Uncertainty Quantification of the Mixing Enthalpy, Excess Heat Capacity and Gibbs Energy parameters of the LiF-KF System using CALPHAD modelling and Polynomial Chaos Expansion

Fleur Rooijakkers (4692217)

TU Delft
Faculty of Applied Sciences
BSc program Applied Physics

Delft
Supervisors:
Dr. A. Smith and Dr. Z. Perkó

1 Abstract

In the recent years, there has been a growing interest in molten salt reactors as a source of energy. To ensure molten salt reactor safety, it is vital to know the thermodynamic properties of the systems involved.

An investigation into the uncertainty of the mixing enthalpy, excess heat capacity and Gibbs energy parameters of the LiF-KF system is presented in this study. The program *FactSage 7.2* [19] is used, which takes optimized Gibbs energy parameters as an input and uses these to calculate phase diagram data and the values of different thermodynamic properties. The uncertainty is quantified using the polynomial chaos expansion, which analyzes the relationship between the input Gibbs energy parameters and the output; which is the phase diagram data, mixing enthalpy and excess heat capacity of the system.

Firstly, an investigation into the accuracy of the polynomial chaos expansion, when applying different settings, is given. Once the most accurate settings are found, this expansion is used to generate many different samples of phase diagrams and the corresponding mixing enthalpy and excess heat capacity values. A margin of 10 Kelvin is then introduced as a maximum deviation from the experimentally determined phase diagram. The input Gibbs energy parameters and the mixing enthalpy and excess heat capacity values, that correlate with the phase diagrams within the margin of the experimentally determined phase diagram, can then be extracted. Once these values are known, the maximum uncertainty half width of the mixing enthalpy and excess heat capacity can be given that is still consistent with sensible phase diagrams.

The found values for the maximum uncertainty half range are 1.65 kJ$\cdot$mol^{-1} for the mixing enthalpy which is a 35.6% deviation from the mixing enthalpy value computed with the original Gibbs energy parameters. For the excess heat capacity, 1.676 J$\cdot$K$^{-1}\cdot$mol^{-1} was found as a maximum uncertainty half range which is a 44.7% deviation from the excess heat capacity value computed with the original Gibbs energy parameters. The one-dimensional uncertainty half range (the uncertainty half range of one parameter assuming all other parameters possess zero uncertainty) of Gibbs energy parameters were calculated. Additionally, scattering plots are generated to illustrate the two- and three-dimensional uncertainty of the different Gibbs energy parameters.

Contents

2 Introduction

Amid the world's quest for renewable energy, nuclear reactors as a source of energy have regained attention. The research into the newer generation of reactors, generation IV reactors, is becoming increasingly pertinent. One of these generation IV reactors is the molten salt reactor [1]. In these types of reactors the nuclear fuel is dissolved in an inorganic liquid (molten salt) which is pumped at low pressure through the primary circuit and reactor vessel, where a fission process takes place. The heat generated by this fission process is transferred to a secondary coolant in a heat exchanger, which is most often also a molten salt [1].

Since the fuel salt serves multiple purposes in the cycle, it must meet multiple constraints [16]. These constraints are: a low melting point, a large thermal expansion coefficient, high thermal and radiolytic stability, low vapor pressure, reasonable hydrodynamic and thermal conductivity, high heat capacity, sufficient dissolution of nuclear fuels and no isotopes/elements with high-parasitic absorption in a neutron spectral region with high neutron flux [16]. Fluoride and chloride salts have a high thermal conductivity, large thermal expansion coefficient and high heat capacity, which make them suitable fuel salt choices [16]. The fluoride salt which will be analyzed in this research is the LiF-KF.

To ensure the safety of the molten salt reactor, it is vital to have an exact knowledge of the thermodynamic behavior of these nuclear fuel systems. An important property is the behavior of the phases of the fuel salt at different temperatures T and compositions X, which can be displayed in a phase diagram. Calculation of phase diagrams (CALPHAD) modeling is a method used to calculate the phase diagrams and thermodynamic properties by minimizing the Gibbs energy of the system. The model takes the input Gibbs energy parameters and uses them to compute the phase diagram and thermodynamic properties of the system. Although the phase diagram data is commonly measured experimentally in the literature, there is often little knowledge of other crucial thermodynamic properties of these systems. It is therefore of importance to know the accuracy of these parameters obtained by CALPHAD modeling.

In this thesis, the polynomial chaos expansion is used to determine the functional relationship between the input Gibbs energy parameters and the output; which is the phase diagram data and the thermodynamic properties mixing enthalpy and excess heat capacity. Using this relation, and the fact that the phase diagram data is known, many different phase diagrams corresponding to different input Gibbs parameters can be generated. The Gibbs parameters and properties associated with sensible phase diagram data can then be filtered. Subsequently, the maximum allowed deviation (uncertainty half range) in the Gibbs energy parameters, the mixing enthalpy and excess heat capacity can be determined that still correspond to sensible phase diagrams.

3 The LiF-KF System

In this study, the LiF-KF system was analyzed. Both the phase diagram and the thermodynamic properties mixing enthalpy and excess heat capacity were evaluated. This chapter provides some principle information about the LiF-KF system and its phase diagram and the mixing enthalpy and excess heat capacity.

LiF-KF is a fluoride salt, which makes it efficient at energy storage and convenient energy transfer material. Pure fluoride salts have high thermodynamic stability up to high temperatures and low vapor pressure [10]. Composing mixtures of these salts makes it possible to influence characteristics of the systems, which makes its use beneficial in molten salt reactors [10].

3.1 Phase Diagram

The phase diagram of the LiF-KF system is very simple with one eutectic point around the composition point X=0.5. This phase diagram is shown in figure 1 together with the experimental values determined by Aukrust et al. (1960) [17].

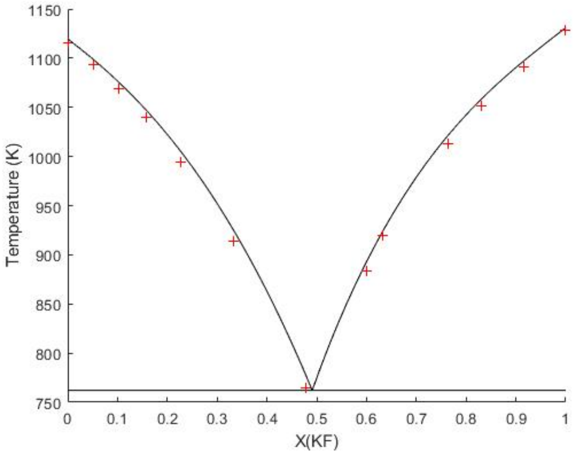

Figure 1: The phase diagram of a LiF-KF system. The vertical axis represents the temperature in Kelvin and the horizontal axis the composition X. The red crosses represent the phase diagram points experimentally determined by Aukrust et al. (1960).

3.2 Mixing Enthalpy

The enthalpy of a system is defined as [11]

$$H = U + pV, \tag{1}$$

where U represents the energy, p the pressure and V the volume of a system. After two miscible liquids are mixed, molecule interaction occurs. Due to this interaction, positive or negative heat is generated. This energy change is defined as the mixing enthalpy, and depends on the mixing ratio [12]. The mixing enthalpy is defined as the difference between the enthalpy of the solution before and after mixing, as shown in equation (2) [18]:

$$\Delta_{mix} H_m = \sum_i x_i (\overline{H}_{m,i} - H_{m,i}), \tag{2}$$

where $\Delta_{mix} H_m$ is the enthalpy of mixing, x_i the molar fraction, $\sum_i x_i \overline{H}_{m,i}$ is the sum of partial molar enthalpy of component i and $\sum_i x_i H_{m,i}$ is the stoichiometric sum of the pure end member's enthalpy of component i [5]. The excess enthalpy of mixing is defined as [18]:

$$\Delta_{mix} H_m^{excess} = \Delta_{mix} H_m - \Delta_{mix} H_m^{ideal} = \Delta_{mix} H_m, \tag{3}$$

where $\Delta_{mix} H_m^{excess}$ is the excess mixing enthalpy and $\Delta_{mix} H_m^{ideal}$ the ideal mixing enthalpy. Since the mixing enthalpy equals zero in ideal mixtures, the excess mixing enthalpy equals the mixing enthalpy as shown in equation (3).

The mixing enthalpy of the LiF-KF system was determined experimentally by Capelli et al. (2013) at 1121 K. These values are plotted in figure 2 together with the mixing enthalpy values as computed using the CALPHAD model.

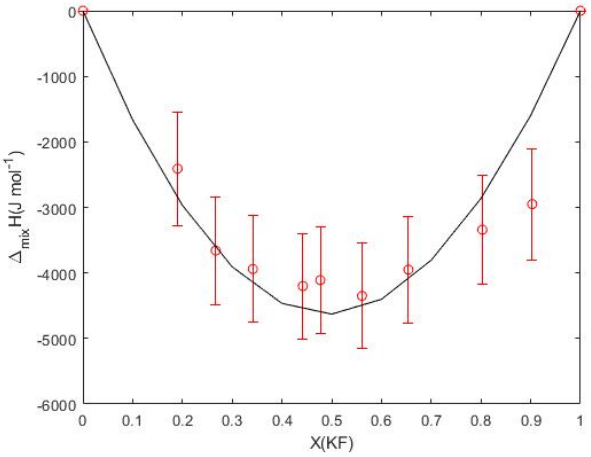

Figure 2: The red points represent the mixing enthalpy values as measured by Capelli et al. (2013) at a temperature of 1121K. This figure includes the corresponding error bars at each point. The solid black line represents the mixing enthalpy values computed using the CALPHAD model.

3.3 Heat Capacity

Knowing heat capacity information of a system is essential since this determines the amount of heat energy stored in a system at a certain temperature. The heat capacity is defined as the derivative of enthalpy with regards to temperature. Using this definition and looking at equation (1), at constant pressure the heat capacity of a system can be calculated using equation (4):

$$C_p = \frac{dH}{dT}. \tag{4}$$

The heat capacity of the LiF-KF system at different composition points has been experimentally determined by Beilmann et al. (2013) using drop calorimetry to measure enthalpy increments [13]. From these results the heat capacity can be calculated using equation (4). The results of this experiment are shown in figure 3.

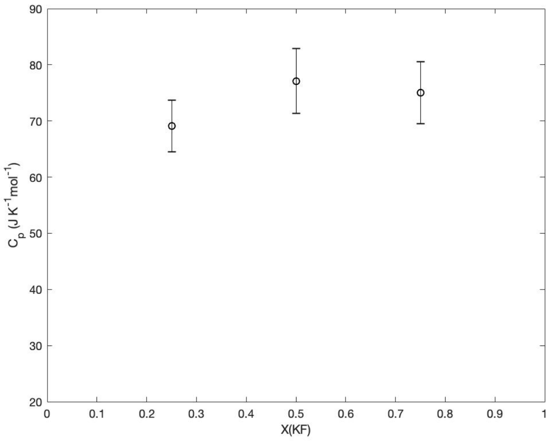

Figure 3: The heat capacity values as measured by Beilmann et al. (2013) at the composition points ($Li_{0.75}K_{0.25}$)F, ($Li_{0.50}K_{0.50}$)F and ($Li_{0.25}K_{0.75}$)F at temperatures ranges 1158 K - 1344 K, 884 K - 1083 K and 1157 K - 1330 K, respectively. This figure includes the corresponding error bars at each point.

By subtracting the lattice contribution from the obtained heat capacity values, the excess heat capacity values could be obtained [18]. These values are plotted in figure 4 together with the excess heat capacity values computed from the CALPHAD model.

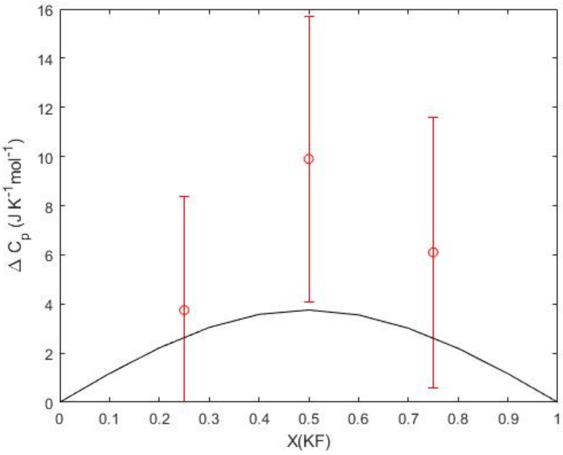

Figure 4: The red points represent the excess heat capacity values as measured by Beilmann et al. (2013) at the composition points $(Li_{0.75}K_{0.25})F$, $(Li_{0.50}K_{0.50})F$ *and* $(Li_{0.25}K_{0.75})F$ *at temperatures ranges 1158 K - 1344 K, 884 K - 1083 K and 1157 K - 1330 K, respectively. These values are plotted with the corresponding error bars. The solid black line represents the excess heat capacity data computed from the CALPHAD model.*

4 Theory

4.1 CALPHAD modelling method

In thermodynamics, the chemical stability of a system are described by the value of the Gibbs energy. The chemical stability increases as the Gibbs energy decreases [1]. By minimizing the Gibbs energy, the phase equilibrium can be determined. In order to construct a temperature-composition (T-X) phase diagram, the Gibbs energy functions of all the phases present in the system have to be known. These functions can be acquired through thermodynamic assessment.

4.1.1 Gibbs Energy

First we analyze the Gibbs energy of a pure element. The Gibbs energy can be derived from the temperature dependence of enthalpy and entropy. This relation can be described using the Helmholtz formula shown in equation (5):

$$G(T) = H(T) - S(T)T. \tag{5}$$

Since it is often difficult to obtain the enthalpy function $H(T)$ and entropy funtion $S(T)$, we can rewrite this equation to obtain equation (6) [2],

$$G(T_0) = \Delta_f H^o(298.15) + \int_{298.15}^{T_0} C_p dT - T_0(S^o(298.15) + \int_{298.15}^{T_0} \frac{C_p}{T} dT), \tag{6}$$

in which $\Delta_f H^o(298.15)$ is the standard enthalpy of formation and $S^o(298.15)$ the standard absolute entropy, both at a standard state temperature of T=298.15 K. This describes the Gibbs function for temperatures higher than 298.15 K. Combining the enthalpy of formation, standard entropy and heat capacity equation, we can also write the Gibbs energy function analytically as in equation (7) [2].

$$G(T) = a + bT + cTln(T) + \sum d_i T^i \tag{7}$$

The coefficients a, b and c are constants which are determined from experimental data.

4.1.2 Gibbs energy of binary (solid or liquid) solutions

When considering binary solutions, we must also take the mixing contribution into account, as can be observed in equation (8).

$$G(T) = x_1 G_1(T) + x_2 G_2(T) + G^{mixing}(T). \tag{8}$$

In which G_1 and G_2 are the Gibbs energy formulas for the pure components and x_1 and x_2 account for the weighted contributions of these components to the total Gibbs energy. $G^{mixing}(T)$ is the mixing contribution of these end-members to the total Gibbs energy. [1] The Gibbs energy of mixing can be divided into the ideal mixing energy and the excess mixing energy. The ideal Gibbs mixing energy can be described by equation (9):

$$G^{mixing,ideal}(T) = x_1 RTln(x_1) + x_2 RTln(x_2), \tag{9}$$

where once again the coefficients x_1 and x_2 account for the weighted contributions of the two components of the system to the ideal mixing Gibbs energy. Additionally, R represents the universal gas constant. Combining equations (8) and (9), equation (10) can be obtained.

$$G(T) = x_1G_1(T) + X_2G_2(T) + x_1RTln(x_1) + x_2RTln(x_2) + G^{excess}. \tag{10}$$

In equation (10), G^{excess} represents the excess Gibbs energy of mixing. This parameter is difficult to determine experimentally, thus optimization is required. Thermodynamic models are used for this optimization. One of these models is the modified quasi-chemical model, used for binary liquid solutions. [1]

4.1.3 Modified Quasi-Chemical Model

When considering binary liquid solutions, the modified quasi-chemical model in the quadruplet approximation proposed by Pelton and coworkers [2] has been used to optimize excess Gibbs energy functions. [1] In this model, atoms A and B are distributed over the sites of a quasi lattice [4]. The second nearest neighbor pair reaction (SNN) for the LiF-KF system is shown schematically in figure 5.

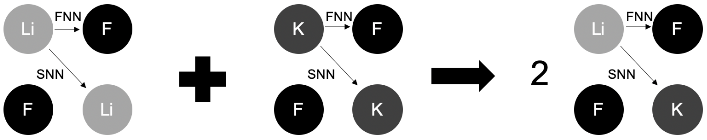

Figure 5: Exchange reaction of the two different cations (Li and K) on the cation sublattice in the SNN approximation [1]. This figure indicates the first nearest neighbors (FNN) as well as the second nearest neighbors (SNN) in the sublattice.

The SNN exchange reaction is given by equation (11):

$$(A - X - A) + (B - X - B) = 2(A - X - B); \Delta g_{AB/X}, \tag{11}$$

in which A and B represent cations and X an anion. [1] In this study, the cations represented as A and B are the Li^+ and K^+ atoms, and the X anion represents the F^- atoms. The $\Delta g_{AB/X}$ represents the Gibbs energy changes as a result of this reaction, which in this study equals $\Delta g_{LiK/F}$. These Gibbs energy changes are given by equation (12),

$$\Delta g_{AB/X} = \Delta g^0_{AB/X} + \sum_{(i+j)\geq 1} g^{ij}_{AB/X}\chi^i_{AB/X}\chi^j_{BA/X}, \tag{12}$$

where $\Delta g^0_{AB/X}$ and $g^{ij}_{AB/X}$ are obtained from optimization of experimental data and are composition independent [1]. The $\chi_{AB/X}$ and the $\chi_{BA/X}$ terms are composition variables defined in equation (13),

$$\chi_{AB/X} = \frac{X_{AA}}{X_{AA} + X_{AB} + X_{BB}}, \tag{13}$$

where X_{AA}, X_{AB} and X_{BB} are the mole fractions of the cation-cation pairs. In this study, for the LiF-KF system the formulas optimized in the literature for the excess Gibbs energy are shown in equations (14), (15) and (16). [5]

$$\Delta g^0_{LiK/F} = -7614.88 + 4.8609712T - 0.0016736T^2 \tag{14}$$

$$g^{10}_{LiK/F} = -958.136 + 1.2552T \tag{15}$$

$$g^{01}_{LiK/F} = -690.36 + 1.96648T \tag{16}$$

4.2 Polynomial Chaos Expansion

The Polynomial Chaos Expansion (PCE) is a series expansion used to approximate the output of a model as a polynomial function of the stochastic input variables. In order to use the polynomial chaos expansion, the stochastic input variables must first be defined. This set of stochastic input variables is denoted as $\boldsymbol{\xi}^N = (\xi_1, \xi_2, \xi_3, ...\xi_N) \in \mathbb{R}^N$, and represent the distinct sources of uncertainty present in the system. The stochasticity of each of these variables corresponds to a probability density funtion $p_{\boldsymbol{\xi}}(\boldsymbol{\xi})$. In this study, this set of stochastic input variables consists of the parameters in the excess Gibbs energy equations (14), (15) and (16). The probability distribution of these stochastic input variables should be chosen to agree with the real physical problem [6]. The responses of interest, of which we want to determine the variability, are given by equation (17):

$$R(\boldsymbol{\xi}) = \sum_{k=0}^{P} r_k \boldsymbol{\Psi}_k(\boldsymbol{\xi}), \tag{17}$$

where r_k represents the polynomial chaos coefficient and $\boldsymbol{\Psi}_k(\boldsymbol{\xi})$ the polynomial chaos basis vectors. Thus, if the basis vectors are defined and the coefficients computed, the dependence of the responses of interest $R(\boldsymbol{\xi})$ on the stochastic input variables $\boldsymbol{\xi}$ can be determined.

4.2.1 Polynomial Chaos Basis Vectors

For each stochastic input variable ξ, polynomials orthogonal to the probability density function of that stochastic input variable should be chosen [7]. This principle is represented in the Wiener-Askey scheme. When considering uniformly distributed variables, Legendre polynomials are a suitable choice. The basis vectors can be written as [8]

$$\boldsymbol{\Psi}_k(\boldsymbol{\xi}) = \prod_{j=1}^{N} \psi_{j,\gamma_{k,j}}(\xi_j), \tag{18}$$

where $\psi_{j,\gamma_{k,j}}(\xi_j) \in (He_{\gamma_{k,j}}(\xi_j), P_{\gamma_{k,j}}(\xi_j), La_{\gamma_{k,j}}(\xi_j), ..)$ and represents different polynomials (Hermite, Legendre, Laguerre,...). $\boldsymbol{\gamma_k}$ represents a multi-index analogous with the different orders $\boldsymbol{\gamma_k} = (\gamma_{k,1}, \gamma_{k,2}, ..., \gamma_{k,N})$. Index j corresponds to the different random variables [6]. One can then choose to include all multidimensional polynomials having a combined order of at most O, this is a full O^{th} order polynomial chaos basis set given by [6]

$$\Gamma(O) = \left\{ \boldsymbol{\Psi}_k(\boldsymbol{\xi}) : \sum_{j=1}^{N} \gamma_{k,j} \leq O \right\}. \tag{19}$$

The number of basis vectors in the expansion in equation (15) can be determined using equation (20):

$$P + 1 = \frac{(N+O)!}{(N!O!)}, \tag{20}$$

where N is the amount of input parameters and O the maximum polynomial order. The full polynomial chaos basis set considers all input parameters and their interactions. However, according to the "sparsity of effects" principle, responses are often dominated by main effects and low order interactions [6,14]. Thus, not all of the basis vectors need to be considered in order to obtain accurate responses. Using sparse PCEs has noise robustness and a lower computational cost as a result [14].

Since the polynomial basis vectors are chosen to always be orthogonal to the probability density functions of the stochastic input parameters, the following relationship holds:

$$\langle \boldsymbol{\Psi}_k(\boldsymbol{\xi}), \boldsymbol{\Psi}_l(\boldsymbol{\xi}) \rangle = \int \int ... \int \boldsymbol{\Psi}_k(\boldsymbol{\xi}), \boldsymbol{\Psi}_l(\boldsymbol{\xi}) p_{\boldsymbol{\xi}}(\boldsymbol{\xi}) d\xi_1 d\xi_2 ... d\xi_N \begin{cases} 0, k \neq l \\ h_k^2, k = l, \end{cases} \tag{21}$$

where h_k^2 is the norm of the k^{th} basis vector.

4.2.2 Polynomial Chaos Coefficient

There are two different methods that can be used to calculate the PC coefficient. The first method is the projection approach. This method reproduces each PC coefficient as a multidimensional integral which can be calculated by simulation or quadrature. The second method is the regression approach, where the PC coefficients are estimated by minimizing the mean square error of the response approximation [19] $(R(\xi) - \sum_j r_j \Psi_j(\xi))^2$. In this section the calculation of the PC coefficient according to the projection approach, using quadratures and cubatures, will be described further.

Using the orthogonality of the basis vectors, the polynomial chaos coefficients can be determined using spectral projection [8]:

$$\begin{aligned} r_k = \frac{\langle R(\boldsymbol{\xi}), \boldsymbol{\Psi}_k(\boldsymbol{\xi}) \rangle}{\langle \boldsymbol{\Psi}_k(\boldsymbol{\xi}), \boldsymbol{\Psi}_k(\boldsymbol{\xi}) \rangle} &= \frac{\int \int ... \int R(\boldsymbol{\xi}) \boldsymbol{\Psi}_k(\boldsymbol{\xi}) p_{\boldsymbol{\xi}}(\boldsymbol{\xi}) d\xi_1 d\xi_2 ... d\xi_N}{\langle \boldsymbol{\Psi}_k(\boldsymbol{\xi}), \boldsymbol{\Psi}_k(\boldsymbol{\xi}) \rangle} \\ = \frac{1}{h_k^2} \int_{a_1}^{b_1} \int_{a_2}^{b_2} ... \int_{a_N}^{b_N} & R(\boldsymbol{\xi}) \prod_{j=1}^{N} \psi_{j,\gamma_{k,j}}(\xi_j) p_{\boldsymbol{\xi}}(\boldsymbol{\xi}) d\xi_1 d\xi_2 ... d\xi_N. \end{aligned} \tag{22}$$

This equation includes the unknown dependence of the response on the input parameters. The numerator in equation (22) can be approximated by a cubature formula which is assembled from one-dimensional quadratures [6],

$$I^{(1)} f = \int_a^b f(\xi_j) p_{\xi_j}(\xi_j) d\xi_j \approx Q_{lev}^{(1)} f = \sum_{i=1}^{n_{lev}} f\left(\xi_{j,lev}^{(i)}\right) w_{lev}^{(i)}. \tag{23}$$

Equation (23) gives the quadrature formula for a general function $f(\xi_j)$, where $\xi_{j,lev}^{(i)} \in [a, b]$ are predefined quadrature points and $w_{lev}^{(i)} \in \mathbb{R}$ weights according to the distribution $p_{\xi_j}(\xi_j)$ and the quadrature rule. Additionally, the accuracy of the quadrature approximation increases with the *lev* index. Using tensorization, this one-dimensional integral can then be extended into a multi-dimensional integral [8,9],

$$\begin{aligned} I^{(N)}f &= \left(Q^{(1)}_{lev1} \otimes Q^{(2)}_{lev2} \otimes ... \otimes Q^{(N)}_{levN}\right) f \\ &= \sum_{i_1=1}^{n_{lev_1}} \sum_{i_2=1}^{n_{lev_2}} ... \sum_{i_N=1}^{n_{lev_N}} \left(w^{(i_1)}_{lev1} w^{(i_2)}_{lev2} ... w^{(i_N)}_{levN}\right) f\left(\xi^{i_1}_{1,lev_1} \xi^{i_2}_{2,lev_2}, ..., \xi^{i_N}_{N,lev_N}\right) \\ &= \sum_{i}^{n} f\left(\boldsymbol{\xi}^{(\boldsymbol{i})}\right) w^{(i)}. \end{aligned} \tag{24}$$

Equation (24) includes a full tensorization to obtain cubature points from the quadrature points. The problem that comes with using the full tensorization is that the number of function assessments required increases with the dimensionality. Smolyak sparse grids can be used to reduce the amount of required function assessments and thus significantly decrease the computational cost. The principle behind these sparse grids is similar to that of the sparse basis vectors. Defining the difference relation $\Delta^{(1)}_{lev} f = Q^{(1)}_{lev} f -^{(1)}_{lev-1} f$, where $Q^{(1)}_0 f = 0$, the quadrature rules can be written as $Q^{(1)}_{lev} f = \sum_{l=1}^{lev} \Delta^{(1)}_{l} f$. After substituting this relation into equation (24), equation (25) is obtained:

$$Q^{(N)}_{lev} f = \sum_{\boldsymbol{l} \in \tau(lev)} \left(\Delta^{(1)}_{l_1} \otimes \Delta^{(1)}_{l_2} \otimes ... \otimes \Delta^{(1)}_{l_N}\right) f = \sum_{\boldsymbol{l} \in \tau(lev)} \Delta^{(N)}_{\boldsymbol{l}} f. \tag{25}$$

In equation (25), the multi-index $\boldsymbol{l} = (l_1, l_2, ..., l_N)$ is introduced to distinguish between different grids, and the set $\tau(lev)$ contains the included multi-indices that depend on the level index $\boldsymbol{lev} = (lev_1, lev_2, ..., lev_N)$ which contains different quadrature levels in different directions. The original Smolyak grid set is given by [6]:

$$\tau^{Smolyak}(lev) = \left\{ \boldsymbol{l} : \sum_{j=1}^{N} l_j \leq lev + N - 1 \right\}, \tag{26}$$

where the maximum grid equals $lev - 1$. Using this set and equations (22), (24) and (25), the following equation for calculating the polynomial chaos coefficients can be obtained:

$$r_k = \frac{1}{h_k^2} \sum_{\boldsymbol{l} \in \tau^{Smolyak}(lev)} \Delta^{(N)}_{\boldsymbol{l}} (R\boldsymbol{\Psi}_k) = \frac{1}{h_k^2} \sum_{i}^{n} R\left(\boldsymbol{\xi}^{(i)}\right) \boldsymbol{\Psi}_k \left(\boldsymbol{\xi}^{(i)}\right) w^{(i)}, \tag{27}$$

the cubature points $\boldsymbol{\xi}^{(i)}$ and corresponding weights $w^{(i)}$ correlate with the grid given by equation (24), determined with the chosen maximum grid order ($lev - 1$). Thereafter, the model can be run using these input values, and the response values $R\left(\boldsymbol{\xi}^{(i)}\right)$ can be obtained. Combining this with the value of the basis vectors $\boldsymbol{\Psi}_k\left(\boldsymbol{\xi}^{(i)}\right)$ and the weights $w^{(i)}$ at those same cubature points, ultimately the polynomial chaos coefficients can be calculated using equation (27).

5 Experimental Method

The Polynomial Chaos expansion (PCE) will be used to determine the uncertainty range of the mixing enthalpy, the excess heat capacity and the excess Gibbs energy parameters. In this study, the program $Factsage 7.2$ [19] is used to perform the thermodynamic calculations and the open source MATLAB openGPC written by Zoltán Perko is used for the polynomial chaos calculations.

5.1 Linking OpenGPC with FactSage 7.2

In order to apply the polynomial chaos expansion to the LiF-KF system, it is necessary to link the thermodynamic calculations executed by FactSage with the polynomial chaos expansion calculations carried out through the openGPC scripts. In order to achieve this, the GPC settings have to be defined first. Based on these settings, the openGPC generates different input values, which in this study are the Gibbs energy parameters. Then, using a MATLAB script, these values are put into a .TXT file and a MACRO is activated using these parameter rows as input value parameters. The MACRO then initiates a calculation of the phase diagram data and equilibrium data in FactSage. The equilibrium data contains calculations of, among other quantities, the mixing enthalpy and the excess heat capacity at different compositions. This data can then be extracted from the calculated files and the openGPC can then analyze the relationship between the input values (Gibbs energy parameters) and the responses calculated by FactSage (Phase diagram points and the chosen quantities in equilibrium files). In this study, the heat capacity and mixing enthalpy data will be extracted from the equilibrium files and thus included in the PCE calculation. Additionally, the phase diagram points are interpolated with a value 0.01 and 0.001 in composition. Ultimately, a GPC class is generated which contains a variety of functions that can be used to assess the uncertainty of mixing enthalpy, the excess heat capacity and the Gibbs energy parameters.

5.2 PCE Validation for Accurate Phase Diagram Computation

When creating the GPC class, the GPC settings must be defined. When generating GPC classes with different GPC Settings, the accuracy of the samples generated by the GPC class with respect to the responses computed by FactSage will vary significantly. For research into the uncertainty, it is vital to generate a GPC class with the highest accuracy. To test this, GPC Classes were generated using different GPC settings. Additionally, the step size used to extract the data can be varied, which affects the accuracy.

In this study, step size values of 0.01 and 0.001 in the composition were tested. In previous research conducted by T. Schuijtvlot [5], a higher maximum polynomial order and an even maximum grid order resulted in a lower uncertainty. Due to FactSage 7.2 memory constraints, the maximum polynomial order that can be used is 6. Therefore, mainly maximum polynomial order 6 and even maximum grid orders were tested with the different interpolations of 0.01 and 0.001. In order to determine the GPC settings which yield the most accurate results, the maximum polynomial order (MPO) and maximum grid order (MGO) were varied and the corresponding errors computed. To eliminate the noise that occurred when using an interpolation of 0.001, both calculation methods (calcMethod) regression and projection were tested. The open GPC settings and their values implemented in this study are displayed in tables 1 and 2.

Table 1: The GPC settings used when generating a GPC class including ΔH^{mixing} data. A,B,D,E,F and G represent the Gibbs energy parameters in equations (14), (5) and (16).

ΔH^{mixing}	
Variable names	A, B, D, E, F, G
Polynomial type	Legendre
Means	- 7614.88, 4.8609712, 958.136, 1.2552, 690.36, 1.96648
Half interval	3807.44, 2.4304856, 479.068, 0.6276, 345.18, 0.98324
MPO	4, 6, 7
MGO	5, 6
calcMethod	projection, regression

Table 2: The GPC settings used when generating a GPC class with ΔC_p data included. This table is similar to table 1, except that when generating the GPC the 'C' value is included in the computations, since it influences the heat capacity. The maximum polynomial order (MPO), maximum grid order (MGO) and calculation method (calcMethod) used are those that have proved most accurate in the mixing enthalpy GPC accuracy calculation.

ΔC_p	
Variable names	A, B, C, D, E, F , G
Polynomial type	Legendre
Means	- 7614.88, 4.8609712, 0.0016736, 958.136, 1.2552, - 690.36, 1.96648
Half interval	3807.44, 2.4304856, 0.0008368, 479.068, 0.6276, 345.18, 0.98324
MPO,MGO, calcMethod	Most accurate

The means are the stochastic input variables, which in this study are $\boldsymbol{\xi}$ = (A,B,C,D,E,F,G), which represent the optimized excess Gibbs energy parameters in equations (14), (15) and (16). A, B and C represent the first, second and third parameters in equation (14), respectively. D and E represent the first and second parameters in equation (15), respectively. Lastly, F and G represent the first and second parameters in equation (16). The third Gibbs energy parameter in equation (14) C relates to the heat capacity, which is considered to have negligible influence on the phase diagram and mixing enthalpy data. For this reason, C is not considered when generating the mixing enthalpy GPC. However, when considering the excess heat capacity, this parameter is of importance since this is the parameter that mainly influences the heat capacity. Since experimental data is considered, the probability distribution outcome is defined as uniform between the bounds [0.5*mean, 1.5*mean], which is why the half interval is defined as 0.5*mean.

Once the different GPC classes were calculated, the mean absolute errors (MAE) and the root mean squared errors (RMSE) in the phase diagram data were computed. The MAE is defined as the average difference between the phase diagram points given by the GPC class and the points computed by FactSage. The RMSE is defined as the square root of the sum of all the separate errors squared divided by the amount of values (the errors are squared before they are averaged). The main difference between these two error indicators is that with the MAE each error carries equal weight, whereas with the RMSE the bigger errors contribute more to the final value of the RMSE than the smaller errors.

5.3 PCE Function Application

Once the GPC class is generated, its functions can be used to quantify the uncertainty. One of the functions that can be used is the sampler function. This function generates many different samples of the phase diagram and mixing enthalpy and excess heat capacity points that result from varying all of the Gibbs energy parameters uniformly. Another function which proved informative is the value function. Whilst the sampler function varies each input parameter and generates corresponding outputs, the value function enables the manual adjustment of input parameters. Using the value function, the dependence of the phase diagram on the different Gibbs energy parameters could independently be determined. Since equation (15) influences the left hand side and equation (16) influences the right hand side of the phase diagram, the parameters of these equations should be coupled when using the value function. This means that when varying the first parameter in equation (15), the first parameter in equation (16) should also be varied. The same coupling applies for the second parameters in equations (15) and (16) (parameters D + F and E + G are coupled).

5.4 Uncertainty Quantification based on Phase Diagram Data

Often the phase diagram data of a system is known experimentally, which is why a method of quantifying the uncertainty based on the phase diagram data is valuable. The method used for obtaining uncertainty in this study is as follows: first, a maximum deviation from the experimentally determined phase diagram of 10K was defined. Then, the functions in the computed GPC class were used to generate a large number of samples. Subsequently, the samples that were within the predefined margin were filtered and the corresponding mixing enthalpy and excess heat capacity data were extracted. From this data the maximum allowed uncertainty range in these quantities that still correspond to sensible phase diagrams (within the 10K margin of the experimentally determined phase diagram) could be determined.

5.4.1 Uncertainty in Mixing Enthalpy and Excess Heat Capacity

Once the mixing enthalpy and excess heat capacity data corresponding to the samples within the 10K margin were filtered, the equations used to quantify the uncertainty half range are as follows:

$$\Delta H^{mixing}_{uncertain}(X) = \frac{H^{mixing}_{max}(X) - H^{mixing}_{min}(X)}{2} + MAE_{GPC} \tag{28}$$

$$\Delta C_{p,uncertain}^{excess}(X) = \frac{C_{p,max}^{excess}(X) - C_{p,min}^{excess}(X)}{2} + MAE_{GPC}, \tag{29}$$

in which the maximum and minimum mixing enthalpy and excess heat capacity were evaluated at the same composition value point X. Additionally, the mean absolute error in the mixing enthalpy and excess heat capacity values generated by the GPC MAE_{GPC} is added. Because this value is very small, it was considered negligent.

5.4.2 Uncertainty assessment of input Gibbs Energy Parameters

The value function of the GPC class enables the variance of only one Gibbs energy parameter. Varying one parameter and filtering the GPC responses, the maximum uncertainty half range in each separate parameter can be assessed. This is the uncertainty in one parameter when assuming that the other parameters all have zero uncertainty. Since parameters D + F and E + G must be adjusted simultaneously (coupled), and the parameter C is considered to have little effect on the phase diagram values, in one dimension the uncertainty half range is calculated only for parameters A and B. This one-dimensional uncertainty half range can be calculated similarly to the uncertainty half range in the mixing enthalpy and excess heat capacity,

$$\xi_{uncertain} = \frac{\xi_{max} - \xi_{min}}{2} + MAE_{GPC}, \tag{30}$$

where ξ represents a Gibbs energy parameter. Additionally, two different parameters can be varied simultaneously and the two-dimensional uncertainty can be determined for different parameter combinations. When looking at the three-dimensional combinations of Gibbs energy parameters, the uncertainty of parameters A and B in combination with the coupled parameters D + F and E + G can also be investigated. Both the two-dimensional and the three-dimensional uncertainty can be displayed using scattering plots. Realistically, all the Gibbs energy parameters possess uncertainty, but this assessment is outside the scope of this research.

6 Results

6.1 Accuracy of the PCE

See table 3 for the uncertainty values for different maximum polynomial orders and step size in composition of 0.01 calculated with calculation method 'projection'.

Table 3: The mean absolute error in the phase diagram (MAE PD) and root mean squared error in the phase diagram (RMSE PD) are displayed for different maximum polynomial orders (MPO) and maximum grid orders (MGO). All the errors are calculated for a step size in composition of 0.01.

MPO	MGO	Step size	calcMethod	MAE PD (K)	RMSE PD (K)
4	5	0.01	projection	0.3155	1.1173
4	6	0.01	projection	0.2615	0.8028
6	6	0.01	projection	0.3941	1.9193
7	6	0.01	projection	0.5742	2.8958

Thus the setting with the smallest deviation from the FactSage responses with the step size of 0.01 in composition is the MPO 4 and MGO 6. This is consistent with the prior research conducted by T. Schuijtvlot [5]. However, by decreasing the step size to 0.001 the error can possibly be decreased even further. This was tested with the settings in table 4. To eliminate oscillations that the GPC analyzes with a step size this small (most likely due to FactSage noise), the calculation method was changed from projection to regression.

Table 4: The mean absolute error in the phase diagram (MAE PD) and root mean squared error in the phase diagram (RMSE PD) are displayed for maximum polynomial order (MPO) 4 and maximum grid order (MGO) 6 and step size in composition of 0.001. The errors corresponding to both calculation methods (calcMethod) projection and regression are displayed. This table also includes the percentile deviations from the mixing enthalpy values calculated with the optimized Gibbs energy parameters in equations (14), (15) and (16).

MPO	MGO	step size	calcMethod	MAE PD	RMSE PD
4	6	0.001	projection	0.2631	0.7796
4	6	0.001	regression	0.0653	0.1567

As can be observed in table 4, the error decreases significantly when decreasing the step size in composition and changing the calculation method to regression. Thus, the optimal GPC settings are MPO4 MGO6 with a step size i composition of 0.001 and calculation method regression. Because of this accuracy, these settings were used for the uncertainty quantification.

6.2 Obtained Uncertainty

Using the determined GPC settings, the uncertainty was quantified based on the phase diagram data. This was done for the input Gibbs energy parameters, the mixing enthalpy and the excess heat capacity.

6.2.1 Mixing Enthalpy

The mixing enthalpy values corresponding to the phase diagrams that are within the 10 K margin of the experimentally determined phase diagram were filtered. Firstly, this was done when varying all the Gibbs energy parameters A, B, D, E, F and G. The 0th, 50th and 100th percentile lines of the filtered mixing enthalpy values are plotted together with the values from Capelli et al. (2014) in figure 6.

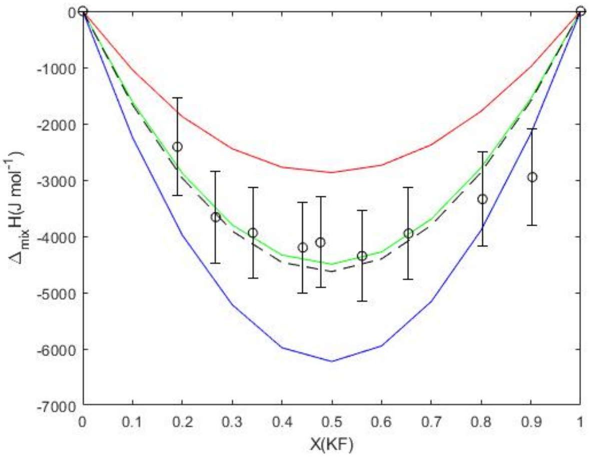

Figure 6: The 0th, 50th and 100th mixing enthalpy percentile lines filtered from 100000 samples, plotted with the experimental values from Capelli et al. (2014). The blue line represents the 0th percentile, the green line the 50th percentile and the red line represents the 100th percentile. The black dashed line represents the mixing enthalpy values computed with the original optimized input Gibbs energy parameters.

The uncertainty half range was determined at each composition point X, using equation (26). The results are shown in table 5.

Table 5: The uncertainty (Unc) half range in the mixing enthalpy values corresponding to the phase diagrams within a 10 K margin of the experimentally determined phase diagram. The uncertainty was calculated at different composition points $X(KF)$, using equation (28). The percentile deviations from the mixing enthalpy values calculated with the original optimized Gibbs energy parameters is also included (Unc (%)).

$X(KF)$	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
Unc (J·mol^{-1})	591.7	1059	1398	1596	1650	1571	1368	1046	593.1
Unc (%)	35.5	35.7	35.8	35.7	35.6	35.7	35.9	36.5	37.2

Thus the maximum uncertainty half range in the mixing enthalpy was found to occur at point X=0.5 and equals 1650 J·mol^{-1}. The maximum percentile deviation from the mixing enthalpy values computed with the original optimized Gibbs parameters occurs at point X=0.9 and equals 37.2 %.

This uncertainty half range and percentile uncertainty can also be determined for the samples generated when only adjusting certain input parameters. The uncertainty half ranges computed for certain parameter combinations at composition point X=0.5 are shown in table 6.

Table 6: The uncertainty (Unc) half range in the mixing enthalpy values corresponding to the phase diagrams within a 10 K margin of the experimentally determined phase diagram. The uncertainty half range was calculated at composition point $X(KF)$ = 0.5, for different adjusted Gibbs energy parameters (Adj. Var.). This table also includes the percentile deviations from the excess heat capacity value at $X(KF)$ = 0.5, calculated with the original optimized Gibbs energy parameters.

Adj. Var.	A	B	DF	EG	ADF	AEG	BDF	BEG
Unc ($\mathrm{Jmol^{-1}}$)	202.2	33.14	147.8	8.029	210.5	396.5	146.8	72.61
Unc (%)	4.4	0.7	3.2	0.2	4.5	8.6	3.2	1.6

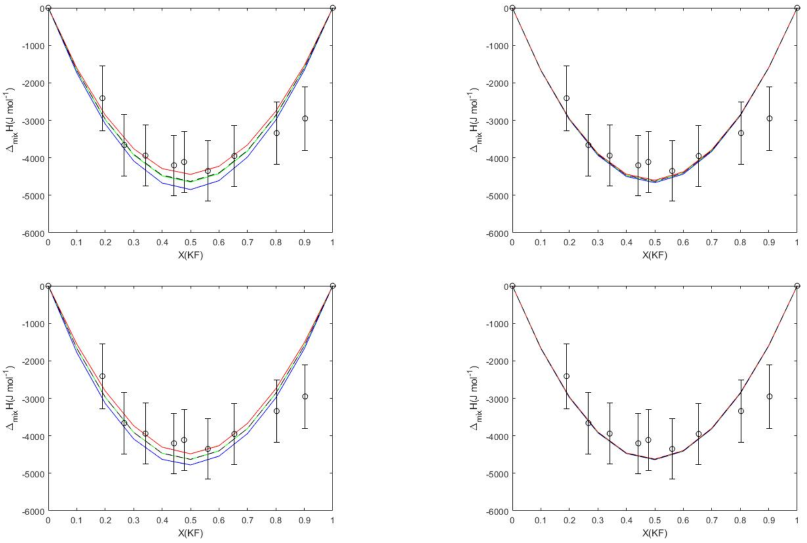

Figure 7: The 0th, 50th and 100th mixing enthalpy percentile lines filtered from 100000 samples, plotted with the experimental values from Capelli et al. (2014). The blue line represents the 0th percentile, the green line the 50th percentile and the red line represents the 100th percentile. The black dashed line represents the mixing enthalpy values computed with the original optimized input Gibbs energy parameters. TOP-LEFT: only the parameter A was adjusted. TOP-RIGHT: only parameter B was adjusted. BOTTOM-LEFT: parameters D and F were adjusted simultaneously. BOTTOM-RIGHT: parameters E and G were adjusted simultaneously

As can be observed in both the plots in figure 7 as the uncertainty (half range) values in table 6, the considered parameters have a small uncertainty compared to the uncertainty when all parameters are varied simultaneously in figure 6. To further investigate which parameters cause this large deviation, the parameter combinations A + B and D + E + F + G were varied simultaneously and the results are plotted in figure 8.

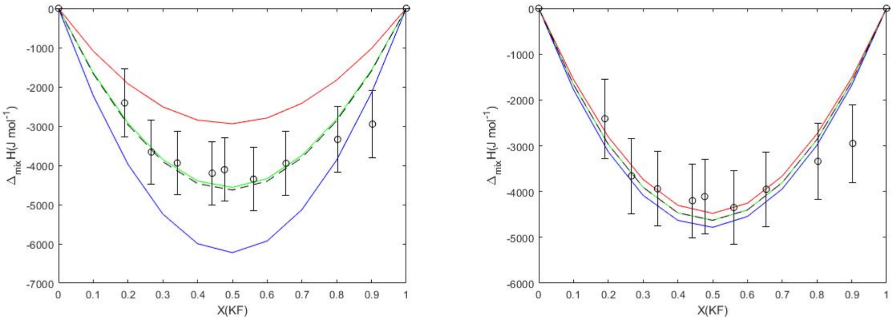

Figure 8: The 0th, 50th and 100th mixing enthalpy percentile lines filtered from 100000 samples, plotted with the experimental values from Capelli et al. (2014). The blue line represents the 0th percentile, the green line the 50th percentile and the red line represents the 100th percentile. The black dashed line represents the mixing enthalpy values computed with the original optimized input Gibbs energy parameters. LEFT: only input Gibbs energy parameters A and B were adjusted simultaneously. RIGHT: only input Gibbs energy parameters D, E, F and G were adjusted simultaneously

The uncertainty half ranges corresponding to the plots in figure 8 at $X = 0.5$ equal 1642 J$\cdot$mol^{-1} and 152.5 J$\cdot$mol^{-1}, respectively. This equals 35.5 % and 3.3 % deviation from the mixing enthalpy values calculated with the original optimized Gibbs energy parameters. Thus it can be observed that the majority of the deviation that occurs in the filtered mixing enthalpy values in figure 6 is due to the combination of the Gibbs energy parameters A and B.

6.2.2 Excess Heat Capacity

Similarly as for the mixing enthalpy, the minimum and maximum filtered excess heat capacity values are plotted in figure 9 along with the experimental values from Beilmann et al. (2013).

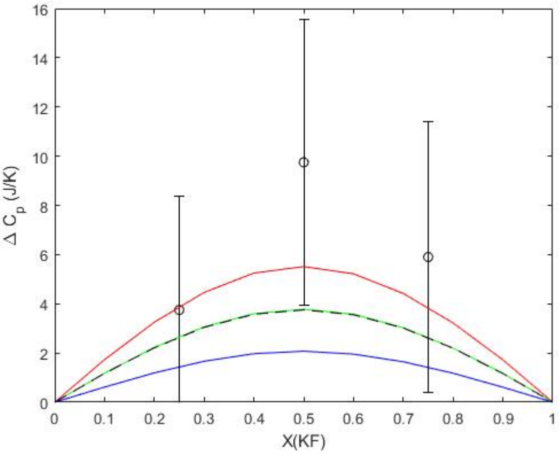

Figure 9: The 0th, 50th and 100th excess heat capacity percentile lines filtered from 100000 GPC samples, plotted with the experimental values from Beilmann et al. (2013) including the error bars in the heat capacity. The blue line represents the 0th percentile, the green line the 50th percentile and the red line represents the 100th percentile. The black dashed line represents the excess heat capacity values corresponding to the original optimized Gibbs energy parameters.

The uncertainty half range was calculated at the different composition points $X(KF)$. The results are shown in table 7.

Table 7: The uncertainty (Unc) half range in the excess heat capacity values corresponding to the phase diagrams within a 10 K margin of the experimentally determined phase diagram.The uncertainty was calculated at different composition points $X(KF)$, using equation (29). This table also includes the percentile deviations from the excess heat capacity values calculated with the optimized Gibbs energy parameters in equations (14), (15) and (16).

$X(KF)$	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
Unc (J·K^{-1}·mol^{-1})	0.5521	1.010	1.367	1.596	1.676	1.594	1.363	1.008	0.5488
Unc (%)	47.1	45.6	44.9	44.6	44.7	44.8	45.2	46.0	47.2

From the data in table 7 it can be seen that the maximum found uncertainty half range is at the composition $X = 0.5$ and equals 1.676 J·K^{-1}·mol^{-1}. The maximum percentile deviation from the excess heat capacity values calculated with the original Gibbs energy parameters occurs at X=0.9 and equals 47.2 %.

Here, the uncertainty half range and percentile uncertainty could also be determined for the samples generated and filtered when only adjusting certain input parameters. The results computed for certain parameter combinations at X=0.5 are shown in table 8.

Table 8: The uncertainty (Unc) half range in the excess heat capacity values corresponding to the phase diagrams within a 10 K margin of the experimentally determined phase diagram. The uncertainty half range was calculated at composition point $X(KF)$ = 0.5, for different adjusted Gibbs energy parameters (Adj. Var.). This table also includes the percentile deviations from the excess heat capacity value at $X(KF)$ = 0.5, calculated with the original optimized Gibbs energy parameters.

Adj. Var.	A	B	C	DF	EG	AC	BC	ABC	CDF
Unc ($\mathrm{J \cdot K^{-1} \cdot mol^{-1}}$)	0.051	0.024	0.69	0.015	0.014	1.48	1.52	1.68	1.14
Unc (%)	1.3	0.63	18.2	0.39	0.37	38.9	40.0	44.2	30.0

In table 8 it can be seen that individually, parameter C causes the largest deviation in the filtered excess heat capacity values. Especially when combining the variance of this parameter with the variance of parameter A and B, this uncertainty half range becomes particularly large. To illustrate the obtained results, some of these filtered excess heat capacity curves are plotted in figure 10.

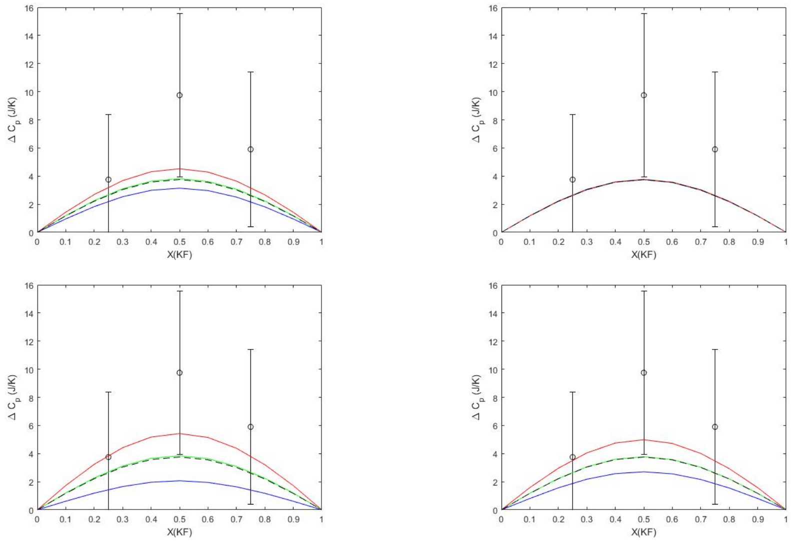

Figure 10: The 0th, 50th and 100th excess heat capacity percentile lines filtered from 100000 samples, plotted with the experimental values from Beilmann et al. (2013). The blue line represents the 0th percentile, the green line the 50th percentile and the red line represents the 100th percentile. The black dashed line represents the excess heat capacity values computed with the original optimized input Gibbs energy parameters. TOP-LEFT: only the parameter C was varied. TOP-RIGHT: only parameters D and F were varied. BOTTOM-LEFT: parameters A, B and C were varied simultaneously. BOTTOM-RIGHT: parameters C, D and F were adjusted simultaneously.

6.2.3 Excess Gibbs Energy parameters

The results obtained after filtering 100000 samples and calculating the one-dimensional uncertainty half range using equation (28) are displayed in table 9.

Table 9: The maximum one-dimensional uncertainty half range in Gibbs energy parameters A and B after filtering 100000 samples. This is the maximum uncertainty in one variable assuming all other parameters have zero uncertainty. The uncertainty half range in % compared to the original optimized Gibbs parameter values is also included.

Gibbs energy parameter	A (J·mol^{-1})	B (J·K^{-1}·mol^{-1})
Uncertainty half range	217.68	0.2828
Uncertainty half range (%)	2.86	5.82

The filtered values of A and B are displayed in histograms in figure 7 to illustrate the distribution of the filtered parameter values.

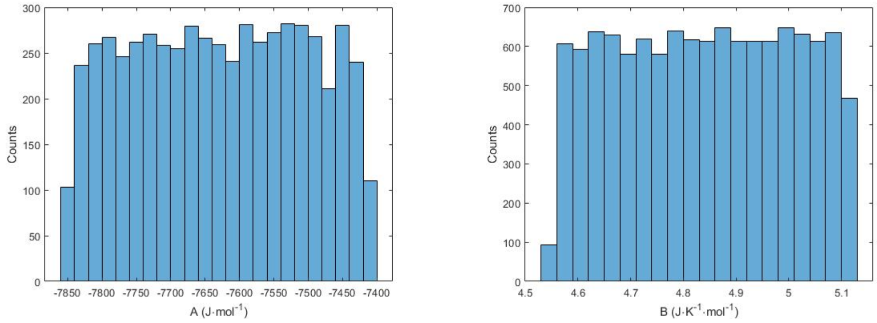

Figure 11: Two histogram plots of the values of the first parameter A and second parameter B in equation (12) that result in phase diagrams within the 10K margin of the experimental phase diagram. These are the histograms obtained when assuming only the parameter in question has an uncertainty.

In the histograms in figure 11 it can be observed that the distribution of the Gibbs energy parameters A and B corresponding to the filtered phase diagrams is virtually uniform within the uncertainty half range.

These two parameters can also be coupled so that both are varied simultaneously and as a result the parameter combinations that produce phase diagrams within the margin could be determined. The results of this filtering for parameters A and B are shown in figure 12.

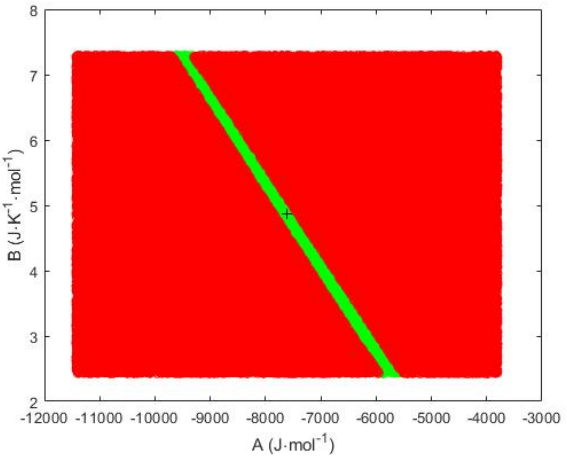

Figure 12: A point plot of the first Gibbs energy parameter A and the second Gibbs energy parameter B in equation (14) from 100000 samples generated by the GPC. The green points represent the parameter combinations that produce phase diagrams within a margin of 10K of the experimentally determined phase diagram, and the red points represent those outside of this phase diagram margin. The black cross represents the original optimized Gibbs energy parameter.

In this figure it can be observed that the parameter combinations within the margin are distributed as a slanting line. Thus, if parameter A becomes more negative, B has to become more positive in order to still produce sensible phase diagram results.

In two dimensions, the two coupled parameter combinations, D + F and E + G, that still give phase diagrams within the margin can be filtered. These are plotted in figure 13.

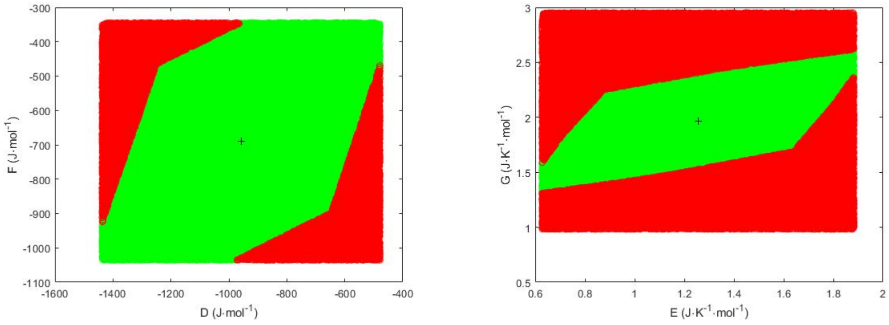

Figure 13: Two-dimensional point plots of 100000 parameter combinations. The green points represent the points within 10K of the experimental phase diagram and the red points represent those outside of this margin. The black cross represents the original optimized Gibbs energy parameter. LEFT: the first Gibbs energy parameters in equations (13) and (14), D and F. RIGHT: the second Gibbs energy parameters in equations (13) and (14), E and G.

When looking at the point plots in figure 13 it can be observed that for both coupled variable combinations, many combinations of these variables result in phase diagrams within the 10K margin. Especially comparing to the point plot in figure 12, the maximum allowed uncertainty in the parameter combinations of A and B seems to be much smaller than in the parameter combinations of coupled variables D + F and E + G.

Four three-dimensional point plots of 100000 points are displayed which illustrate the maximum allowed uncertainty range of parameters A and B in combination with the coupled parameters D + F and E + G.

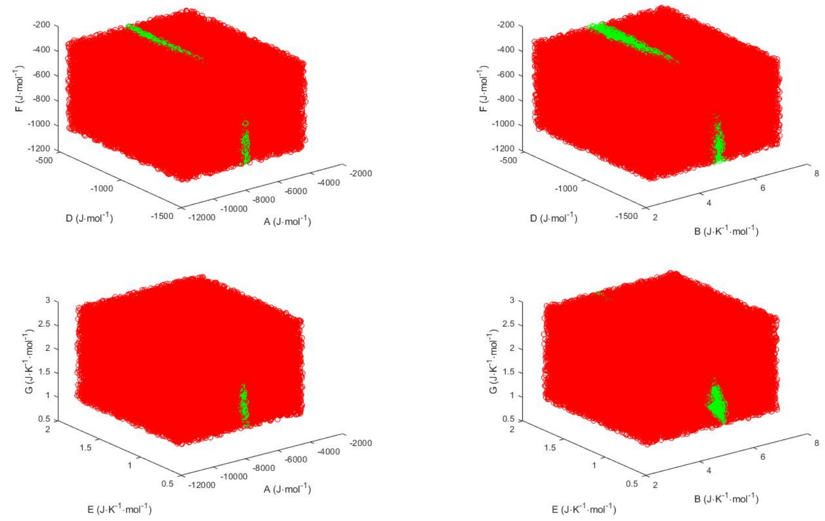

Figure 14: Four three-dimensional point plots of parameters A and B in combination with the coupled parameters D + F and E + G. 100000 points are plotted of which the green points represent parameter combinations that correspond to a phase diagram within the 10K margin and the red points represent those outside of the 10K margin.

The planes observed in the point plots in figure 14 match the two-dimensional planes plotted in figure 13. From figure 14 it can be observed that these green planes only exist across a small range of parameters A and B. In the plots with the coupled parameters with parameter A (A-D-F and A-E-G plots), this range is even smaller than the plots with parameter B (B-D-F and B-E-G plots). Thus, the maximum allowed uncertainty range again seems to be relatively smaller in the parameters A and B than in the coupled parameters D + F and E + G. Additionally, the maximum allowed uncertainty range seems to be relatively smaller in parameter A compared to parameter B.

7 Discussion

7.1 Plausibility of Final Results

7.1.1 Accuracy of the PCE

The GPC class that was ultimately chosen had GPC settings MPO 4, MGO 6, calculation method regression and step size 0.001 in composition. This resulted in a GPC Class which was able to generate phase diagrams that only deviated slightly from the responses calculated by FactSage. See figure 15 for a GPC response plotted together with a FactSage response for the same input Gibbs energy parameters.

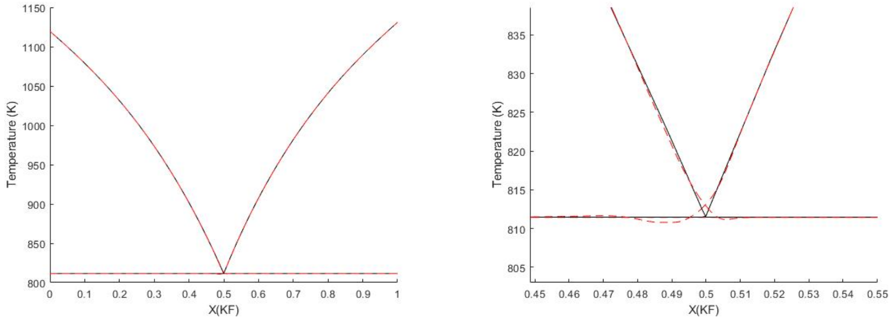

Figure 15: A phase diagram generated by GPC (red dashed line) plotted with a phase diagram calculated by FactSage (black line). The x-axis represents the composition X and the y-axis the temperature T in Kelvin. The left plot shows the whole phase diagram and the right plot zooms in on the eutectic point.

In this figure it can be observed that the GPC response only diverges slightly from the FactSage response around the eutectic point. Thus, the PCE used in this study can be considered quite accurate.

7.1.2 Mixing Enthalpy

The found maximum uncertainty in the mixing enthalpy when varying all the Gibbs energy parameters simultaneously is equal to 1.65 kJ·mol^{-1} at $X = 0.5$. This equals a 35.6% deviation from the mixing enthalpy values computed with the original optimizeed Gibbs energy parameters. This value for the mixing enthalpy is nearly consistent with the results in previous research by T. Schuijtvlot [5], which was 1.5 kJ·mol^{-1}. This slight deviation is most likely due to the difference in the GPC settings with which the GPC was constructed. The error in the GPC used in this study is smaller than in the GPC generated by T. Schuijtvlot, and the uncertainty is quantified using double the amount of samples. Because of these two reasons, the mixing enthalpy uncertainty determined in this study is likely more precise than the result found by T. Schuijtvlot. Additionally, thee found 50th percentile lines from the filtered data corresponds well with the mixing enthalpy values computed with the original optimized input Gibbs energy parameters.

However, when comparing the results from varying all parameters with the experimental values determined by Capelli et al. (2014), there is a large deviation from these experimental values. When varying Gibbs energy parameters A, B, D + F and E + G, these deviations were much smaller than the deviation that results when varying all parameters. Only when parameters A and B were varied simultaneously, the uncertainty half range was 35.5%, which is almost equally as large as the half range found when all parameters were varied simultaneously. Thus, in order to constrain the uncertainty half range in the mixing enthalpy, the variance of parameters A and B should be limited.

7.1.3 Excess heat capacity

The found maximum uncertainty half range when varying all Gibbs energy parameters equals 1.676 J$\cdot$K$^{-1}\cdot$mol^{-1} at composition value X=0.5, which is a 44.7% deviation from the excess heat capacity values calculated with the optimized Gibbs energy parameters. This deviation is large. When looking at the samples obtained after varying Gibbs energy parameters individually, it was observed that varying parameter C resulted in the largest deviation of 18.2%. Additionally, when varying parameter C together with parameters A and B the uncertainty half range increases even more to 44.2%. Thus the large deviation that occurs in the filtered excess heat capacity values is primarily due to the parameters A, B and C, with C being particularly responsible. Therefore, in order to constrain the uncertainty half range in the excess heat capacity values, the variance of parameters A, B and particularly C should be limited.

Comparing the values for the excess mixing enthalpy found in this study with the values found experimentally by Beilmann et al. (2013), there is a significant deviation. As can be observed in figure 6, the point at composition X=0.5 deviates exceptionally from the filtered GPC responses. This is quite odd, since the GPC responses do correspond nearly perfectly with the data computed by FactSage, and the 50th percentile lines overlap very closely with the excess heat capacity values computed with the original optimized input Gibbs energy parameters. A possible explanation for the deviation is that, at the point X=0.5 particularly, the values from Beilman et al. (2013) were measured at a temperature of 884 K - 1083 K. This temperature range is significantly lower than the temperature ranges at points X=0.25 and X=0.75, which are 1158 K - 1344 K and 1157 K -1330 K, respectively. Thus the fact that these measurements were conducted at a lower temperature explains the deviation from the values computed by FactSage in this study.

7.1.4 Excess Gibbs Energy Parameters

For the excess Gibbs energy parameters, the found values for the one-dimensional uncertainty in A and B equal 217.68 J$\cdot$mol^{-1} and 0.2828 J$\cdot$K$^{-1}\cdot$mol^{-1}, respectively. This is a 2.86% and 5.82% deviation from the original optimized Gibbs parameters, respectively. This is a relatively small deviation. Additionally, point plots were generated illustrating the joint two- and three-dimensional uncertainty in A and B and the coupled parameters D + F and E + G. The conclusion was drawn that the maximum allowed uncertainty in parameters A and B (particularly in parameter A) is relatively much smaller than in the coupled parameters D + F and E + G. Also, in the two-dimensional point plot for parameters A and B, it can be observed that when parameter A becomes more negative, B has to become more positive in order to still produce sensible phase diagram results. Thus, in order to produce sensible phase diagrams, the value of parameters A and B should remain close to the original values whilst the values for D, E, F and G can vary much more.

When varying the different Gibbs energy parameters and plotting the corresponding phase diagrams, the phase diagrams corresponding to the variance of parameters A and B were indeed much more deviant. It is therefore consistent with the found results.

7.2 Research Limitations

One limitation experienced during the research is the fact that an overflow occurred in FactSage after conducting around 20000 calculations consecutively (the exact quantity depends among other things on the amount of other programs running). This limited the maximum grid order with which the GPC could be computed to 6, since higher grid orders require more samples to be calculated subsequently in FactSage. A possible solution is to conduct the FactSage calculations in different parts. Also, if the calculations could be conducted on a computer which has more memory or a newer version of FactSage, higher grid orders could be tested without having to separate the calculations. Another downside of using the PCE is that finding the most accurate GPC is a time consuming process, since it requires generating many different GPCs with different settings.

7.3 Further Research

There is much opportunity for further research on this topic. One possibility would be to extend the Gibbs energy parameter uncertainty quantification into all 6 dimensions (or 7, if the heat capacity Gibbs parameter C has significant effect on the output). Additionally, the uncertainty of more thermodynamic parameters in the system can be tested. The method of uncertainty quantification can also be extended to other, more complex systems, such as the LiF-BeF system.

8 References

[1] O. Benes. Thermodynamics of molten salt for nuclear applications. Institute of chemical technology, Prague, 2008.

[2] M. Perrut. Thermodynamic modeling by the calphad method and its applications to innovative materials. AerospaceLab, 2015, p. 1-11. ffhal-01193159f

[3] P. Mekala. CALPHAD Modeling of Phase Diagram Boundaries for Binary 2-Amino-2-Methyl-1,3-Propanediol [AMPL] and Tris(hydroxymethyl) Aminomethane [TRIS] system. University of Nevada, Reno, 2010.

[4] A.D. Pelton, S.A. Degterov, G. Eriksson, C. Robelin, and Y. Dessureal. The Modified Quasichemical Model I—Binary Solutions. Volume 31B, August 2000.

[5] T. L. Schuijttvlot. Uncertainty assessment of the CALPHAD model parameters in the LiF-KF molten salt system using polynomial chaos expansion. Delft University of Technology, 2019. Unpublished.

[6] Z. Perkó, S. R. van der Voort, S. van de Water, et al. Supplemental Material for Fast and Accurate Sensitivity Analysis of IMPT Treatment Plans Using Polynomial Chaos Expansion. Institute of Physics and Engineering in Medicine, 2016.

[7] D. Xiu, G. E. Karniadakis. The Wiener–Askey Polynomial Chaos for Stochastic Differential Equations. SIAM J. Sci. Comput., 24(2), 619–644, 2006.

[8] Z. Perkó, L. Gilli1, D. Lathouwers, J. Kloosterman. Grid and basis adaptive polynomial chaos techniques for sensitivity and uncertainty analysis. Journal of Computational Physics, Volume 260, 1 March 2014, Pages 54-84.

[9] X. Wu, W. Zhang, S. Song, Z. Ye. Sparse grid-based polynomial chaos expansion for aerodynamics of an airfoil with uncertainties. Chinese Journal of Aeronautics,Volume 31, Issue 5, May 2018, Pages 997-1011.

[10] O. Benes, R.J.M. Konings. Thermodynamic properties and phase diagrams of fluoride salts for nuclear applications. Journal of Fluorine Chemistry, Volume 130, Issue 1, January 2009, Pages 22-29.

[11] S. R. Addison. Heat Capacity, Specific Heat, and Enthalpy. January 22, 2001.

[12] Determination of the mixing enthalpy of binary fluid mixtures. PHYWE series of publications, Laboratory Experiments, Chemistry. PHYWE SYSTEME GMBH Co. KG, D-37070 Göttingen.

[13] M. Beilmann, O. Benes, E. Capelli, V. Reuscher,R. J. M. Konings, and Th. Fanghanel. Excess Heat Capacity in Liquid Binary Alkali-Fluoride Mixtures. Inorganic Chemistry, 2013, 52, 24042411.

[14] N. Lüthen, S. Marelli1 and B. Sudret. Sparse Polynomial Chaos Expansions: Literature Survey and Benchmark. ETH Zürich, Zürich, Switzerland, February 5, 2020.

[15] E. Capelli, O. Benes, M. Beilmann, R.J.M. Konings. Thermodynamic investigation of the lif–thf4 system. The Journal of Chemical Thermodynamics, 58:110–116, 2013. doi: https://doi.org/10. 1016/j.jct.2012.10.013.

[16] D. E. Holcomb, G. F. Flanagan, B. W. Patton, J. C. Gehin, R. L. Howard, T. J. Harrison. Fast Spectrum Molten Salt Reactor Options. OAK RIDGE NATIONAL LABORATORY, July 2011.

[17] E. Aukrust, B. Bj orge, H. Flood, and T. Førland. Activities in molten salt mixtures of potassium- lithium-halide mixtures: A preliminary report. Annals of the New York Academy of Sciences, 79 (11):830–837, 1960. doi: https://doi.org/10.1111/j.1749-6632.1960.tb42757.x. Data points obtained with GetData Graph Digitizer version 2.26.0.20.

[18] A. L. Smith, R.J.M. Konings. Advances in Nuclear Fuel Chemistry, part A, Reaction kinetics and chemical thermodynamics of nuclear materials. © 2020 Elsevier Ltd.

[19] Bale, C.W., Bélisle, E., Chartrand, P., Decterov, S.A., Eriksson, G., Gheribi, A.E., Hack, K., Jung, I.H., Kang, Y.B., Melançon, J., Pelton, A.D., Petersen, S., Robelin, C., Sangster, J., Spencer, P., Van Ende, M.A.: FactSage thermochemical software and databases, 2010-2016. Calphad 54, 35 (2016).

A Appendix

A.1 MATLAB Scripts

A.1.1 userFunction.m

```
function [responses, IDs, status] = userFunction(physicalInput,ModelDetails)
%This function accepts physicalinput data from openGPC. It then transfers
%it to FactSage7.0 for calculations and retrieved the data from FactSage7.0
%and sends it back to openGPC.
%V1.0 Y2019-M8-D23 Made by Tristan Lo c Schuijtvlot
%V2.0 Y2020-M8-D24 Edited by Fleur Rooijakkers

%NOTE!!! CHAR Vectors NEED TO BE CHANGED BEFORE USE. Make sure all things
%with an @@ are changed according to your workspace.
%==known issues==
%-FactSage sometimes does not save anything at all
%-FactSage sometimes forgets to save 1 or more files
%-FactSage runs out of memory for no reason
% %% Folder Preparation
% %turn pause on
pause('on')
%Define the user designated results folder of FactSage7.0
%Folder where Phasediagrams made by the macro will be saved
ResultFolder = 'C:\fleurrooijakkers\Results';
%Saves current work folder path
OldPath = pwd;
%Defines the crash line. This is the line where FactSage crashes when
%reading a .txt file after a number of calculations.
% This depends on the amount of decimal/parameter and the amount of calcs.
% Here wirh 15 decimals per parameter are written to a .txt
%(for 7 parameters). FactSage7.0 will crash after 2369 calcs upon reading
%line 237.
% THIS MEANS THAT FACTSAGE CAN CRASH DEPENDING ON byte/line size.
%CrashLine = 200; %200 For safety, max of 70k calculations reached
cd(ResultFolder)
%Makes sure that the result folder has no old .figs. This has been
%effective to stop FactSage7.0 from NOT saving files
if ~isempty(dir('PHAS*.fig'))
    system('del * /Q')
    pause(20)
end
%Safety pause because of slow computer file processing.
%Going one folder up CHANGE THIS TO FOLDER CONTAINING FSInput.txt
cd('..')
%Makes sure that the folder has no old input parameters .txt. This has been
%effective to make FactSage7.0 read correct files
if ~isempty(dir('FSInput*.txt'))
    system('del FSInput*.txt /Q')
    pause(20)
```

```
end
%Amount of FSInput files for FactSage (FS has a READ LINE limit till line CrashLine)
%nFiles = ceil(size(physicalInput,1)/CrashLine);
% Amount of calculations that need to be done
nSamples = size(physicalInput,1);
%writing the amound of calculations to a .txt for FactSage7.0
dlmwrite('C:\fleurrooijakkers\nSamples.txt',nSamples,' ');
%Deviding the input parameters files per crash line amount.
zz= 1;
% for a = 1:nFiles
%     if nSamples - CrashLine*(a-1) > CrashLine
%         RStep = CrashLine;
%     else
%         RStep = nSamples - CrashLine*(a-1);
%     end
    %Writing FactSage Input .txt files.
FSInFile = sprintf('C:\\fleurrooijakkers\\FSInput.txt'); %EDIT: \\FSInput%d.txt',a)
fileID = fopen(FSInFile,'w');
fprintf(fileID,'%.15f %.15f -0.001674400000000 %.15f %.15f %.15f %.15f\r\n', physicalInput(1:nSamples,:)'); %EDIT: physicalInput(zz:zz+RStep-1,:), C = -0.001674400000000
fclose(fileID);
%    zz = zz +RStep;
% end


%Going to the FactSage folder. This is needed to excecute macro(.mac) files
cd('C:\FactSage');
pause(0.2) %Safety Pause

%Excecuting the FactSage7.0 .mac file for phase diagram calculations
%@@ CHANGE THIS TO THE .mac YOU ARE USING. This runs a command in CMD
system('EquiSage.exe /PhaseDiagram /MACRO Macros\PhasLiFKF_macro_read.mac');

%Crude check to see if FactSage saved any .fig files. Could be improved.
cd('C:\fleurrooijakkers\Results');
if isempty(dir('PHAS*.FIG'))
    cd('C:\FactSage');
    system('EquiSage.exe /PhaseDiagram /MACRO Macros\PhasLiFKF_macro_read.mac'); %@@ CHANGE THIS TO THE .mac YOU ARE USING. This runs a command in CMD
end
cd('C:\FactSage');
%Excecuting the FactSage7.0 .mac file for the Equilib module calculations
%@@ CHANGE THIS TO THE .mac YOU ARE USING. This runs a command in CMD

system('EquiSage.exe /EQUILIB /MACRO  Macros\EquiLiFKF_macro.mac');

%Crude check to see if FactSage saved any .txt files. Could be improved.
cd('C:\fleurrooijakkers\Results');
if isempty(dir('EQUI*.TXT'))      %EDIT: changed from EQH to EQUI
```

```
    cd('C:\FactSage');
    system('EquiSage.exe /EQUILIB /MACRO  Macros\EquiLiFKF_macro.mac'); %@@ CHANGE THIS TO THE .mac YOU ARE USING. This runs a command in CMD
end
%Returns to the starting work folder
cd(OldPath);
%Wanted stepsize and  in the phase diagram using linear interpolation
x_inc = 0.001;
x_res = 0 : x_inc : 1;
% Extracting data from created .fig and .txt files. z = data cell, a =
% preallocation array.
%@@CHANGE THIS IF: You have a different resultsfolder!
[z,a] = readPhaseEqui(ResultFolder,size(physicalInput,1));
%obtaining the amount of phase lines found in the .fig files
global PhaseLines
%Memory preallocation
responses = zeros(a(1), a(2)+ 3*length(x_res));

%Code for data extraction from data cells obtain from
[x,y] = size(z);
for i = 1:x
    n=1;
    for j = 1:y
        [xx,yy] = size(z{i,j});
        for k = 1:xx
            for l = 1:yy
                %Extracting phase diagram values
                if j == 1
                    Extr_Data = z{i,j}{k,l}';
                    Extr_Data(1,1) = 0; %Because Factsage uses 1e-5 and 0.99999 for 0 and 1
                    Extr_Data(1,end) = 1;
                    responses(i,n:n-1+length(x_res))=interp1(Extr_Data(1,:), Extr_Data(2,:), x_res);
                    n = n+length(x_res); %add if statement for diff between phase and equi extraction!
                end
            end
        end
        %Extracting Equilib values
        if j == 2
             %HERE (:,2) MEANS TO EXTRACT H_DELTA!
            responses(i,n:n-1+length(z{i,j}(:,6))) = z{i,j}(:,6)'; %EDIT z{i,j}(:,2) --> z{i,j}(:,6)
            n = n+length(z{i,j}(:,6));
        end
    end
end


```

```matlab
%ID and Status update for openGPC
for i=1:length(physicalInput(:,1))
    IDs(i,1)={['ID',datestr(datetime(datetime,'Format','yyyyMMdd-HHmmSSSSSSSSS'),'yyyymmdd-HHMMSSFFF'),...
        ' S',num2str(i)]};
    pause(1e-3);
    if all(~isnan(responses(i,:)))
        status(i,1)={'Fetched'};
    else
        status(i,1)={'Error'};
    end


end
end
```

A.1.2 readPhaseEqui.m

```matlab

function [Data,PreA] = readPhaseEqui(resultPath,SampleAmount)
%V1.0 Y2019-M8-D23 Made by Tristan Lo c Schuijtvlot, based on the code of an unknown student.
%Feel free to do anything with the code. Just give some credit to me!

%This Function reads MARKED phase diagram .FIG and equilib .TXT files.
%THESE MARKERS ARE PHASE* AND EQH* WHERE THE * IS A NUMBER FROM 1 TO N. N
%IS THE AMOUNT OF FILES! This was done on purpose to work with openGPC.

%Input:
%-resultPath is the path of the designated results folder, where the target
%files are located
%-SampleAmount is an optional command to set the amount of PHASE* and EQH*
%files.
%Output:
% -Data, a cell containing phase diagram line data in (x,1) and equilib
% data in (x,2). x is here the number of the corresponding file.
% -PreA, saves data for further preallocation of memory. (1) is the amount
% of files extracted, (2) is the amount of

% SUMMARY
%This matlab script looks for the Factsage7.0 figure files (PHASE*.FIG) and
%the Equilib module text (EQH*.txt) files in the user designated Results
%Folder. It first looks into the .fig file for the lines where the
%coordinates of the phase diagram are stored, reads the coordinates and
%puts stores it in N cells. N corresponds to the amount of found phase
%lines in the phase diagram and stores it in a cell 'data' at position
%{x,1}. Where X is corresponds to the Xth file. making it ready to be
%exported.

%Then it looks at the equilib module text file (corresponding to the Xth
32 %phase file) and finds the lines where the thermodynamic data is stored.
33 %The script puts this in a matrix and also stores it in the data cell at
34 %position {x,2}
35 
36 %NOTE THIS SCRIPT CURRENTLY ONLY LOOKS FOR LINES, NOTHING MAY BE ADDED TO
37 %THE .FIG GRAPH LIKE SYMBOLS AND TEXT! It is way better to add this data
38 %via matlab and not via FactSage7.0 as this can be very buggy with macros.
39 
40 %Data cells work like the following. A column at (x,1) gives the data
41 %obtained from the phase diagram (a 1x3 cell for LiF-KF) In this cell there
42 %are other cells that each give a set of points to plot a phase diagram
43 %line. In these phase line cells, row 1 is the Composition and row 2 the
44 %Temperature
45 % The other cell in gives the data at (x,2) is obtained from the Factsage7.0
46 % 7.0 Equilib module. Where it is organized in a matrix. The column are
47 % organized as Composition, DELTA H, DELTA G, DELTA V, DELTA S, DELTA Cp.
48 % NOTE that more data can be extracted from the equilib .txt file but this
49 % was not relevant for current research.
50 
51 
52 
53 %--Troubleshooting--
54 %-Is your work folder currently in Folder/Scripts?
55 %-Is there a Results Folder? -Do the Files names correspond
56 %to the names in the script?
57 %-Are there any files in the Results folder?
58 %-Do any other files have the same names as the expected .fig (PHASE*) and
59 %.txt (EQH*) files?
60 %-Make sure that the .fig file has nothing added manually like data points
61 
62 %--Edited by Fleur rooijakkers--
63 %line 107: Changed '000END' to '001CMT   DEL/FIN activated'
64 %line 135: Changed 'EQH%d' to 'EQUI%d'
65 %% Setting parameters
66 
67 %checks is the results folder path is a character vector
68 if ischar(resultPath) ~= 1
69     error('Please use a char as input!\nCurrent input class: %s',class(resultPath))
70 end
71 
72 %set a global variable for the max amount of lines
73 %found in the phase diagram. More convienent to put global
74 clear PhaseLines
75 global PhaseLines
76 
77 oldPath = cd; %Get path of current workfolder, to return when done.
78 %Setting current folder as the Results Folder
79 cd(resultPath)
80 %Checking if the Sample Amount is set
81 if ~exist('SampleAmount','var')
```

```
    FIG = dir('PHAS*.FIG');
    a2 = numel(FIG); %Get amount of figs
else
    a2 = SampleAmount;
end

PhaseLines = 0;
delimiterIn = ' '; %setting delimiter of the .FIG files
headerlinesIn = 9999;
%the Data variable is a cell which contains the values from the Phasediagram and Equilib per column respectively. One set of calculations per row.
%Preallocation
Data = cell(a2,2);
PreA = [a2 0];
%for loop to extract PHASx.fig and EQH*.txt
for x = 1:a2

%% Obtaining data from Phasefigs, i.e .fig files
%   Works on any .fig file that has no manual figure additions.

    %sets the file name to PHASEx, where x is a integer
    filename = sprintf('PHAS%d.FIG',x);
    %imports the Phase*.fig where * is a wildcard
    FIG = importdata(filename,delimiterIn,headerlinesIn);
    %looks for the lines that mark the solution/line data in the .fig file
    index = find(strncmp(FIG,'701LIN',5));
    %Finds the end markers of the data in the file.
    %Only works if PHASE figure has NOTHING added
    index(numel(index)+1) = find(strcmp(FIG,'001CMT   DEL/FIN activated')); %Changed from '000END'
    A1 = cell(1,numel(index)-1); %Memory allocation
    B1 = cell(1,numel(index)-1);
    %Split the "FIG cell" up in usable parts
    for i = 1:numel(A1)
        A1{i} = cell2mat(FIG(index(i)+1:index(i+1)-1));
    end
    Vl = 0;
    %Converting strings to numbers. Note: can probably be optimized
    for i = 1:numel(A1)
        for j = 1:size(A1{i},1)
            B1{i}(j,:) = str2num(A1{i}(j,13:size(A1{i},2)));
        end
    end

    Data{x,1} = B1;
    PhaseLines = numel(B1);
% UNCOMMENT FOR PLOTS! WHEN RAM IS  4GB DO NOT PLOT WHEN AMOUNT OF PLOTS>20!
%      figure(); hold on %plotting a phase diagram for i = 1:numel(B1)
%          plot(B1{i}(:,1),B1{i}(:,2)) %Know that Factsage plots multiple
%          lines for the phase diagram. Excatly the same method here
%      end hold off
% Could be made into an IF statement

%% Obtaining data from equilib

     % Finds the EQHx.txt file where x is an interger
    filename2 = sprintf('EQUI%d.TXT',x); %Edited: Changed to EQUI from EQH
    %Loading file as a Cell
    TXT = importdata(filename2,delimiterIn,headerlinesIn);
    %look for string where the fraction of reactant "A" is located
    findA = strfind(TXT,'is ');
    %Stores the indexes of the TXT Cell where "A" data is located
    indexCellA = find(~cellfun('isempty', findA));
    %Memory allocation, this Matrix will store all values from Equilib
    Values = zeros(length(indexCellA),6);
    for i = 1:numel(indexCellA)
        Values(i,1) = str2num(TXT{indexCellA(i)} ...
            (findA{indexCellA(i)} + 3:end));%A values place in first column
    end

    %looks for string where data is located, NOTE The data is 3 lines
    %lower and the Delta values are twice given in the .txt
    findDValues = strfind(TXT, ' DELTA H        DELTA G'); %EDIT ' DELTA H DELTA G' --> S          Cp
    %Same progress above at "A data
    indexCellDValues = find(~cellfun('isempty', findDValues));
    for i = 1:numel(indexCellDValues)
        Values(i,2:end) = str2num(TXT{indexCellDValues(i)+3});
    end
    Vl = Vl + i;

    Data{x,2} = Values; %Stores data in the data cell for later extraction
%   UNCOMMENT IF YOU DO WANT A PLOT!
%     figure(); plot(Values(:,1),Values(:,2),'-'); %Plot DeltaH
%    Could be made into an IF statement
    if PreA(2) < Vl
%          Save the number of needed rows and colums (just H_delta)
%          for Memory Preallocation
        PreA = [a2 Vl];
    end
end
    cd(oldPath)%returning to starting Work folder

end
```

A.1.3 errorcheck.m

```
function [ErrorIndicators] = errorcheck(GPCClass,IntegrationData)
%% Plot 1 line

```

```
InputVar = IntegrationData.ErrorScenarios.All.abscissas(10,:);
InputVarR = GPCClass.Details.means+ InputVar.*GPCClass.Details.stds;
FSResp = IntegrationData.ErrorScenarios.All.responses(10,:);
GPCResp = GPCClass.value(InputVarR)

x_res = 0:0.001:1;
n=1;
figure(); xlabel('X(KF)'); ylabel('Temperature (K)')
for k = 1:3
    hold on
    plot(x_res,FSResp(:,n:n+length(x_res)-1),'k',x_res,GPCResp(:,n:n+length(x_res)-1),'--r') %x_res, Samples(:,n:n+length(x_res)-1),'xk')
    n= n+length(x_res);
end
%% Generation of Error Indicators of the PCE
InputVar = IntegrationData.ErrorScenarios.All.abscissas;
InputVarR = GPCClass.Details.means+ InputVar.*GPCClass.Details.stds;
Responses = IntegrationData.ErrorScenarios.All.responses;
GPCSamp = GPCClass.value(InputVarR);

%Saving relevant information
MaxP = GPCClass.Details.maxPolOrder;%Max Polynomial Order of GPCClass
MaxGO = GPCClass.Details.maxGridLevel;%Max Grid Order of GPCClass
ExtraLvl = GPCClass.Details.nExtraLevels;%Numb of extra levels of GPCClass
%Calculating the error
Err = Responses - GPCSamp; %Error between GPCClass and FactSage
ErrP = Err(:,1:3003); %Error phase diagram part
ErrE = Err(:,3004:3014);%Error mixing enthalpy part
%Calculating the Mean Absolute error
MAEP = sum(abs(ErrP),'all')/numel(ErrP); %Mean Absolute Error phase diagram part
MAEE = sum(abs(ErrE),'all')/numel(ErrE); %mixing enthalpy part
MAE = [MAEP,MAEE];
%Calculating the Root Mean Squared Error
RMSEP = sqrt(sum(ErrP.^2,'all')/numel(ErrP));%Root Mean Squared Error phase diagram part
RMSEE = sqrt(sum(ErrE.^2,'all')/numel(ErrE));%mixing enthalpy part
RMSE = [RMSEP,RMSEE];
%Calculating Mean Error and Mean Squared Error
%Mean Error of all calulations per point of the 314 responses
ME = sum(Err,1)/size(Err,1);
MSE = sum(Err.^2,'all')/numel(Err); %Mean Squared Error ALL part


%Making the Error Indicators structure
ErrorIndicators = struct('Error',Err,'MeanSquaredError',MSE,...
     'RootMeanSquareError',RMSE,'MeanAbsoluteError', ...
    MAE,'MaximumPolynomialOrder',MaxP,'MaximumGridOrder',...
    MaxGO, 'ExtraLevels',ExtraLvl,'MeanError',ME);
```

A.1.4 HmixPhaseMargin.m

```
function [H_mix,H_err] = HmixPhaseMargin(GPCClass, Samples)
%%
%This function filters the samples that are within the defined margin (in K) of
%the phase diagram and then returns the corresponding mixing enthalpy data points,
%Gibbs energy parameters and the uncertainty in these mixing enthalpy
%points (difference between the minimum and maximum values)

%%
%cd('C:\frooijakkers\opengpc\src');
%Generates GPC samples and corresponding input variables (Gibbs energy parameters)
%[Samples,InputVar] = GPCClass.sample(SampAmount);
%Unnormalizing the input variables
%InputVar = GPCClass.Details.means+ x.*GPCClass.Details.stds

%original phase diagram points
Phase_control = [762.053773000000,762.053773000000, ...,
    762.053773000000,762.053773000000,0,-1665.96000000000,
-2970.60000000000,-3908.58000000000,-4465.36000000000,
-4629.22000000000,-4402.99000000000,-3806.14000000000,
-2864.68000000000,-1596.61000000000,0];%Phase control values with interpolation 0.01
Phase_control = [762.035971000000,762.035971000000,...,
762.035971000000,762.035971000000,762.035971000000,
762.035971000000,762.035971000000]; %Phase control values with interpolation 0.001
Margin = 10;
PhaseLines = 3;
H_len = 11; %amount of mixing enthalpy rows in Samples
col = size(Samples,2)- H_len; %Amount of phase diagram columns in samples
%H_mix = zeros(1,H_len); %Initiate mixing enthalpy matrix
%Gibbs = zeros(1,size(InputVar,2)); %Initiate Gibbs energy matrix
j = 1;
h = 1;

for i = 1 : size(Samples,1)
    if all(abs(Samples(i,1:col) - Phase_control(1:col)) < Margin)
%         Phas(j,:) = Samples(i, :);
        H_mix(j,:) = Samples(i,(col+1):size(Samples,2));
%         Gibbs(j,:) = InputVar(i,:);
        j = j + 1;
    else
%         Phasr(h,:) = Samples(i, :);
%         Gibbsr(h,:) = InputVar(i,:);
        h = h + 1;
    end
end

%H_mix = Phas(:,3004:3014);
%
H_err = abs(max(H_mix(:,6)) - min(H_mix(:,6)))/2;
```

```
% for c=1:size(Gibbs,2)
%     G_err(c) = abs(max(Gibbs(:,c)) - min(Gibbs(:,c)))/2;
% end

%GibbsErr = abs(max(Gibbs(:,pos)) - min(Gibbs(:,pos)))/2
prc100S = prctile(H_mix,100,1);
prc50S = prctile(H_mix,50,1);
prc0S = prctile(H_mix,0,1);

%Plot the filtered mixing enthalpy values
x = 0:0.1:1;
x_exp = [0.25,0.50,0.75];
Exp = [3.75, 9.75, 5.9];
Err = [4.6, 5.8, 5.5];
lim = [0,16]

errorbar(x_exp, Exp,Err, 'ko')
ylim(lim)
hold on
plot(x,prc0S,'b',x,prc50S,'g',x,prc100S,'r')
xlabel('X(KF)')
ylabel('\Delta C_p (J/K)')
hold on
y = GPCClass.value(GPCClass.Details.means)
plot(x,y(3004:3014),'--k')
% x_res = 0:0.001:1
% plot(x,Phas(:,3004:3014))
% hold on
%
% X = [0, 0.4771,0.5609,0.4420,0.6542,0.1901,0.2669,0.8028,0.3415,0.9029, 1]
% H = [0, -4110,-4350,-4200,-3950,-2410,-3660,-3340,-3940,-2950,0]
% Err = [0, 808,800,805,814,867,823,836,814,850,0]
%
%plot(X,H,'ko')
% errorbar(X,H,Err,'ko')
% xlabel('X(KF)')
% ylabel('\Delta_{mix}H(J mol^{-1})')
%plot(x,Phase_control(3004:3014),'-o')
% errorbar(x,Phase_control(304:314),H_err)
% n=1;
% figure(); title('PhasDia'); xlabel('mol fraction'); ylabel('Temperature (K)')
% for k = 1:PhaseLines
%     hold on
%     plot(x_res,Phas(:,n:n+length(x_res)-1),'g',x_res,Phase_control(:,n:n+length(x_res)-1),'ko') %,x_res,Phasr(:,n:n+length(x_res)-1),'r' %x_res, Samples(:,n:n+length(x_res)-1),'xk')
%     n= n+length(x_res);
% end
```

A.1.5 Value.m

```
function [H_mix,H_err] = Value(GPCClass,nSamples,pos)
%%

[Samples,InputVar] = GPCClass.sample(nSamples);
InputVar = GPCClass.Details.means+ InputVar.*GPCClass.Details.stds;

x = repmat(GPCClass.Details.means,nSamples,1);

if pos == 5 || pos == 7
    x(:,7) = InputVar(:,7);
    x(:,5) = InputVar(:,5);
elseif pos == 4 || pos == 6
    x(:,4) = InputVar(:,4);
    x(:,6) = InputVar(:,6);
else
    x(:,pos) = InputVar(:,pos);
end
% x(:,3) = InputVar(:,3);
% x(:,2) = InputVar(:,2);
disp(x)
Samp = GPCClass.value(x);
[H_mix,H_err] = HmixPhaseMargin(GPCClass, Samp)
```

A.1.6 pointPlot.m

```
function [Gibbs,Gibbsr] = pointPlot(GPCClass, nSamples, var1, var2)

[Samples,InputVar] = GPCClass.sample(nSamples);
InputVar = GPCClass.Details.means+ InputVar.*GPCClass.Details.stds;

x = repmat(GPCClass.Details.means,nSamples,1);

% if var1 == 3
%     var2 = 5
% elseif var1 == 5
%     var2 = 3
% elseif var1 == 4
%     var2 = 6
% elseif var1 == 6
%     var2 = 4
% end
%
% if var2 == 3
%     var1 = 5
% elseif var2 == 5
%     var1 = 3
% elseif var2 == 4
%     var1 = 6
% elseif var2 == 6
%     var1 = 4
% end
% if var1 == 3 || var1 == 5
%     x(:,3) = InputVar(:,3);
%     x(:,5) = InputVar(:,5);
% elseif var1 == 4 || var1 == 6
%     x(:,4) = InputVar(:,4);
%     x(:,6) = InputVar(:,6);
% else
%     x(:,var1) = InputVar(:,var1);
% end
%
% if var2 == 3 || var2 == 5
%     x(:,3) = InputVar(:,3);
%     x(:,5) = InputVar(:,5);
% elseif var2 == 4 || var2 == 6
%     x(:,4) = InputVar(:,4);
%     x(:,6) = InputVar(:,6);
% else
%     x(:,var2) = InputVar(:,var2);
% end

x(:,var1) = InputVar(:,var1);
x(:,var2) = InputVar(:,var2);

Samples = GPCClass.value(x);
Phase_control = [762.053773000000,762.053773000000, ...,
    762.053773000000,762.053773000000,0,-1665.96000000000,
-2970.60000000000,-3908.58000000000,-4465.36000000000,
-4629.22000000000,-4402.99000000000,-3806.14000000000,
-2864.68000000000,-1596.61000000000,0];%Phase control values with interpolation 0.01
Phase_control = [762.035971000000,762.035971000000,...,
762.035971000000,762.035971000000,762.035971000000,
762.035971000000,762.035971000000]; %Phase control values with interpolation 0.001
Margin = 10;
PhaseLines = 3;
H_len = 11; %amount of mixing enthalpy rows in Samples
col = size(Samples,2)- H_len; %Amount of phase diagram columns in samples
%H_mix = zeros(1,H_len); %Initiate mixing enthalpy matrix
%Gibbs = zeros(1,size(InputVar,2)); %Initiate Gibbs energy matrix
j = 1;
h = 1;

for i = 1 : nSamples
    if all(abs(Samples(i,1:col) - Phase_control(1:col)) < Margin)
%         H_mix(j,:) = Samples(i,(col+1):size(Samples,2));
        Gibbs(j,:) = x(i,:);
        j = j + 1;
    else
        Gibbsr(h,:) = x(i,:);
        h = h + 1;
    end
end

% for c=1:size(Gibbs,2)
%     G_err(c) = abs(max(Gibbs(:,c)) - min(Gibbs(:,c)))/2;
% end

%Point plot
% x = Gibbs(:,var1)
% y = Gibbs(:,var2)
% z = Gibbs(:,var3)
%
% xr = Gibbsr(:,var1);
% yr = Gibbsr(:,var2);
% zr = Gibbsr(:,var3);
% plot(x,y,z,'og',xr,yr,zr,'or')
% xlabel('E (J\cdotK^{-1}\cdotmol^{-1})')
% ylabel('G (J\cdotK^{-1}\cdotmol^{-1})')
% hold on
%plot(GPCClass.Details.means(var1),GPCClass.Details.means(var2),'+k')
%(J\cdotK^{-1}\cdotmol^{-1})
%(J\cdotmol^{-1})
```