Learning-Augmented Streaming Algorithms for Approximating Boolean Max-CSPs

Contact:
Yinhao Dong: yhdong@mail.ustc.edu.cn
Xiaoyang Xu: xiaoyangxu@cs.ucsb.edu

Yinhao Dong* Pan Peng* Ali Vakilian† Santhoshini Velusamy‡ Xiaoyang Xu§

*University of Science and Technology of China †Virginia Tech ‡University of Waterloo §University of California, Santa Barbara

Boolean Maximum Constraint Satisfaction Problems (Max-CSPs)

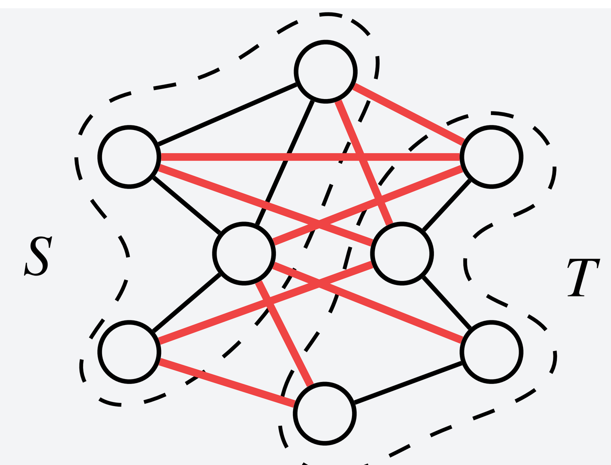
- **Variables:** $x_1, x_2, \dots, x_n$ taking values in $\{-1, 1\}$.
- **Constraints:** $(f, \mathbf{j})$ where $f : \{-1,1\}^k \to \{0,1\}$ and $\mathbf{j} = (j_1, \dots, j_k) \in [n]^k$.
  Example: $f(a,b) = \neg a \wedge b$ and $\mathbf{j} = (3,8)$, read as $\neg x_3 \wedge x_8$.
- **Value:** Given a Boolean Max-CSP instance $\Psi = \{(f_i, \mathbf{j}_i)\}_{i\in[m]}$, its **value** is the largest number of satisfied constraints over all assignments. Formally,

$$\mathsf{val}_\Psi := \max_{\boldsymbol{\sigma}\in\{-1,1\}^n} |\{(f,\mathbf{j}) \in \Psi : f(\sigma_{j_1}, \dots, \sigma_{j_k}) = 1\}| \in [0, m].$$

- α**-approx** $(0 < \alpha \le 1)$**:** $\alpha \cdot \mathsf{OPT} \le \mathsf{ALG} \le \mathsf{OPT}$, where $\mathsf{OPT} = \mathsf{val}_\Psi$.

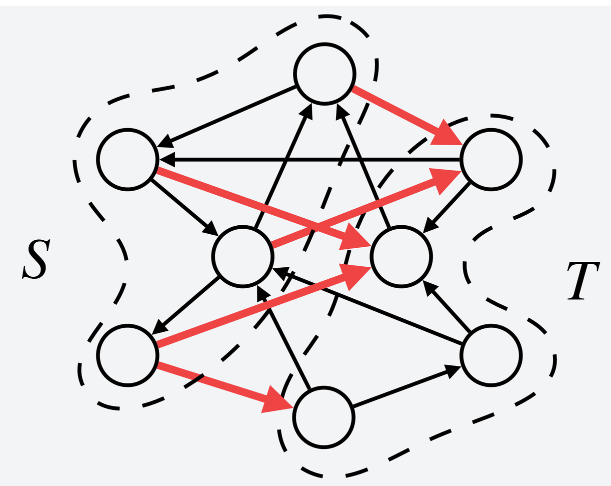
Max-CUT as a CSP

- **Variables:** $x_i = 1 \Leftrightarrow i \in S$, $x_i = -1 \Leftrightarrow i \in T$
- **Constraints:** $(i,j) \in E \Leftrightarrow x_i \oplus x_j \in \Psi$
- **Value:** val_Ψ = max cut value



Max-DICUT as a CSP

- **Variables:** $x_i = 1 \Leftrightarrow i \in S$, $x_i = -1 \Leftrightarrow i \in T$
- **Constraints:** $(i,j) \in E \Leftrightarrow x_i \wedge \neg x_j \in \Psi$
- **Value:** val_Ψ = max directed cut (dicut) value



CSPs in the Streaming Model

- Ψ is given as input to the algorithm as a stream of constraint insertions/deletions.
  - Insertion-Only Stream: consists of constraint insertion only
  - Dynamic Stream: has both insertions and deletions
- **Obs:** Storing the assignment requires $\Omega(n)$ bits of space!
- **Goal:** In one pass, using small space ($o(n)$ bits), estimate the value of Ψ.

Max-CUT: $1/2$-approx, $O(\log n)$ space (trivial). LB: $(1/2+\eta)$-approx, $\Omega_\eta(n)$ space [KK19]

Max-DICUT: $(1/2-\eta)$-approx, $n^{1-\Omega_\eta(1)}$ space [ABFS26]. LB: $(4/9+\eta)$-approx, $\Omega_\eta(\sqrt{n})$ space [CGV20]; $(1/2+\eta)$-approx, $\Omega_\eta(n)$ space [KK19] (Space measured in *bits* in this block)

Learning-Augmented Algorithms with Noisy Predictions [CdG+24, GMM25]

- $\boldsymbol{\sigma}^* \in \{-1,1\}^n$ denotes some fixed but unknown optimal assignment.
- Algorithm has oracle access to a prediction vector $\mathbf{Y} \in \{-1,1\}^n$.
- ε**-accurate:** Y_i is independently correct w.p. slightly better than random guessing:

$$\forall i \in [n], \qquad \Pr[Y_i = \sigma_i^*] = 1/2 + \varepsilon \qquad (0 < \varepsilon \le 1/2).$$

Our Results

Theorem: For any $\eta > 0$, with ε-accurate predictions, there is a single-pass streaming algo that achieves a $(1-\eta)$-approx to the value of any Boolean Max-kCSP w.p. $2/3$.
- $k = 2$: $\mathrm{poly}(1/\varepsilon, 1/\eta)$ space, insertion-only; $\mathrm{poly}(1/\varepsilon, 1/\eta, \log n)$ space, dynamic.
- $k \ge 3$: space has exponential dependence on $k, 1/\varepsilon, 1/\eta$. (Space measured in *words*)

Remark: Significantly improves upon [DPV25]: from Max-CUT to any Boolean Max-CSP, and from $(1/2+\Omega(\varepsilon^2))$ to $(1-\eta)$-approx, with extra $\mathrm{poly}(1/\eta)$ space dependence.

Recall: Streaming Max-CUT with Predictions [DPV25]

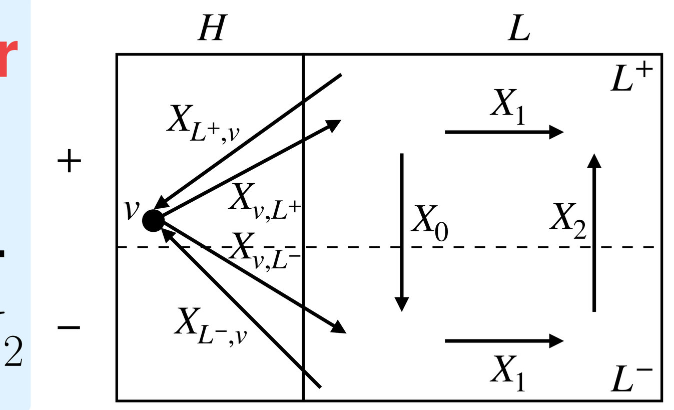
$(1/2+\Omega(\varepsilon^2))$-approx, $\mathrm{poly}(1/\varepsilon)$ space (insertion), $\mathrm{poly}(1/\varepsilon, \log n)$ space (dynamic)

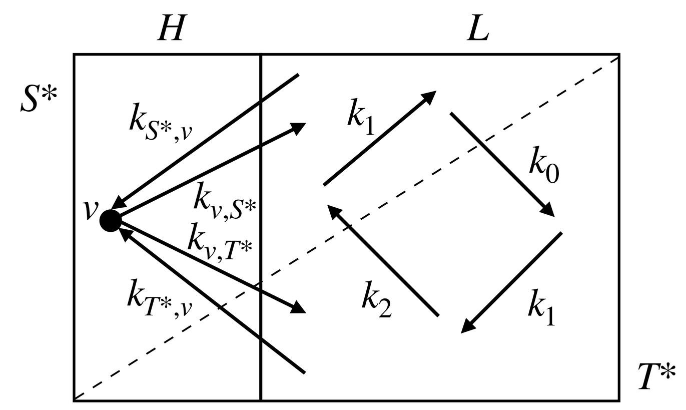
- **Obs:** If max degree $\Delta = O(\varepsilon^2 m)$, simply **follow the predictions**!
  - Count edges (i,j) with $Y_i \ne Y_j$. Bounded variance gives concentration.
- **General graphs:** divide vertices into H (high-deg) and L (low-deg); return max of:
  - Estimate 1: follow predictions on L to get (L^+, L^-); assign vertices in H greedily.
  - Estimate 2: cut (H, L).
- **Streaming:** sampling + sketching techniques to estimate H, L and incident edges.

Streaming Max-DICUT with Predictions

$(1-\eta)$-approx, $\mathrm{poly}(1/\varepsilon, 1/\eta)$ space (insertion), $\mathrm{poly}(1/\varepsilon, 1/\eta, \log n)$ space (dynamic)

- Simply following predictions is not enough; predictions actually give us more!
- **Obs:** $(\mathbb{E}[X_0], \mathbb{E}[X_1], \mathbb{E}[X_2])$ is a **linear transformation** of (k_0, k_1, k_2).
- **Decoding:** Solve the linear system.
  $\widetilde{k}_0 := f_0(\varepsilon) \cdot X_0 + f_1(\varepsilon) \cdot X_1 + f_2(\varepsilon) \cdot X_2$





- Similar decoding for the contribution of high-deg vertices H: obtain $\widetilde{k}_{S^*,v}$ and $\widetilde{k}_{v,T^*}$.
- Greedily assign high-deg vertices to get final estimator $\widetilde{k}_{0(L)} + \sum_{v\in H} \max\{\widetilde{k}_{S^*,v}, \widetilde{k}_{v,T^*}\}$.
- **Extension to Max-kCSPs:** reduce to Max-kAND; use a similar decoding method.

Open Problems

- Can a similar theorem be proved in other prediction models?
- Generalize the prediction model to q-ary alphabets? (Currently Boolean, $q = 2$)

References

[ABFS26] Amir Azarmehr, Soheil Behnezhad, Shane Ferrante, and Mohammad Saneian. Half-Approximating Maximum Dicut in the Streaming Setting. In *STOC'26*.
[CdG+24] Vincent Cohen-Addad, Tommaso d'Orsi, Anupam Gupta, Euiwoong Lee, and Debmalya Panigrahi. Learning-Augmented Approximation Algorithms for Maximum Cut and Related Problems. In *NeurIPS'24*.
[CGV20] Chi-Ning Chou, Alexander Golovnev, and Santhoshini Velusamy. Optimal Streaming Approximations for all Boolean Max-2CSPs and Max-kSAT. In *FOCS'20*.
[DPV25] Yinhao Dong, Pan Peng, and Ali Vakilian. Learning-Augmented Streaming Algorithms for Approximating MAX-CUT. In *ITCS'25*.
[GMM25] Suprovat Ghoshal, Konstantin Markarychev, and Yury Markarychev. Constraint Satisfaction Problems with Advice. In *SODA'25*.
[KK19] Michael Kapralov and Dmitry Krachun. An Optimal Space Lower Bound for Approximating MAX-CUT. In *STOC'19*.