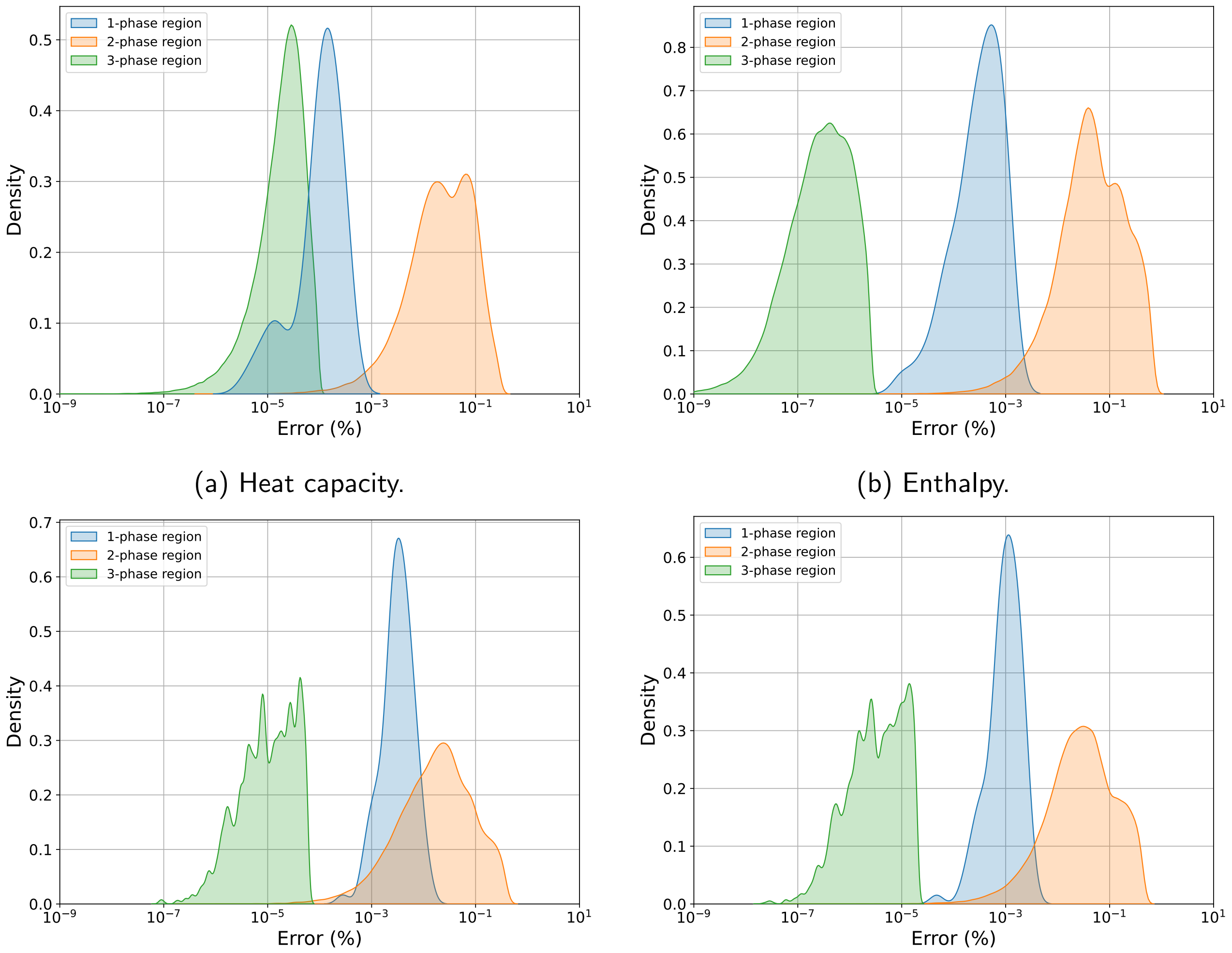
Figure B.40: Distribution of acceleration factors for the $Al-Fe_2O_3-SiO_2$ system.



(a) Phase composition.

(b) Phase fraction.

Figure B.41: Distribution of phase composition and phase fraction difference for the $Al-Fe_2O_3-SiO_2$ system.



(a) Heat capacity.

(b) Enthalpy.

(c) Entropy.

(d) Gibbs energy.

Figure B.42: Distribution of physical and thermochemical property errors for the $Al-Fe_2O_3-SiO_2$ system.