



HAL
open science

Distance Geometry in Linearizable Norms

Claudia d'Ambrosio, Leo Liberti

► **To cite this version:**

Claudia d'Ambrosio, Leo Liberti. Distance Geometry in Linearizable Norms. Geometric Science of Information Third International Conference, GSI 2017, Paris, France, November 7-9, 2017, Proceedings, pp.830-837, 2017, 978-3-319-68444-4. 10.1007/978-3-319-68445-1_95 . hal-02105291

HAL Id: hal-02105291

<https://hal.science/hal-02105291>

Submitted on 20 Apr 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Distance geometry in linearizable norms

Claudia D’Ambrosio¹ and Leo Liberti^{1*}

LIX CNRS (UMR7161), École Polytechnique, 91128 Palaiseau, France
{dambrosio,liberti}@lix.polytechnique.fr

Abstract. Distance Geometry puts the concept of distance at its center. The basic problem in distance geometry could be described as drawing an edge-weighted undirected graph in \mathbb{R}^K for some given K such that the positions for adjacent vertices have distance which is equal to the corresponding edge weight. As we are unaware of any work in this field using any other norm but ℓ_2 , we move some first steps using the ℓ_1 and ℓ_∞ norms: we discuss worst-case complexity, propose mixed-integer linear programming formulations, and sketch a few heuristic ideas.

Keywords: distance geometry, norms, mathematical programming

1 Introduction

We discuss the following basic problem in Distance Geometry (DG)

DISTANCE GEOMETRY PROBLEM (DGP). Given an integer $K > 0$ and a simple, edge-weighted, undirected graph $G = (V, E, d)$, where $d : E \rightarrow \mathbb{R}_+$, determine whether there exists a *realization function* $x : V \rightarrow \mathbb{R}^K$ such that:

$$\forall \{i, j\} \in E \quad \|x_i - x_j\| = d_{ij}, \quad (1)$$

where $\|\cdot\|$ is either the ℓ_1 or the ℓ_∞ norm. We assume all along that, without loss of generality, G is connected, otherwise it suffices to realize the disconnected components independently. Moreover, given a realization $x \in \mathbb{R}^K$ providing the solution of a DGP instance with any norm, any *congruence* (i.e., rotation, translation, reflection) of x in \mathbb{R}^K is also a solution.

Most existing work concerning the DGP focuses on the ℓ_2 (or Euclidean) norm. In this paper, we move some first steps in the direction of the ℓ_1 and ℓ_∞ norms, which we call *linearizable norms*, since their unit spheres are polyhedral.

DG in the ℓ_2 norm recently received a lot of attention [13, 6] due to its widespread use in engineering and science applications, such as, for example, finding the structure of proteins from Nuclear Magnetic Resonance (NMR) interatomic distances [19] and many others. It was shown in [17] that the DGP (with the Euclidean norm and $K = 1$) is **NP**-hard, by reduction from PARTITION.

The lack of attention to DG in other norms stems from a scarcity of applications. Yet, recently, we were made aware of applications for both of the

* Partly supported by the ANR “Bip:Bip” project under contract ANR-10-BINF-0003.

linearizable norms. The DGP with the ℓ_1 norm arises in the positioning of mobile sensors in urban areas [2]. The DGP with the ℓ_∞ or with the ℓ_1 can be used in order to fill a hypercube with a pre-fixed number of “well-distributed” points, which is relevant in the design of experiments [4, 18].

The rest of this paper is organized as follows. We recall some notions relating to linearizable norms in Sect. 2. We propose two new formulations for linearizable norms in Sect. 3, and prove that the DGP in linearizable norms is **NP**-complete for any K . Lastly, we sketch some new ideas for solving the linearizable norm DGP using heuristics in Sect. 4.

2 Known results for ℓ_1 and ℓ_∞ norms

Complexity: Since the $\ell_1, \ell_2, \ell_\infty$ norms coincide for $K = 1$, the reduction from the PARTITION problem given in [17] shows that the DGP is **NP**-complete for these three norms in $K = 1$. Since a realization in one dimension can be embedded isometrically in any number of dimensions, this shows by inclusion that the DGP is **NP**-hard for these three norms. We show in Thm. 2 that the ℓ_1 and ℓ_∞ norm variants are also **NP**-complete for $K > 1$. This strikes a remarkable difference with the Euclidean norm DGP, for which the status of membership in **NP** is currently unknown [1].

Isometric embeddings: For the ℓ_∞ norm, the isometric embedding problem can be solved in a very elegant way [9]: any finite metric (X, d) with $X = \{x_1, \dots, x_n\}$ can be embedded in \mathbb{R}^n using the ℓ_∞ norm by means of the *Fréchet embedding*: $\forall i, j \leq n \quad T(x_i) = (d(x_i, x_1), \dots, d(x_i, x_n))$. The proof is short and to the point: for any $i, j \leq n$ we have:

$$\|T(x_i) - T(x_j)\|_\infty = \max_{k \leq n} |d(x_i, x_k) - d(x_j, x_k)| \leq \max_{k \leq n} d(x_i, x_j) = d(x_i, x_j)$$

by the triangle inequality on the given metric d . Moreover, the maximum over k of $|d(x_i, x_k) - d(x_j, x_k)|$ is obviously achieved when $k \in \{i, j\}$, in which case $|d(x_i, x_k) - d(x_j, x_k)| = d(x_i, x_j)$. So we have $\|T(x_i) - T(x_j)\|_\infty = d(x_i, x_j)$ as claimed. We remark that (X, d) need not be given explicitly: the distance matrix is enough. We also remark that a Fréchet embedding can be constructed for any given square symmetric matrix A : if the Fréchet embedding of A is infeasible w.r.t. A , it means that A is not a valid distance matrix.

For the ℓ_1 norm there no such general result is known. It is known that ℓ_2 metric spaces consisting n points can be embedded in a vector space of $O(n)$ dimensions in ℓ_1 norm almost isometrically [15, §2.5] (the “almost” refers to a multiplicative distortion measure of the form: $(1 - \varepsilon)\|x\|_2 \leq \|T(x)\|_1 \leq (1 + \varepsilon)\|x\|_2$ for some $\varepsilon \in (0, 1)$).

3 MILP formulations for linearizable norms

The DGP is a nonlinear feasibility problem. As such, it can be modelled by means of Mathematical Programming (MP), which is a formal language for describing

optimization problems. When the norm is linearizable, it is possible to replace nonlinear functions by piecewise linear forms, which are routinely modelled in MP using binary variables and linear forms. This yields MPs of the Mixed-Integer Linear Programming (MILP) class. A convenient feature of MILP is that solution technology is very advanced — MILP solvers are currently at the forefront for their generality and empirical efficiency. To the best of our knowledge, no MILP formulations have ever been proposed for the DGP in linearizable norms. We give here two MILP formulations for ℓ_1 and ℓ_∞ norms.

We first re-write Eq. (1) as follows: $\min_x \sum_{\{i,j\} \in E} \left| \|x_i - x_j\|_p - d_{ij} \right|$, for $p \in \{1, \infty\}$. Obviously, even if the equation above always has a feasible solution, it has global optimal value zero if and only if the global optimum is a solution of Eq. (1). The MILP formulations below can be solved using any off-the-shelf MILP solver, such as, e.g., CPLEX [12].

The ℓ_1 norm. For $p = 1$ we write:

$$\min_x \sum_{\{i,j\} \in E} \left| \sum_{k \leq K} |x_{ik} - x_{jk}| - d_{ij} \right|. \quad (2)$$

The MILP reformulation we propose is the following:

$$\left. \begin{array}{l} \min_{x,s,t,z} \\ \forall \{i,j\} \in E \quad -s_{ij} \leq \sum_{k \leq K} (t_{ijk}^+ + t_{ijk}^-) - d_{ij} \leq s_{ij} \\ \forall k \leq K, \{i,j\} \in E \quad t_{ijk}^+ - t_{ijk}^- = x_{ik} - x_{jk} \\ \forall k \leq K, \{i,j\} \in E \quad t_{ijk}^+ \leq d_{ij} z_{ijk} \\ \forall k \leq K, \{i,j\} \in E \quad t_{ijk}^- \leq d_{ij} (1 - z_{ijk}) \\ \forall k \leq K \quad \sum_{i \in V} x_{ik} = 0 \\ \forall \{i,j\} \in E \quad s_{ij} \in [0, d_{ij}] \\ \forall k \leq K, \{i,j\} \in E \quad t_{ijk}^+, t_{ijk}^- \in [0, d_{ij}] \\ \forall k \leq K, \{i,j\} \in E \quad z_{ijk} \in \{0, 1\}. \end{array} \right\} \quad (3)$$

Additional variables $s_{ij} \geq 0$, for each $\{i,j\} \in E$, are non-negative decision variables $s_{ij} \geq 0$ that represent the outermost absolute value of (2) thanks to the well known fact that $\min |f|$ is equivalent to $\min_{\hat{f} \geq 0} \hat{f}$ subject to $-\hat{f} \leq f \leq \hat{f}$. Moreover, additional slack and surplus variables t^+, t^- were introduced to reformulate the innermost absolute value terms. In order to do this, they are subject to complementarity constraints $t_{ijk}^+ t_{ijk}^- = 0$, $\forall k \leq K, \{i,j\} \in E$, that in (3) were linearized by adding binary variables z and the two sets of standard “big-M” constraints that links the t and the z variables.

Note that, as each realization can be translated at will, constraint $\sum_{i \in V} x_i = 0$ can be safely added. It means that we can, without loss of generality, impose that the barycenter is zero and it is not necessary but useful in practice, see [5]

for detailed empirical results. Again for practical efficiency, we let $U = \sum_{\{i,j\} \in E} d_{ij}$ and use it to bound x , so that for each $i \in V$ and $k \leq K$, we have $x_{ik} \in [-U, U]$.

Proposition 1. *Eq. (3) is a valid formulation for the DGP using the ℓ_1 norm. The proof is given in the Appendix.*

The ℓ_∞ norm. For $p = \infty$ we write: $\min_x \sum_{\{i,j\} \in E} \left| \max_{k \leq K} |x_{ik} - x_{jk}| - d_{ij} \right|$.

The MILP reformulation that we propose is the following:

$$\left. \begin{array}{l}
 \min_{x,w,z,s,t} \quad \sum_{\{i,j\} \in E} s_{ij} \\
 \forall \{i,j\} \in E, k \in K \quad t_{ijk}^+ + t_{ijk}^- - d_{ij} \leq s_{ij} \\
 \forall \{i,j\} \in E, k \in K \quad t_{ijk}^+ + t_{ijk}^- + s_{ij} \geq d_{ij} w_{ijk} \\
 \forall \{i,j\} \in E \quad \sum_{k \leq K} w_{ijk} \geq 1 \\
 \forall k \leq K, \{i,j\} \in E \quad t_{ijk}^+ - t_{ijk}^- = x_{ik} - x_{jk} \\
 \forall k \leq K, \{i,j\} \in E \quad t_{ijk}^+ \leq d_{ij} z_{ijk} \\
 \forall k \leq K, \{i,j\} \in E \quad t_{ijk}^- \leq d_{ij} (1 - z_{ijk}) \\
 \forall k \leq K \quad \sum_{i \in V} x_{ik} = 0 \\
 \forall \{i,j\} \in E \quad s_{ij} \in [0, d_{ij}] \\
 \forall k \leq K, \{i,j\} \in E \quad t_{ijk}^+, t_{ijk}^- \in [0, d_{ij}] \\
 \forall k \leq K, \{i,j\} \in E \quad w_{ijk}, z_{ijk} \in \{0, 1\}
 \end{array} \right\} \quad (4)$$

where the outermost and the innermost absolute values are reformulated as before and the barycenter constraint is added.

At this point similarities with the ℓ_1 norm stop. Note that the first three set of constraints model the linearization of constraints $-s_{ij} \leq \max_{k \leq K} (t_{ijk}^+ + t_{ijk}^-) - d_{ij} \leq s_{ij}$, $\forall \{i,j\} \in E$. The center and right-hand-side can be reformulated as the first set of constraints. The left-hand-side and the center is more complicated as it corresponds to a non-convex constraint. For linearizing it, we need to add auxiliary binary variables w and the second and third sets of constraints that express the fact that *at least one out of K components satisfies the constraint* (that component being the maximum, of course).

Proposition 2. *Eq. (4) is a valid formulation for the DGP using the ℓ_∞ norm. The proof is given in the Appendix.*

The DGP with linearizable norms is in NP. Our MILP formulations make it easy to prove the following.

Theorem 1. *The DGP in linearizable norms is NP-complete.*

Proof. It was already remarked in Sect. 2 that the DGP in linearizable norms is NP-hard by reduction from PARTITION to the DGP in $K = 1$, where the three norm $\ell_1, \ell_2, \ell_\infty$ coincide. Now by Prop. 3 and 4 we know that certificates to the

DGP in linearizable norms are rational for rational input, since the MP formulations in (3) and (4) can be solved by a Branch-and-Bound (BB) algorithm, each node of which involves the solution of a Linear Program (LP), which is known to be in **NP**. Simple BB implementations will continue branching until the incumbent, found by solving the LP at some node, is proven optimal. So the certificate (solution) provided by the BB is polynomially sized, as claimed. \square

3.1 Computational results

Both formulations (Eq. (3) and (4)) belong to the MILP class, and can be solved by several existing MILP solvers. We employ CPLEX 12.6.2 [12] on a MacBook Pro mid-2015 running Darwin 15.5.0 on a (virtual) quad-core i7 at 3.1GHz with 16 GB RAM.

We generated a set of (feasible) random DGP instances in both norms as follows: for each cardinality value $n = |V|$ of the vertex set of G ranging over $\{10, 15, 20, 25, 30, 35, 40\}$ we sampled n points in the plane, bounded by a box. We then added a Hamiltonian cycle to the edge set, in order to guarantee connectedness. Lastly, we add the remaining edges with an Erdős-Rényi generation process [7] having probability s ranging over $\{0.1, 0.2, 0.3, 0.5, 0.8\}$. This yields 35 instances per norm $p \in \{1, \infty\}$.

We deployed CPLEX on these 70 instances with a time limit of 10 minutes enforced on the “wall clock”, meaning the actual elapsed time. We used this measure instead of the user time since CPLEX exploits all four processor cores, which means that the system time taken for parallel execution tasks is essential. Since the running time is limited, we do not always find feasible solutions. We evaluate the error by employing two well known measures: the scaled Mean Distance Error (MDE) and the scaled Largest Distance Error (LDE).

$$\text{MDE}(x) = \frac{1}{|E|} \sum_{\{i,j\} \in E} \frac{||x_i - x_j||_p - d_{ij}}{d_{ij}}, \quad \text{LDE}(x) = \max_{\{i,j\} \in E} \frac{||x_i - x_j||_p - d_{ij}}{d_{ij}}.$$

Intuitively, the scaled MDE gives an idea of the percentual average discrepancy of the given realization from the given data. The scaled LDE gives an idea of the percentual average worst error over all edges. Obviously, the LDE is generally higher than the MDE.

The detailed results are shown in Table 1 of the Appendix. Each row of the table reports the instance statistics, followed by scaled MDE and LDE scores, as well as by the wall clock CPU time taken by CPLEX running on Eq. (3) and Eq. (4).

It appears clear that DGP instances for linearizable norms can be solved in practice up to at least $n = 40$, though the densest instances are more difficult with the ℓ_∞ norm. A good ℓ_1 norm solution (with MDE and LDE smaller than 10^{-4}) has been obtained using CPLEX on Eq. (3) for a 50 vertex instance with $s = 0.8$ in just over 304s of wall clock time, but no solution was obtained within 10 minutes of wall clock time for a 60 vertex instance and $s = 0.8$. This seems to indicate the need for heuristic methods (discussed below) to address larger instance sizes.

4 Heuristic ideas

In this section we provide some ideas to design heuristic methods for the DGP problem with ℓ_∞ and ℓ_1 norm and report computational results.

4.1 Solving the DMCP in the ℓ_∞ norm

A problem related to DGP, the DISTANCE MATRIX COMPLETION PROBLEM (DMCP), asks whether a partially defined matrix can be completed to a (full) distance matrix in a specified norm. Although a solution of the DMCP is a distance matrix, finding the missing distance values usually entails finding a realization that is consistent with the given values: this realization is then used to compute the missing entries in the given partial matrix. In this sense, realizations provide certificates for both DGP and DMCP. In the DMCP, however, differently from the DGP, the dimensionality K of the embedding space is not given — realizations into \mathbb{R}^K for any $K > 0$ will provide a feasibility certificate for DMCP instances.

In the following we propose a heuristic for DMCP that will be the starting point for a heuristic for DGP. Our first proposal concerns the exploitation of the Fréchet embedding to “approximately solve” the DMCP (rather than the DGP) in the ℓ_∞ norm. Our algorithm works as follows:

1. let A' be the $n \times n$ weighted adjacency matrix of G , where off-diagonal zeros are to be considered as “unspecified entries”
2. complete A' to a full symmetric matrix A using the Floyd-Warshall all-shortest-paths algorithm [16]
3. output the realization $x : V \rightarrow \mathbb{R}^n$ given by the Fréchet embedding $x_i = A_i$ for each $i \in V$.

By “approximately solve” we mean that the output realization x is generally not going to be a valid certificate for the given DMCP instance, but that its MDE and LDE error measures are hopefully going to be low enough. The worst-case complexity of this heuristic is dominated by the Floyd-Warshall algorithm, which is generally $O(n^3)$. Experimentally, we found that this heuristic is useless for sparse graphs (where the reconstruction of A' has the highest chances of being wrong), but is both fast and successful for dense graphs, notably those with $s = 0.8$, which represented the hardest instances in the tests. Thus, in Table 2 of the Appendix, we only report results on instances with $s = 0.8$ and n varying in $\{5 + 5\ell \mid 1 \leq \ell \leq 17\}$. This heuristic took 2.42s to find *all* realizations for 17 random graphs. The cumulative scaled MDE over all instances is 0.019, with cumulative scaled LDE at 9.61 mostly due to one bad outlier. Overall, it found MDE and LDE smaller than 10^{-4} for 11 over 17 instances. In conclusion, it is both fast and effective.

4.2 Solving the DGP in the ℓ_∞ norm

The second idea turns the above heuristic into a method for solving DGP in ℓ_∞ norm for a given dimensionality K : it consists in selecting the K columns

from the realization $x \in \mathbb{R}^n$ obtained in Sect. 4.1 which best match the given distances. For this purpose, we solve the following MILP (based on Eq. (4)):

$$\left. \begin{array}{l}
 \min_{w,y,s} \quad \sum_{\{i,j\} \in E} s_{ij} \\
 \forall \{i,j\} \in E, k \leq n \quad |x_{ik} - x_{jk}|y_k - s_{ij} \leq d_{ij} \\
 \forall \{i,j\} \in E, k \leq n \quad |x_{ik} - x_{jk}|y_k + s_{ij} \geq d_{ij}w_{ijk} \\
 \forall \{i,j\} \in E \quad \sum_{k \leq K} w_{ijk} \geq 1 \\
 \sum_{k \leq n} y_k = K \\
 \forall \{i,j\} \in E \quad s_{ij} \in [0, d_{ij}] \\
 \forall k \leq K, \{i,j\} \in E \quad w_{ijk}, y_k \in \{0, 1\}.
 \end{array} \right\} \quad (5)$$

In Eq. (5), note that x are no longer decision variables, but constants (output of the Fréchet heuristic for the DMCP). The new decision variables $y \in \{0, 1\}^n$ decide whether coordinate k should be in the realization in \mathbb{R}^K or not. Since solving MILP takes exponential time, and we would like the heuristic to be fast, we set a 120s time limit on CPLEX. The results, for instances up to $n = 70$ with s fixed at 0.8, are presented in Table 2 in the Appendix. The average scaled MDE error is non-negligible but still acceptable (around 0.18) while the average scaled LDE is disappointingly close to 0.9. However, this heuristic allows us to solve instances which the MP formulations of Sect. 3 cannot solve.

4.3 Solving the DGP in the ℓ_1 norm

At first, we tested a Variable Neighbourhood Search (VNS) [11] type algorithm with neighbourhoods defined by centers given by infeasible realization x' , and radii given by the maximum number of coordinates of x' that are allowed to change during a BB search. This constraint is enforced by a Local Branching (LB) [8] mechanism added to Eq. (3). The centers are computed using the probabilistic constructive heuristic proposed in [14], which extends the concept of Fréchet embeddings to sets. Unfortunately, this idea yielded very poor results, both quality-wise and in terms of CPU time. We only report it to prevent other researchers from pursuing this same direction.

The second idea we tested is based on alternating solutions between two continuous reformulations of Eq. (2), denoted \mathcal{A} (Eq. (2) with $|\cdot|$ replaced by $\sqrt{(\cdot)^2 + \varepsilon}$) and \mathcal{B} ((3) with nonlinear complementarity constraints): we randomly sample an infeasible realization as a starting point for \mathcal{A} , then use \mathcal{A} 's solution as a starting point to \mathcal{B} . We repeat this loop, updating the “best solution found so far”, until a CPU time-based termination condition (set to 600s) becomes true. We used SNOPT [10] to solve \mathcal{A} and IPOPT [3] to solve \mathcal{B} . The results, obtained on 85 instances with $n \in \{10, 15, 20, \dots, 90\}$ of any density $s \in \{0.1, 0.2, 0.3, 0.5, 0.8\}$, count 13 failures (generally instances with high density where SNOPT had convergence issues) and 8 feasible realizations (scaled MDE/LDE scores $< 10^{-4}$). We report in Table 3 of the Appendix only the results for the instances ended with no failures. Overall, this heuristic displayed average scaled MDE and LDE of 0.13 and 1.16.

References

1. N. Beeker, S. Gaubert, C. Glusa, and L. Liberti. Is the distance geometry problem in **NP**? In A. Mucherino, C. Lavor, L. Liberti, and N. Maculan, editors, *Distance Geometry: Theory, Methods, and Applications*. Springer, New York, 2013.
2. W.-Y. Chiu and B.-S. Chen. Mobile positioning problem in Manhattan-like urban areas: Uniqueness of solution, optimal deployment of BSs, and fuzzy implementation. *IEEE Transactions on Signal Processing*, 57(12):4918–4929, 2009.
3. COIN-OR. *Introduction to IPOPT: A tutorial for downloading, installing, and using IPOPT*, 2006.
4. C. D’Ambrosio, G. Nannicini, and G. Sartor. MILP models for the selection of a small set of well-distributed points. working paper.
5. C. D’Ambrosio, K. Vu, C. Lavor, L. Liberti, and N. Maculan. New error measures and methods for realizing protein graphs from distance data. Technical Report submit-1604830, arXiv, 2016.
6. I. Dokmanić, R. Parhizkar, J. Ranieri, and M. Vetterli. Euclidean distance matrices: Essential theory, algorithms and applications. *IEEE Signal Processing Magazine*, 1053-5888:12–30, Nov. 2015.
7. P. Erdős and A. Rényi. On random graphs i. *Publicationes Mathematicae (Debrecen)*, 6:290–297, 1959.
8. M. Fischetti and A. Lodi. Local branching. *Mathematical Programming*, 98:23–37, 2005.
9. M. Fréchet. Sur quelques points du calcul fonctionnel. *Rendiconti del Circolo Matematico di Palermo*, 22:1–74, 1906.
10. P.E. Gill. *User’s guide for SNOPT version 7.2*. Systems Optimization Laboratory, Stanford University, California, 2006.
11. P. Hansen and N. Mladenović. Variable neighbourhood search: Principles and applications. *European Journal of Operations Research*, 130:449–467, 2001.
12. IBM. *ILOG CPLEX 12.6 User’s Manual*. IBM, 2014.
13. L. Liberti, C. Lavor, N. Maculan, and A. Mucherino. Euclidean distance geometry and applications. *SIAM Review*, 56(1):3–69, 2014.
14. J. Matousek. On the distortion required for embedding finite metric spaces into normed spaces. *Israel Journal of Mathematics*, 93:333–344, 1996.
15. J. Matoušek. Lecture notes on metric embeddings. Technical report, ETH Zürich, 2013.
16. K. Mehlhorn and P. Sanders. *Algorithms and Data Structures*. Springer, Berlin, 2008.
17. J. Saxe. Embeddability of weighted graphs in k -space is strongly **NP**-hard. *Proceedings of 17th Allerton Conference in Communications, Control and Computing*, pages 480–489, 1979.
18. K. Vu, C. D’Ambrosio, Y. Hamadi, and L. Liberti. Surrogate-based methods for black-box optimization. *International Transactions in Operational Research*, to appear.
19. K. Wüthrich. Protein structure determination in solution by nuclear magnetic resonance spectroscopy. *Science*, 243:45–50, 1989.

A Proofs

Proposition 3. *Eq. (3) is a valid formulation for the DGP using the ℓ_1 norm.*

Proof. Assume there exists x^*, s^*, t^*, z^* feasible and optimal in Eq. (3). Then, by suitably projecting these sets of variables on the subspace spanned by the x coordinates, one finds that x^* is a valid solution to Eq. (2), and hence also to the DGP instance, which therefore must be feasible. Conversely, assume that the DGP instance is feasible: then there must be a certificate x^* which satisfies Eq. (2); by carrying out the steps of the reformulation as detailed above, we can construct s^*, t^*, z^* satisfying Eq. (3). \square

Proposition 4. *Eq. (4) is a valid formulation for the DGP using the ℓ_∞ norm.*

Proof. The proof of Prop. 4 has the same structure as the proof of Prop. 3. The only detail worthy of note is that, at the optimum, the binary w_{ijk} variables select for each edge $\{i, j\} \in E$ at least one component k such that $|x_{ik} - x_{jk}| \geq d_{ij}$: and if *one* component has that property, the component of *maximum value* must also have that property, which establishes the correctness of the non-convex part of the ℓ_∞ norm constraint (*). \square

Theorem 2. *The DGP in linearizable norms is **NP**-complete.*

The proof is given in the Appendix.

Proof. It was already remarked in Sect. 2 that the DGP in linearizable norms is **NP**-hard by reduction from PARTITION to the DGP in $K = 1$, where the three norm $\ell_1, \ell_2, \ell_\infty$ coincide. Now by Prop. 3 and 4 we know that certificates to the DGP in linearizable norms are rational for rational input, since the MP formulations in (3) and (4) can be solved by a Branch-and-Bound (BB) algorithm, each node of which involves the solution of a Linear Program (LP), which is known to be in **NP**. Simple BB implementations will continue branching until the incumbent, found by solving the LP at some node, is proven optimal. So the certificate (solution) provided by the BB is polynomially sized, as claimed. \square

B Detailed Tables

Instance n	s	ℓ_1 norm			ℓ_∞ norm		
		MDE	LDE	CPU t	MDE	LDE	CPU t
10	0.1	6.66E-17	1.83E-16	0.06	2.98E-17	1.63E-16	0.12
10	0.2	8.19E-08	1.09E-06	0.07	4.94E-08	6.91E-07	0.07
10	0.3	6.93E-07	5.15E-06	0.11	7.64E-07	6.99E-06	0.14
10	0.5	5.58E-07	4.75E-06	0.3	6.25E-07	5.70E-06	0.28
10	0.8	1.50E-06	1.21E-05	0.36	1.52E-06	8.52E-06	1.48
15	0.1	8.17E-17	4.22E-16	0.05	1.85E-17	2.07E-16	0.06
15	0.2	6.05E-07	7.18E-06	1.8	1.71E-07	3.66E-06	1.6
15	0.3	7.01E-07	5.52E-06	1.61	1.86E-06	1.73E-05	1.89
15	0.5	1.52E-06	1.35E-05	0.65	1.87E-06	1.29E-05	5.94
15	0.8	1.85E-06	1.16E-05	1.86	1.86E-06	8.09E-06	5.64
20	0.1	5.62E-07	4.75E-06	5.71	1.70E-08	6.63E-07	4.31
20	0.2	1.23E-06	1.84E-05	2.07	1.23E-06	1.95E-05	13.05
20	0.3	1.20E-06	9.28E-06	5.27	1.95E-06	1.17E-05	14.05
20	0.5	1.86E-06	1.92E-05	2.32	2.63E-06	1.97E-05	9.88
20	0.8	2.06E-06	1.55E-05	5.54	2.72E-06	1.08E-05	53.63
25	0.1	8.52E-07	9.48E-06	4.37	7.47E-07	1.04E-05	4.4
25	0.2	1.88E-06	3.24E-05	5.97	8.10E-07	7.67E-06	16.39
25	0.3	1.84E-06	1.54E-05	6.53	1.77E-06	9.37E-06	46.54
25	0.5	1.88E-06	1.33E-05	15.89	2.67E-06	1.68E-05	63.48
25	0.8	2.19E-06	1.65E-05	14.68	7.36E-04	3.45E-02	600.09
30	0.1	1.23E-06	1.90E-05	11.29	9.61E-07	3.50E-05	25.59
30	0.2	1.74E-06	4.19E-05	8.76	1.34E-06	1.20E-05	79.36
30	0.3	1.45E-06	1.64E-05	5.51	2.17E-06	1.13E-05	42.43
30	0.5	2.70E-06	2.32E-05	22.22	2.79E-06	1.80E-05	51.67
30	0.8	2.42E-06	3.91E-05	19.67	0.0824708	0.925715	600.33
35	0.1	6.37E-07	1.07E-05	35.54	9.50E-07	1.85E-05	504.89
35	0.2	1.83E-06	1.28E-05	23.34	1.77E-06	1.07E-05	17.87
35	0.3	1.90E-06	1.83E-05	32.55	2.56E-06	1.99E-05	36.82
35	0.5	2.44E-06	1.49E-05	33.24	3.32E-06	1.62E-05	358.36
35	0.8	3.03E-06	4.81E-05	13.57	0.137722	1	600.33
40	0.1	1.45E-06	1.71E-05	33.97	1.06E-06	1.09E-05	566.18
40	0.2	2.35E-06	4.23E-05	25.49	2.24E-06	6.62E-05	172.54
40	0.3	1.93E-06	1.99E-05	21.74	2.40E-06	1.81E-05	79.45
40	0.5	2.05E-06	4.16E-05	64.74	2.54E-06	5.90E-05	600.11
40	0.8	3.08E-06	1.70E-04	80.86	0.312365	1	600.15

Table 1. Results on instances for the DGP in ℓ_1 and ℓ_∞ norms formulated as MILPs.

n	MDE	LDE	CPU t
10	2.286187E-07	1.930254E-06	0.002
15	4.879686E-07	4.824712E-06	0.004
20	6.895810E-07	6.353593E-06	0.008
25	7.991818E-07	5.997201E-06	0.013
30	7.213860E-07	7.283439E-06	0.022
35	1.438231E-02	4.075396E+00	0.047
40	9.289797E-07	2.514727E-05	0.048
45	1.044442E-06	1.336336E-05	0.062
50	8.630696E-07	1.631295E-05	0.087
55	1.196788E-06	3.177958E-05	0.127
60	1.121936E-06	4.064804E-05	0.140
65	6.429619E-04	4.537939E-01	0.185
70	1.424837E-03	9.845047E-01	0.215
75	1.319162E-06	4.709088E-05	0.298
80	1.131649E-03	1.803801E+00	0.321
85	2.770027E-04	5.781805E-01	0.382
90	8.615075E-04	1.721325E+00	0.468

n	MDE	LDE	CPU t
10	8.563691E-02	5.568593E-01	28.87
15	1.127657E-01	8.739597E-01	63.04
20	8.838678E-02	6.704091E-01	80.66
25	1.556823E-01	9.845021E-01	82.46
30	8.102230E-02	9.166102E-01	97.07
35	1.805664E-01	9.613564E-01	127.03
40	1.614281E-01	9.990920E-01	164.49
45	2.473848E-01	9.953499E-01	231.05
50	3.023465E-01	9.970702E-01	320.09
55	1.662879E-01	9.953521E-01	303.27
60	3.023465E-01	9.970702E-01	361.52
65	2.405345E-01	9.990544E-01	551.41
70	3.020851E-01	9.890115E-01	718.16

Table 2. Results on instances for the DMCP (left) and DGP (right) in ℓ_∞ norm: heuristic approach.

n	s	MDE	LDE	CPU t	CPU t best sol
10	0.1	0.000000E+00	0.000000E+00	0.30	0.30
10	0.2	1.290000E-02	1.329000E-01	120.49	0.07
10	0.3	4.400000E-03	5.660000E-02	120.64	111.55
10	0.5	1.800000E-02	1.765000E-01	120.72	0.80
10	0.8	2.770000E-02	2.758000E-01	120.00	2.00
15	0.1	0.000000E+00	0.000000E+00	3.84	3.84
15	0.2	4.640000E-02	3.172000E-01	120.36	116.72
15	0.3	6.340000E-02	4.219000E-01	120.75	1.59
15	0.5	8.300000E-02	1.000000E+00	120.79	86.81
15	0.8	3.740000E-02	2.314000E-01	120.06	95.06
20	0.1	3.670000E-02	6.148000E-01	120.90	28.48
20	0.2	1.009000E-01	9.554000E-01	120.07	1.65
20	0.3	1.297000E-01	9.995000E-01	120.30	1.99
20	0.5	1.506000E-01	1.907300E+00	120.86	1.25
20	0.8	3.530000E-01	1.000000E+00	121.25	1.01
25	0.1	7.980000E-02	1.000000E+00	120.07	110.30
25	0.2	1.596000E-01	8.961000E-01	120.93	1.23
25	0.3	1.813000E-01	1.011000E+00	120.70	5.41
25	0.5	7.000000E-04	2.040000E-02	120.30	106.30
25	0.8	3.380000E-01	1.000000E+00	121.40	7.75
30	0.1	1.111000E-01	9.917000E-01	120.66	20.26
30	0.2	1.983000E-01	1.000000E+00	121.14	15.01
30	0.3	1.990000E-01	1.000000E+00	120.72	106.70
30	0.5	1.110000E-02	2.233000E-01	120.38	13.21
35	0.1	1.126000E-01	1.000000E+00	120.92	2.64
35	0.2	2.055000E-01	1.070000E+00	120.43	51.05
35	0.3	2.296000E-01	2.163100E+00	120.71	23.29
35	0.5	2.402000E-01	8.525000E-01	120.91	3.89
35	0.8	9.000000E-03	2.702000E-01	121.61	71.38
40	0.1	1.864000E-01	9.790000E-01	120.78	1.25
40	0.2	2.557000E-01	2.302400E+00	120.58	20.97
40	0.3	1.430000E-01	1.000000E+00	120.78	4.64
40	0.5	1.270000E-02	5.133000E-01	120.15	15.24
40	0.8	4.025000E-01	9.899000E-01	121.08	33.50
45	0.1	1.973000E-01	1.000000E+00	602.41	141.86
45	0.2	2.101000E-01	1.000000E+00	602.91	7.66
45	0.3	1.642000E-01	9.063000E-01	14007.03	5.51
45	0.5	0.000000E+00	0.000000E+00	602.48	113.32
50	0.1	1.916000E-01	1.000000E+00	601.27	2.48
50	0.3	3.300000E-03	6.150000E-02	600.35	59.85
50	0.5	0.000000E+00	0.000000E+00	602.78	16.08
50	0.8	0.000000E+00	0.000000E+00	605.49	405.00
55	0.1	2.339000E-01	1.312800E+00	602.36	12.04
55	0.2	1.293000E-01	1.156500E+00	601.38	237.93
55	0.3	1.800000E-02	1.831000E-01	602.87	23.11
60	0.1	2.427000E-01	4.894700E+00	602.15	141.55
60	0.2	2.960000E-02	4.482000E-01	602.53	406.05
60	0.3	4.200000E-03	1.596000E-01	605.23	28.47
60	0.8	0.000000E+00	1.000000E-04	2363.12	2363.12
65	0.1	1.548000E-01	9.515000E-01	601.35	20.35
65	0.2	7.860000E-02	9.359000E-01	601.45	554.45
65	0.3	1.430000E-02	5.405000E-01	624.85	58.08
65	0.5	3.952000E-01	1.000000E+00	607.52	599.71
70	0.1	2.611000E-01	3.027700E+00	602.80	51.53
70	0.2	1.280000E-01	1.294800E+00	604.24	577.02
70	0.3	0.000000E+00	0.000000E+00	601.85	601.85
75	0.1	2.092000E-01	1.190300E+00	601.55	7.00
75	0.2	8.140000E-02	1.839200E+00	602.42	78.33
75	0.3	2.891000E-01	2.428910E+01	680.06	392.29
75	0.5	4.054000E-01	9.739000E-01	644.90	223.76
75	0.8	4.180000E-01	1.000000E+00	799.64	799.64
80	0.1	2.786000E-01	2.431000E+00	601.54	262.93
80	0.2	4.740000E-02	5.413000E-01	601.55	22.08
80	0.3	3.630000E-01	1.000000E+00	614.46	488.08
80	0.5	3.421000E-01	1.000000E+00	677.27	385.06
80	0.8	0.000000E+00	0.000000E+00	1238.01	1238.01
85	0.2	3.240000E-02	1.000000E+00	615.05	94.76
85	0.3	2.000000E-04	2.060000E-02	602.50	83.70
90	0.1	2.964000E-01	1.000000E+00	605.59	16.87
90	0.2	6.700000E-03	3.279000E-01	605.21	241.30
90	0.3	1.000000E-03	1.259000E-01	2553.13	2553.13
90	0.5	3.356000E-01	1.000000E+00	798.64	770.28

Table 3. Results on instances for the DGP in ℓ_1 norm: heuristic approach.