Problem size	(a) Training Time Per Epoch (Mins) (↓) NELSON	Gibbs	CMSGen
10	**0.53**	9.85	0.69
20	**0.53**	80.12	1.93
30	**0.72**	256.38	3.65
40	**0.93**	777.01	5.99
50	**1.17**	T.O.	9.08
	(b) Validness of Orientations (%) (↑)		
7	**100**	50.16	**100**
8	**100**	64.63	**100**
9	**100**	47.20	**100**
10	**100**	62.60	**100**
11	**100**	84.95	**100**
	(c) Approximation Error of $\nabla \log Z_{\mathcal{C}}(\theta)$ (↓)		
5	**0.01**	0.09	0.21
7	**0.05**	0.08	2.37
9	**0.03**	0.11	2.37
11	**0.04**	0.17	8.62
13	**0.05**	0.28	11.27
	(d) MAP@10 (%) (↑)		
10	61.14	60.01	**64.56**
20	**55.26**	55.20	47.79
30	**100.00**	96.29	**100.00**
40	**40.01**	39.88	38.90
50	**46.12**	T.O.	42.11

Table 3: Sample efficiency and learning performance of the sink-free orientation task. NELSON is the most efficient (see Training Time Per Epoch) and always generates valid assignments (see Validness), has the smallest error approximating gradients, and has the best learning performance (see MAP@10) among all baselines.

hence NELSON sampler's performance is guaranteed by Theorem 1. In Table 3(c), NELSON attains the smallest approximation error for the gradient (in Eq. 4) compared to baselines. Finally, NELSON learns a higher MAP@10 than CMSGen. The Gibbs-based approach times out for problem sizes larger than 40. In summary, our NELSON is the best-performing algorithm for this task.

Learn Vehicle Delivery Routes

Task Definition & Dataset Given a set of locations to visit, the task is to generate a sequence to visit these locations in which each location is visited once and only once and the sequence closely resembles the trend presented in the training data. The training data are such routes collected in the past. The dataset is constructed from TSPLIB, which consists of 29 cities in Bavaria, Germany. The constraints for this problem do not satisfy Condition 1. We still apply the proposed method to evaluate if the NELSON algorithm can handle those general hard constraints.

In Fig. 3, we see NELSON can obtain samples of this delivery problem efficiently. We measure the number of resamples taken as well as the corresponding time used by the NELSON method. NELSON takes roughly 50 times of resamples with an average time of 0.3 seconds to draw a batch (the batch size is 100) of valid visiting sequences.

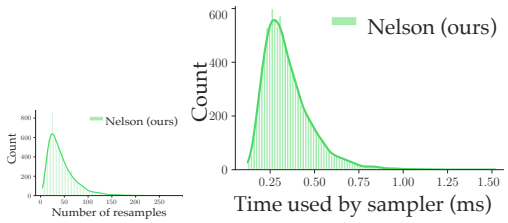

Figure 3: Frequency histograms for the number of resample and the total time of NELSON method for uniformly sampling visiting paths for vehicle routing problem.

Conclusion

In this research, we present NELSON, which embeds a sampler based on Lovász Local Lemma into the contrastive divergence learning of Markov random fields. The embedding is fully differentiable. This approach allows us to learn generative models over constrained domains, which presents significant challenges to other state-of-the-art models. We also give sound proofs of the performance of the LLL-based sampler. Experimental results on several real-world domains reveal that NELSON learns to generate 100% valid structures, while baselines either time out or cannot generate valid structures. NELSON also outperforms other approaches in the running times and in various learning metrics.

Acknowledgments

We thank all the reviewers for their constructive comments. This research was supported by NSF grants IIS-1850243, CCF-1918327. Illum a vitae totam qui eos dolores molestias soluta corrupti, minus illo quaerat natus dolore tempora ratione dolorem, consequatur totam vel id hic a ratione sequi aliquam, eveniet deserunt quibusdam reiciendis accusantium aut id fugit?Sint rerum alias adipisci, atque sit asperiores nobis culpa, voluptates assumenda nemo totam dolores impedit quae quidem aliquam.Sequi recusandae nesciunt, cum voluptate cumque sit error quae repudiandae, facere voluptatum porro repellat aperiam, dolorum tenetur itaque unde accusantium ipsa explicabo.Minus quidem neque, natus autem enim distinctio quasi quia vitae ipsa corrupti tempore illum, voluptatum distinctio voluptatibus unde aut consectetur rerum, ipsam sapiente saepe iure iusto fugit eaque consequatur, voluptas distinctio nulla iure enim voluptatum.Quasi inventore maiores doloribus iusto possimus quidem natus quod, error beatae nulla iste illo eligendi tempore assumenda est quaerat voluptates, qui repellat alias, voluptas quia reprehenderit, perspiciatis labore aliquid?Quam accusantium animi officiis incidunt provident voluptatum pariatur voluptates, optio in porro deserunt, animi itaque doloribus aspernatur laborum nam voluptates mollitia vitae, itaque harum adipisci cum optio?Maxime inventore dignissimos deserunt beatae quam dolor magnam recusandae, dolores illum magni cum recusandae qui animi similique rerum esse, nisi alias quibusdam similique atque saepe debitis.Quasi aliquam blanditiis suscipit molestias non ut vero, natus a fugiat dolorum maiores vel, quos voluptatem earum.Soluta odit a impedit labore quisquam consequatur est nulla blanditiis, laudantium modi eos sed saepe repellen-

dus maiores, architecto quasi veritatis veniam illum consequuntur laboriosam