REVIEWER #1

We thank the reviewer for reading our manuscript and giving helpful comments that will allow us to clarify our work.

The authors present a modification of their code MAGEMIN to allow for more efficient Gibbs free energy minimization. They state that the advantages of this modification arise from the new minimization algorithm being "unconstrained" as opposed to the "constrained" algorithm used previously. This distinction is misleading and the authors should change the paper to more clearly state the change in strategy. Simply put, the new algorithm is NOT "unconstrained".

We agree with the reviewer that our reformulation is not strictly unconstrained but bound-constrained with the bound being that parameters should be positive. We updated the title of the manuscript accordingly ("A bound-constrained formulation for complex solution phase minimization") and clarified the terminology of the inequality constraints to non-linear inequality constraints.

The central problem is simply stated: minimize the Gibbs free energy of a phase as a function of the fractions of cations X_i, I=1,n on one or more crystallographic sites. The minimization is subject to two constraints:
1) The cation fractions must sum to 1:
sum_i^n X_i = 1
and
2) they must be non-negative
X_i >= 0
Constraint (1) can be implemented either by performing a constrained minimization for which (1) forms an auxiliary statement to the minimization problem OR by minimizing over the null space of the constraint (1). The latter approach is appealing because it reduces the dimensionality of the problem (by the number of crystallographic sites), and avoids the auxiliary statement of constraint.

We agree with the general description of the problem. However, the solution phase formulation as described by Holland et al. (2018) and all previous publications of the THERMOCALC group are given in terms of compositional variables, which within bounds, can results in site fraction < 0.0. There is thus a need to reformulate the solution phase using mixing site constraints. As discussed by the reviewer below, for other (simpler) thermodynamic databases, this may not be necessary.

The primary focus of this paper is on the implementation of the null space approach. And it is on the basis of the removal of the auxiliary statement (1) that they describe their modified algorithm as "unconstrained".

The primary focus of the paper, is to reformulate the compositional and order variable formulation used by THERMOCALC a-x models and subsequently, as moted by the reviewer, remove the auxiliary statement.

However, it is NOT unconstrained. The reason is constraint (2). This must still be dealt with: the minimization must still be subject to the bounds described by constraint 2. Thus the minimization is still constrained, not unconstrained.

We agree that the problem is still bound-constrained and corrected this throughout the manuscript.

In other words, they have removed an equality constraint, but NOT the inequality constraint.

We appreciate the opportunity to clarify our intent. Our intention was not to suggest the removal of the bound constraints of the minimization problem.

Our contribution is that we reformulate the non-linear inequality constraints of the site fraction, which are expressed as functions of compositional and order variables (e.g., Holland et al., 2018), into equality constraints expressed as functions of site fraction variables. Subsequently, we eliminate these equality constraints using a nullspace approach, leaving only the bound constraints on the site fraction variables.

This approach is adopted because the site fraction formulation presented in Holland et al. (2018) incorporates non-linear inequality constraints that restrict the hypercube defined by the 'compositional variables.' In other words, the hypercube defined by the bound constraints of the compositional variables does not fully represent the feasible domain. For instance, certain combinations of compositional variables can result in negative site fractions (even within bounds), thus necessitating the use of these additional inequality constraints. The focus of the paper is the elimination of these inequality constraints.

From the reviewer remarks we understand that this was insufficiently clear in the text, so we will update the manuscript accordingly in the revised version.

Of course the authors DO end up applying the inequality constraint, although this application is somewhat buried in the details (Eq. 32).

Each dimension of the problem is indeed bounded, but as described above, we no longer must invoke a solver with (internal) inequality constraints.

My argument is not with the method itself, nor primarily with the claims of greater efficiency (although see below). It is with what I think is a misleading description of the modification. I implore them to characterize their modification more clearly and correctly. Some further comments.

We hope that the revised version clarifies this better.

1) Readers may get the mistaken impression that the null space approach is new in the context of petrological Gibbs free energy minimization codes. It is not, and an example of another minimization code that uses the null space approach is HeFESTo (Stixrude and Lithgow-Bertelloni, 2011). Nor do the authors of HeFESTo claim priority for the null space approach, but it is the closest implementation known to this reviewer in terms of intended application (i.e. to petrology).

We agree with the reviewer that the nullspace approach is not a novel concept in the context of petrological Gibbs free energy minimization and we did not intent to present it as such. Indeed, this method forms the foundation of the solution phase formalism in THERMOCALC, particularly in the formulation of “compositional variables”, which have been used since at least White et al. (2000). These variables are used to parametrize mixing site charge neutrality, enforce the condition that the sum of site fractions equals 1, and ensure that the sum of endmember proportions also equals 1.

2) The claimed superiority of their new implementation to SLSQP is puzzling to this reviewer. One reason for the source of puzzlement is that HeFESTo uses SLSQP and finds a very high (essentially perfect) rate of success and precision as documented in Stixrude and Lithgow-Bertelloni (2021).

This is likely because the problem solved using SLSQP in HeFESTo is only bound-constrained, while the one solved in MAGEMin also includes internal inequality constraints. This is why we wanted to reformulate the problem in a similar manner as in Stixrude and Lithgow-Bertelloni (2021).

Perhaps the reason is that HeFESTo uses the null space approach for the equality constraints and relies on the constraint facility of SLSQP only for the inequality constraint. Maybe in the current paper, the authors are instead relying on SLSQP to take care of the equality AND inequality constraints.

We think that there is a misunderstanding coming from having a different solution phase formalism in mind here.
While the solution phase formalism described in Stixrude and Lithgow-Bertelloni (e.g., 2021, as well as earlier references) is indeed constrained solely by bounds (referred to by the reviewer as inequality constraints), the site fraction formulation presented in Holland et al. (2018) is natively different. It incorporates non-linear inequality constraints that internally restrict the hypercube defined by the compositional variables bounds. For example, certain combinations of compositional variables (within bounds) can still result in a negative site fraction. Therefore, additional non-linear inequality constraints are required. Our manuscript shows how to take those into account before passing the problem to the innermost solver.

Therefore, to avoid confusion in the literature, I suggest the following test:
Perform the SLSQP minimization(s) again, but by combining SLSQP with the null space approach.

The results of the SLSQP algorithm presented in the manuscript already used the nullspace approach as the solution phase formalism presented in Holland et al., 2018 ensures that the mixing site charge neutrality, the sum (site fraction) = 1 and the sum of endmember proportion = 1 by parameterizing the solution space.

Moreover, using the SLSQP approach on bound-constrained only problems reduces to using a bounded BFGS approach.

3) A final comment on SLSQP. The authors state that SLSQP sometimes fails because it violates inequality constraints and then cannot return to the feasible space. But this should not be true. According to NLOPT documentation SLSQP is guaranteed to respect inequality constraints at all intermediate steps of the minimization.

We respectfully think that the reviewer is mixing bound-constraints and (non-)linear inequality constraints. While bound constraints are indeed always respected, this is not necessarily true for non-linear inequality constraints.

The NLopt documentation indicates the following (https://nlopt.readthedocs.io/en/latest/NLopt_Tutorial/):

"In principle, we don't need the bound constraint $x_2 \geq 0$*, since the nonlinear constraints already imply a positive-*x_2 *feasible region. However, NLopt doesn't guarantee that, on the way to finding the optimum, it won't violate the nonlinear constraints at some intermediate steps, while it does guarantee that all intermediate steps will satisfy the bound constraints. So, we will explicitly impose* $x_2 \geq 0$ *in order to ensure that the* $\sqrt{x_2}$ *in our objective is real."*

So, in other words, the nonlinear constraints are not always guaranteed to be satisfied.

Perhaps the problem is that SLSQP sometimes does venture into the space where one or more X_i are exactly zero. If this is the case, it is a problem easily solved, by setting the inequality constraint instead to:
X_i >= epsilon
very much like their own implementation of the inequality constraint.

As pointed out by the reviewer, having a bound-constraint for a site fraction exactly equal to zero would pose a problem (as log(0.0) is undefined for the configurational entropy term) and we indeed added a small epsilon in the code to avoid this (see https://github.com/ComputationalThermodynamics/SandBox/blob/main/GradientBasedMinimizers/unconstrained_CG_BFGS/functions/gradient_method.jl L99 and L237-245).

We further clarified this in the text LX

REVIEWER #2

This is plainly a useful and important contribution. The essential motivation and results are clear. Nonetheless, before publication some minor revisions are needed to make the presentation as clear and unambigious as it can be.

Lines 40-48: People are starting to explore a new cheap work-around, namely neural net emulators of thermodynamic models. See, for example, https://agu.confex.com/agu/agu24/meetingapp.cgi/Paper/1708197 … just an abstract at this point, but maybe worth mentioning.

We added the reference, and a couple more on the ongoing development of neural networks

Line 70: Fix grammar: "…inequality constraint in gradient-based methods results in…"

Fixed

Line 77: "…avoids the need to express …"

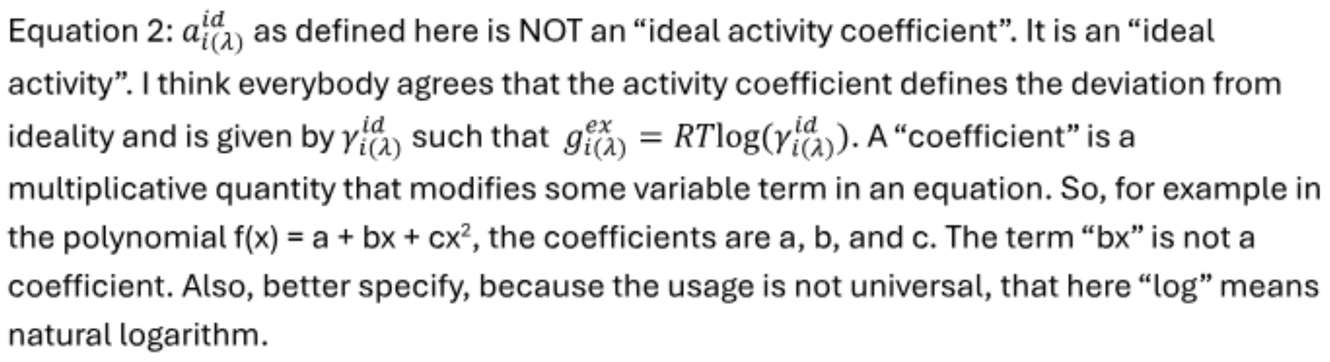
Fixed

Line 79: Just "conjugate", not "conjugated"

Fixed

Line 88: "combinations", plural

Fixed



Equation 2: $a_{i(\lambda)}^{id}$ as defined here is NOT an "ideal activity coefficient". It is an "ideal activity". I think everybody agrees that the activity coefficient defines the deviation from ideality and is given by $\gamma_{i(\lambda)}^{id}$ such that $g_{i(\lambda)}^{ex} = RT\log(\gamma_{i(\lambda)}^{id})$. A "coefficient" is a multiplicative quantity that modifies some variable term in an equation. So, for example in the polynomial $f(x) = a + bx + cx^2$, the coefficients are a, b, and c. The term "bx" is not a coefficient. Also, better specify, because the usage is not universal, that here "log" means natural logarithm.

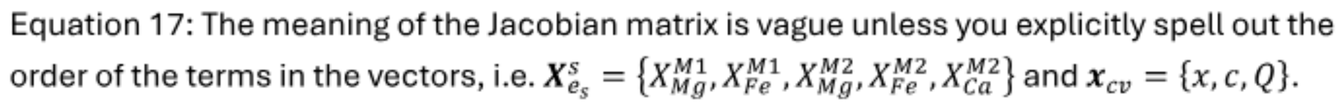
We agree and corrected to "activity" term and change log to "ln".

Line 113: “functions”

Corrected

Line 138: c is defined incorrectly. Obviously, it is equal to X_{Ca}^{M2}, not X_{Fe}^{M1} !

Thank you for spotting it. Fixed accordingly.



Equation 17: The meaning of the Jacobian matrix is vague unless you explicitly spell out the order of the terms in the vectors, i.e. $\boldsymbol{X}_{e_s}^{s} = \{X_{Mg}^{M1}, X_{Fe}^{M1}, X_{Mg}^{M2}, X_{Fe}^{M2}, X_{Ca}^{M2}\}$ and $\boldsymbol{x}_{cv} = \{x, c, Q\}$.

We agree and updated the manuscript accordingly.

Equation 21: Again, make up your mind: x or xcv?

Fixed to Xcv

Equation 22: What is b? Here it appears to be {1, 1}, but this is never stated. Equation 18 is not the set of equality constraints, it is just the definition of the matrix A (which, again, ought to have a distinctive symbol to indicate it is a matrix).

We clarified this part, by introducing the equality constraint equations Ax = b, introducing the constraint vector b and properly addressing the definition of coefficient matrix A.
We also now distinguish the symbols for matrices and vectors by making them bold.

Line 183: It is not obvious (to those that don’t do QR decompositions for a living) that Q is a matrix whose columns are q1 … qm, or that these are ordered into Q1 and Q2 where Q1 has n columns corresponding to the image and Q2 has m–n columns corresponding to the nullspace. Please say that.

We improved this part accordingly

Do you really need to introduce the matrix F? Isn’t it just Q2? Or for that matter, Nz as the nullspace matrix of A when A is already the nullspace matrix of the Jacobian?

We corrected this part and use Nz for all instances.

Bottom line: I really think equations 17-27 could be presented in a manner that is less confusing and introduces fewer throw-away symbols.

We improved the presentation and reduced the number of symbols

Lines 196-198: The sentence beginning “Considering ...” is not a sentence.

Corrected

Line 262: I am confused about whether this is a valid test of the acceleration available from this algorithm, if the global minimum hyperplane first has to be calculated by MAGEmin (i.e., using the old, slow algorithm!).

The objective here is to evaluate how much a change in the orientation of the Gibbs hyperplane affects the minimization of the solution phases with respect to it. Importantly, this assessment does not require the Gibbs hyperplane to correspond to a global minimum; it can be performed for any arbitrary Gibbs hyperplane, defined by a given set of oxide chemical potentials.

MAGEMin operates in a manner similar to Theriak. First, an initial guess for the Gibbs hyperplane is made (step 1). Next, the solution phase models are minimized with respect to this hyperplane (step 2). The resulting minimized solutions are then used to update the orientation of the Gibbs hyperplane (step 3). Steps 2 and 3 are iterated until convergence, i.e., until the Gibbs hyperplane ceases to change. Between successive iterations, the change in the hyperplane's orientation can be quantified using $\Delta\Gamma$. In this example, we evaluate the performance of the new optimizer with respect to the of magnitude of $\Delta\Gamma$.

We find that the inequality constraint optimizer does not depend on this magnitude while the new bound-constrained optimizer does.

Figure 2 caption: “Least”, not “Levast”; “Conjugate Gradient”, not “conjugated gradiend”; “method presented in”, not “present in”; “number of minimizations”, not “minimization”.

Corrected

Line 299: What is G?

We added the definition of Gamma: chemical potential of the pure components of the systems (oxides)

Line 310: The conclusion section starts here, give it a section header

We added the header