A Hybrid Quantum-Classical Algorithm for Robust Fitting Supplementary Material

Anh-Dzung Doan [1] Michele Sasdelli [1] David Suter [2] Tat-Jun Chin [1]
[1] School of Computer Science, The University of Adelaide, South Australia
[2] Centre for AI&ML, School of Science, Edith Cowan University, Western Australia

A. Proof of claim 1

By construction, $f(\mathcal{I}^*) = 0$. Given an infeasible basis $\mathcal{B}$, by monotonicity

$$1 = f(\mathcal{B}) \leq f(\mathcal{I}^* \cup \mathcal{B}) = 1. \tag{1}$$

Thus, $\mathcal{B}$ must contain a point not in $\mathcal{I}^*$.

B. Proof of claim 2

Recall that an independent set of a hypergraph is a subset of the vertices where none of the members of the subset form a hyperedge [2, Chapter 2].

Given $f(\mathcal{I}) = 0$, we have that $f(\mathcal{A}) = 0$ for all $\mathcal{A} \subseteq \mathcal{I}$ due to monotonicity. Hence, the vertices in $\mathcal{I}$ do not form hyperedges, i.e., $\mathcal{I}$ is an independent set.

Let $\mathcal{I}$ be an independent set of H, and $\Omega = \{\mathcal{S}_1, \mathcal{S}_2, \dots\}$ be all δ-subsets of $\mathcal{C}$. By construction,

$$g(\mathcal{S}_\ell) \leq \epsilon \quad \forall \mathcal{S}_\ell \in \Omega \tag{2}$$

since there are no hyperedges in $\mathcal{I}$. Let $\mathcal{S}_* \in \Omega$ such that

$$g(\mathcal{S}_\ell) \leq g(\mathcal{S}_*) \leq \epsilon \quad \forall \mathcal{S}_\ell \in \Omega. \tag{3}$$

Suppose $g(\mathcal{I}) > \epsilon$: from [1, 6], there is a support set $\bar{\mathcal{S}}$—which is also a basis and hence a δ-subset—of $\mathcal{I}$ such that

$$g(\mathcal{I}) = g(\bar{\mathcal{S}}), \tag{4}$$

which implies that $\bar{\mathcal{S}} \in \Omega$ but $\bar{\mathcal{S}} \neq \mathcal{S}_*$, and

$$g(\bar{\mathcal{S}}) > \epsilon \geq g(\mathcal{S}_*), \tag{5}$$

which contradicts (3). Thus, we must have that $g(\mathcal{I}) \leq \epsilon$, i.e., $\mathcal{I}$ is a consensus set.

C. Reformulating QUBO for D-Wave solvers

Recall the QUBO (19)

$$Q_\lambda(A) = \min_{\mathbf{v} \in \{0,1\}^{N+\delta' M}} \begin{bmatrix}\mathbf{v}^T & 1\end{bmatrix} (\mathbf{J} + \lambda \mathbf{H}_A^T \mathbf{H}_A) \begin{bmatrix}\mathbf{v}^T & 1\end{bmatrix}^T$$

Let $\mathbf{Q} = \mathbf{J} + \lambda \mathbf{H}_A^T \mathbf{H}_A$, and denote q_{ij} is the element in i^{th}-row and j^{th}-column of $\mathbf{Q}$

D-Wave solvers accept $\mathbf{Q}$ as a upper-triangular matrix, which can be obtained by following procedure

- For every i and j, if $j > i$, then $q_{ij} = q_{ij} + q_{ji}$
- For every i and j, if $j < i$, then $q_{ij} = 0$

We attain the new QUBO with upper-triangular $\mathbf{Q}$

$$Q_\lambda(A) = \min_{\mathbf{v} \in \{0,1\}^{N+\delta' M}} \begin{bmatrix}\mathbf{v}^T & 1\end{bmatrix} \mathbf{Q} \begin{bmatrix}\mathbf{v}^T & 1\end{bmatrix}^T$$

which, however, still cannot be applied to D-Wave solvers since the variables are in the form $\begin{bmatrix}\mathbf{v}^T & 1\end{bmatrix}$. However, this formulation can be rewritten using a simple derivation.

Suppose

$$\mathbf{v} = \begin{bmatrix} v_1 & v_2 & v_3 \end{bmatrix} \tag{6}$$

$$\mathbf{Q} = \begin{bmatrix} q_{11} & q_{12} & q_{13} & q_{14} \\ 0 & q_{22} & q_{23} & q_{24} \\ 0 & 0 & q_{33} & q_{34} \\ 0 & 0 & 0 & q_{44} \end{bmatrix}. \tag{7}$$

We then take the last column of $\mathbf{Q}$ except the last element q_{44}, which yields $\mathbf{q} = \begin{bmatrix} q_{14} & q_{24} & q_{34} \end{bmatrix}^T$. Next, we add $\mathbf{q}$ to the diagonal of $\mathbf{Q}$

$$\mathbf{Q}' = \begin{bmatrix} q_{11} + q_{14} & q_{12} & q_{13} \\ 0 & q_{22} + q_{24} & q_{23} \\ 0 & 0 & q_{33} + q_{34} \end{bmatrix}. \tag{8}$$

Since $\mathbf{v}$ is a binary vector, i.e., $v_i^2 = v_i$, we can obtain

$$\begin{bmatrix}\mathbf{v}^T & 1\end{bmatrix} \mathbf{Q} \begin{bmatrix}\mathbf{v}^T & 1\end{bmatrix}^T = \mathbf{v}^T \mathbf{Q}' \mathbf{v} + q_{44}. \tag{9}$$

Therefore, we get the new QUBO

$$Q_\lambda(A) = \min_{\mathbf{v} \in \{0,1\}^{N+\delta' M}} \mathbf{v}^T \mathbf{Q}' \mathbf{v} + \text{constant}, \tag{10}$$

which can be directly applied to D-Wave solvers.

D. Spectral gap

Computation of spectral gap. Recall matrix $\mathbf{Q}'$ of QUBO (10). For simplicity, let

$$n = N + \delta' M \tag{11}$$

thus $\mathbf{Q}'$ is an *upper-triangular matrix* of the size $n \times n$, and denote $q'_{i,j}$ is the element in i^{th}-row and j^{th}-column of $\mathbf{Q}'$.

The QUBO problem (10) is firstly converted to Ising problem [5]

$$h_i = \frac{q'_{ii}}{2} + \sum_{j=1}^{n} \frac{q'_{ij}}{4}, \tag{12}$$

$$J_{ij} = \frac{q'_{ij}}{4} \tag{13}$$

for all $i \in \{1, \dots, n\}$ and all $i < j$. In QPU, h_i are termed biases, and J_{ij} are called couplings. Then, biases and couplings are normalised such that $h_i \in [-2, 2]$ and $J_{ij} \in [-1, 1]$, since D-Wave limits the value range of biases in $[-2, 2]$ and couplings in $[-1, 1]$ [4]. Next, the initial and final Hamiltonians are computed

$$H_{\text{init}} = \sum_i \hat{\sigma}_x^{(i)}, \tag{14}$$

$$H_{\text{final}} = \sum_i h_i \hat{\sigma}_z^{(i)} + \sum_{i<j} J_{ij} \hat{\sigma}_z^{(i)} \hat{\sigma}_z^{(j)} \tag{15}$$

where,

$$\hat{\sigma}_x^{(i)} = \overbrace{\mathbf{I} \otimes \mathbf{I} \otimes \cdots \otimes \mathbf{I} \otimes \sigma_x \otimes \mathbf{I} \otimes \cdots \otimes \mathbf{I}}^{N}, \quad i^{\text{th}} \text{ position}$$

$$\hat{\sigma}_z^{(i)} = \overbrace{\mathbf{I} \otimes \mathbf{I} \otimes \cdots \otimes \mathbf{I} \otimes \sigma_z \otimes \mathbf{I} \otimes \cdots \otimes \mathbf{I}}^{N}, \quad i^{\text{th}} \text{ position}$$

$$\hat{\sigma}_z^{(i)} \hat{\sigma}_z^{(j)} = \overbrace{\mathbf{I} \otimes \cdots \otimes \sigma_z \otimes \mathbf{I} \otimes \cdots \otimes \mathbf{I} \otimes \sigma_z \otimes \cdots \otimes \mathbf{I}}^{N}, \quad i^{\text{th}} \text{ position}, \ j^{\text{th}} \text{ position}$$

$$\mathbf{I} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad \sigma_x = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \quad \sigma_z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix},$$

the Hamiltonian of the quantum computer is represented as

$$H(s) = (1 - s) H_{\text{init}} + s H_{\text{final}}, \tag{16}$$

where $s \in [0, 1]$ is the normalised annealing time.

For a particular s, $H(s)$ is a $2^N \times 2^N$ matrix, which is then decomposed to obtain the smallest and second smallest eigenvalues. The eigenspectra of smallest and second smallest eigenvalues respectively represent the ground state and the first excited (high) energy state. The minimum gap between those two eigenspectra represents the spectral gap.

Results. The synthetic data with the setting same as that of Sec. 6.1 is generated, where $N = 5$, outlier ratio $= 0.4$, $\lambda \in [0.1, 100]$, and A containing all hyperedges E.

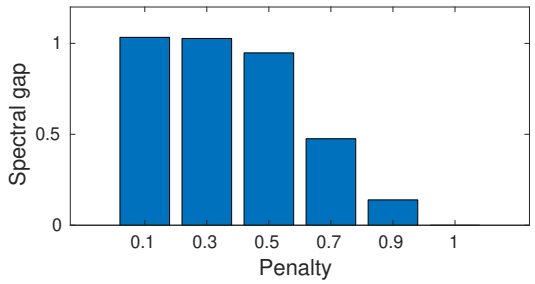


Figure 1. Spectral gap.

As the penalty λ increases, the spectral gap quickly reduces (Fig. 1). This implies that the probability of the quantum system remains in the ground state during the annealing time decreases with larger λ.

The eigenvalues of specific penalty λ is also shown in Fig. 2, which indicates the decrease of spectral gap with larger penalty λ.

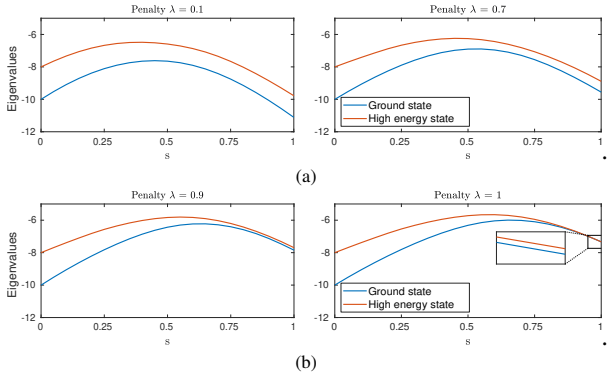


Figure 2. Eigenvalues of ground state and first excited (high) energy state for every penalty value λ, where the minimum gap between two eigenspectrums is the spectral gap.

E. Benefit of penalty decay

Fig. 3 shows the comparison between fixing λ and decaying λ in fundamental matrix estimation. The decay parameters are set same as those in Sec. 6.2.1. If the penalty λ is large ($\lambda = 1$), Alg. 1 can quickly find a consensus set but the error bounds cannot be tightened. By contrast, if the penalty λ is small ($\lambda = 0.02$), Alg. 1 will require more iterations to find a consensus set. Therefore, decaying penalty λ is a reasonable strategy that helps Alg. 1 quickly find a consensus set and efficiently tighten the error bound.

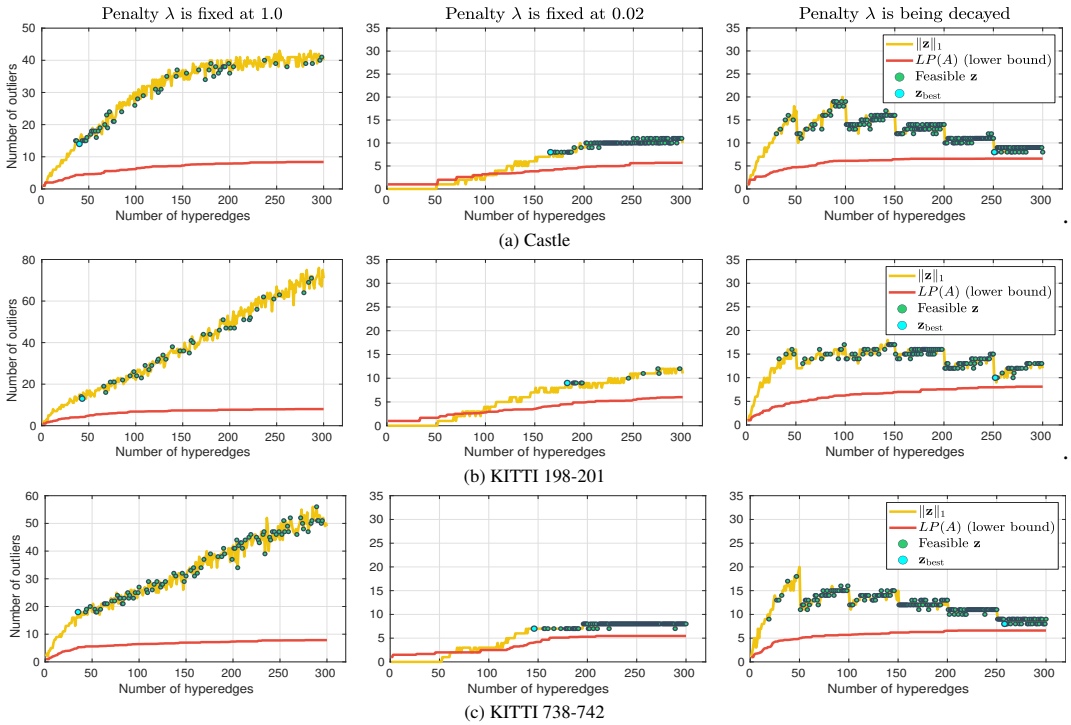


Figure 3. Comparing fixed penalty λ to penalty λ being decayed. The error bound cannot be tightened with a large penalty (left), while Alg. 1 requires more iterations to find a consensus set (middle). Therefore, decaying penalty can quickly find a consensus set and efficiently tighten the error bound (right).

F. Minor embedding

Before quantum annealing, minor embedding should be performed to embed $\mathbf{Q}'$ (Eq. (10)) to the QPU topology.

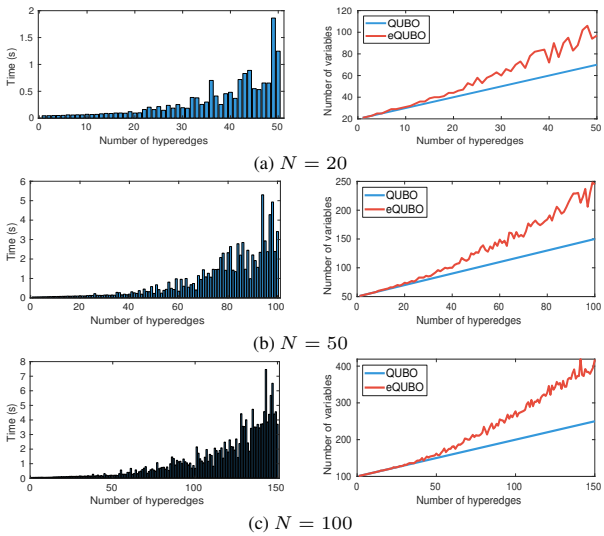


Figure 4. The embedding time (left) and total number of variables (right) of QUBO (10) before and after embedding to QPU topology (respectively denoted as QUBO and eQUBO).

To investigate the minor embedding, synthetic data with the setting same as that of Sec. 6.1 is generated, where $N = 20, 50, 100$ and outlier ratio $= 0.2$. In every iteration of Alg. 1, we measure the embedding time and total number of variables of QUBO (10) before and after embedding (see Fig. 4). In all cases, the embedding time and number of variables increase as more hyperedges are sampled. Also see Fig. 5 for the visualisation of the embedding on the QPU.

G. More experimental results on real data

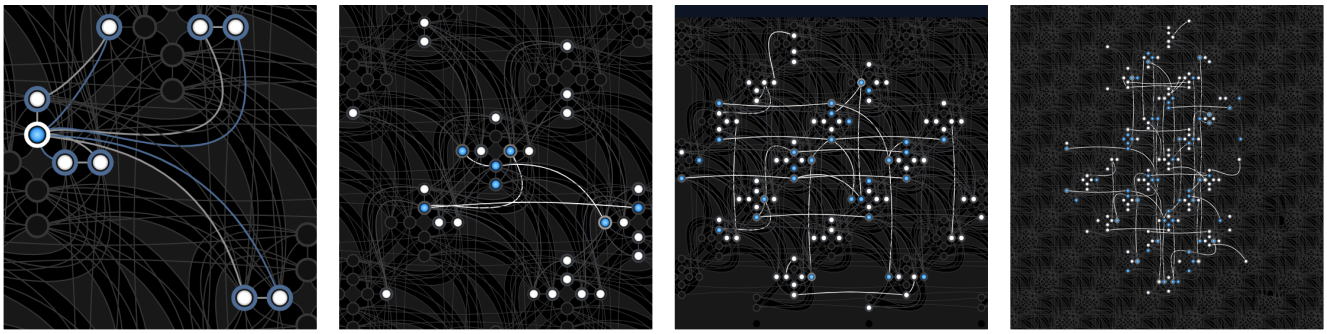
Fundamental matrix estimation. Fig. 6 shows the intermediate outputs of Alg. 1-F on Zoom, Valbonne and KITTI 104-108. A same conclusion as Sec. 6.2.1 can be drawn.

Triangulation. Fig. 7 shows the Alg. 1-F's intermediate outputs on Nikolai point 534, Linkoping point 14 and Tower point 3. A same conclusion as Sec. 6.2.2 can be drawn.

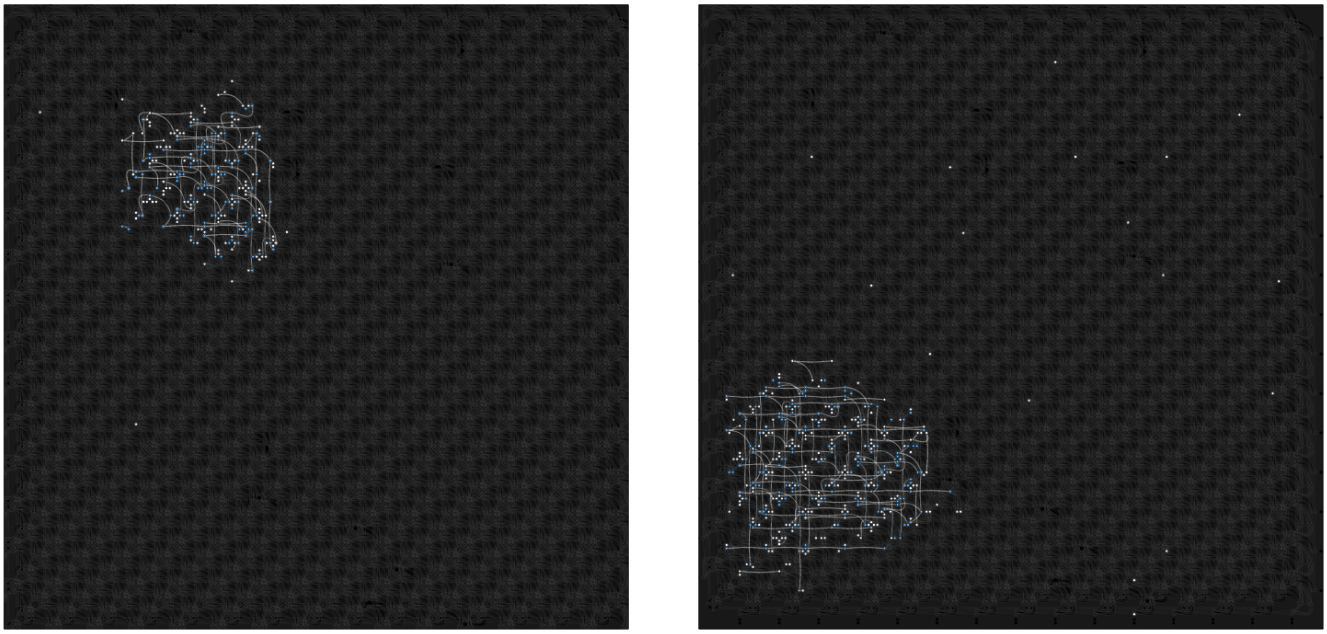
References

[1] Nina Amenta, Marshall Bern, and David Eppstein. Optimal point placement for mesh smoothing. *Journal of Algorithms*, 1999.

[2] Alain Bretto. *Hypergraph Theory: An Introduction*. Springer International Publishing, 2013.



(a) (From left to right) N=5 (# hyperedges = 4), N=10 (# hyperedges = 16), N = 20 (# hyperedges = 50), and N = 30 (# hyperedges = 80).



(b) (From left to right) N=50 (# hyperedges = 100), and N=100 (# hyperedges = 150)

Figure 5. Visualisation of the embedding using D-Wave problem inspector [3]. Each node represents a physical qubit. In each node, the colors in the outer rings represents the signs of qubit biases measured in the lowest energy state; and the inner colors represent the solutions. Edges represent the coupling strengths.

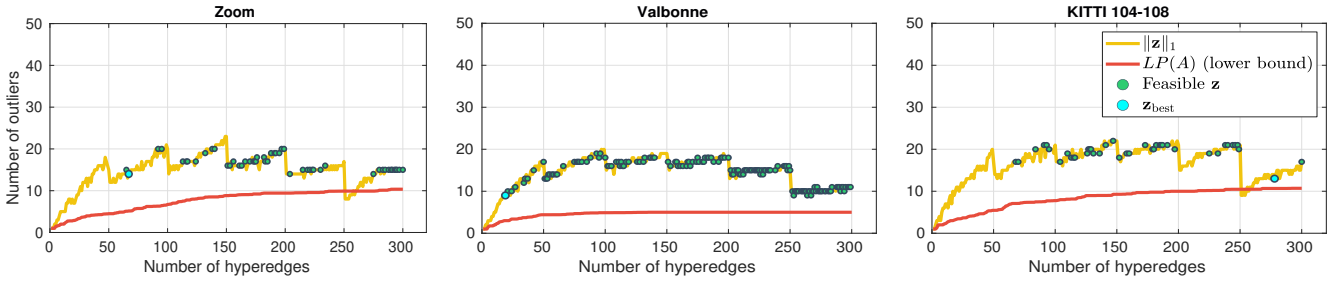


Figure 6. Fundamental matrix estimation, where number of outliers $\|\mathbf{z}\|_1$ and lower bound $LP(A)$, plotted across the iterations of Alg. 1-F.

[3] D-Wave Systems, Inc. D-Wave Problem Inspector. https://docs.ocean.dwavesys.com/projects/inspector/en/latest/index.html. Accessed: 2021-11-14.

[4] D-Wave Systems, Inc. Solver Properties. https://docs.dwavesys.com/docs/latest/c_solver_properties.html. Accessed: 2021-11-14.

[5] Hristo N Djidjev, Guillaume Chapuis, Georg Hahn, and Guillaume Rizk. Efficient combinatorial optimization using quantum annealing. *arXiv preprint arXiv:1801.08653*, 2018.

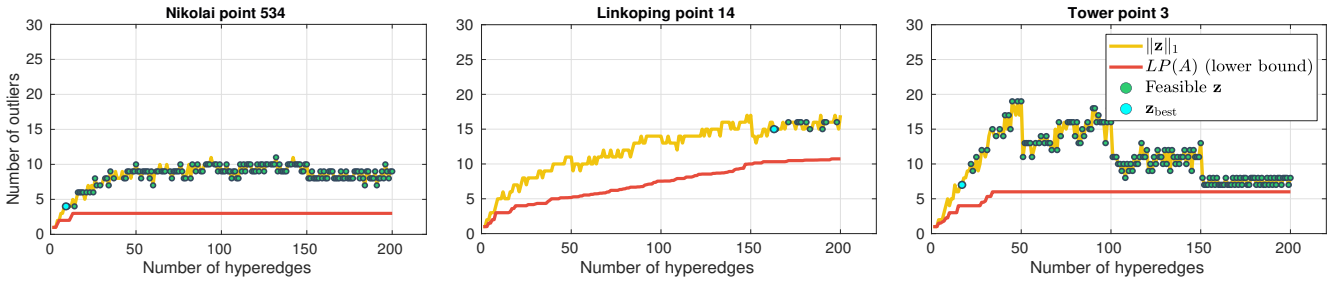


Figure 7. Multi-view triangulation, where number of outliers $\|\mathbf{z}\|_1$ and lower bound $LP(A)$ plotted across the iterations of Alg. 1-F.

[6] David Eppstein. Quasiconvex programming. *Combinatorial and Computational Geometry*, 2005.