### What are CSPs?

- **Constraint Satisfaction Problems (CSPs)** are an important class of optimization problems.
- Every CSP is characterized by a finite family \( \mathcal{F} \) of constraint functions.
- An instance of CSP(\( \mathcal{F} \)) consists of a finite set of variables, and constraints from \( \mathcal{F} \) applied on the variables.
- **Value** of a CSP instance: Maximum fraction of constraints that can be satisfied by any assignment to the variables.
- Examples include Max-CUT, Max-DICUT, Unique Games, Max-k-SAT, Max-q-Coloring.

### CSPs in streaming settings

- **Insertion-only setting**: Constraints appear one-by-one in a stream.
  - Algorithm for CSP: Using only polylog storage space, compute the value of the CSP instance.

- **Dynamic setting**: Constraints are either added or deleted one-by-one in a stream.

- **NP-Hard for most CSPs!**

- **Approximation**-algorithm for CSP: Using only polylog storage space, compute the value of the CSP instance approximately.

### Objectives

To study the fine-grained approximability of every CSP in streaming settings, *i.e.*, answer the following problem for all \( 1 > \gamma > \beta > 0 \):

#### \((\gamma, \beta)-distinguishability problem for CSP(\mathcal{F})\)

Does there exist a polylog space streaming algorithm that can distinguish every instance of CSP(\( \mathcal{F} \)) with value **at least** \( \gamma \) from every instance of CSP(\( \mathcal{F} \)) with value **at most** \( \beta \), with probability at least 0.9?

### Results

- **Dichotomy theorem in the dynamic setting**: For every pair \( 1 > \gamma > \beta > 0 \), we define two closed, convex, and bounded sets \( K^\gamma_\mathcal{F} \) and \( K^\beta_\mathcal{F} \) and prove that
  - **If the sets do not intersect**, yes, there exists such a dynamic algorithm!
  - **If the sets intersect**, no, there is no such dynamic algorithm!

- **Approximation resistance in the Insertion-only setting**: If \( \mathcal{F} \) “weakly supports one-wise independence,” then there is no “non-trivial” insertion-only streaming approximation algorithm for CSP(\( \mathcal{F} \)). Examples include Max-CUT, Unique Games, Max-q-Coloring.

### Proof techniques

- Dynamic streaming algorithm when \( K^\gamma_\mathcal{F} \) and \( K^\beta_\mathcal{F} \) do not intersect:
  - Consider the normal of the separating hyperplane: \( \lambda = (\lambda_1, \lambda_2, \ldots, \lambda_k) \).
  - Compute the \( \ell_1, \infty \)-norm of a “\( \lambda \)-bias matrix” of the instance in the dynamic setting.
  - The value of this norm distinguishes instances with value **at least** \( \gamma \) from instances with value **at most** \( \beta \).

- **\( \Omega(\sqrt{n}) \) space lower bound when \( K^\gamma_\mathcal{F} \) and \( K^\beta_\mathcal{F} \) intersect**:
  - Follows from the hardness of a one-way communication game.

### Future Directions

- A dichotomy theorem for every CSP in the insertion-only streaming setting.
- A dichotomy theorem for every CSP in the insertion-only streaming setting where the constraints are randomly ordered.

Based on Approximability of all finite CSPs in the dynamic streaming setting, Chi-Ning Chou, Alexander Golovnev, Madhu Sudan, and Santhoshini Velusamy. To appear in FOCS 2021.