Reviewers’ comments:

Contributing Editor: The second referee is requesting numerical comparisons with reference [42]. Please do this, and thoroughly address each point raised by both referees.

Reviewer #2:

The manuscript deals with a particular yet broad class of constrained DC programs. Since the authors assume that the nonlinear constraint has gradient Lipschitz continuous, the family of problems under consideration can be seen as a special category of DC-constrained DC-problems.
The constraint function's assumptions permit an interpolation step that ensures the feasibility of all iterates produced by the algorithms.

The manuscript is well-motivated, and the mathematical developments are sound. In addition, a numerical section illustrates the algorithms’ performance on two well-known problems.

Assessments

The manuscript combines, without acknowledging it, two well-known algorithms in the mathematical programming community, namely, the Support Hyperplane Method and Proximal Linearized Method for DC programming. The resulting algorithms are interesting, and the mentioned combination seems new to us. However, the manuscript has at least two serious drawbacks: 1) it does not acknowledge related ideas/methods already known in the literature; 2) the (strong) assumptions are poorly stated, given the impression that the presented methodology requires less than it needs. In what follows, we discuss these two points in detail.

1) Related Methods.

The retraction strategy employed by the authors is nothing but an interpolation step, already present in the Support Hyperplane Method (SHM) by Veinott (1967) (see reference [A]). Moreover, in contrast to the setting of the proposed Algorithm 1, the work [A] (and more recently its regularized version in [B]) does not require the constraint function(s) to be convex, but the feasible set. This weaker assumption allows the nonlinear constraint function to have only generalized convexity properties (e.g., quasi-convexity, alpha-convexity [B]). That being said, we believe it is possible to weaken the convexity assumption on the constraint functions in Algorithm 1 with only a few modifications in the convergence analysis. Could the authors confirm that?

Apart from the interpolation step, what are the main differences between the proposed methodologies and paper [42]?

Although Algorithm 1 employs the SHM's interpolation step, it differs substantially from SHM due to the DC structure. However, Algorithm 1, as well as Algorithm 2, handle the DC structure similarly to the Proximal Linearized Method for DC programming [C,D,E]: at every iteration, the concave part is linearized, and a quadratic term is added to form the objective function in the convex subproblem. In particular, the method in [E] seems more general than Algorithms 1 and 2 because the former allows for DC-constrained DC programs. Therefore, we kindly ask the authors to put their methodology in perspective with [E].

That being said, we see the methodology presented in this manuscript as a "linearized proximal method with support hyperplane for DC programs."
Indeed, the above seems a better title for the work because it connects the core ideas in the manuscript.

2) Assumptions.

The assumptions are poorly stated. For instance, Assumption 3.1 (announced in the key Theorems 3.1 and 3.2) asserts that g_i is convex and a Slater point exists. However, the authors need much more than this: g_i needs to be differentiable (rightly stated in the proofs and other parts), and the Slater point must be known. Note that knowing a Slater point is much more restrictive than only assuming its existence.
We kindly ask the authors to fix this issue.

Minor comments.

a) In the Introduction, the authors related their methodology with Sequential Quadratic Programming. However, this link is unclear since this manuscript deals with nonsmooth DC programs, and SQP is for (twice) differentiable problems. Could the authors please give more details on such a connection? Is it related to the sequential DC programming ideas from [F]?

b) As far as we can tell, Definition 2.3 is about criticality and not stationarity. Stationarity in DC programming is a strong condition, and most algorithms are only ensured to compute critical points. Please see [E] for the differences between these two conditions.

c) Theorem 3.2. Please recall that u^k = \tilde u.

d) Assumption 4.1. The notation \xi has already been employed for the subgradient of P_2.

e) Assumption 4.2. The Slater point must be known.

f) Subproblem (4.6): Should it be \sigma^x ?

g) Please compare the algorithms with the method presented in [42], by the third author.

References

[A] A. Veinott. The supporting hyperplane method for unimodal programming, Operations Research, 15 (1967), pp. 147-152.

[B] W. van Ackooij and W. de Oliveira. Convexity and optimization with copulae structured probability constraints. Optimization, 65 (2016) pp. 1349-1376.

[C] Sun, W., Sampaio, R. J. B., & Candido, M. A. B. Proximal point algorithm for minimization of DCFunctions. Journal of Computational Mathematics,21 (2003), 451-462.

[D] Souza, J.C.O. et al . Global convergence of a proximal linearized algorithm for difference of convex functions. Optim Lett 10 (2016), 1529-1539

[E] Pang, J.S., Razaviyayn, M., Alvarado, A.: Computing B-stationary points of nonsmooth DC programs. Math. Oper. Res. 42 (2017)

[F] W. de Oliveira. Sequential Difference-of-Convex Programming. Journal of Optimization Theory and Applications, 186 (2020) pp. 936-959.