Algorithmic Regularity Lemmas and Applications

László Miklós Lovász
Massachusetts Institute of Technology

Proving and Using Pseudorandomness
Simons Institute for the Theory of Computing

Joint work with Jacob Fox and Yufei Zhao

March 8, 2017

Szemerédi's Regularity Lemma

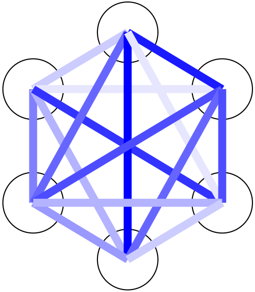
Szemerédi's regularity lemma

Roughly speaking, in any graph, the vertices can be partitioned into a bounded number of parts, such that the graph is "random-like" between almost all pairs of parts.

Szemerédi's Regularity Lemma

Szemerédi's regularity lemma

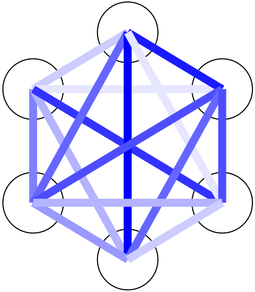
Roughly speaking, in any graph, the vertices can be partitioned into a bounded number of parts, such that the graph is "random-like" between almost all pairs of parts.



Szemerédi's Regularity Lemma

Szemerédi's regularity lemma

Roughly speaking, in any graph, the vertices can be partitioned into a bounded number of parts, such that the graph is "random-like" between almost all pairs of parts.

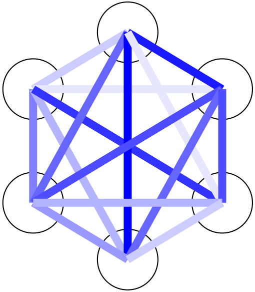


- Very important tool in graph theory

Szemerédi's Regularity Lemma

Szemerédi's regularity lemma

Roughly speaking, in any graph, the vertices can be partitioned into a bounded number of parts, such that the graph is "random-like" between almost all pairs of parts.



- Very important tool in graph theory
- Gives a rough structural result for all graphs

Regularity of Sets

Let X and Y be two sets of vertices in a graph G.

Regularity of Sets

Let X and Y be two sets of vertices in a graph G.
$e(X, Y)$: number of pairs of vertices in $X \times Y$ that have an edge between them.

Regularity of Sets

Let X and Y be two sets of vertices in a graph G.
$e(X, Y)$: number of pairs of vertices in $X \times Y$ that have an edge between them.
$d(X, Y) = \frac{e(X,Y)}{|X||Y|}$.

Regularity of Sets

Let X and Y be two sets of vertices in a graph G.
$e(X, Y)$: number of pairs of vertices in $X \times Y$ that have an edge between them.
$d(X, Y) = \frac{e(X,Y)}{|X||Y|}$.

Definition

Given a graph G and two sets of vertices X and Y, we say the pair (X, Y) is ϵ-*regular* if for any $X' \subset X$ with $|X'| \geq \epsilon|X|$, $Y' \subset Y$ with $|Y'| \geq \epsilon|Y|$, we have

$$\left| d(X', Y') - d(X, Y) \right| \leq \epsilon.$$

Regularity of Sets

Let X and Y be two sets of vertices in a graph G.
$e(X, Y)$: number of pairs of vertices in $X \times Y$ that have an edge between them.
$d(X, Y) = \frac{e(X,Y)}{|X||Y|}$.

Definition

Given a graph G and two sets of vertices X and Y, we say the pair (X, Y) is ϵ-*regular* if for any $X' \subset X$ with $|X'| \geq \epsilon|X|$, $Y' \subset Y$ with $|Y'| \geq \epsilon|Y|$, we have

$$\Big| d(X', Y') - d(X, Y) \Big| \leq \epsilon.$$

Roughly says graph between X and Y is "random-like".

Szemerédi's Regularity Lemma

Definition

Given a partition $\mathcal{P}$ of the set of vertices V, we say it is *equitable* if the size of any two parts differs by at most one.

Szemerédi's Regularity Lemma

Definition

Given a partition $\mathcal{P}$ of the set of vertices V, we say it is *equitable* if the size of any two parts differs by at most one.

Definition

Given an equitable partition $\mathcal{P}$ of the set of vertices V, it is ϵ-*regular* if all but $\epsilon|\mathcal{P}|^2$ pairs are ϵ-regular.

Szemerédi's Regularity Lemma

Definition

Given a partition $\mathcal{P}$ of the set of vertices V, we say it is *equitable* if the size of any two parts differs by at most one.

Definition

Given an equitable partition $\mathcal{P}$ of the set of vertices V, it is *ϵ-regular* if all but $\epsilon|\mathcal{P}|^2$ pairs are ϵ-regular.

Szemerédi's regularity lemma

For every $\epsilon > 0$, there is an $M(\epsilon)$ such that for any graph $G = (V, E)$, there is an equitable, ϵ-regular partition of the vertices into at most $M(\epsilon)$ parts.

Regularity Lemma Proof Sketch

Definition

For a vertex partition $\mathcal{P} : V = V_1 \cup V_2 \cup \ldots \cup V_k$, define the *mean square density*:

$$q(\mathcal{P}) = \sum_{i,j} p_i p_j d(V_i, V_j)^2,$$

where $p_i = \frac{|V_i|}{|V|}$.

Regularity Lemma Proof Sketch

Definition

For a vertex partition $\mathcal{P} : V = V_1 \cup V_2 \cup \ldots \cup V_k$, define the *mean square density*:

$$q(\mathcal{P}) = \sum_{i,j} p_i p_j d(V_i, V_j)^2,$$

where $p_i = \frac{|V_i|}{|V|}$.

- Between 0 and 1.

Regularity Lemma Proof Sketch

Definition

For a vertex partition $\mathcal{P} : V = V_1 \cup V_2 \cup \ldots \cup V_k$, define the *mean square density*:

$$q(\mathcal{P}) = \sum_{i,j} p_i p_j d(V_i, V_j)^2,$$

where $p_i = \frac{|V_i|}{|V|}$.

- Between 0 and 1.
- If we refine the partition, it cannot decrease.

Regularity Lemma Proof Sketch

Definition

For a vertex partition $\mathcal{P} : V = V_1 \cup V_2 \cup \ldots \cup V_k$, define the *mean square density*:

$$q(\mathcal{P}) = \sum_{i,j} p_i p_j d(V_i, V_j)^2,$$

where $p_i = \frac{|V_i|}{|V|}$.

- Between 0 and 1.
- If we refine the partition, it cannot decrease.
- If a partition into k parts is not ϵ-regular, can divide each piece into at most 2^{k+1} parts, according to worst case sets, to get an increase of ϵ^5 (then make equitable).

Algorithmic Regularity

Alon-Duke-Lefmann-Rödl-Yuster (1994)

If a pair (X, Y) is not ϵ-regular, find a pair of subsets that show they are not $\epsilon^4/16$-regular, in time $O_\epsilon(n^{\omega+o(1)})$. Implies tower height at most $T(\epsilon^{-20})$. $(\omega < 2.373)$

Algorithmic Regularity

Alon-Duke-Lefmann-Rödl-Yuster (1994)

If a pair (X, Y) is not ϵ-regular, find a pair of subsets that show they are not $\epsilon^4/16$-regular, in time $O_\epsilon(n^{\omega+o(1)})$. Implies tower height at most $T(\epsilon^{-20})$. ($\omega < 2.373$)

Frieze-Kannan (1999)

Regularity lemma algorithmically, through a spectral approach.

Algorithmic Regularity

Alon-Duke-Lefmann-Rödl-Yuster (1994)

If a pair (X, Y) is not ϵ-regular, find a pair of subsets that show they are not $\epsilon^4/16$-regular, in time $O_\epsilon(n^{\omega+o(1)})$. Implies tower height at most $T(\epsilon^{-20})$. ($\omega < 2.373$)

Frieze-Kannan (1999)

Regularity lemma algorithmically, through a spectral approach.

Kohayakawa-Rödl-Thoma (2003)

Faster algorithmic lemma, running time $O_\epsilon(n^2)$.

Algorithmic Regularity

Alon-Duke-Lefmann-Rödl-Yuster (1994)

If a pair (X, Y) is not ϵ-regular, find a pair of subsets that show they are not $\epsilon^4/16$-regular, in time $O_\epsilon(n^{\omega+o(1)})$. Implies tower height at most $T(\epsilon^{-20})$. ($\omega < 2.373$)

Frieze-Kannan (1999)

Regularity lemma algorithmically, through a spectral approach.

Kohayakawa-Rödl-Thoma (2003)

Faster algorithmic lemma, running time $O_\epsilon(n^2)$.

Alon-Naor (2006)

Polynomial-time algorithm, at most $T(O(\epsilon^{-7}))$ parts.

Algorithmic Regularity

Even though only a tower-type number is guaranteed, most graphs have a much smaller regularity partition. Previous algorithms may not find it.

Algorithmic Regularity

Even though only a tower-type number is guaranteed, most graphs have a much smaller regularity partition. Previous algorithms may not find it.

Fischer-Matsliah-Shapira (2010)

Randomized algorithm which runs in time $O_{\epsilon,k}(1)$, if there is an ϵ-regular partition with k parts, finds 2ϵ-regular partition with at most k parts.

Algorithmic Regularity

Even though only a tower-type number is guaranteed, most graphs have a much smaller regularity partition. Previous algorithms may not find it.

Fischer-Matsliah-Shapira (2010)

Randomized algorithm which runs in time $O_{\epsilon,k}(1)$, if there is an ϵ-regular partition with k parts, finds 2ϵ-regular partition with at most k parts.

Folklore/Tao blog post (2010)

Randomized algorithm in time $O_{\epsilon}(1)$, ϵ-regular partition.

Finding a regular partition

Fox-L.-Zhao

An $O_{\epsilon,\alpha}(n^2)$-time *deterministic* algorithm which, given ϵ, α, k and a graph G on n vertices that has an ϵ-regular partition with k parts, gives a $(1+\alpha)\epsilon$-regular partition into k parts.

Finding a regular partition

Fox-L.-Zhao

An $O_{\epsilon,\alpha}(n^2)$-time *deterministic* algorithm which, given ϵ, α, k and a graph G on n vertices that has an ϵ-regular partition with k parts, gives a $(1+\alpha)\epsilon$-regular partition into k parts.

An intermediate result is testing regularity.

Finding a regular partition

Fox-L.-Zhao

An $O_{\epsilon,\alpha}(n^2)$-time *deterministic* algorithm which, given ϵ, α, k and a graph G on n vertices that has an ϵ-regular partition with k parts, gives a $(1+\alpha)\epsilon$-regular partition into k parts.

An intermediate result is testing regularity.

Fox-L.-Zhao

An $O_{\epsilon,\alpha,k}(n^2)$-time *deterministic* algorithm which, given ϵ, α and a graph G between sets X, Y of size n, outputs either

- that (X, Y) are ϵ-regular.
- a pair of subsets $U \subset X$, $W \subset Y$ that show that (X, Y) are not $(1-\alpha)\epsilon$-regular, i.e. $|U| \geq (1-\alpha)\epsilon|X|$, $|W| \geq (1-\alpha)\epsilon|Y|$, and $|d(X, Y) - d(U, W)| > (1-\alpha)\epsilon$.

Frieze-Kannan (weak) regularity lemma

Definition

Given a partition $\mathcal{P} = \{V_1, V_2, ..., V_k\}$ of the set of vertices V, it is *Frieze-Kannan ϵ-regular* (FK-ϵ-regular) if for any pair of sets $S, T \subseteq V$, we have

$$\left| e(S,T) - \sum_{i,j=1}^{k} d(V_i, V_j)|S \cap V_i||T \cap V_j| \right| \leq \epsilon |V|^2$$

Frieze-Kannan (weak) regularity lemma

Definition

Given a partition $\mathcal{P} = \{V_1, V_2, ..., V_k\}$ of the set of vertices V, it is *Frieze-Kannan ϵ-regular* (FK-ϵ-regular) if for any pair of sets $S, T \subseteq V$, we have

$$\left| e(S,T) - \sum_{i,j=1}^{k} d(V_i, V_j)|S \cap V_i||T \cap V_j| \right| \leq \epsilon |V|^2$$

Frieze-Kannan regularity lemma

Let $\epsilon > 0$. Every graph has a Frieze-Kannan ϵ-regular partition with at most $2^{2/\epsilon^2}$ parts.

Frieze-Kannan (weak) regularity lemma

Definition

Given a partition $\mathcal{P} = \{V_1, V_2, ..., V_k\}$ of the set of vertices V, it is *Frieze-Kannan ϵ-regular* (FK-ϵ-regular) if for any pair of sets $S, T \subseteq V$, we have

$$\left| e(S,T) - \sum_{i,j=1}^{k} d(V_i, V_j)|S \cap V_i||T \cap V_j| \right| \leq \epsilon |V|^2$$

Frieze-Kannan regularity lemma

Let $\epsilon > 0$. Every graph has a Frieze-Kannan ϵ-regular partition with at most $2^{2/\epsilon^2}$ parts.

Proof similar: refine by worst case sets, mean square density increases by ϵ^2.

Counting Lemma

Definition

Given two (possibly weighted) graphs G_1 and G_2 on the same vertex set V, we define their *cut distance*

$$d_{\square}(G_1, G_2) = \frac{1}{|V|^2} \max_{S,T \subseteq V} |e_{G_1}(S,T) - e_{G_2}(S,T)|.$$

Counting Lemma

Definition

Given two (possibly weighted) graphs G_1 and G_2 on the same vertex set V, we define their *cut distance*

$$d_\square(G_1, G_2) = \frac{1}{|V|^2} \max_{S,T \subseteq V} |e_{G_1}(S,T) - e_{G_2}(S,T)|.$$

Partition $\mathcal{P}$ is FK-ϵ-regular if and only if $d_\square(G, G_{\mathcal{P}}) \leq \epsilon$.

Counting Lemma

Definition

Given two (possibly weighted) graphs G_1 and G_2 on the same vertex set V, we define their *cut distance*

$$d_\square(G_1, G_2) = \frac{1}{|V|^2} \max_{S,T \subseteq V} |e_{G_1}(S, T) - e_{G_2}(S, T)|.$$

Partition $\mathcal{P}$ is FK-ϵ-regular if and only if $d_\square(G, G_\mathcal{P}) \leq \epsilon$.

Counting lemma

Given two graphs G_1 and G_2 on the same vertex set, for any graph H on k vertices, we have

$$|\hom(H, G_1) - \hom(H, G_2)| \leq e(H) d_\square(G_1, G_2) n^k.$$

Algorithmic Frieze-Kannan

Dellamonica-Kalyanasundaram-Martin-Rödl-Shapira

Give a deterministic algorithm which finds a Frieze-Kannan ϵ-regular partition

- in time $\epsilon^{-6} n^{\omega + o(1)}$ into at most $2^{O(\epsilon^{-7})}$ parts (2012)
- in time $O(2^{2^{\epsilon^{-O(1)}}} n^2)$ into at most $2^{\epsilon^{-O(1)}}$ parts (2015)

Algorithmic Frieze-Kannan

Dellamonica-Kalyanasundaram-Martin-Rödl-Shapira

Give a deterministic algorithm which finds a Frieze-Kannan ϵ-regular partition

- in time $\epsilon^{-6} n^{\omega+o(1)}$ into at most $2^{O(\epsilon^{-7})}$ parts (2012)
- in time $O(2^{2^{\epsilon^{-O(1)}}} n^2)$ into at most $2^{\epsilon^{-O(1)}}$ parts (2015)

Dellamonica-Kalyanasundaram-Martin-Rödl-Shapira

There is an $n^{\omega+o(1)}$-time algorithm which, given $\epsilon > 0$, an n-vertex graph G and a partition $\mathcal{P}$ of $V(G)$, either:

1. Correctly states that $\mathcal{P}$ is FK-ϵ-regular;
2. Finds sets S, T which witness the fact that $\mathcal{P}$ is not FK-$\epsilon^3/1000$-regular.

Algorithmic Frieze-Kannan

Corollary

There is an $\epsilon^{-O(1)} n^{\omega+o(1)}$-time algorithm which, given $\epsilon > 0$, an n-vertex graph G, outputs $t \leq \epsilon^{-O(1)}$, subsets $S_1, S_2, \ldots, S_t, T_1, T_2, \ldots, T_t \subset V(G)$ and real numbers $c_1, c_2, \ldots, c_t$ such that

$$d_{\square}(G, d(G)K_{V(G)} + c_1 K_{S_1,T_1} + c_2 K_{S_2,T_2} + \ldots + c_t K_{S_t,T_t}) \leq \epsilon.$$

Algorithmic Frieze-Kannan

Corollary

There is an $\epsilon^{-O(1)} n^{\omega + o(1)}$-time algorithm which, given $\epsilon > 0$, an n-vertex graph G, outputs $t \leq \epsilon^{-O(1)}$, subsets $S_1, S_2, ..., S_t, T_1, T_2, ..., T_t \subset V(G)$ and real numbers $c_1, c_2, ..., c_t$ such that

$$d_\square(G, d(G)K_{V(G)} + c_1 K_{S_1,T_1} + c_2 K_{S_2,T_2} + ... + c_t K_{S_t,T_t}) \leq \epsilon.$$

Can also do in time $2^{2^{\epsilon^{-O(1)}}} n^2$.

Counting subgraphs

Algorithmic problem

Count the number of copies of a graph H in a graph G on n vertices.

Counting subgraphs

Algorithmic problem

Count the number of copies of a graph H in a graph G on n vertices.

Special case: is there a single copy?

Counting subgraphs

Algorithmic problem

Count the number of copies of a graph H in a graph G on n vertices.

Special case: is there a single copy?

Even for K_k, Zuckerman showed NP-hard to approximate the size of the largest clique within a factor $n^{1-\epsilon}$, building on an earlier result of Hastad.

Counting subgraphs

Algorithmic problem

Count the number of copies of a graph H in a graph G on n vertices.

Special case: is there a single copy?

Even for K_k, Zuckerman showed NP-hard to approximate the size of the largest clique within a factor $n^{1-\epsilon}$, building on an earlier result of Hastad.

How fast can we approximate the count within an additive $\epsilon n^{|V(H)|}$?

Counting subgraphs

Algorithmic problem

Count the number of copies of a graph H on k vertices in a graph on n vertices, up to an error of at most ϵn^k.

Counting subgraphs

Algorithmic problem

Count the number of copies of a graph H on k vertices in a graph on n vertices, up to an error of at most ϵn^k.

A simple randomized algorithm gives 99% certainty:

Counting subgraphs

Algorithmic problem

Count the number of copies of a graph H on k vertices in a graph on n vertices, up to an error of at most ϵn^k.

A simple randomized algorithm gives 99% certainty:

Sample $10/\epsilon^2$ random k-sets of vertices.

Counting subgraphs

Algorithmic problem

Count the number of copies of a graph H on k vertices in a graph on n vertices, up to an error of at most ϵn^k.

A simple randomized algorithm gives 99% certainty:

Sample $10/\epsilon^2$ random k-sets of vertices.

What about deterministic algorithms?

Counting subgraphs

Algorithmic problem

Count the number of copies of a graph H on k vertices in a graph on n vertices, up to an error of at most ϵn^k.

A simple randomized algorithm gives 99% certainty:

Sample $10/\epsilon^2$ random k-sets of vertices.

What about deterministic algorithms?

Can use algorithmic regularity lemma.

Counting subgraphs

Algorithmic problem

Count the number of copies of a graph H on k vertices in a graph G on n vertices, up to an error of at most ϵn^k.

Counting subgraphs

Algorithmic problem

Count the number of copies of a graph H on k vertices in a graph G on n vertices, up to an error of at most ϵn^k.

Duke-Lefmann-Rödl (1996)

Can be done in time $2^{(k/\epsilon)^{O(1)}} n^{\omega+o(1)}$.

Counting subgraphs

Algorithmic problem

Count the number of copies of a graph H on k vertices in a graph G on n vertices, up to an error of at most ϵn^k.

Duke-Lefmann-Rödl (1996)

Can be done in time $2^{(k/\epsilon)^{O(1)}} n^{\omega+o(1)}$.

Fox-L.-Zhao (2017)

Can be done in time $O_H(\epsilon^{-O(e(H))} n + \epsilon^{-O(1)} n^{\omega+o(1)})$.

Counting subgraphs

Algorithmic problem

Count the number of copies of a graph H on k vertices in a graph G on n vertices, up to an error of at most ϵn^k.

Duke-Lefmann-Rödl (1996)

Can be done in time $2^{(k/\epsilon)^{O(1)}} n^{\omega+o(1)}$.

Fox-L.-Zhao (2017)

Can be done in time $O_H(\epsilon^{-O(e(H))} n + \epsilon^{-O(1)} n^{\omega+o(1)})$.

Corollary

We can approximate the count of K_{1000} in a graph on n vertices within an additive $n^{1000-10^{-6}}$ in time $O(n^{2.4})$.

Counting subgraphs proof sketch

Fox-L.-Zhao (2017)

Can count the number of copies of a graph H on k vertices in a graph G on n vertices, up to an error of at most ϵn^k in time $O_H(\epsilon^{-O(e(H))} n + \epsilon^{-O(1)} n^{\omega + o(1)})$.

Counting subgraphs proof sketch

Fox-L.-Zhao (2017)

Can count the number of copies of a graph H on k vertices in a graph G on n vertices, up to an error of at most ϵn^k in time $O_H(\epsilon^{-O(e(H))}n + \epsilon^{-O(1)}n^{\omega+o(1)})$.

Apply algorithmic Frieze-Kannan: In time $\epsilon^{-O(1)}n^{\omega+o(1)}$, get

$$G' = d(G)K_{V(G)} + c_1 K_{S_1,T_1} + c_2 K_{S_2,T_2} + \dots + c_t K_{S_t,T_t}$$

and $d_\square(G, G') \leq \epsilon/e(H)$, $t \leq \epsilon^{-O(1)}$.

Counting subgraphs proof sketch

Fox-L.-Zhao (2017)

Can count the number of copies of a graph H on k vertices in a graph G on n vertices, up to an error of at most ϵn^k in time $O_H(\epsilon^{-O(e(H))} n + \epsilon^{-O(1)} n^{\omega+o(1)})$.

Apply algorithmic Frieze-Kannan: In time $\epsilon^{-O(1)} n^{\omega+o(1)}$, get

$$G' = d(G) K_{V(G)} + c_1 K_{S_1,T_1} + c_2 K_{S_2,T_2} + \ldots + c_t K_{S_t,T_t}$$

and $d_\square(G, G') \leq \epsilon / e(H)$, $t \leq \epsilon^{-O(1)}$.

This means that the count is off by at most ϵn^k in G'.

Counting subgraphs proof sketch

Fox-L.-Zhao (2017)

Can count the number of copies of a graph H on k vertices in a graph G on n vertices, up to an error of at most ϵn^k in time $O_H(\epsilon^{-O(e(H))} n + \epsilon^{-O(1)} n^{\omega+o(1)})$.

Apply algorithmic Frieze-Kannan: In time $\epsilon^{-O(1)} n^{\omega+o(1)}$, get

$$G' = d(G) K_{V(G)} + c_1 K_{S_1,T_1} + c_2 K_{S_2,T_2} + \ldots + c_t K_{S_t,T_t}$$

and $d_\square(G, G') \leq \epsilon / e(H)$, $t \leq \epsilon^{-O(1)}$.

This means that the count is off by at most ϵn^k in G'.

We can compute $\hom(H, G')$ by computing a sum of $(t+1)^{e(H)}$ terms.

Algorithmic regularity proof sketch

Fox-L.-Zhao

An $O_{\epsilon,\alpha}(n^2)$-time deterministic algorithm which, given ϵ, α and a graph G between sets X, Y of size n, outputs either

- that (X, Y) are ϵ-regular.
- a pair of subsets $U \subset X$, $W \subset Y$ that show that (X, Y) are not $(1 - \alpha)\epsilon$-regular.

Algorithmic regularity proof sketch

Fox-L.-Zhao

An $O_{\epsilon,\alpha}(n^2)$-time deterministic algorithm which, given ϵ, α and a graph G between sets X, Y of size n, outputs either

- that (X, Y) are ϵ-regular.
- a pair of subsets $U \subset X$, $W \subset Y$ that show that (X, Y) are not $(1-\alpha)\epsilon$-regular.

Algorithmic Frieze-Kannan: $t \le (\alpha\epsilon)^{-O(1)}$, G' with $d_\square(G, G') \le \alpha\epsilon^3/4$,

$$G' = d(G)K_{V(G)} + c_1 K_{S_1,T_1} + c_2 K_{S_2,T_2} + \ldots + c_t K_{S_t,T_t}.$$

Algorithmic regularity proof sketch

Fox-L.-Zhao

An $O_{\epsilon,\alpha}(n^2)$-time deterministic algorithm which, given ϵ, α and a graph G between sets X, Y of size n, outputs either

- that (X, Y) are ϵ-regular.
- a pair of subsets $U \subset X$, $W \subset Y$ that show that (X, Y) are not $(1-\alpha)\epsilon$-regular.

Algorithmic Frieze-Kannan: $t \leq (\alpha\epsilon)^{-O(1)}$, G' with $d_\square(G, G') \leq \alpha\epsilon^3/4$,

$$G' = d(G)K_{V(G)} + c_1 K_{S_1,T_1} + c_2 K_{S_2,T_2} + \ldots + c_t K_{S_t,T_t}.$$

Can check a bounded number of cases based on the sizes of the intersection of U, W with X, Y and each S_i, T_i. Check feasibility and whether the density is off.

Algorithmic regularity proof sketch

Fox-L.-Zhao

An $O_{\epsilon,\alpha}(n^2)$-time algorithm which, given ϵ, α and a graph G between sets X, Y of size n, outputs either

- that (X, Y) are ϵ-regular.
- a pair of subsets $U \subset X$, $W \subset Y$ that show that (X, Y) are not $(1 - \alpha)\epsilon$-regular.

Algorithmic regularity proof sketch

Fox-L.-Zhao

An $O_{\epsilon,\alpha}(n^2)$-time algorithm which, given ϵ, α and a graph G between sets X, Y of size n, outputs either

- that (X, Y) are ϵ-regular.
- a pair of subsets $U \subset X$, $W \subset Y$ that show that (X, Y) are not $(1 - \alpha)\epsilon$-regular.

Corollary

An $O_{\epsilon,\alpha,k}(n^2)$-time algorithm which, given $\epsilon, \alpha, k > 0$, graph G on n vertices, and a k-part partition $\mathcal{P}$ of the vertices, either:

- correctly states that $\mathcal{P}$ is $(1 + \alpha)\epsilon$-regular.
- correctly states that $\mathcal{P}$ is not ϵ-regular.

Algorithmic regularity proof sketch

Fox-L.-Zhao

An $O_{\epsilon,\alpha}(n^2)$-time deterministic algorithm which, given ϵ, α, k and a graph G on n vertices that has an ϵ-regular partition with k parts, gives a $(1+\alpha)\epsilon$-regular partition into k parts.

Algorithmic regularity proof sketch

Fox-L.-Zhao

An $O_{\epsilon,\alpha}(n^2)$-time deterministic algorithm which, given ϵ, α, k and a graph G on n vertices that has an ϵ-regular partition with k parts, gives a $(1+\alpha)\epsilon$-regular partition into k parts.

Apply algorithmic Frieze-Kannan to obtain $t \le (\alpha\epsilon/k)^{O(1)}$, G' such that $d_\square(G, G') \le \alpha\epsilon/(10k^2)$, and

$$G' = d(G)K_{V(G)} + c_1 K_{S_1,T_1} + c_2 K_{S_2,T_2} + \ldots + c_t K_{S_t,T_t}.$$

Algorithmic regularity proof sketch

Fox-L.-Zhao

An $O_{\epsilon,\alpha}(n^2)$-time deterministic algorithm which, given ϵ, α, k and a graph G on n vertices that has an ϵ-regular partition with k parts, gives a $(1+\alpha)\epsilon$-regular partition into k parts.

Apply algorithmic Frieze-Kannan to obtain $t \le (\alpha\epsilon/k)^{O(1)}$, G' such that $d_\square(G, G') \le \alpha\epsilon/(10k^2)$, and

$$G' = d(G)K_{V(G)} + c_1 K_{S_1,T_1} + c_2 K_{S_2,T_2} + \ldots + c_t K_{S_t,T_t}.$$

Can work with G'. Need to check $2^{2^{(k/\alpha\epsilon)^{O(1)}}}$ possible partitions. For each one, either get not $(1+\alpha/2)\epsilon$-regular, or $(1+3\alpha/4)\epsilon$-regular. Second case must happen for a partition.

Conclusion

Dellamonica, Kalyanasundaram, Martin, Rödl and Shapira developed an algorithmic Frieze-Kannan regularity lemma.

Conclusion

Dellamonica, Kalyanasundaram, Martin, Rödl and Shapira developed an algorithmic Frieze-Kannan regularity lemma.

It actually gives a bit more than just a partition: it gives a finite sum structure.

Conclusion

Dellamonica, Kalyanasundaram, Martin, Rödl and Shapira developed an algorithmic Frieze-Kannan regularity lemma.

It actually gives a bit more than just a partition: it gives a finite sum structure.

We can use this to count the number of copies of a small graph H in a graph G efficiently.

Conclusion

Dellamonica, Kalyanasundaram, Martin, Rödl and Shapira developed an algorithmic Frieze-Kannan regularity lemma.

It actually gives a bit more than just a partition: it gives a finite sum structure.

We can use this to count the number of copies of a small graph H in a graph G efficiently.

We can also use this to more efficiently find and test regularity of sets and of partitions.

Conclusion

Dellamonica, Kalyanasundaram, Martin, Rödl and Shapira developed an algorithmic Frieze-Kannan regularity lemma.

It actually gives a bit more than just a partition: it gives a finite sum structure.

We can use this to count the number of copies of a small graph H in a graph G efficiently.

We can also use this to more efficiently find and test regularity of sets and of partitions.

Questions

- Faster algorithmic regularity lemmas?
- With what additive error can we count subgraphs?