that $|\beta_2| \leq 2^{w(M')+r|E|}$. See full version for the details. The main argument is that since these matchings are not BestCSMs, they have signature strictly worse than the BestCSMs and contribute a much smaller amount to the coefficient. Then we have that $|\beta_1| - |\beta_2| > 0$. Thus, with probability at least $\frac{1}{2}$, the coefficient of π is non-zero. □

Running time: In both of these cases, the overall running time is primarily decided by the step where we check the coefficient of π. This can be done with multi-variate polynomial interpolation. For example, through the approach in [8], we can do this in time n^{r^2}. The other manipulations of the determinant and the instance require at most $O(n^4)$ operations, giving an overall runtime of $O(n^{r^2+4})$. The arithmetic on large numbers adds an overhead of $O(n^r)$ for the case of cumulative better signatures, giving an overall runtime of $O(n^{r^2+r+4})$. When $r = O(1)$, the algorithm runs in polynomial time in either case.

4 HARDNESS RESULTS IN THE FIXED QUOTA SETTING

We prove our hardness results when every post $p \in \mathcal{P}$ has a unit quota, and every applicant has strict preferences over the posts. All our hardness results are based on the hardness of a special case of labelled perfect matching defined in [16]. We note that there is a rich literature on similar problems under various other names such as rainbow matching (for example, see [13]). We show that a variant of the problem in [16], that we call LABELLED MATCHING, is also NP-hard. We define this as follows.

Problem 4.1 (LABELLED MATCHING). *Given a bipartite graph $G = (A \cup B, E)$ with $|A| = |B| = n$, and $E = E_1 \cup E_2 \cup \ldots \cup E_r$ be a partition of edges such that every vertex is adjacent to at most one edge from each E_i, does the instance admit a matching M such that $|E_i \cap M| \geq 1$ for every $i \in [r]$?*

Proposition 4.2. LABELLED MATCHING *is NP-hard even on 2-regular graphs.*

Now we go back to our original problem of EXACT-SIGN-Q. We show the hardness reduction from LABELLED MATCHING to EXACT-SIGN-Q to prove our hardness part of Theorem 1.1. We defer the proof to the full version of the paper. The main idea is to treat the partition of edges $E_1, E_2, \ldots, E_r$ from LABELLED MATCHING as the ranks for the applicants. However, this might cause there to be 'gaps' in the preference lists of applicants. For example, a vertex incident to an edge from E_1 and E_3 but not E_2 would have a rank-1 and rank-3 post, but not a rank-2 post. This can be avoided by adding dummy applicants and posts in a careful way. Next, we show the hardness of CUM-SIGN-Q. We show a reduction from EXACT-SIGN-Q.

Proof of Theorem 1.2(Hardness). Given an instance of EXACT-SIGN-Q on a graph G with input signature $\rho = (\rho_1, \rho_2, \ldots, \rho_r)$ with $r \geq 2$, we reduce to an instance of CUM-SIGN-Q with $2r + 1$ ranks. We assume without loss of generality that $V = \mathcal{A} \cup \mathcal{P}$ with $|\mathcal{A}| = |\mathcal{P}| = n$ and $\sum_{1 \leq j \leq r} \rho_j = n$. Let the applicants and posts be numbered from 1 to n.

Let $G = (V, E_1 \cup E_2 \ldots \cup E_r)$ where E_j is the set of rank-j edges. We construct a graph H as follows. Make r copies of the vertex set $V = \mathcal{A} \cup \mathcal{P}$ times to get graphs $G_1, G_2, \ldots, G_r$ where G_j has vertex set $(\mathcal{A}^{(j)} \cup \mathcal{P}^{(j)})$ and only contains rank-j edges, i.e. E_j. Let the edges of E_j have rank-$2j$ in G_j. Let $a_i^{(j)}$ represent the copy of a_i in G_j and similarly let $p_i^{(j)}$ represent the copy of p_i in G_j. We add $(r-1) \cdot n$ dummy posts and applicants

$$D_P = \{d_p^{(1,1)}, \ldots, d_p^{(1,n)}, d_p^{(2,1)} \ldots, d_p^{(r-1,n)}\}$$

$$D_A = \{d_a^{(1,1)}, \ldots, d_a^{(1,n)}, d_a^{(2,1)} \ldots, d_a^{(r-1,n)}\}$$

such that for every i, j, $d_p^{(1,i)}, \ldots, d_p^{(r-1,i)}$ are all connected to copies of $a_i^{(j)}$ through rank-1 edges and for every i, j, $d_a^{(1,i)}, \ldots, d_a^{(r-1,i)}$ are all connected to $p_i^{(j)}$ through rank-$2j+1$ edges.

Lemma 4.3. *There is a matching M in G with $\sigma(M) = \rho$ iff there is a matching M' in H such that $\sigma(M') \geq_C ((r-1) \cdot n, \rho_1, n - \rho_1, \rho_2, n - \rho_2, \ldots, n - \rho_r) = \pi$.*

Given M in G, it is easy to recover M' in H. We do the following: for all rank-j edges matched in G, match the corresponding edge in G_j in H. For all the unmatched vertices, match them to dummy applicants or posts. It can be observed that $\sigma(M') = \pi \geq_C \pi$.

Now suppose we have a matching M in H with $\sigma(M) = \pi' \geq_C \pi$. We will show that $\pi' = \pi$. This can be done via a series of claims whose proofs we defer to the full version.

Claim 4.4. *M is a perfect matching for H.*

Claim 4.5. *For every $a_i \in \mathcal{A}$, exactly $r-1$ copies of a_i are matched to the dummy posts D_P. For every $p_i \in \mathcal{P}$, exactly $r-1$ copies of p_i are matched to the dummy applicants D_A.*

Claim 4.6. *For every $j \geq 1$, $\pi'_{2j} \geq \rho_j$ and $\pi'_{2j} + \pi'_{2j+1} = n$.*

Lemma 4.7. *$\pi' = \pi$.*

Now, we need to show that there is a matching in G that achieves signature ρ. We do this in the following way: for every applicant $a \in \mathcal{A}$, match it to $p \in \mathcal{P}$ such that some copy of a is matched to some copy of p in M. From Claim 4.5, this gives us a matching since every applicant is matched to exactly one non-dummy post and vice versa. Consider Lemma 4.7 and the observation that rank-$2j$ edges in H correspond to rank-j edges in G, the corresponding matching has signature ρ. This concludes the proof of Lemma 4.3. We observe that the above constructed instances have ties and gaps in the preference lists. These can be removed via some minor bookkeeping. This can be done in a manner similar to that in the proof of hardness part of Theorem 1.1. We defer these details to the full version. This then concludes the proof of hardness part of Theorem 1.2. □

5 EXPERIMENTS

We present the empirical evaluation of the cumulative better signature for the fixed quota setting as well as the cost-based setting. We report results on available real-world data sets as well as synthetically generated data sets. The experiments were conducted on a laptop running on a 64-bit Windows 10 Home edition and equipped with an Intel Core i5-8250U CPU @1.60GHz and 8GB of RAM. We used IBM ILOG CPLEX Optimization Studio 20.1 with Python APIs

Table 1: Fixed quota setting results on the real-world data sets. All values are given in percentage. The cells with background green meet the input requirement whereas the ones with background pink fail to meet the input requirement.

Data set	Input requirement			CSM			RMM			FM		
	#rank 1	#top 3 ranks	size	#rank 1	#top 3 ranks	size	#rank 1	#top 3 ranks	size	#rank 1	#top 3 ranks	size
Real-1	≥ 60	≥ 85	≥ 90	60.04	85.00	91.23	66.03	81.41	85.96	48.73	84.48	93.82
Real-2	≥ 60	≥ 80	≥ 85	60.02	80.01	85.55	61.88	78.02	82.99	51.82	80.59	88.74
Real-3	≥ 70	≥ 90	≥ 95	70.01	90.02	95.49	73.02	87.55	93.13	60.84	90.45	97.42

Table 2: Cost-based setting results on the real-world data sets. All values are given in percentage.

Data set	Input requirement			CF-1		CF-2		CF-3	
	#rank 1	#top 3 ranks	size	max violation	total violation	max violation	total violation	max violation	total violation
Real-1	≥ 65	≥ 85	≥ 90	**3.21**	14.81	**5.37**	10.25	**10.73**	17.73
Real-2	≥ 65	≥ 80	≥ 85	**3.97**	17.38	**4.43**	13.33	**4.06**	14.86
Real-3	≥ 75	≥ 90	≥ 95	**5.04**	10.46	**4.56**	12.34	**3.65**	11.21

to solve integer linear programs. The results on synthetic data sets are deferred to the full version.

Real-world data sets: The data sets Real-1, Real-2 and Real-3 are obtained from the elective allocation at an educational institution for three different periods. Each data set has around 2000 students (applicants in our model) and 100 courses (posts in our model). Each course has an upper bound on the number of students it can take, and every student has a strict preference ordering over the courses the student is interested in. A student needs to be assigned to at most one course.

Fixed quota setting: For each data set we select an input signature ρ such that the instance admits a matching which is cumulatively better than ρ. The signature is selected such that it is practically appealing in real-world applications. However, we observe that, neither the rank-maximal matching nor the fair matching are able to meet the requirement in terms of the signature. Table 1 shows our results in this setting. For the data set Real-1 suppose the requirement is to match at least 60% students to their rank-1 courses, at least 85% students to one of their top 3 courses, and the size of the matching must be at least 90% (see row-1 column Input requirement Table 1). The size of the rank-maximal matching (RMM) is 86% thus not meeting the requirement of 90% students being matched. Similarly, the fair matching (FM) matches only 49% students to their rank-1 courses thus failing to meet the 60% requirement of rank-1 matches. See row-1 columns RMM and FM in Table 1. The RMM and FM are obtained by implementing the known algorithms in [10, 11] respectively. The CSM is obtained by an linear linear program formulation (CSM ILP). By choice of the signature, the CSM satisfies the requirement.

Cost-based setting: We begin with an instance of the fixed quota setting. We obtain the cost for every post p by defining a function which takes as input the input quota $q(p)$ and the number of applicants $\#N(p)$ who applied to p. In fact we define three natural cost functions called CF-1, CF-2, CF-3. Using these we derive different instances of the cost-based setting from a single instance of the fixed quota setting. For an instance in the fixed quota setting, we select a signature ρ such that the input fixed quota instance *does not* admit a matching that is cumulatively better than ρ. For each of the three instances derived from this instance, we use our algorithm in Theorem 1.4 and compute a matching with minimum cost. The absolute cost of the matching obtained is not of significance. Since we started with a fixed quota setting instance, we measure the violation in the upper quota for every course, and we report the *max violation* (bold font) and *total violation*. Table 2 gives our results for the real-world data sets. We observe that the maximum violation for any course is around 5% (except for Real-1 with cost function CF-3) and the total violation is around 15%. We believe these are acceptable values in practice in order to meet the signature requirement of the allocation. We briefly describe our cost functions below:

- CF-1: $c(p) = max\{0, (\#N(p) - q(p))\}$
- CF-2: linear cost model that assign costs in non-decreasing order of the ratio $\frac{\#N(p)}{q(p)}$.
- CF-3: $c(p) = \frac{\#N(p).LCM}{q(p)}$, where LCM represents the least common multiple of all the quotas.

We make the following observations from our experiments:

- The cumulatively better signature allows us to express requirements which the input instance may admit and is not captured by the standard measures like rank-maximality or fairness.
- The cost based setting experiments open up the possibility of converting a fixed quota setting instance to a cost-based instance by selecting an appropriate cost function. Thus, although the input instance may not admit a matching with the desired signature, we may be able to achieve the same by violating the quotas by a small value.

ACKNOWLEDGMENTS

This work was partially supported by the grant CRG/2019/004757. We thank Girija Limaye for fruitful discussions that helped us shape our work in the initial phases. We also thank the anonymous reviewers for their valuable feedback.

REFERENCES

[1] David J Abraham, Katarína Cechlárová, David F Manlove, and Kurt Mehlhorn. 2004. Pareto optimality in house allocation problems. In *International Symposium on Algorithms and Computation*. Springer, 3–15.

[2] Ravindra K Ahuja, Thomas L Magnanti, and James B Orlin. 1988. Network flows. (1988).

[3] Noga Alon, Tao Jiang, Zevi Miller, and Dan Pritikin. 2003. Properly colored subgraphs and rainbow subgraphs in edge-colorings with local constraints. *Random Structures & Algorithms* 23, 4 (2003), 409–433. https://doi.org/10.1002/rsa.10102 arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1002/rsa.10102

[4] Katarína Cechlárová, Pavlos Eirinakis, Tamás Fleiner, Dimitrios Magos, David Manlove, Ioannis Mourtos, Eva Oceláková, and Baharak Rastegari. 2016. Pareto optimal matchings in many-to-many markets with ties. *Theory of Computing Systems* 59, 4 (2016), 700–721.

[5] Katarína Cechlárová and Tamás Fleiner. 2017. Pareto optimal matchings with lower quotas. *Mathematical Social Sciences* 88 (2017), 3–10.

[6] Alan Frieze and Michael Krivelevich. 2008. On rainbow trees and cycles. *Electronic Journal of Combinatorics* (2008), 59.

[7] Karthik Gajulapalli, James Liu, Tung Mai, and Vijay V Vazirani. 2019. Stability-Preserving, Time-Efficient Mechanisms for School Choice in Two Rounds. *arXiv e-prints* (2019), arXiv–1904.

[8] M Hecht, Bevan L Cheeseman, Karl B Hoffmann, and Ivo F Sbalzarini. 2017. A quadratic-time algorithm for general multivariate polynomial interpolation. *arXiv preprint arXiv:1710.10846* (2017).

[9] Hadi Hosseini, Vijay Menon, Nisarg Shah, and Sujoy Sikdar. 2021. Necessarily Optimal One-Sided Matchings. In *Proceedings of Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021*. AAAI Press, 5481–5488. https://ojs.aaai.org/index.php/AAAI/article/view/16690

[10] Chien-Chung Huang, Telikepalli Kavitha, Kurt Mehlhorn, and Dimitrios Michail. 2016. Fair matchings and related problems. *Algorithmica* 74, 3 (2016), 1184–1203.

[11] Robert W Irving, Telikepalli Kavitha, Kurt Mehlhorn, Dimitrios Michail, and Katarzyna E Paluch. 2006. Rank-maximal matchings. *ACM Transactions on Algorithms (TALG)* 2, 4 (2006), 602–610.

[12] Fuhito Kojima, Parag A Pathak, and Alvin E Roth. 2013. Matching with couples: Stability and incentives in large markets. *The Quarterly Journal of Economics* 128, 4 (2013), 1585–1632.

[13] Van Bang Le and Florian Pfender. 2014. Complexity results for rainbow matchings. *Theoretical Computer Science* 524 (2014), 27–33. https://doi.org/10.1016/j.tcs.2013.12.013

[14] Girija Limaye and Meghana Nasre. 2021. Stable Matchings with Flexible Quotas. *arXiv preprint arXiv:2101.04425* (2021).

[15] David F. Manlove and Colin T. S. Sng. 2006. Popular Matchings in the Capacitated House Allocation Problem. In *Algorithms – ESA 2006*, Yossi Azar and Thomas Erlebach (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 492–503.

[16] Jérôme Monnot. 2005. The labeled perfect matching in bipartite graphs. *Inform. Process. Lett.* 96, 3 (2005), 81–88.

[17] Ketan Mulmuley, Umesh V. Vazirani, and Vijay V. Vazirani. 1987. Matching is as Easy as Matrix Inversion. In *Proceedings of the Nineteenth Annual ACM Symposium on Theory of Computing (STOC '87)*. 345–354.

[18] Meghana Nasre, Prajakta Nimbhorkar, and Nada Pulath. 2019. Classified Rank-Maximal Matchings and Popular Matchings - Algorithms and Hardness. In *Graph-Theoretic Concepts in Computer Science - 45th International Workshop, WG 2019, Vall de Núria, Spain, June 19-21, 2019, Revised Papers (Lecture Notes in Computer Science, Vol. 11789)*, Ignasi Sau and Dimitrios M. Thilikos (Eds.). Springer, 244–257. https://doi.org/10.1007/978-3-030-30786-8_19

[19] Katarzyna Paluch. 2013. Capacitated rank-maximal matchings. In *International Conference on Algorithms and Complexity*. Springer, 324–335.

[20] Ignacio Rios, Tomas Larroucau, Giorgiogiulio Parra, and Roberto Cominetti. 2021. Improving the Chilean college admissions system. *Operations Research* (2021).

[21] Scottish Foundation Allocation Scheme. 2011 - 2022. *Matching in Practice,European network for research on matching practices in education and related markets @online.* https://www.matching-in-practice.eu/the-scottish-foundation-allocation-scheme-sfas/