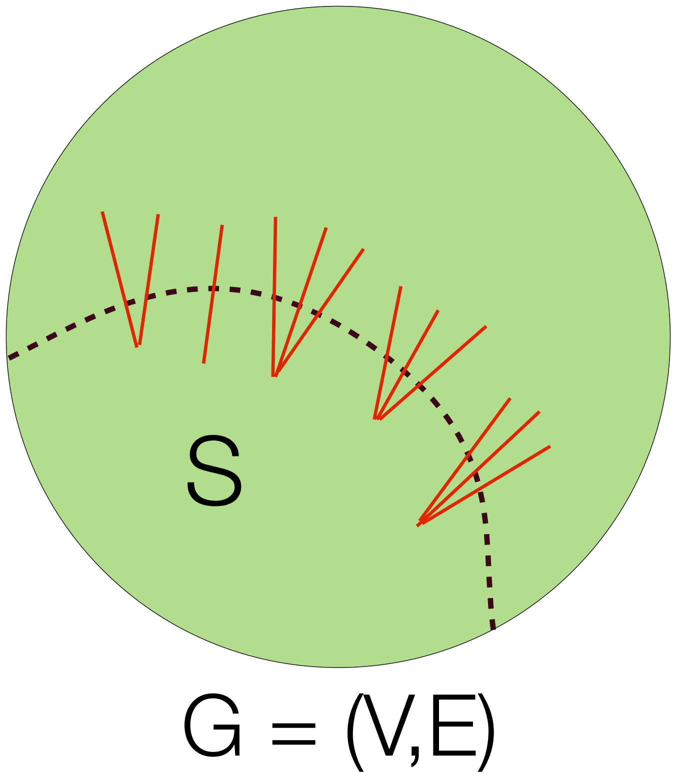
# Reductions Between Expansion Problems

Prasad Raghavendra
Georgia Tech.

David Steurer
MSR New England

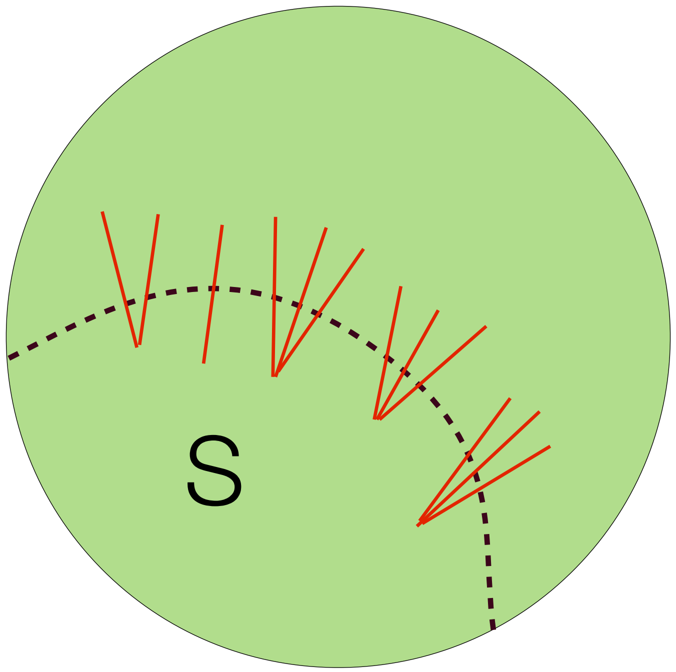
Madhur Tulsiani
Princeton University

# Graph Expansion




d-regular

# Graph Expansion




G = (V,E)

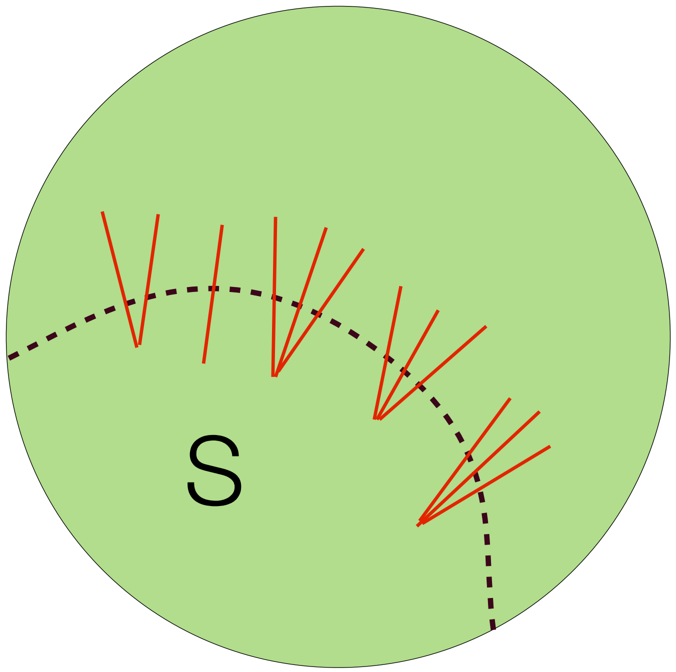
d-regular

$$\Phi_G(S) = \frac{E(S, V \setminus S)}{d|S|}$$

$$= \frac{\mathbf{P}_{(x,y) \in E}(x \in S,\ y \notin S)}{\mathbf{P}_{(x,y) \in E}(x \in S)}$$

$$\Phi_G = \min_{|S| \leq n/2} \{\Phi_G(S)\}$$

# Graph Expansion




G = (V,E)

d-regular

$$\Phi_G(S) = \frac{E(S, V \setminus S)}{d|S|}$$

$$= \frac{\mathbf{P}_{(x,y)\in E}(x \in S,\ y \notin S)}{\mathbf{P}_{(x,y)\in E}(x \in S)}$$

$$\Phi_G = \min_{|S| \leq n/2} \{\Phi_G(S)\}$$

- Expansion measures the probability of a random edge crossing a set S.
- Approximating the expansion of a graph is important for algorithms and also a fundamental problem in complexity.

# Expansion at different scales

# Expansion at different scales

- Define the measure of a set by the fraction of edges landing in it.

$$\mu(S) = \mathbf{P}_{(x,y) \in E}(x \in S)$$
$$= |S|/n \text{ (for regular graphs)}$$

- $\Phi_G(\delta) = \min_{\mu(S) = \delta} \{\Phi_G(S)\}$

# Expansion at different scales

- Define the measure of a set by the fraction of edges landing in it.

$$\mu(S) = \mathbf{P}_{(x,y) \in E}(x \in S)$$
$$= |S|/n \text{ (for regular graphs)}$$

- $\Phi_G(\delta) = \min_{\mu(S) = \delta} \{\Phi_G(S)\}$

- $\Phi_G$ measures the minimum expansion over all scales.

# Expansion at different scales

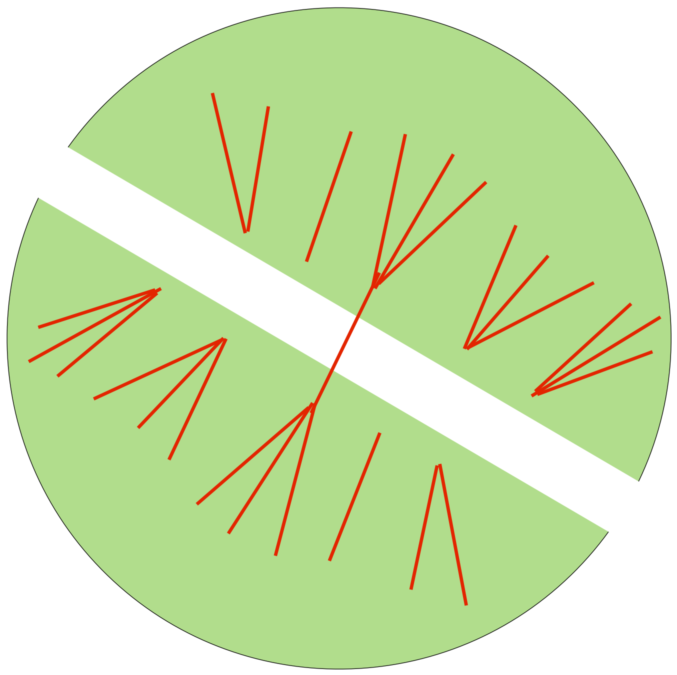
- Define the measure of a set by the fraction of edges landing in it.

$$\mu(S) = \mathbf{P}_{(x,y) \in E}(x \in S)$$
$$= |S|/n \text{ (for regular graphs)}$$

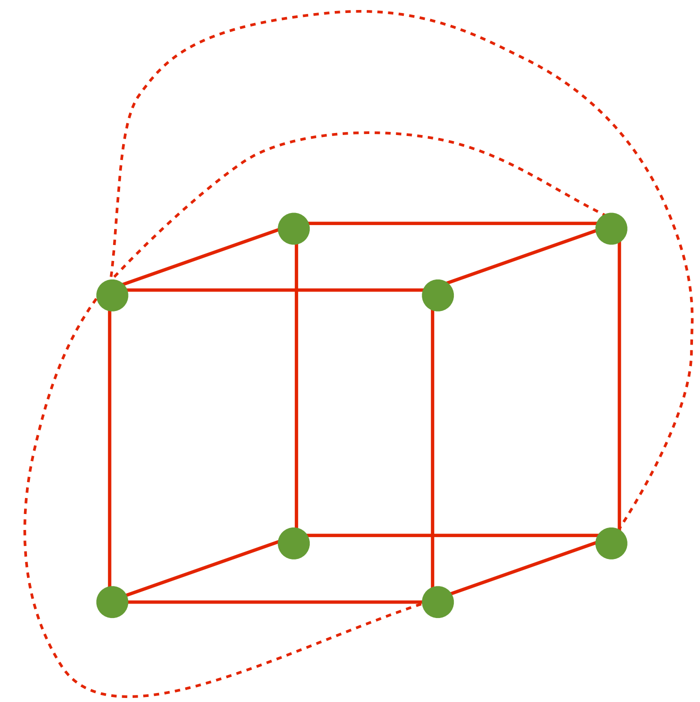
- $\Phi_G(\delta) = \min_{\mu(S) = \delta} \{\Phi_G(S)\}$

- $\Phi_G$ measures the minimum expansion over all scales.

- The expansion profile of a graph may look very different at different scales.

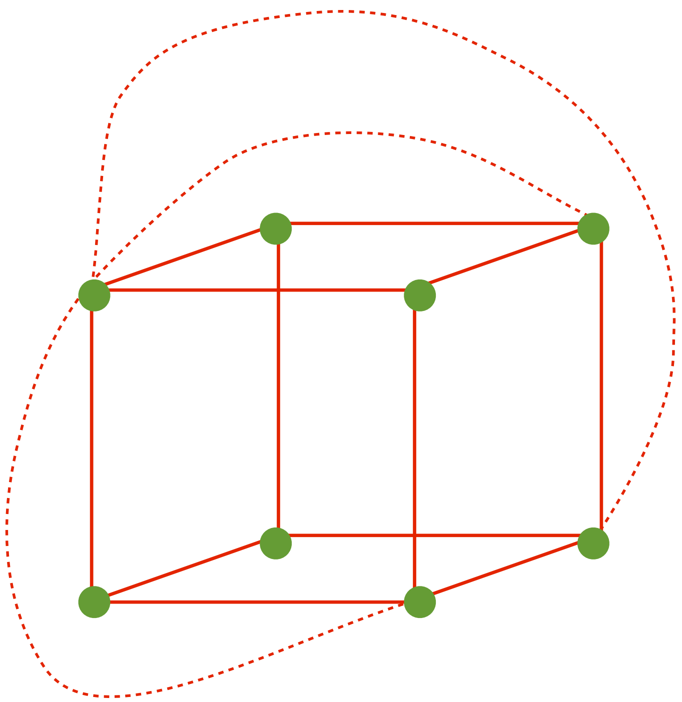


# An example: the noise graph on $\{0,1\}^n$



# An example: the noise graph on $\{0,1\}^n$

- Connect $x, y \in \{0,1\}^n$ with weight
  $\epsilon^{H(x,y)} (1-\epsilon)^{n-H(x,y)}$

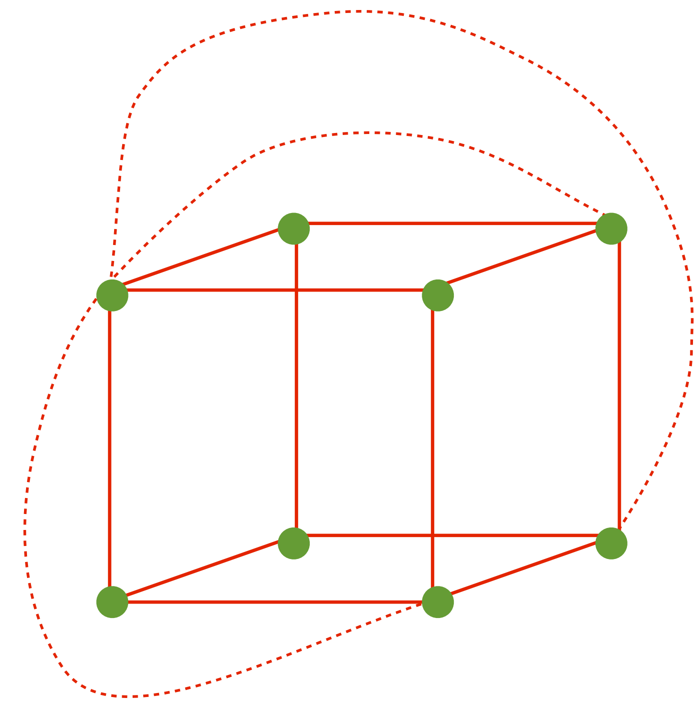


# An example: the noise graph on $\{0,1\}^n$

- Connect $x, y \in \{0,1\}^n$ with weight $\epsilon^{H(x,y)} (1-\epsilon)^{n-H(x,y)}$

- Best cut with $\mu(S) = 1/2$ is subcube of dimension n-1.

$$\Phi_G(S) = \epsilon$$



# An example: the noise graph on $\{0,1\}^n$

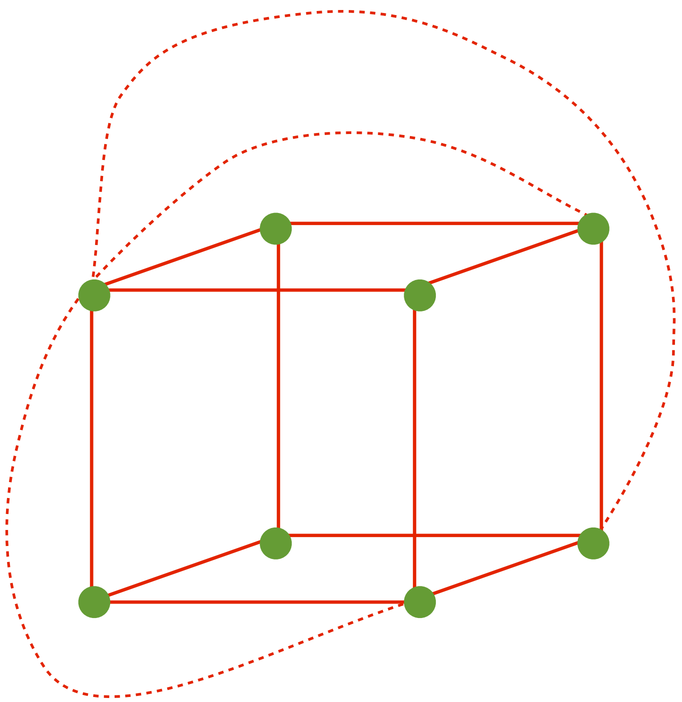
- Connect $x, y \in \{0,1\}^n$ with weight $\epsilon^{H(x,y)} (1-\epsilon)^{n-H(x,y)}$

- Best cut with $\mu(S) = 1/2$ is subcube of dimension n-1.

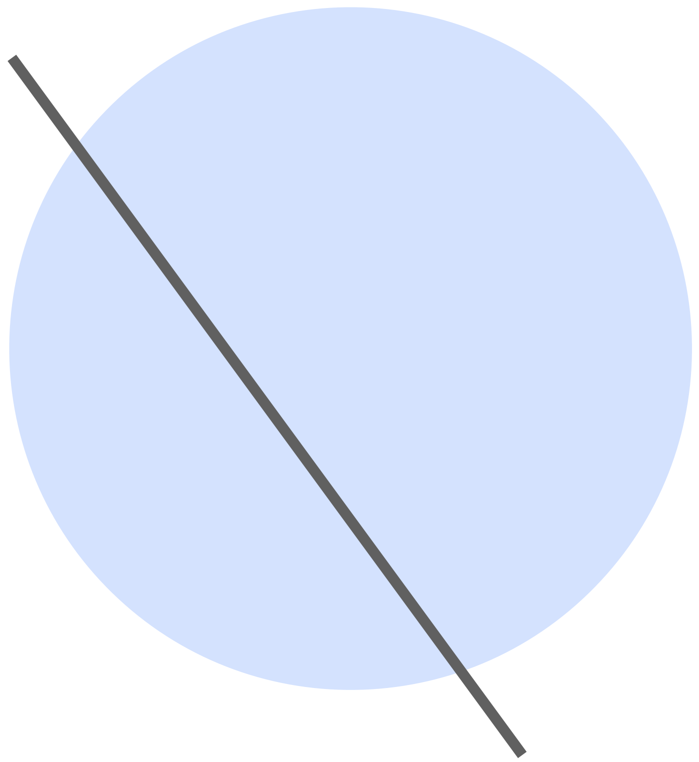
$$\Phi_G(S) = \epsilon$$

- Best cut with $\mu(S) = \delta$ is subcube of dimension $n - \log(1/\delta)$.

$$\Phi_G(S) \approx \epsilon \log(1/\delta)$$

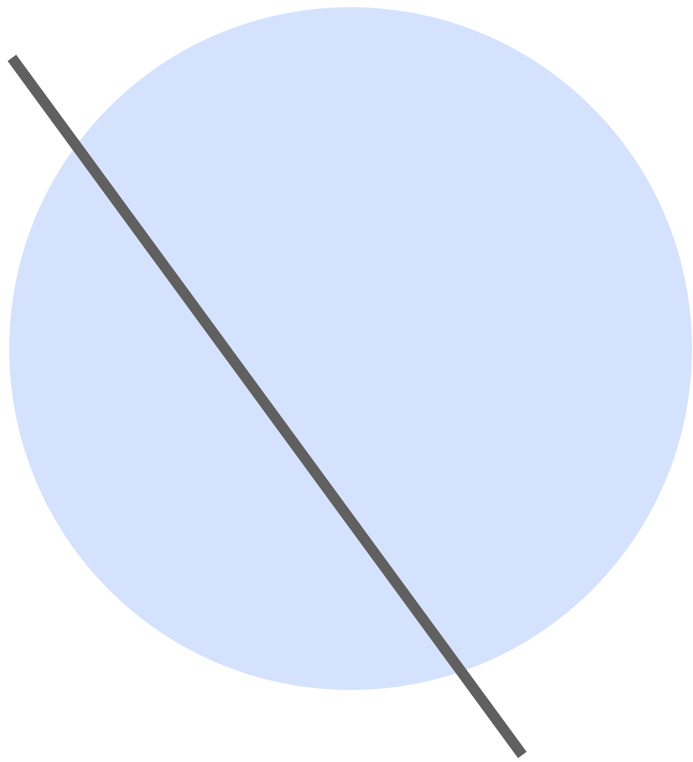


# Cuts not aligned with any direction



# Cuts not aligned with any direction

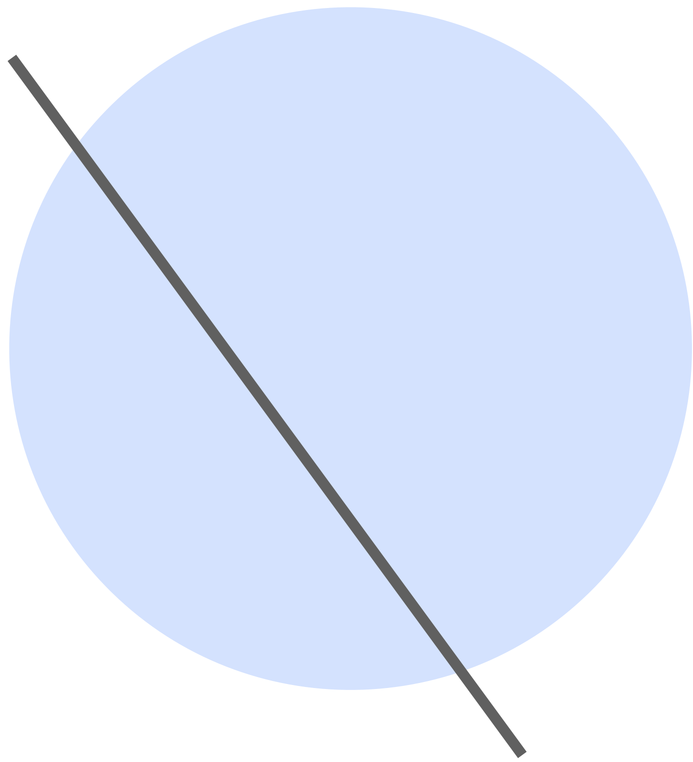
- Cuts “far from direction cuts” are like cuts on the sphere.



# Cuts not aligned with any direction

- Cuts “far from direction cuts” are like cuts on the sphere.
- Best cut with μ(S) = 1/2 is majority [MOO 05]

$\Phi_G(S) = \Omega(\epsilon^{1/2})$



# Cuts not aligned with any direction

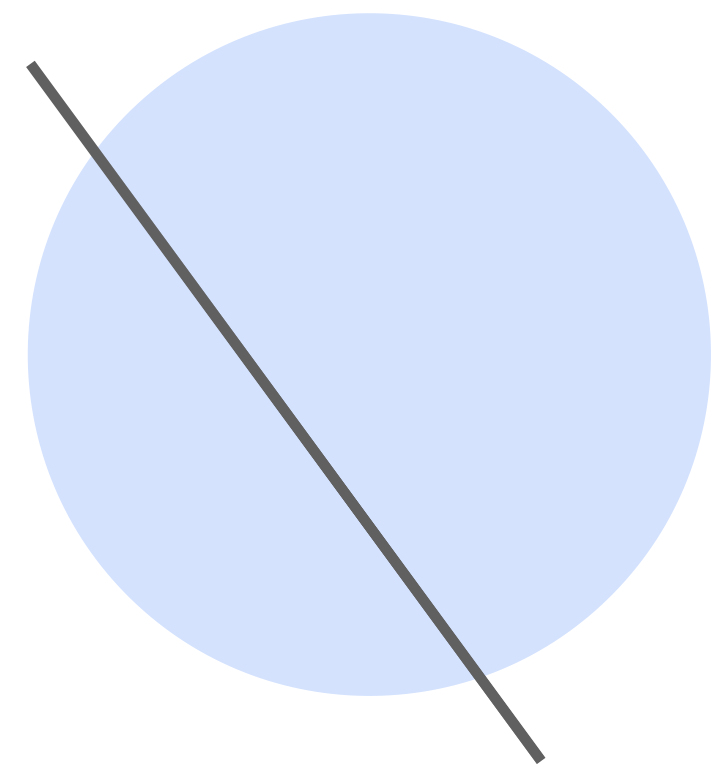
- Cuts “far from direction cuts” are like cuts on the sphere.

- Best cut with μ(S) = 1/2 is majority [MOO 05]

$$\Phi_G(S) = \Omega(\epsilon^{1/2})$$

- Best cut with μ(S) = δ is threshold function with mean δ.

$$\Phi_G(S) \approx 1 - (1/\delta)\ \Gamma_{1-2\epsilon}(\delta) \approx 1 - \delta^{\epsilon/(1-\epsilon)}$$



# The Small-Set Expansion problem

# The Small-Set Expansion problem

- [RS 10] Studied the problem of approximating the expansion for small sets.

- Showed a reduction to Unique Games.

# The Small-Set Expansion problem

- [RS 10] Studied the problem of approximating the expansion for small sets.

- Showed a reduction to Unique Games.

- SSE($\eta$,$\delta$): Given graph G, distinguish between the following cases:
  - $\exists S$ with $\mu(S) = \delta$ such that $\Phi_G(S) \leq \eta$
  - $\forall S$ with $\mu(S) = \delta$, $\Phi_G(S) \geq 1-\eta$

# The Small-Set Expansion problem

- [RS 10] Studied the problem of approximating the expansion for small sets.

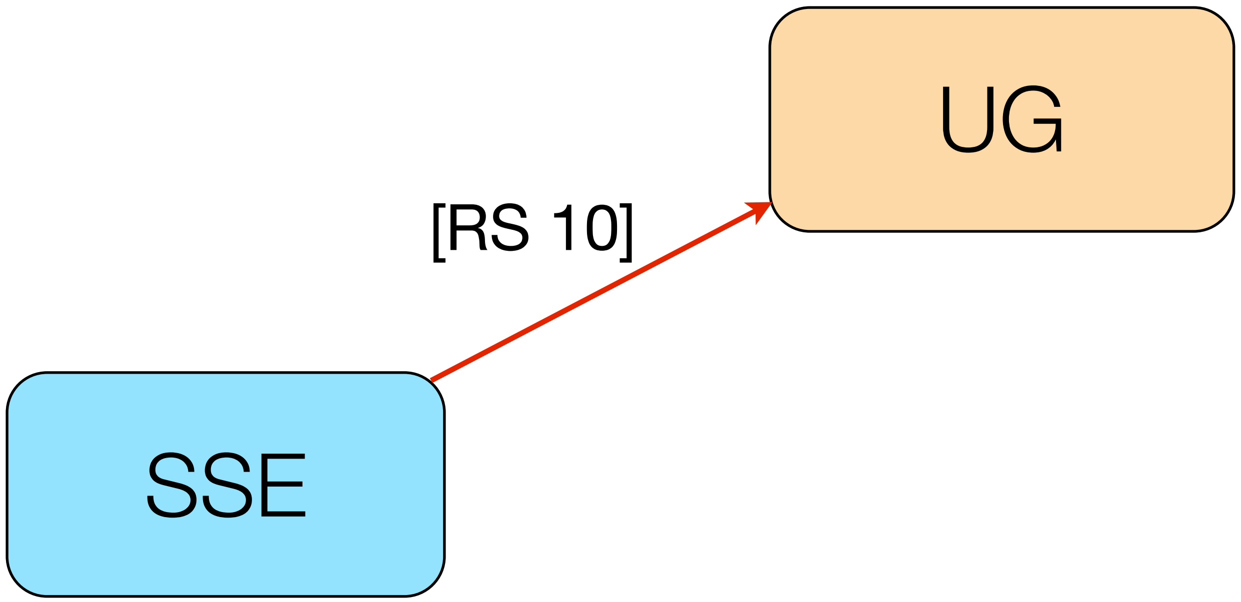
- Showed a reduction to Unique Games.

- SSE($\eta$,$\delta$): Given graph G, distinguish between the following cases:
  - $\exists S$ with $\mu(S) = \delta$ such that $\Phi_G(S) \leq \eta$
  - $\forall S$ with $\mu(S) = \delta$, $\Phi_G(S) \geq 1-\eta$

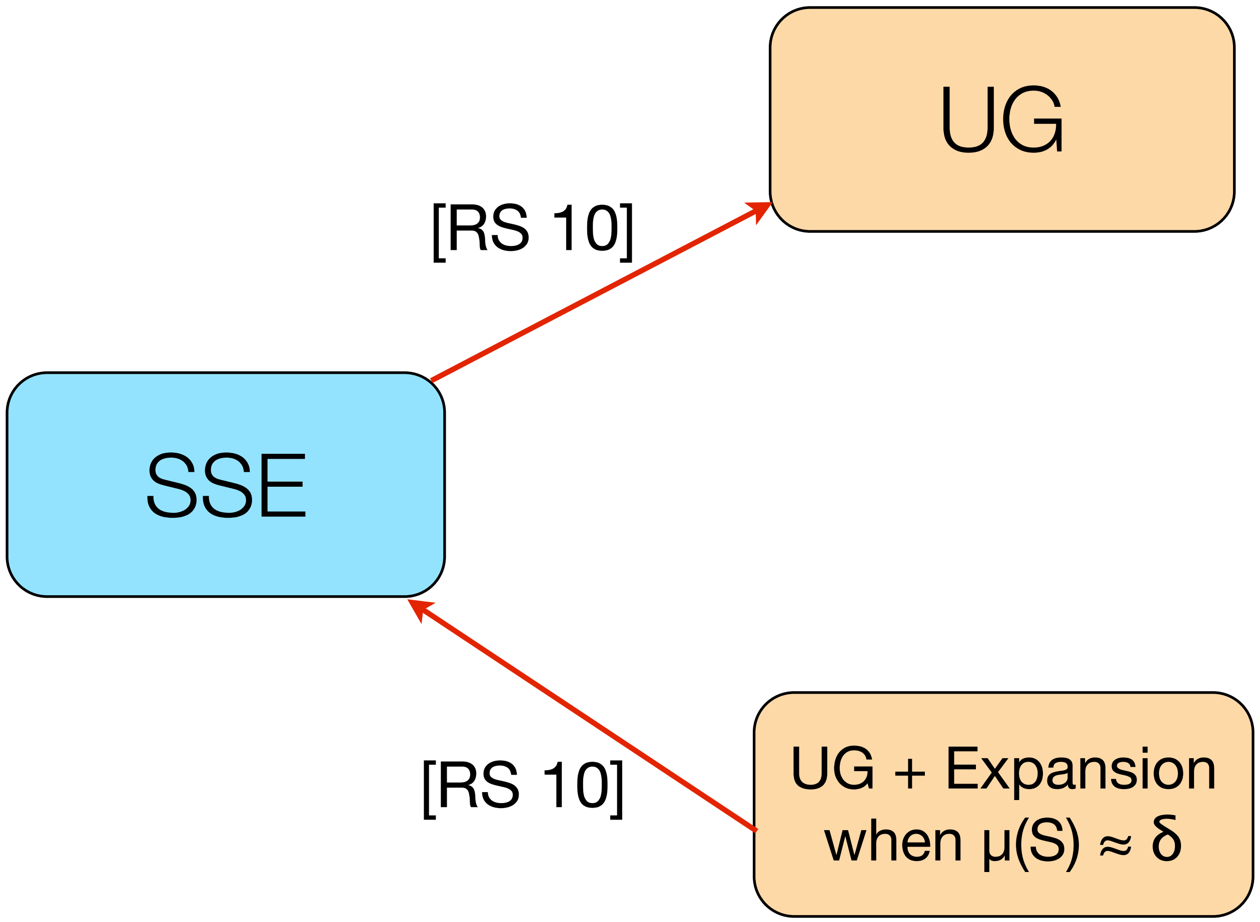
- SSE-Conjecture: $\forall \eta$ $\exists \delta$ such that SSE($\eta$,$\delta$) is hard.

# The big-picture

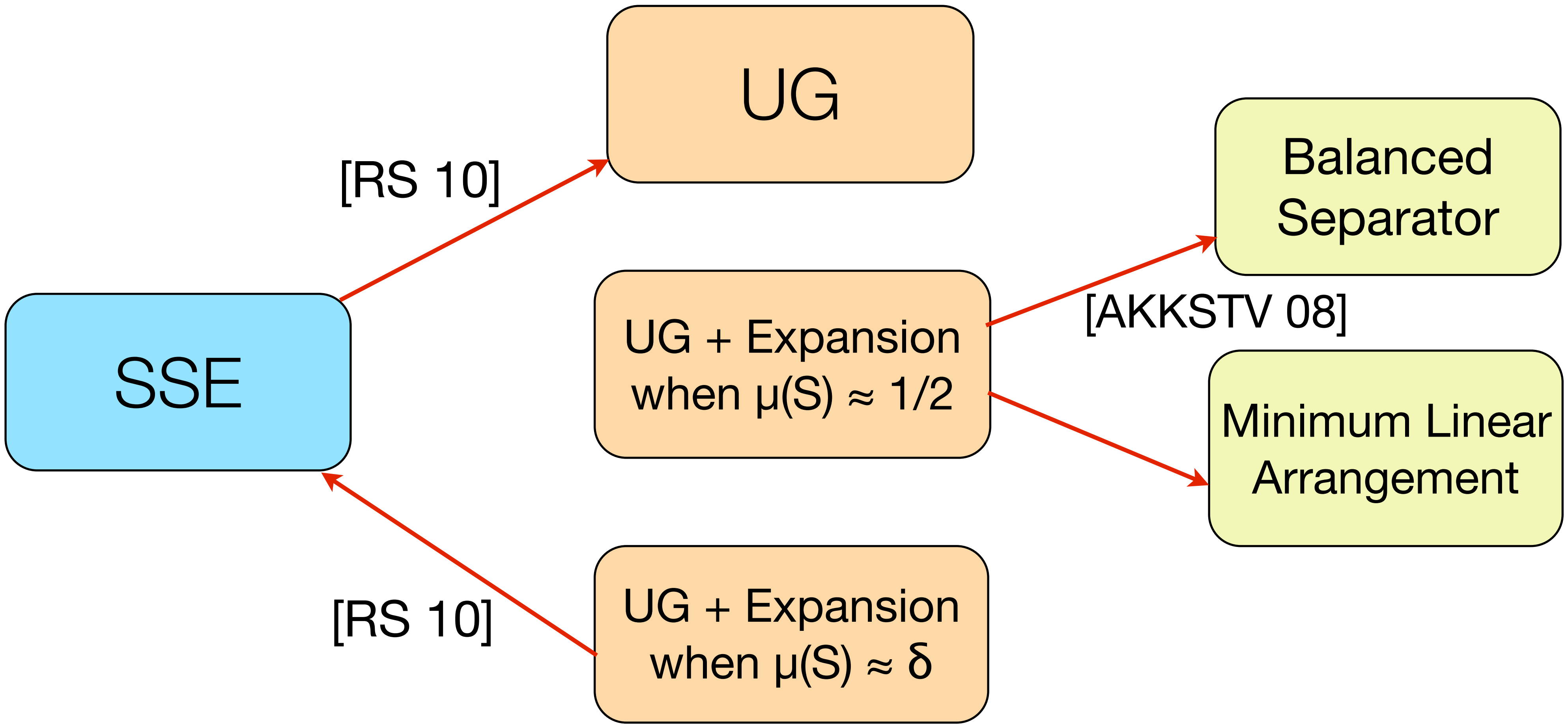
# The big-picture



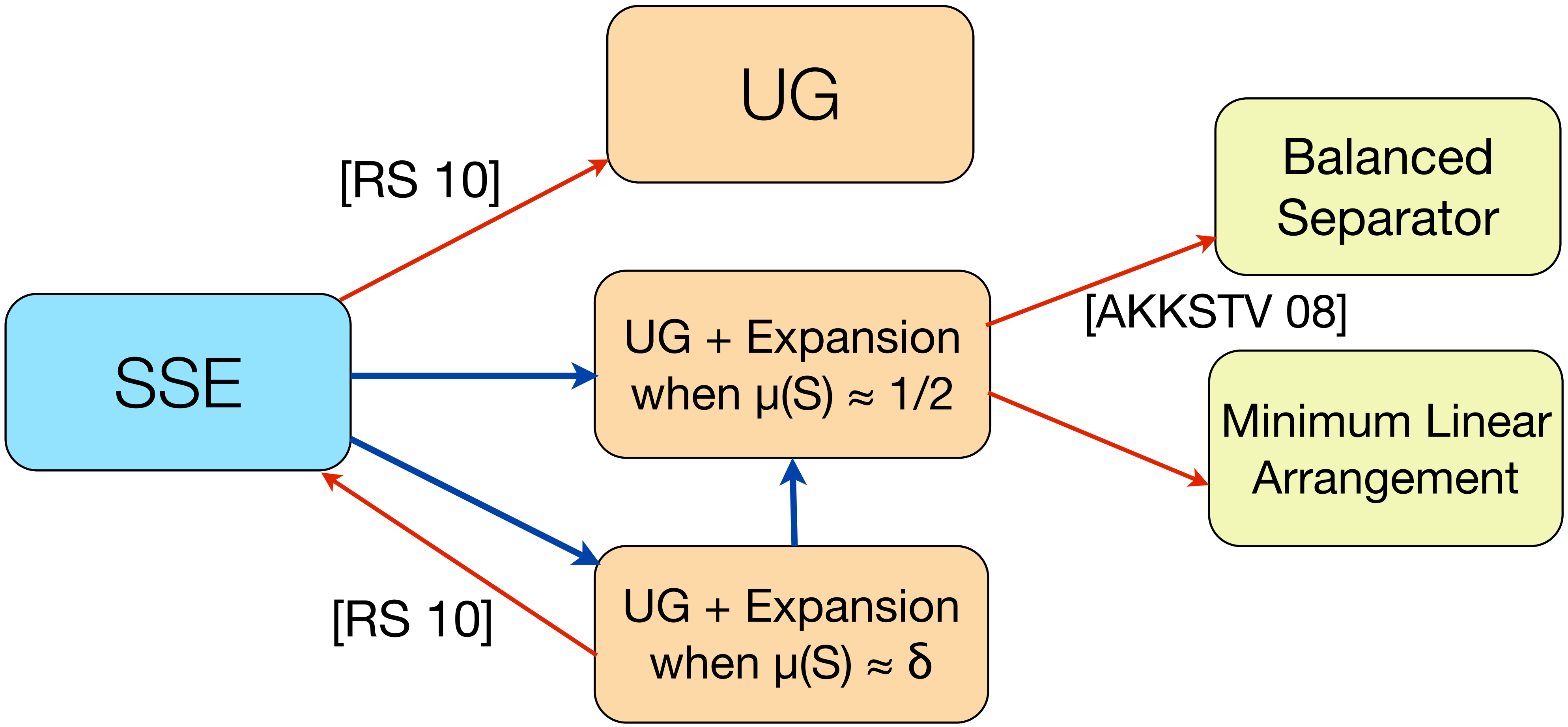
# The big-picture



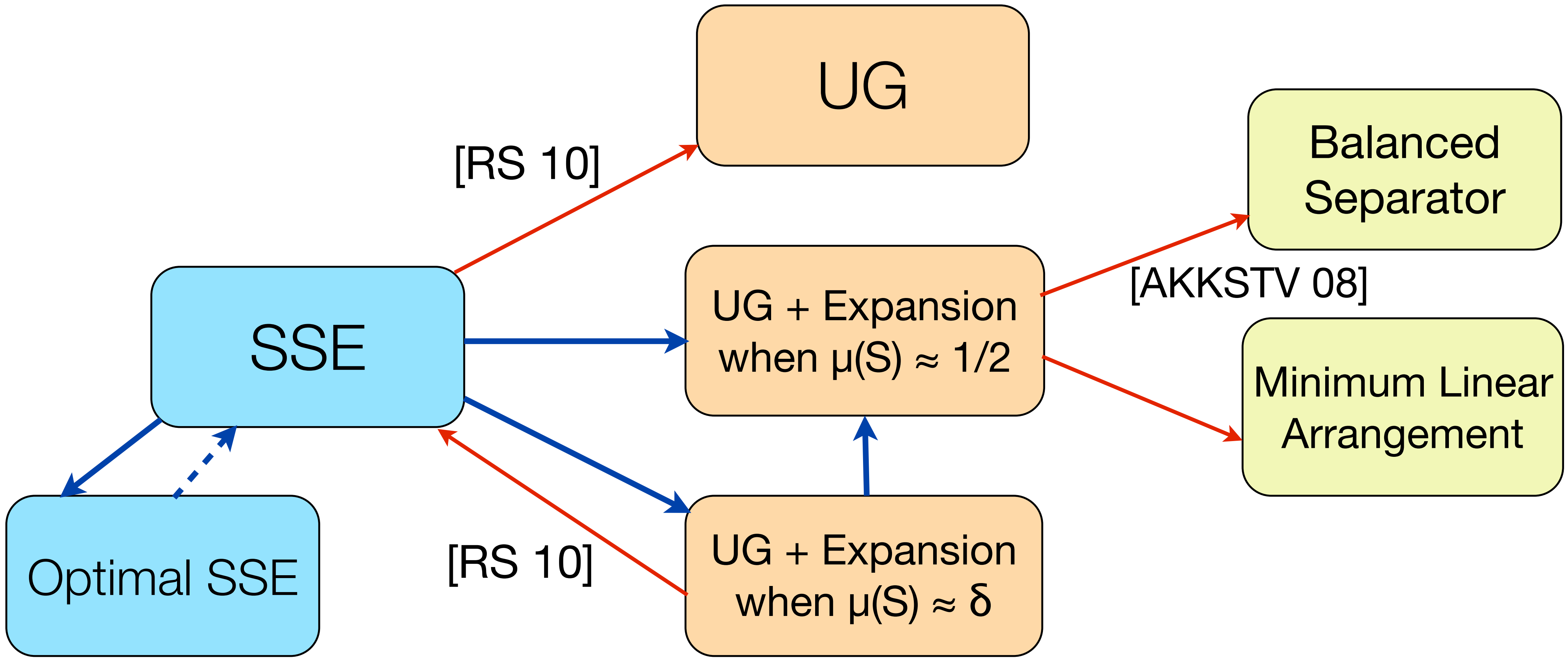
# The big-picture



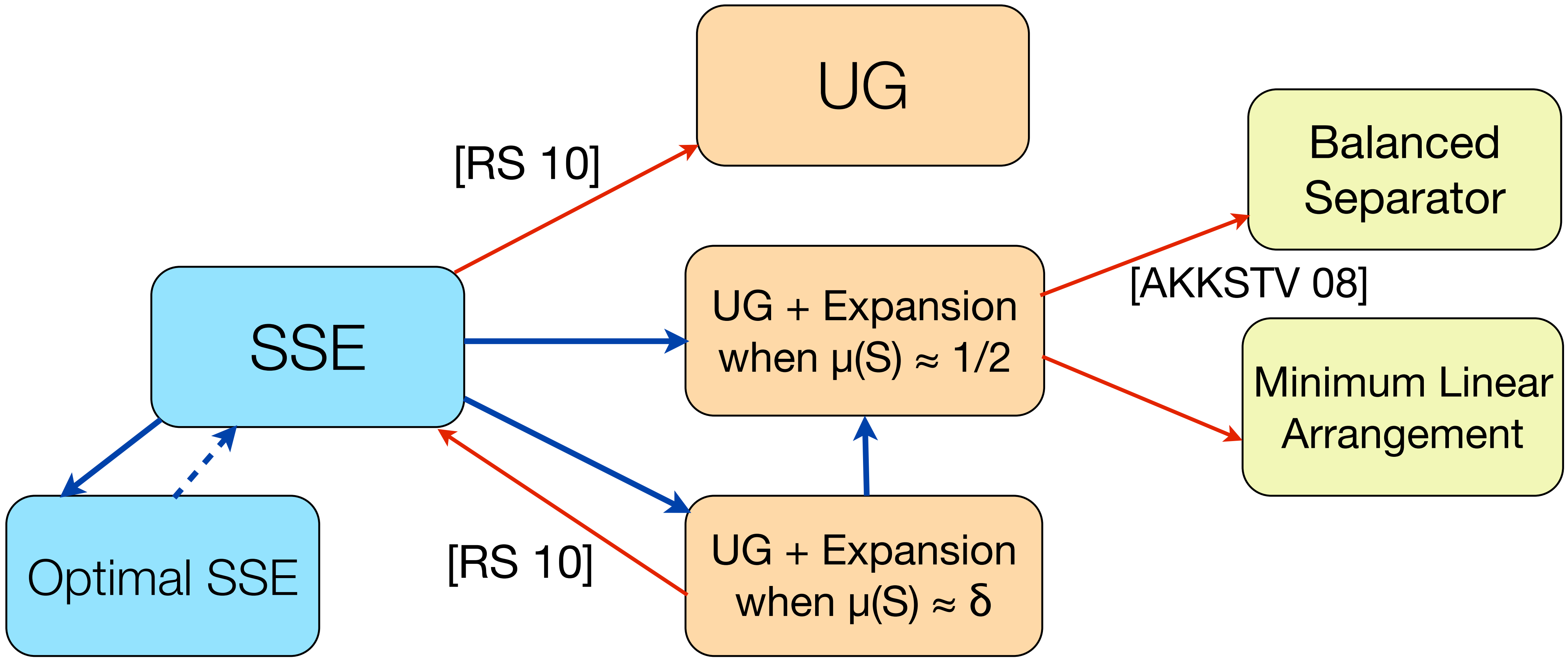
# The big-picture



# The big-picture



# The big-picture




- Can reduce problem about expansion at scale $\delta$ to any scale $> \delta$.
- Similar to alphabet reduction for Unique Games by [KKMO 04] (UG alphabet $\approx 1/\delta$).

# Main result

# Main result

For all $\epsilon$, $\delta > 0$ and $q \in N$, the following is SSE-hard.

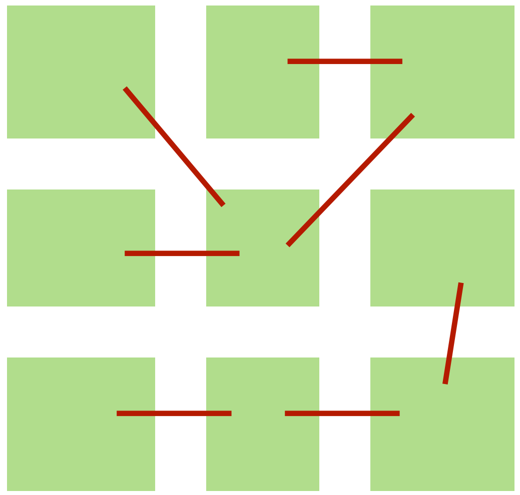
Given $H = (V_H, E_H)$, distinguish between the cases:

# Main result

For all $\epsilon$, $\delta > 0$ and $q \in N$, the following is SSE-hard.

Given $H = (V_H, E_H)$, distinguish between the cases:

- $V_H$ can be partitioned in to equal pieces $S_1, \ldots, S_q$ s.t. $\forall t$ $\Phi_G(S_t) \leq \epsilon + o(\epsilon)$.

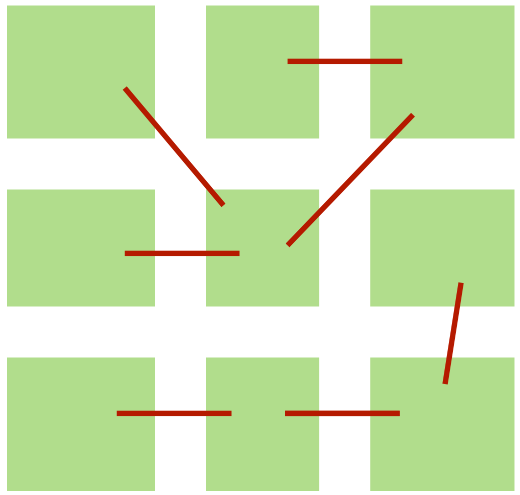


# Main result

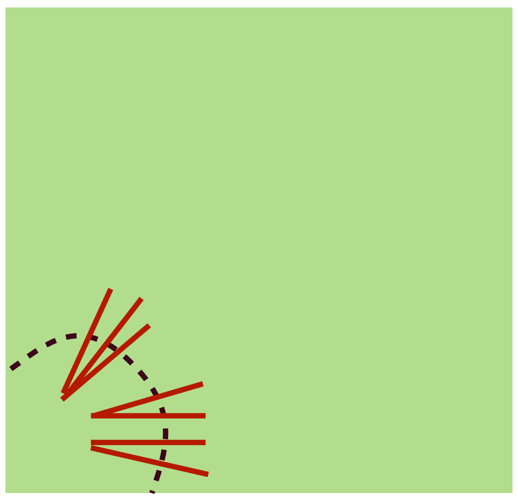
For all $\epsilon, \delta > 0$ and $q \in N$, the following is SSE-hard.

Given $H = (V_H, E_H)$, distinguish between the cases:

- $V_H$ can be partitioned in to equal pieces $S_1, \ldots, S_q$ s.t. $\forall$ t $\Phi_G(S_t) \leq \epsilon + o(\epsilon)$.
- For all $S \subseteq V_H$, with $\mu(S) \geq \delta$ $\Phi_G(S) \geq 1 - (1/\mu(S)) \Gamma_{1-\epsilon/2}(\mu(S)) - o(\epsilon)$



vs.



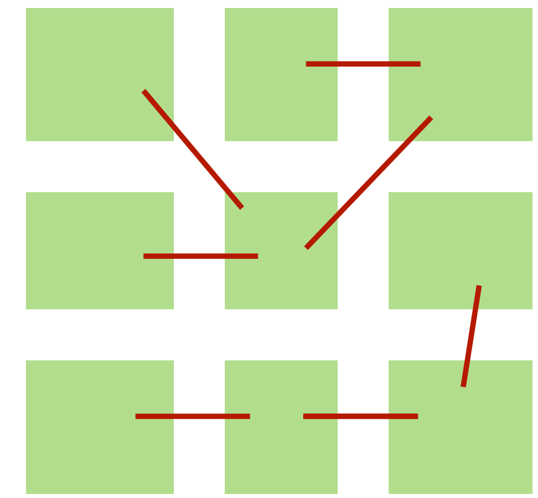
# Main result

For all $\epsilon, \delta > 0$ and $q \in \mathbb{N}$, the following is SSE-hard.

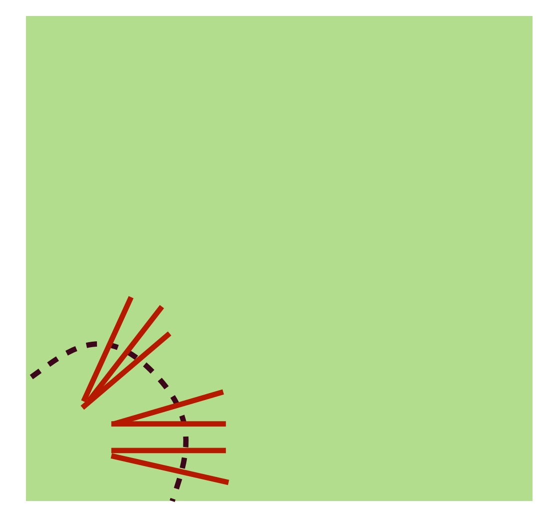
Given $H = (V_H, E_H)$, distinguish between the cases:

- $V_H$ can be partitioned in to equal pieces $S_1, \ldots, S_q$ s.t. $\forall\, t$ $\Phi_G(S_t) \leq \epsilon + o(\epsilon)$.

- For all $S \subseteq V_H$, with $\mu(S) \geq \delta$
 $\Phi_G(S) \geq 1 - (1/\mu(S))\, \Gamma_{1-\epsilon/2}(\mu(S)) - o(\epsilon)$



vs.



Matches (upto constants) the algorithmic guarantee of [RST 10]

# Application to Balanced Separator

# Application to Balanced Separator

Balanced Separator: Minimize $\Phi_G(S)$ over all sets S with $1/4 \leq \mu(S) \leq 1/2$.

# Application to Balanced Separator

Balanced Separator: Minimize $\Phi_G(S)$ over all sets S with $1/4 \leq \mu(S) \leq 1/2$.

Simply taking $q = 2$ and $\delta = 1/4$ gives the hardness of distinguishing for $H = (V_H, E_H)$:

- There exists S with $\mu(S) = 1/2$ and $\Phi_G(S) \leq \epsilon + o(\epsilon)$
- For all S with $1/4 \leq \mu(S) \leq 1/2$, $\Phi_G(S) \geq \Omega(\epsilon^{1/2})$

# Application to Balanced Separator

Balanced Separator: Minimize $\Phi_G(S)$ over all sets S with $1/4 \leq \mu(S) \leq 1/2$.

Simply taking $q = 2$ and $\delta = 1/4$ gives the hardness of distinguishing for $H = (V_H, E_H)$:

- There exists S with $\mu(S) = 1/2$ and $\Phi_G(S) \leq \epsilon + o(\epsilon)$
- For all S with $1/4 \leq \mu(S) \leq 1/2$, $\Phi_G(S) \geq \Omega(\epsilon^{1/2})$

Also gives hardness for Minimum Bisection since $\mu(S) = 1/2$ in the first case.

# Application to Minimum Linear Arrangement

# Application to Minimum Linear Arrangement

Problem: Find an ordering $\pi$ of the vertices to minimize the average length of an edge = $E_{(x,y) \in E} |\pi(x) - \pi(y)|$.

# Application to Minimum Linear Arrangement

Problem: Find an ordering $\pi$ of the vertices to minimize the average length of an edge = $E_{(x,y) \in E} |\pi(x) - \pi(y)|$.

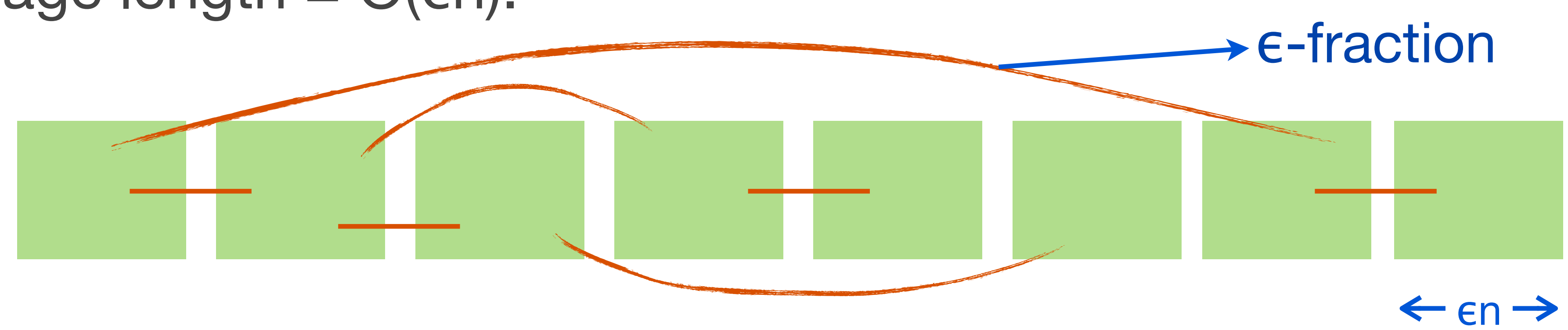
Taking $q = 1/\epsilon$ gives the following two cases:

# Application to Minimum Linear Arrangement

Problem: Find an ordering π of the vertices to minimize the average length of an edge = $E_{(x,y) \in E} |\pi(x) - \pi(y)|$.

Taking q = 1/ϵ gives the following two cases:

- Parts of size ϵn with ≤ ϵ fraction of edges going between parts. Average length = O(ϵn).

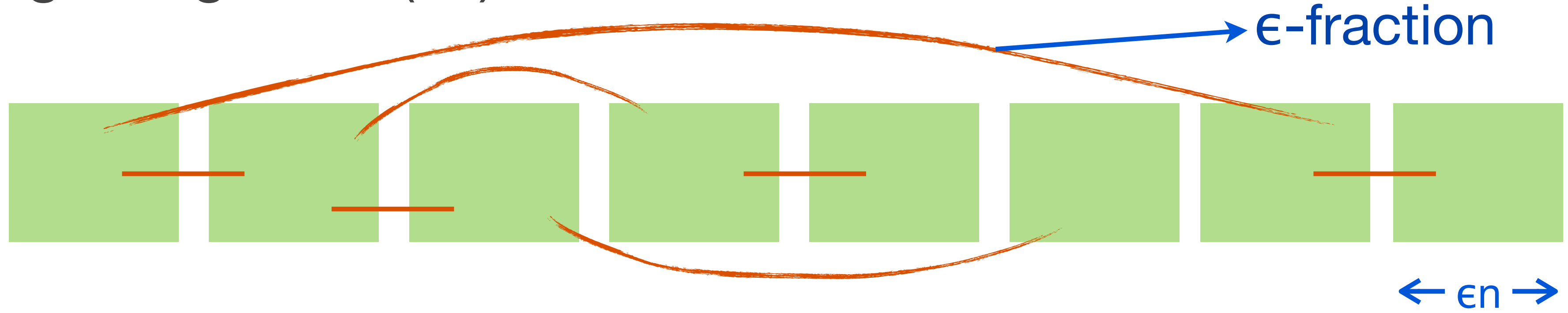


# Application to Minimum Linear Arrangement

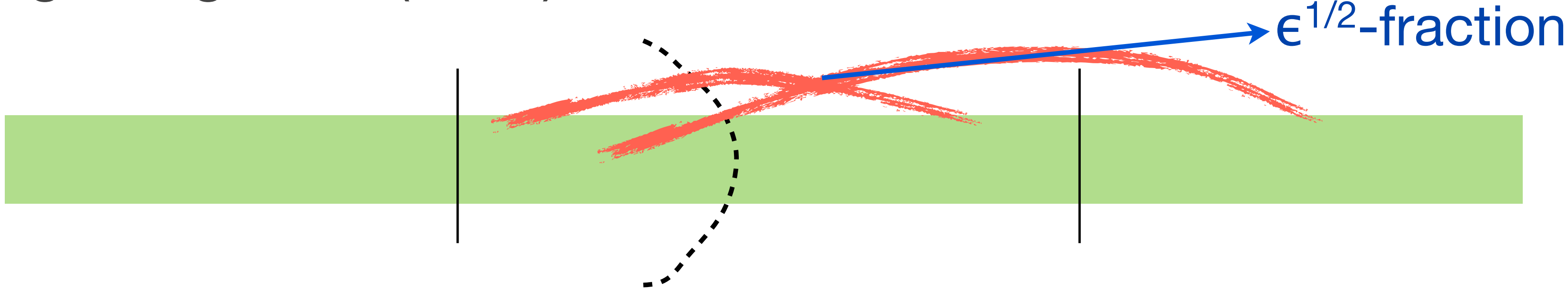
Problem: Find an ordering π of the vertices to minimize the average length of an edge = $E_{(x,y) \in E} |\pi(x) - \pi(y)|$.

Taking q = 1/ϵ gives the following two cases:

- Parts of size ϵn with ≤ ϵ fraction of edges going between parts. Average length = O(ϵn).




- $\Omega(\epsilon^{1/2})$ fraction of edges crossing every cut on middle n/3 vertices. Average length = $\Omega(\epsilon^{1/2}\, n)$.



# Reducing to an expansion problem

# Unique Games

# Unique Games

- Given: Graph G=(V,E), alphabet R and for each edge (u,v), a permutation constraint $\pi_{uv} : [R] \to [R]$.

  Find: A labeling $L : V \to [R]$ satisfying for as many edges (u,v) as possible

  $\pi_{uv}(L(u)) = L(v)$

- Can also be viewed as a two-prover game (Alice and Bob get one vertex each from a randomly chosen edge).

# Unique Games

- Given: Graph G=(V,E), alphabet R and for each edge (u,v), a permutation constraint $\pi_{uv}$ : [R] → [R].

  Find: A labeling L : V → [R] satisfying for as many edges (u,v) as possible

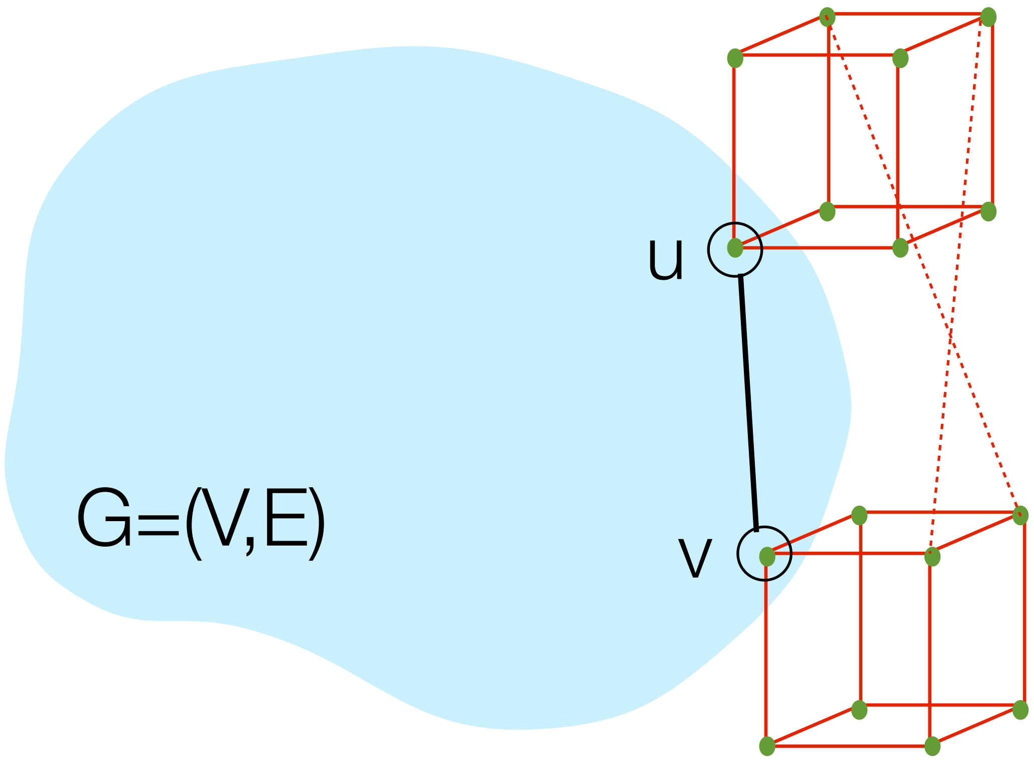
  $\pi_{uv}(L(u)) = L(v)$

- Can also be viewed as a two-prover game (Alice and Bob get one vertex each from a randomly chosen edge).

- UGC: Hard to distnguish between the cases when
  - At least 1-η fraction of edges can be satisfied
  - At most η fraction of edges can be satisfied.

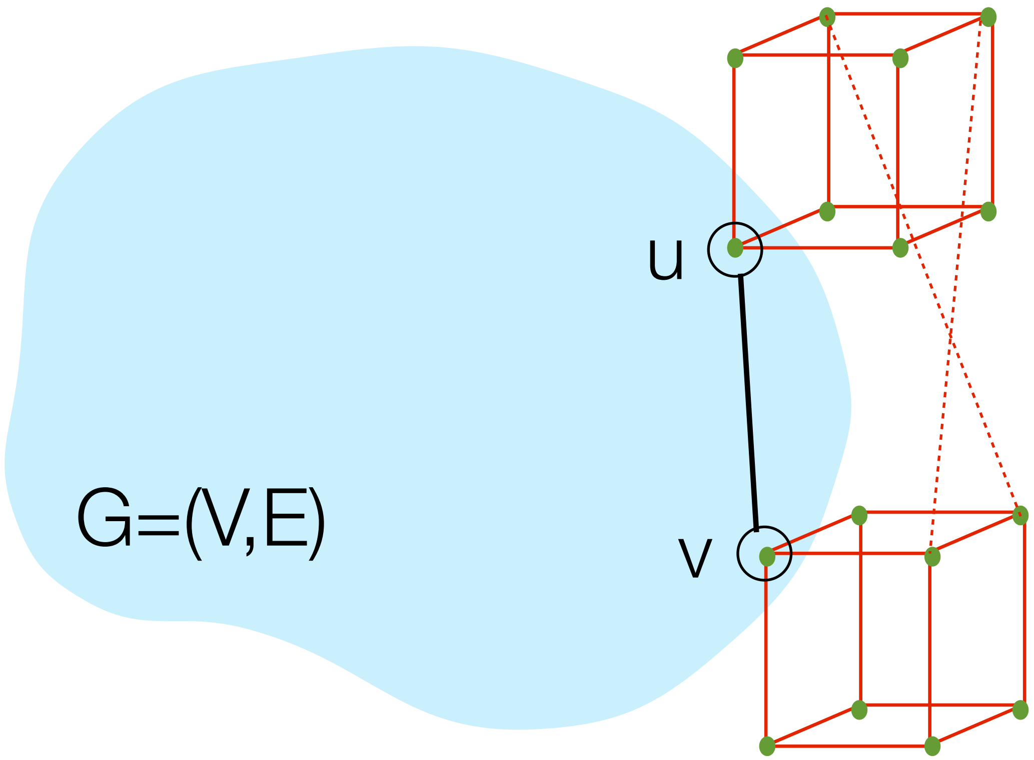
# (Attempted) Reduction to Balanced Separator

# (Attempted) Reduction to Balanced Separator




- Attach $\{0,1\}^R$ to each vertex of G
- Pick $x, y \in \{0,1\}^R$ with $\epsilon$-noise (weight = $\epsilon^{H(x,y)} (1-\epsilon)^{R-H(x,y)}$ )
- Connect $(u, \pi_{uv}(x))$ to $(v,y)$ with above weight. (send $(u, \pi_{uv}(x))$ to Alice, $(v,y)$ to Bob)
- Ask for a balanced cut on new graph.

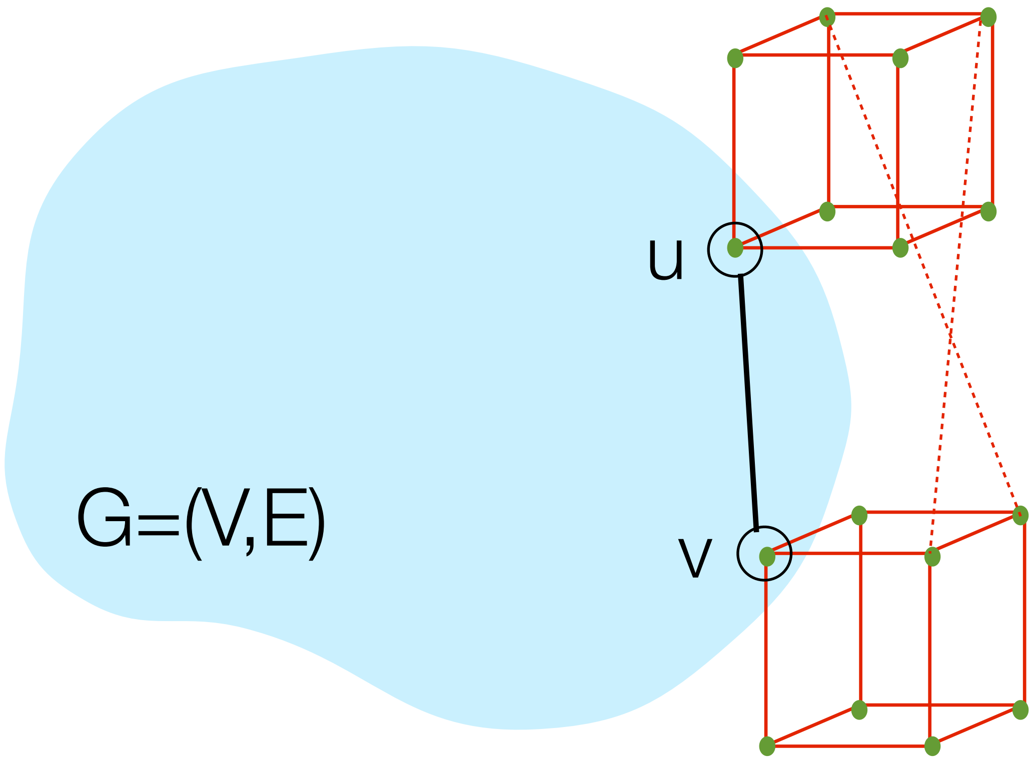
# (Attempted) Reduction to Balanced Separator




- Attach $\{0,1\}^R$ to each vertex of G
- Pick x, y $\in \{0,1\}^R$ with $\epsilon$-noise (weight = $\epsilon^{H(x,y)} (1-\epsilon)^{R-H(x,y)}$ )
- Connect $(u, \pi_{uv}(x))$ to $(v,y)$ with above weight. (send $(u, \pi_{uv}(x))$ to Alice, $(v,y)$ to Bob)
- Ask for a balanced cut on new graph.

- When good labeling exists, for each u, pick R-1 dimensional sub-cube with $x_{L(u)} = 1$.
- If $\pi_{uv}(L(u)) = L(v)$, then only $\epsilon$-fraction of edges between cubes of u and v are cut.

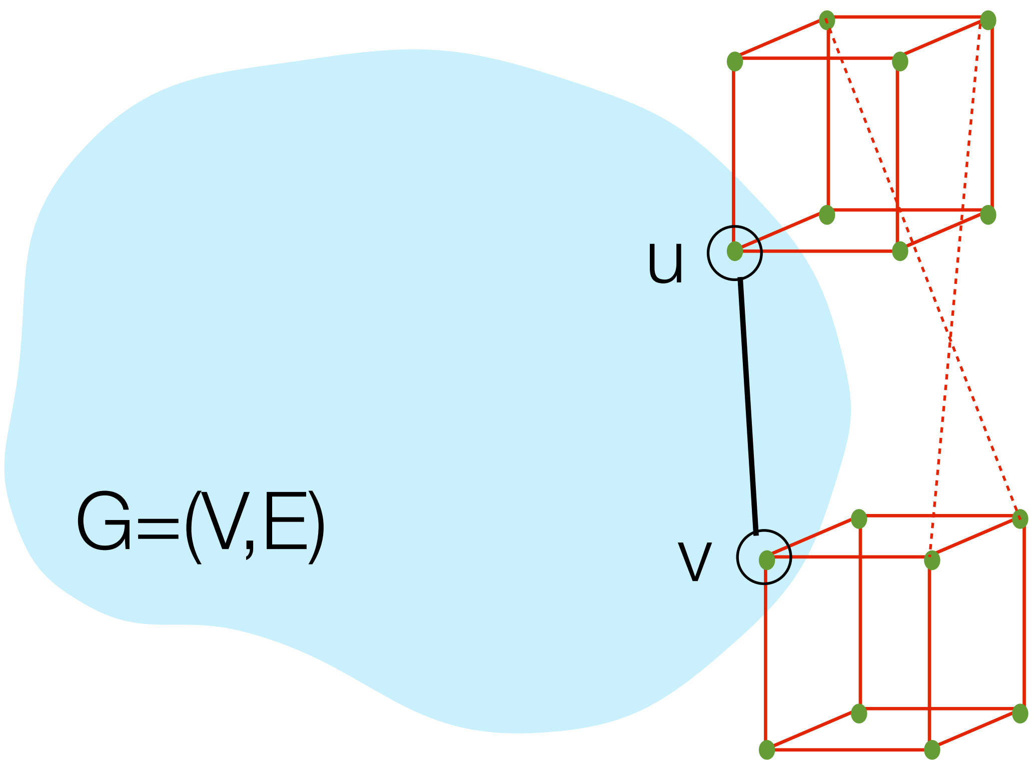
# (Attempted) Reduction to Balanced Separator



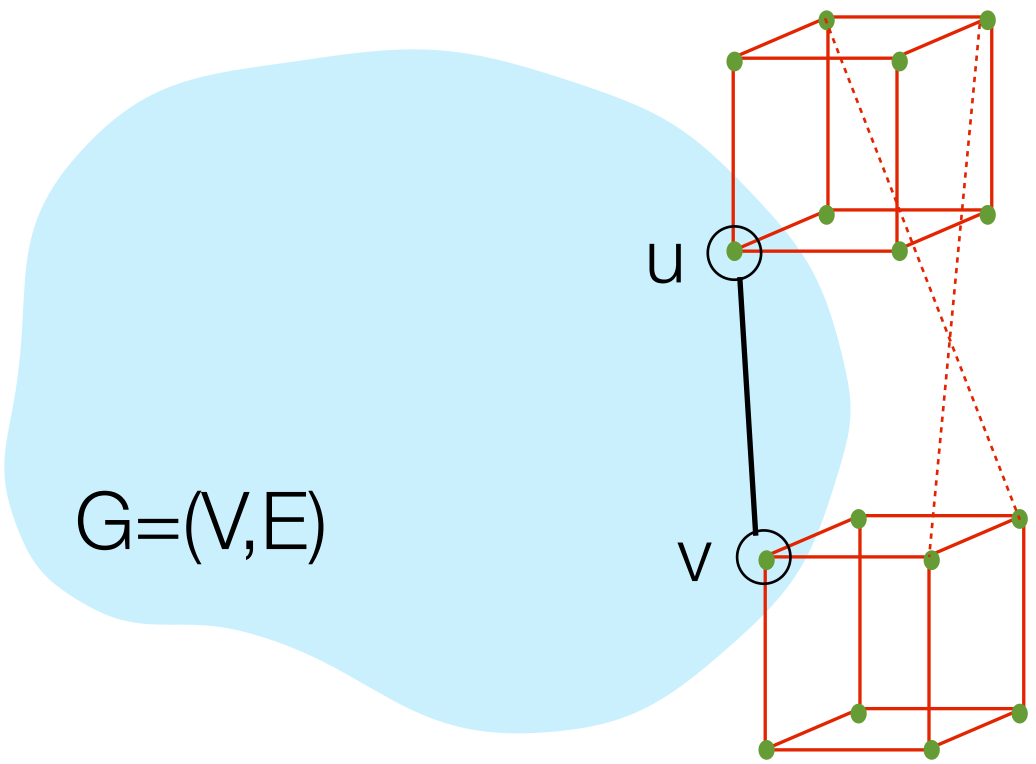

- Attach $\{0,1\}^R$ to each vertex of G
- Pick x, y $\in \{0,1\}^R$ with $\epsilon$-noise (weight = $\epsilon^{H(x,y)} (1-\epsilon)^{R-H(x,y)}$ )
- Connect $(u, \pi_{uv}(x))$ to $(v,y)$ with above weight. (send $(u, \pi_{uv}(x))$ to Alice, $(v,y)$ to Bob)
- Ask for a balanced cut on new graph.

- When good labeling exists, for each u, pick R-1 dimensional sub-cube with $x_{L(u)} = 1$.
- If $\pi_{uv}(L(u)) = L(v)$, then only $\epsilon$-fraction of edges between cubes of u and v are cut.
- If labeling satisfies $1-\eta$ fraction, then value of cut in graph is $\leq \epsilon + \eta$.

# Problems with the reduction

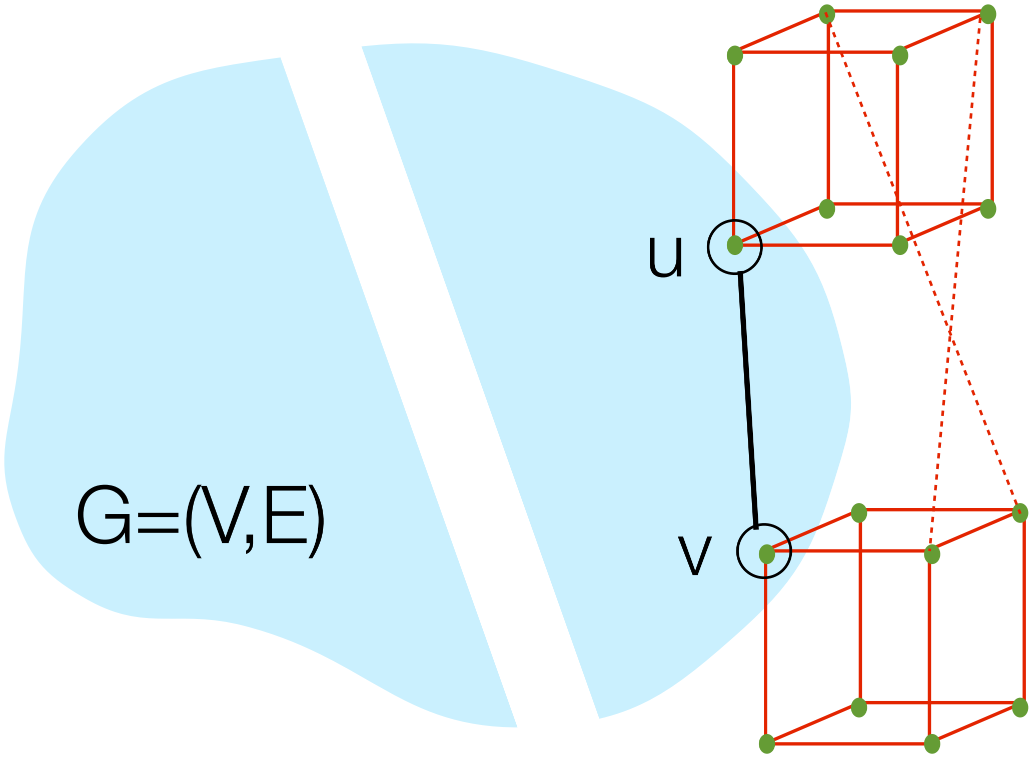


# Problems with the reduction



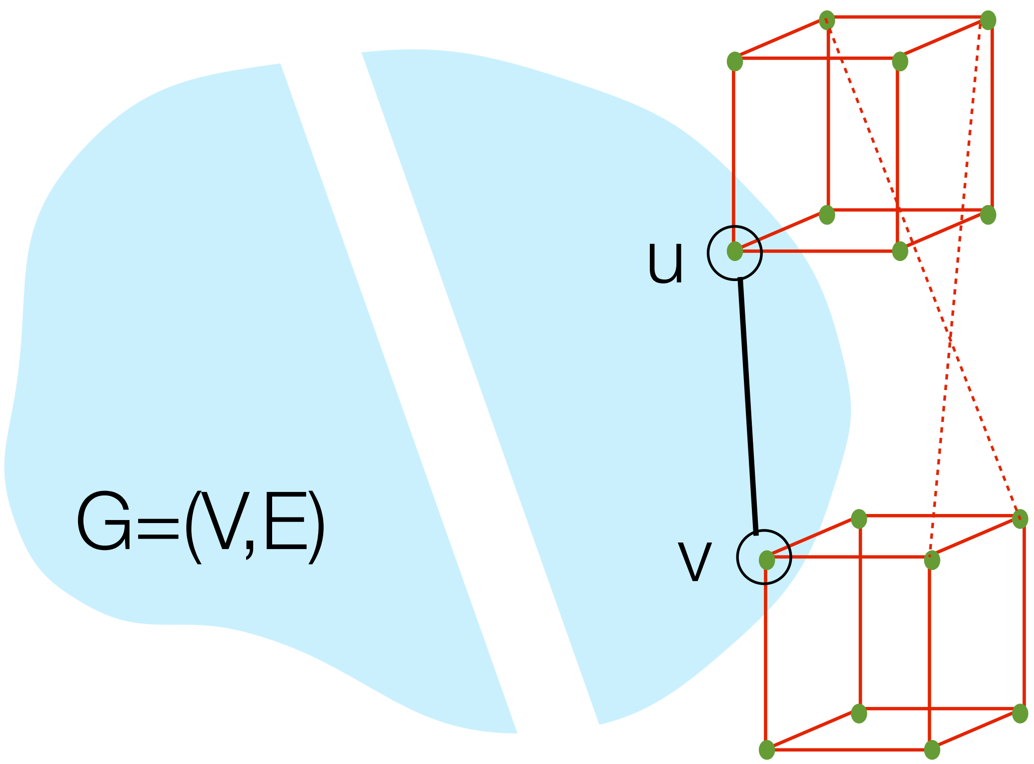

- Even when no good labeling exists, there might be a way to cut which does not cut the hypercubes at all.

# Problems with the reduction



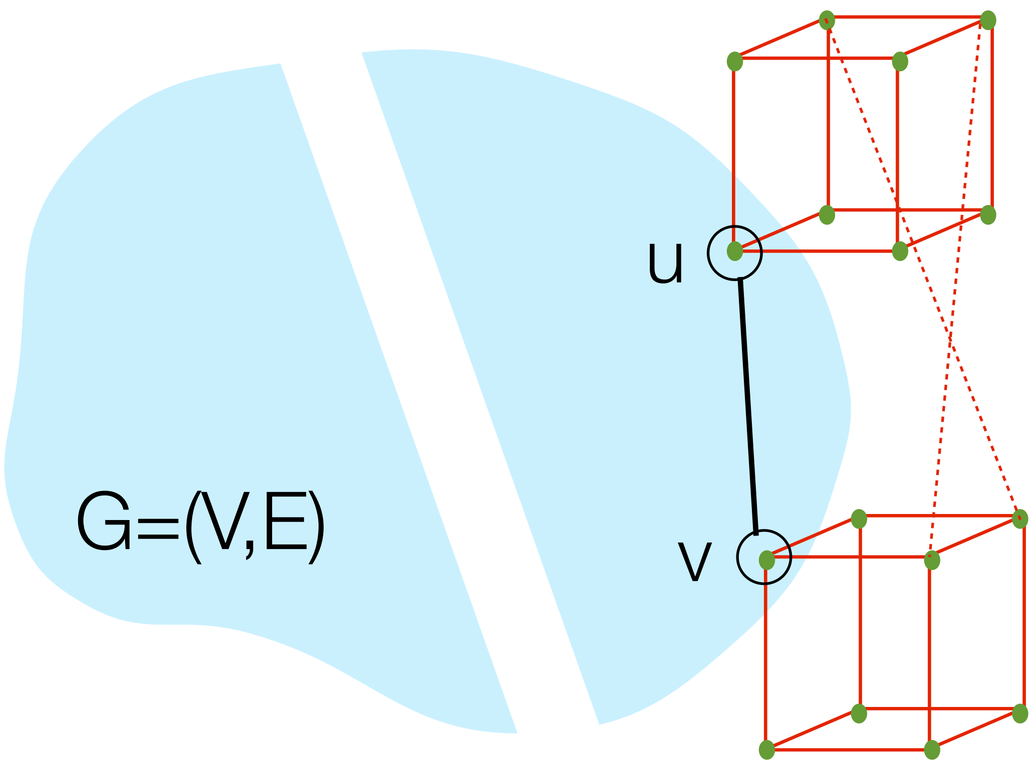

- Even when no good labeling exists, there might be a way to cut which does not cut the hypercubes at all.

# Problems with the reduction




- Even when no good labeling exists, there might be a way to cut which does not cut the hypercubes at all.
- Reduction preserves structure of starting graph.
- If starting UG does not have expansion, neither does the new graph.

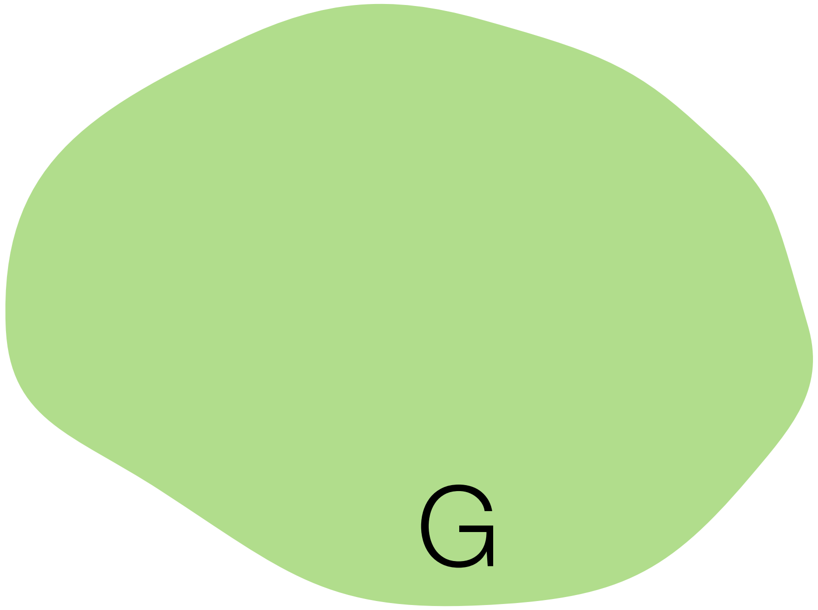
# Problems with the reduction



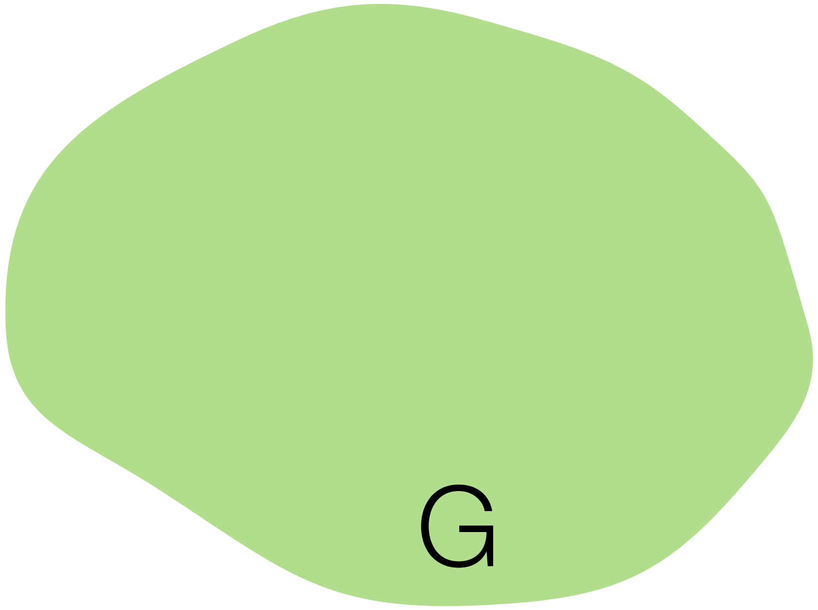

- Even when no good labeling exists, there might be a way to cut which does not cut the hypercubes at all.
- Reduction preserves structure of starting graph.
- If starting UG does not have expansion, neither does the new graph.

- For UGs obtained by reducing from Small-Set Expansion, can introduce a special folding operation, which does not preserve structure of UG-graph.
- Even though the graph for UG may not have expansion, the graph for Balanced Separator does.

# From SSE to Unique Games [RS 10]

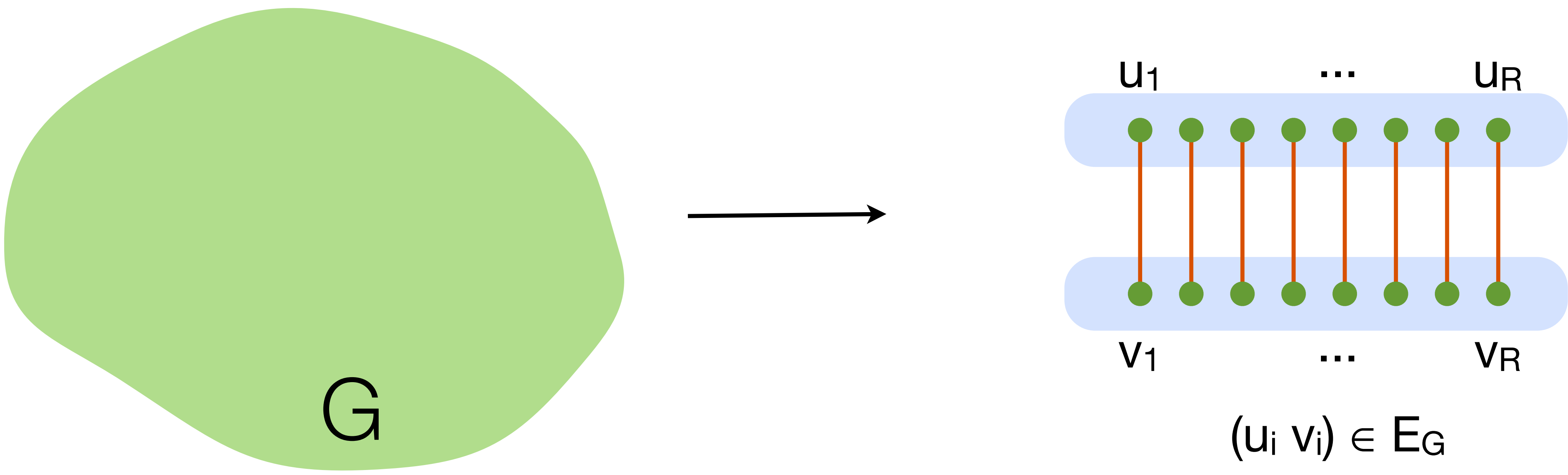


# From SSE to Unique Games [RS 10]



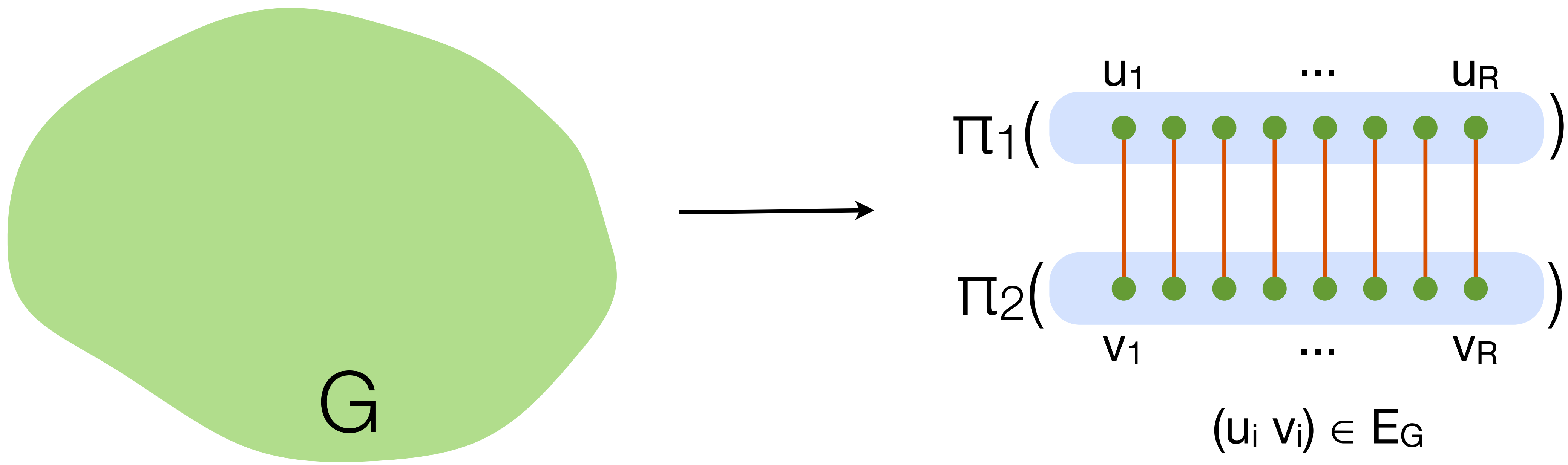

- UG is defined on $G^R$ and has alphabet size R.

# From SSE to Unique Games [RS 10]



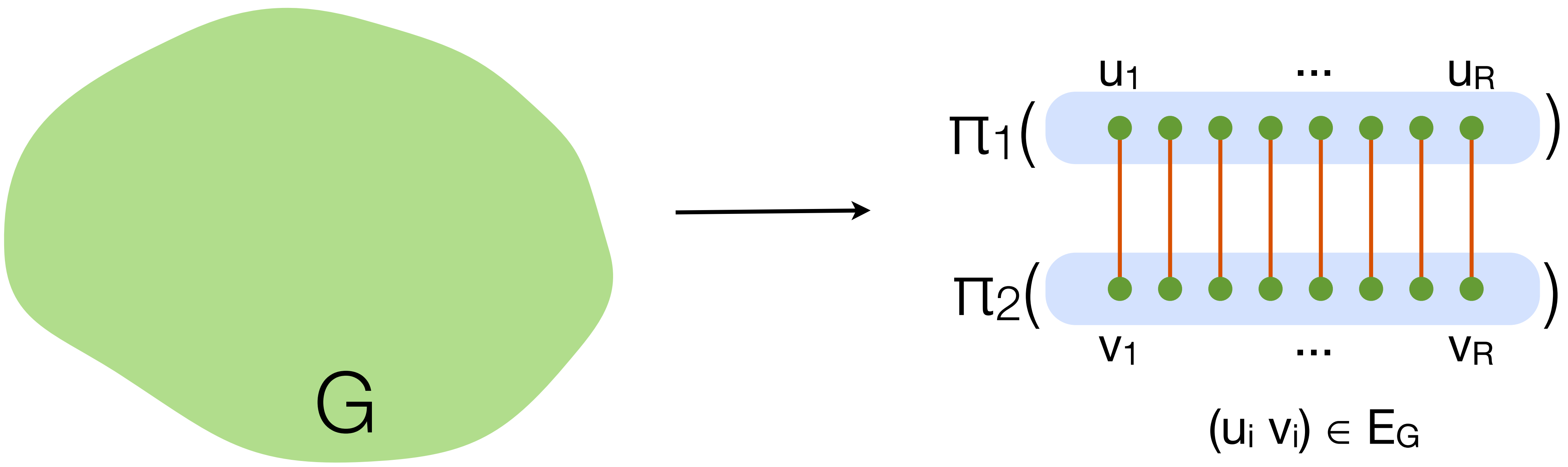

- UG is defined on $G^R$ and has alphabet size R.
- Pick $(A,B) \sim E_G{}^R$

# From SSE to Unique Games [RS 10]




- UG is defined on $G^R$ and has alphabet size R.
- Pick (A,B) ~ $E_G^R$
- Pick random permutations $\pi_1$, $\pi_2$ from [R] to [R]
- Connect $\pi_1$(A) to $\pi_2$(B). Call them u,v. (send $\pi_1$(A) to Alice, $\pi_2$(B) to Bob)

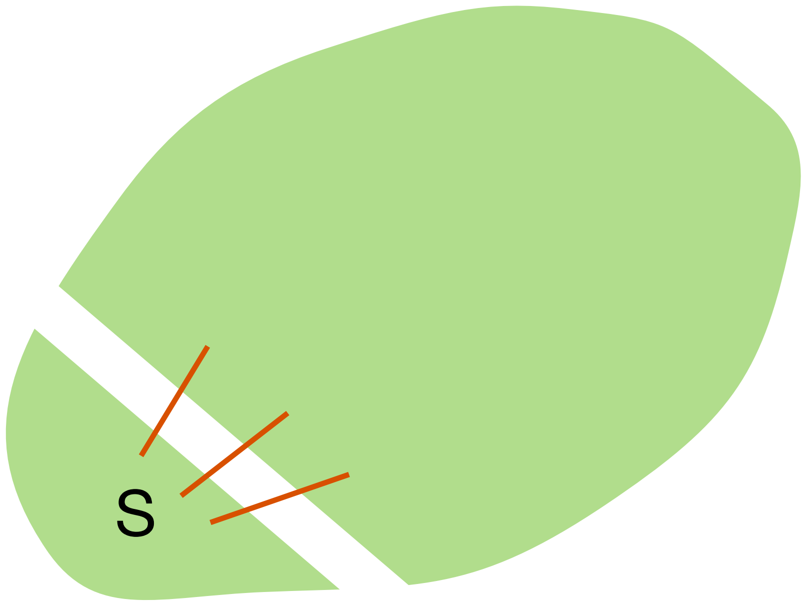
# From SSE to Unique Games [RS 10]



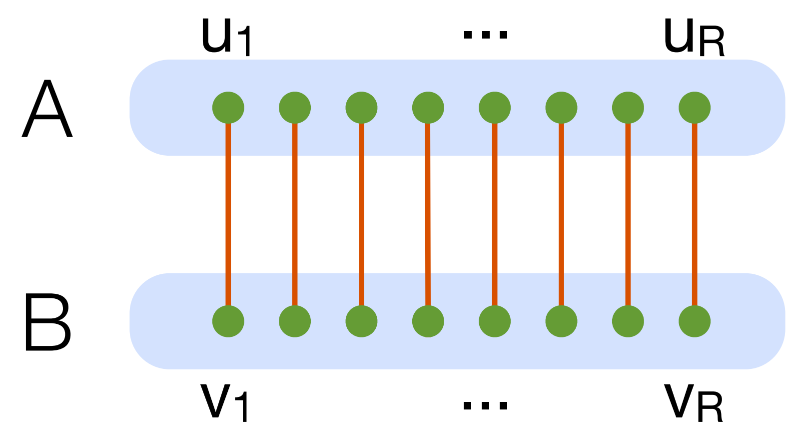

- UG is defined on $G^R$ and has alphabet size R.
- Pick (A,B) ~ $E_G{}^R$
- Pick random permutations $\pi_1$, $\pi_2$ from [R] to [R]
- Connect $\pi_1$(A) to $\pi_2$(B). Call them u,v. (send $\pi_1$(A) to Alice, $\pi_2$(B) to Bob)
- Labeling is required to find an edge in G. Constraint on (u,v) is

$$\pi_1^{-1}(L(u)) = \pi_2^{-1}(L(v))$$

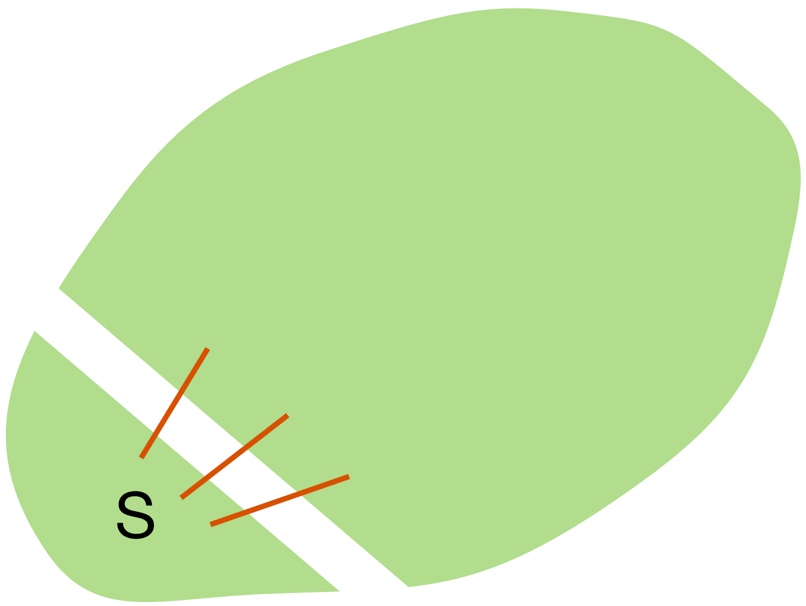
# Why the SSE reduction works



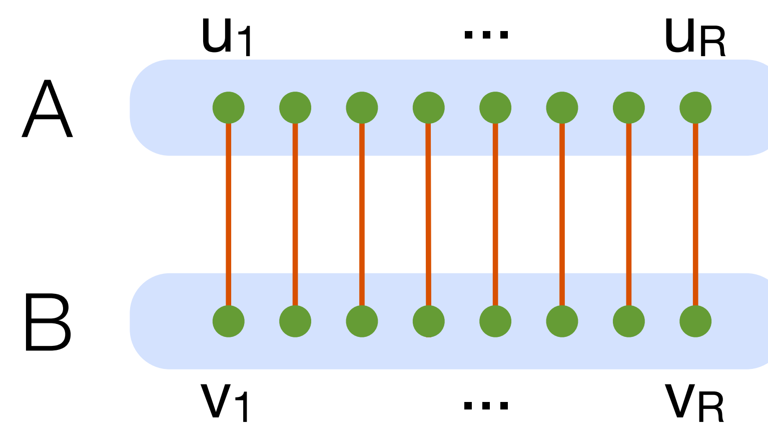

$\mu(S) = 1/R, \;\; \Phi_G(S) \leq \eta$



# Why the SSE reduction works

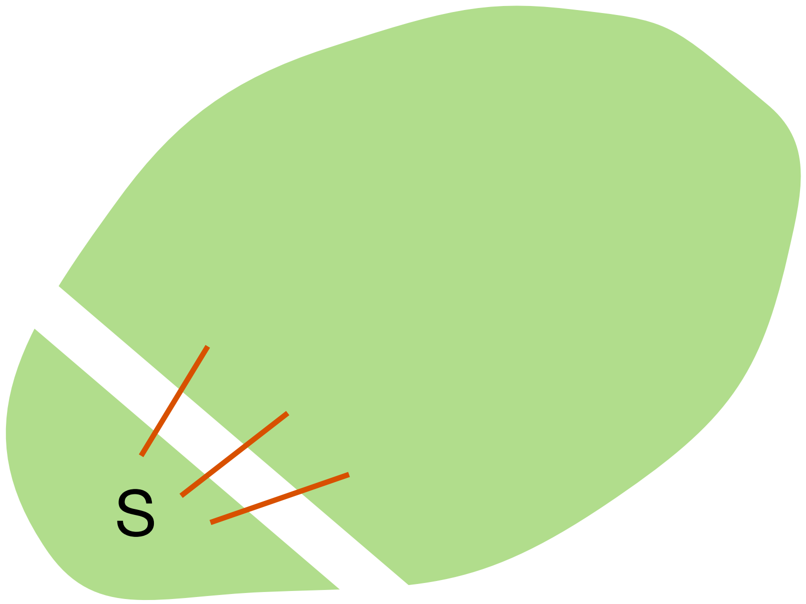



$\mu(S) = 1/R, \;\; \Phi_G(S) \leq \eta$

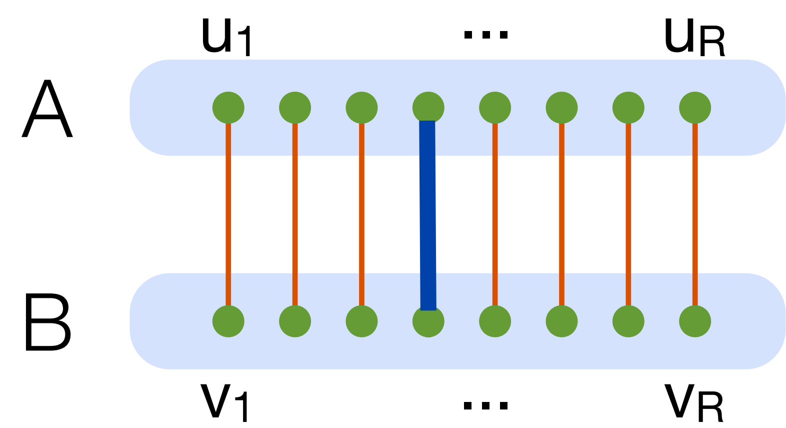



- P(A has exactly one S-vertex) = $(1 - 1/R)^{R-1} \approx 1/e$

# Why the SSE reduction works

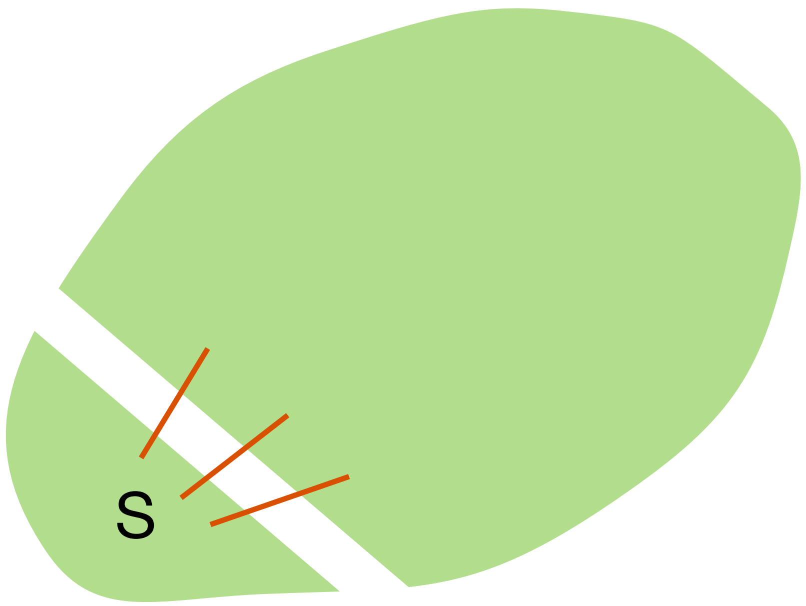



$\mu(S) = 1/R, \ \Phi_G(S) \leq \eta$

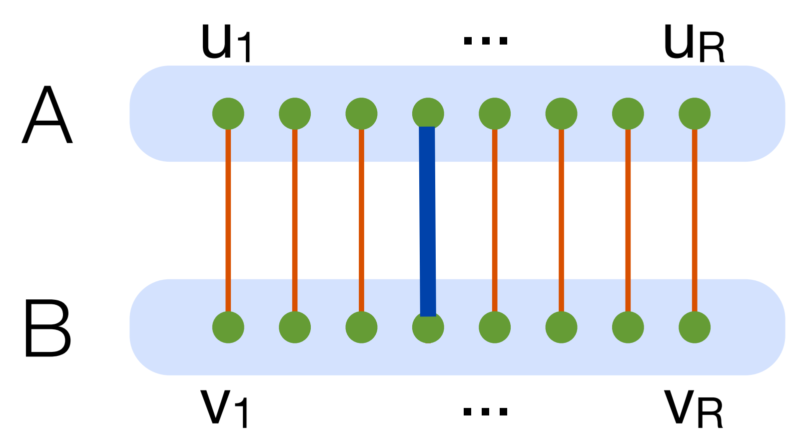



- P(A has exactly one S-vertex) = $(1 - 1/R)^{R-1} \approx 1/e$
- If $u_i$ is in S, so is $v_i$ (with probability $1-\eta$)

# Why the SSE reduction works

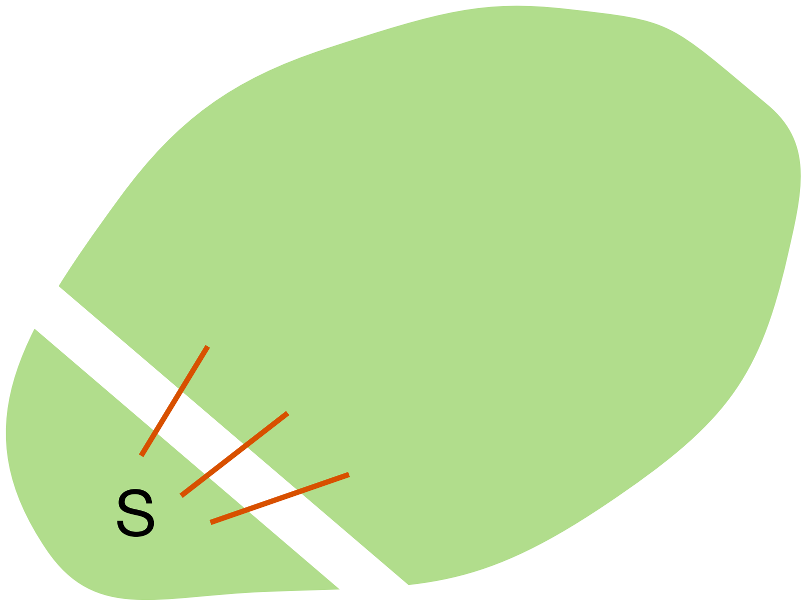



$\mu(S) = 1/R, \;\; \Phi_G(S) \le \eta$

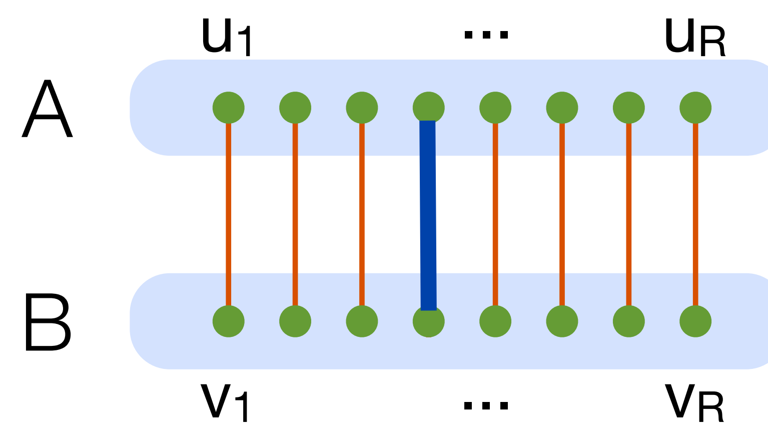



- P(A has exactly one S-vertex) = $(1 - 1/R)^{R-1} \approx 1/e$
- If $u_i$ is in S, so is $v_i$ (with probability $1-\eta$)
- $L(\pi_1(A))$ = index of S-vertex in $\pi_1(A)$
  $L(\pi_2(B))$ = index of S-vertex in $\pi_2(B)$

# Why the SSE reduction works

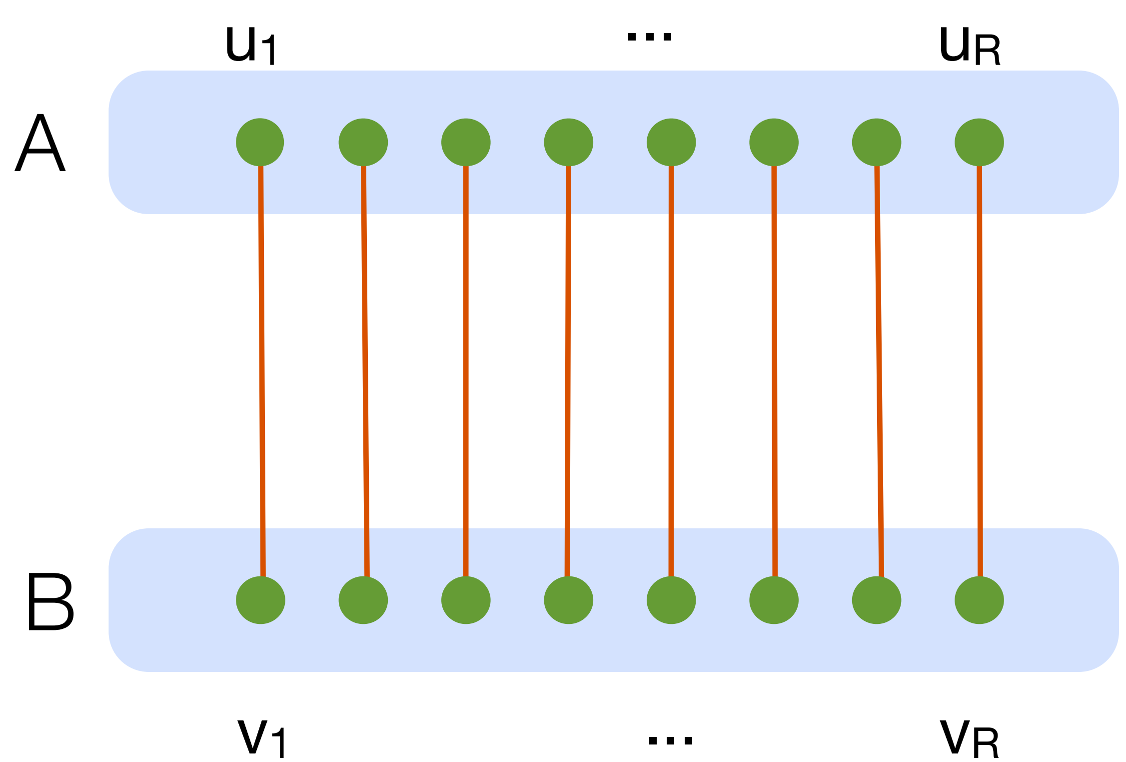



$\mu(S) = 1/R, \ \Phi_G(S) \le \eta$

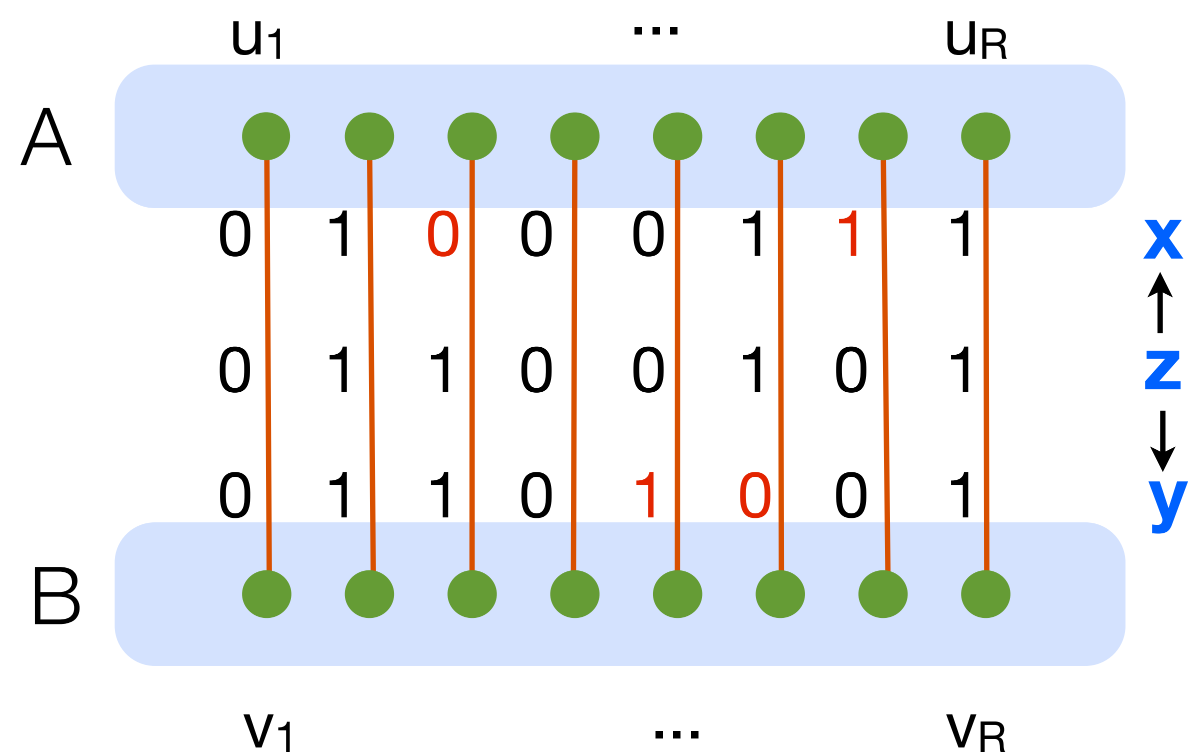



- P(A has exactly one S-vertex) = $(1 - 1/R)^{R-1} \approx 1/e$
- If $u_i$ is in S, so is $v_i$ (with probability $1-\eta$)
- $L(\pi_1(A))$ = index of S-vertex in $\pi_1(A)$
  $L(\pi_2(B))$ = index of S-vertex in $\pi_2(B)$
- Works with probability $(1-\eta)/e$

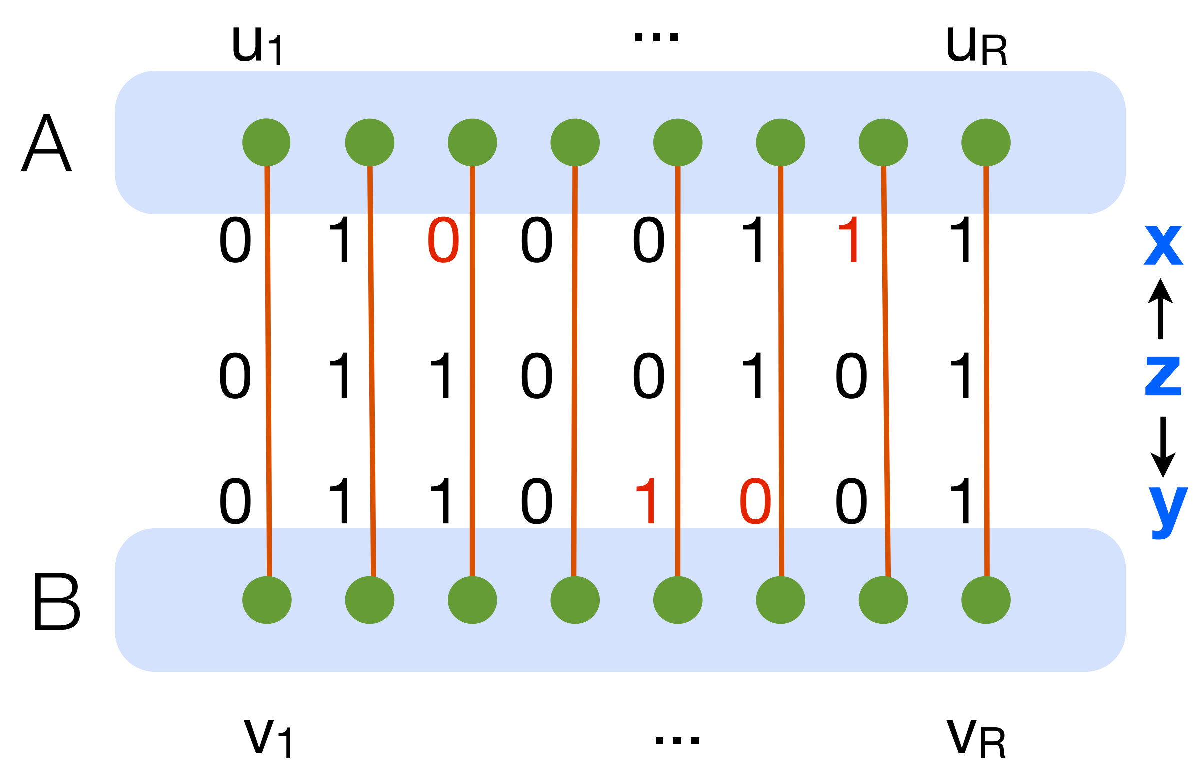
# Combining the two reductions



# Combining the two reductions



# Combining the two reductions

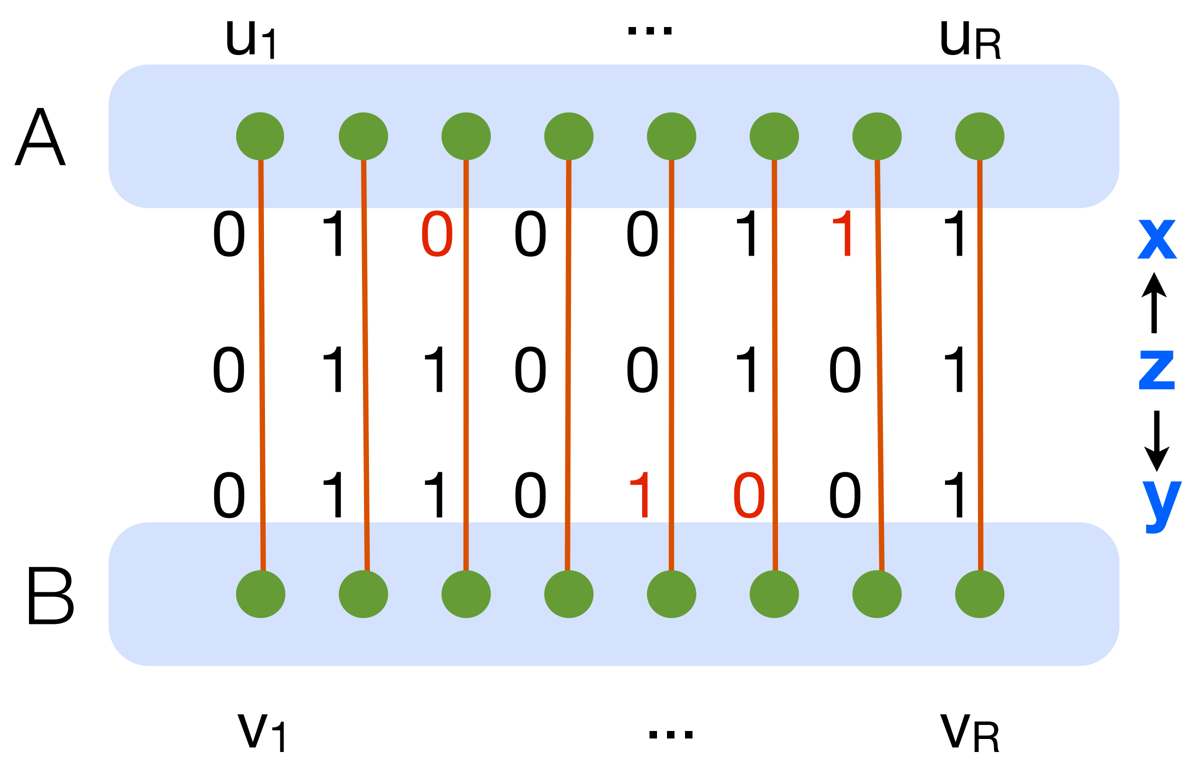



Pick random $z \in \{0,1\}^R$

Generate x, y from z using independent $\epsilon$-noise.

Connect $(\pi_1(A), \pi_1(x))$ to $(\pi_2(B), \pi_2(y))$.

# Combining the two reductions



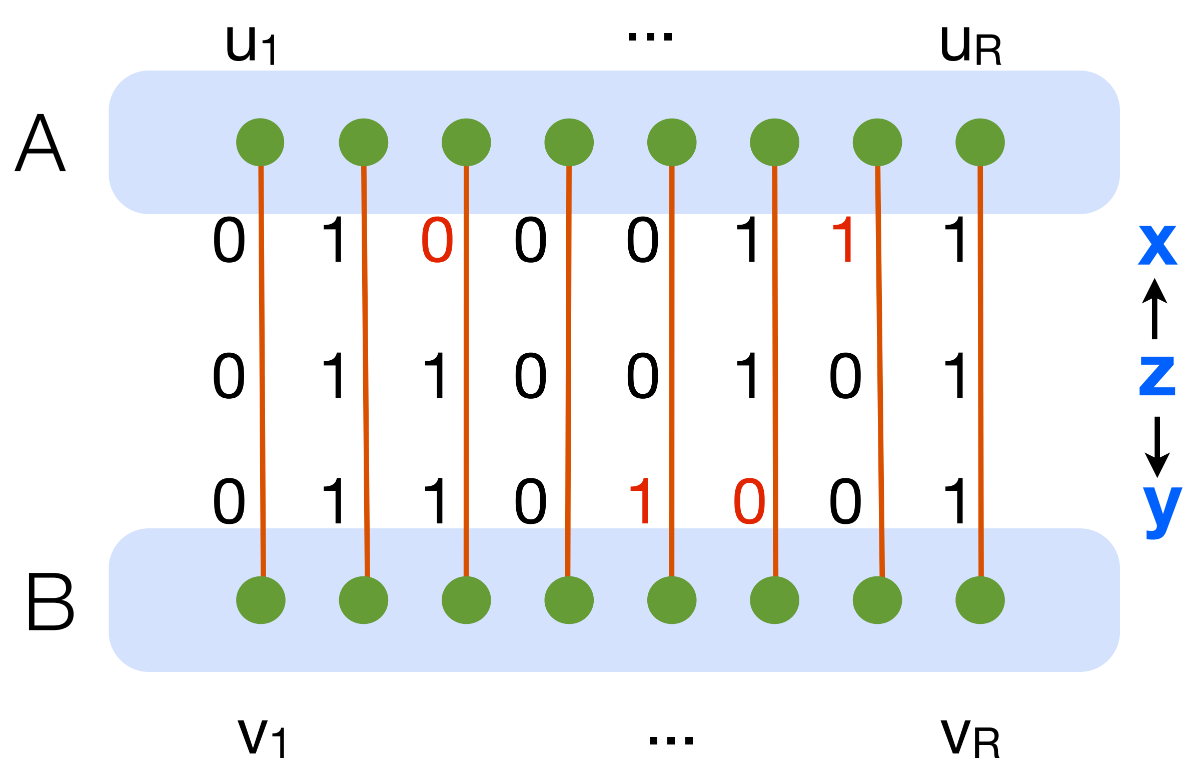

Pick random $z \in \{0,1\}^R$

Generate x, y from z using independent $\epsilon$-noise.

Connect $(\pi_1(A), \pi_1(x))$ to $(\pi_2(B), \pi_2(y))$.

- Think of bits as picking out subsets of A and B.

# Combining the two reductions



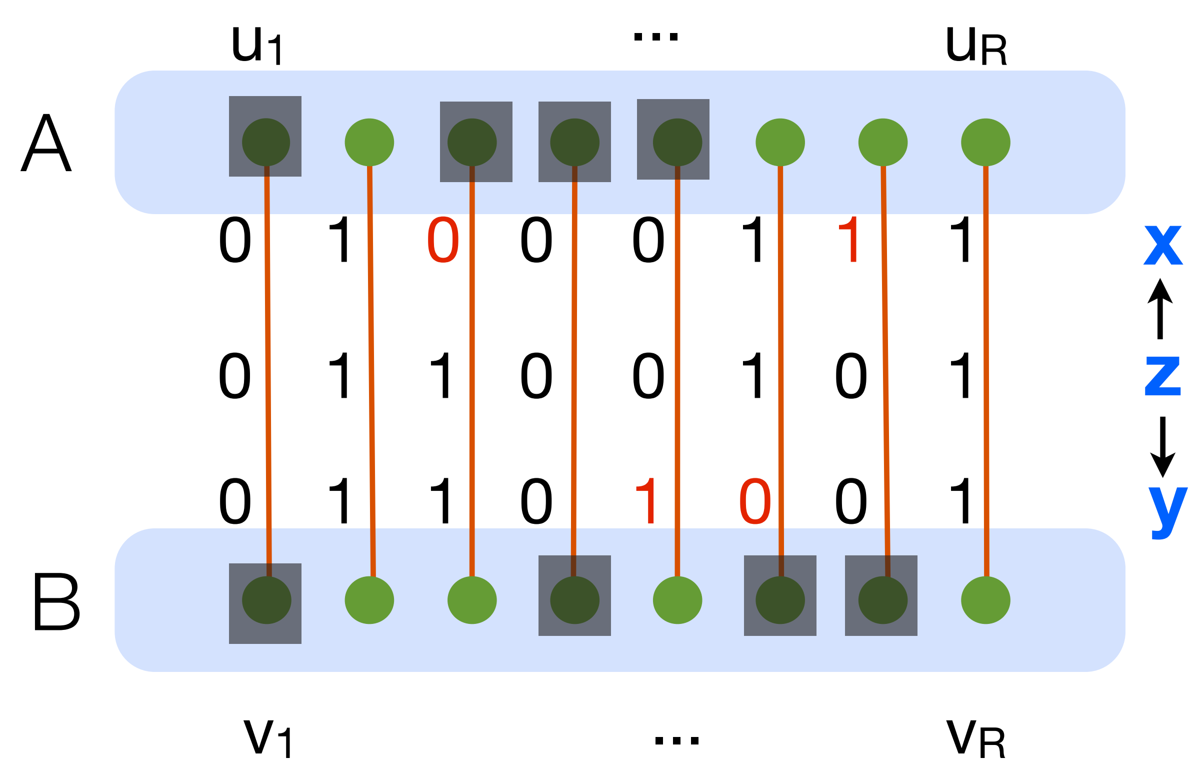

Pick random $z \in \{0,1\}^R$

Generate x, y from z using independent $\epsilon$-noise.

Connect $(\pi_1(A), \pi_1(x))$ to $(\pi_2(B), \pi_2(y))$.

- Think of bits as picking out subsets of A and B.
- Modification: Only send these subsets. Hide elements of A (or B) in positions where x (resp. y) are 0 (send $\perp$ instead).

# Combining the two reductions




Pick random $z \in \{0,1\}^R$

Generate x, y from z using independent $\epsilon$-noise.

Connect $(\pi_1(A), \pi_1(x))$ to $(\pi_2(B), \pi_2(y))$.

- Think of bits as picking out subsets of A and B.
- Modification: Only send these subsets. Hide elements of A (or B) in positions where x (resp. y) are 0 (send $\perp$ instead).
- Corresponds to a folding operation on the graph obtained after reduction.

# Is that allowed?

# Is that allowed?

- Alice and Bob still get tuples of size ≈ R/2. Each tuple still contains exactly one S-vertex with constant probability (≈ $1/2e^{1/2}$).

# Is that allowed?

- Alice and Bob still get tuples of size ≈ R/2. Each tuple still contains exactly one S-vertex with constant probability (≈ $1/2e^{1/2}$).
- If A contains an S-vertex which is not deleted, then so does B with probability 1-2ϵ.

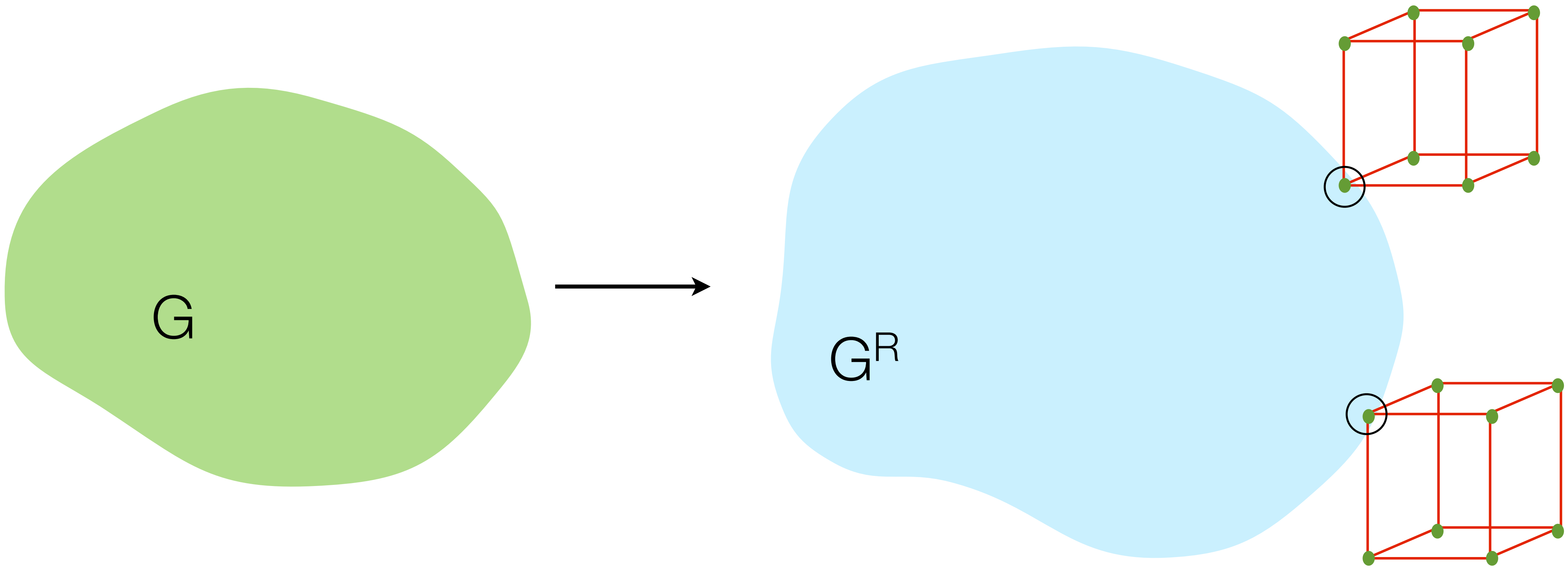
# Is that allowed?

- Alice and Bob still get tuples of size ≈ R/2. Each tuple still contains exactly one S-vertex with constant probability (≈ $1/2e^{1/2}$).
- If A contains an S-vertex which is not deleted, then so does B with probability 1-2ϵ.
- Can still play the corresponding Unique Game. Also, balanced cut with expansion O(ϵ + η) by choosing all tuples which have a unique S-vertex.

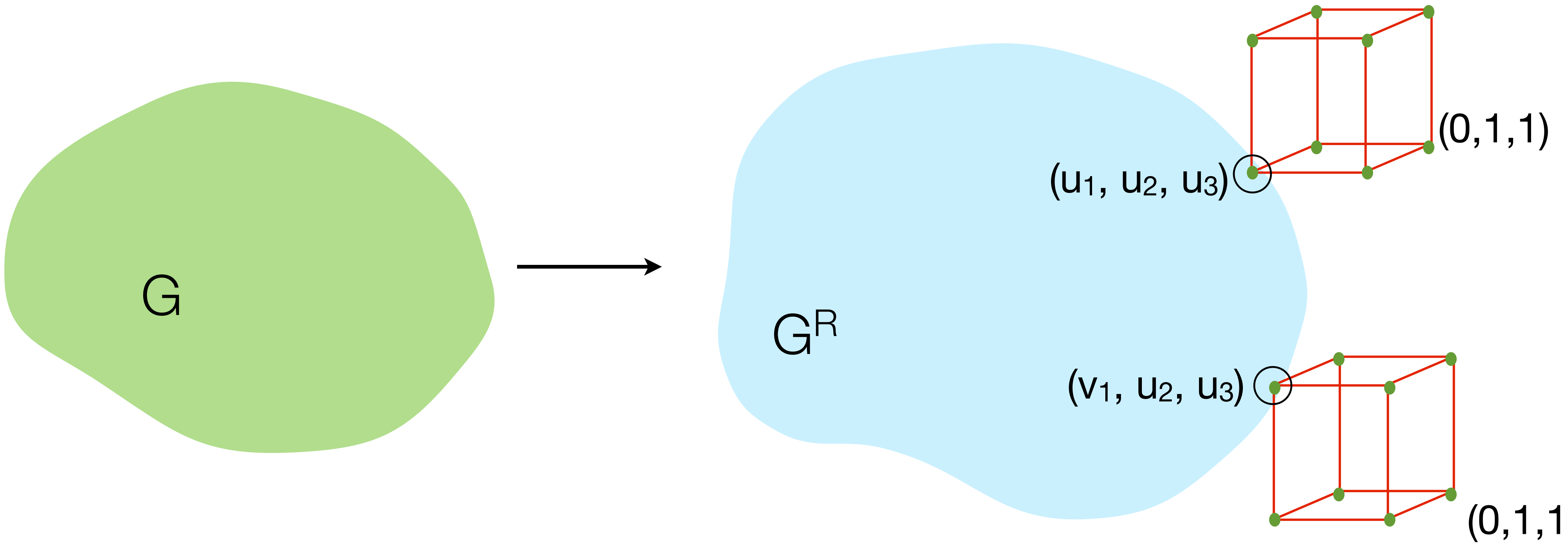
# Is that allowed?

- Alice and Bob still get tuples of size ≈ R/2. Each tuple still contains exactly one S-vertex with constant probability (≈ $1/2e^{1/2}$).
- If A contains an S-vertex which is not deleted, then so does B with probability 1-2ϵ.
- Can still play the corresponding Unique Game. Also, balanced cut with expansion O(ϵ + η) by choosing all tuples which have a unique S-vertex.
- Modification only uses the fact that the strategy is about membership in a set.

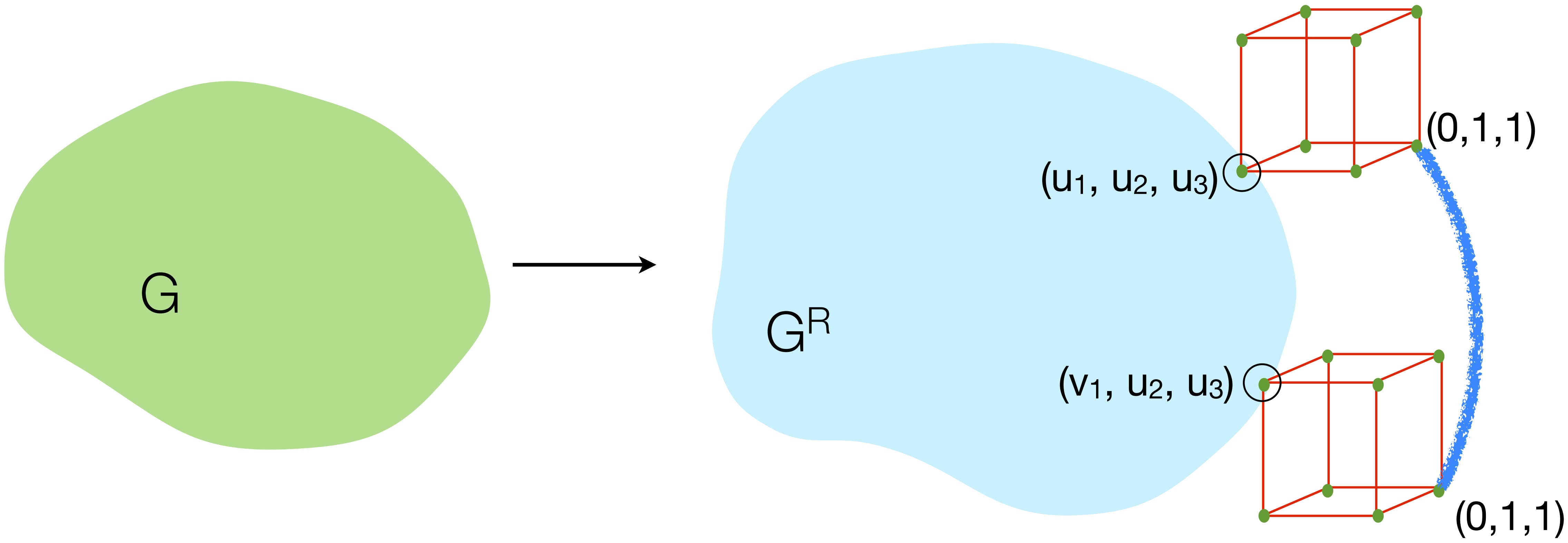
# The resulting graph



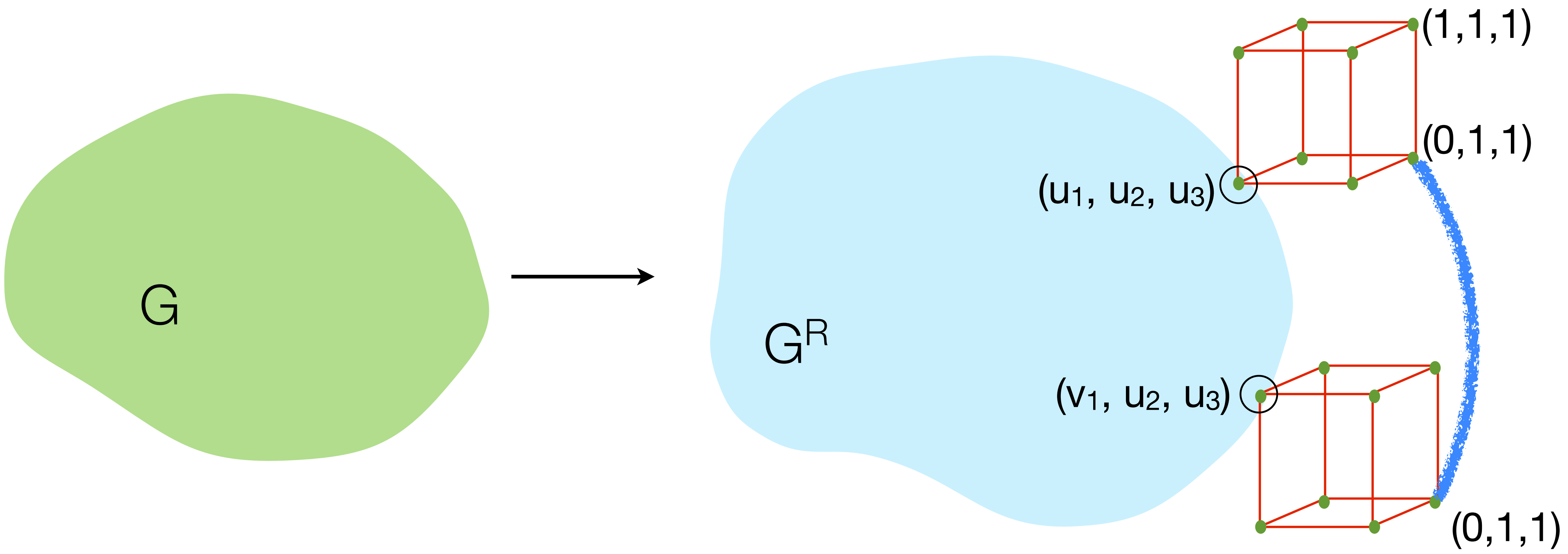
# The resulting graph



# The resulting graph

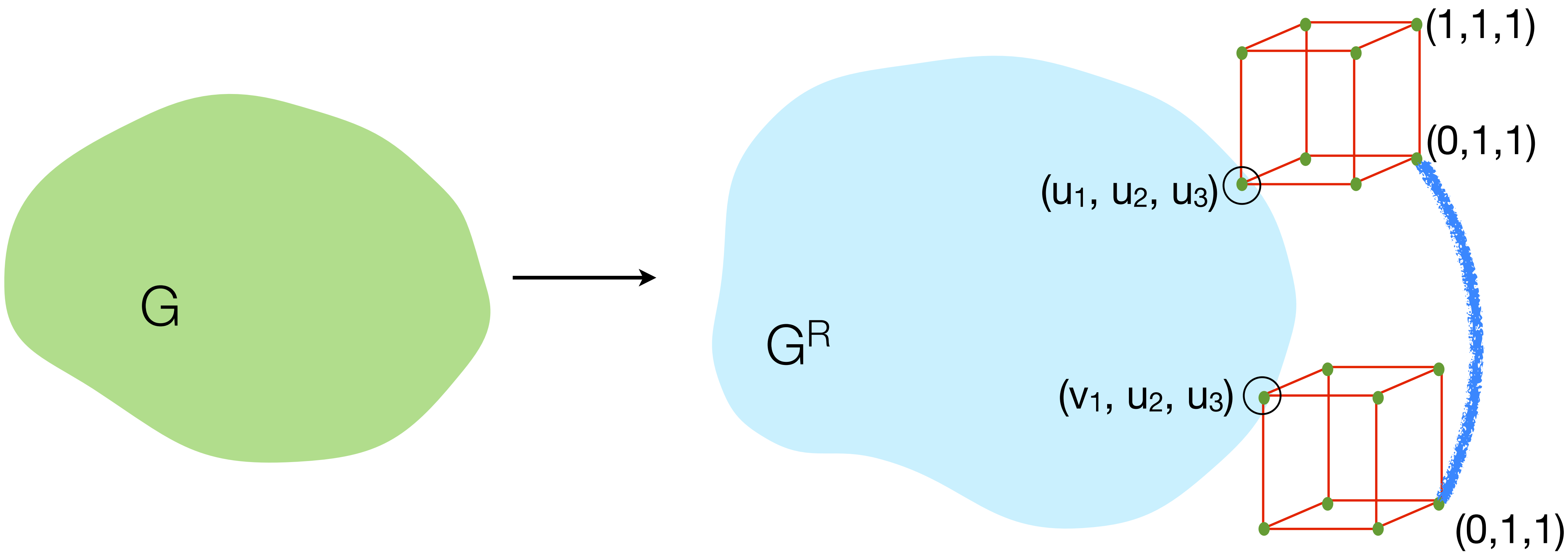


# The resulting graph



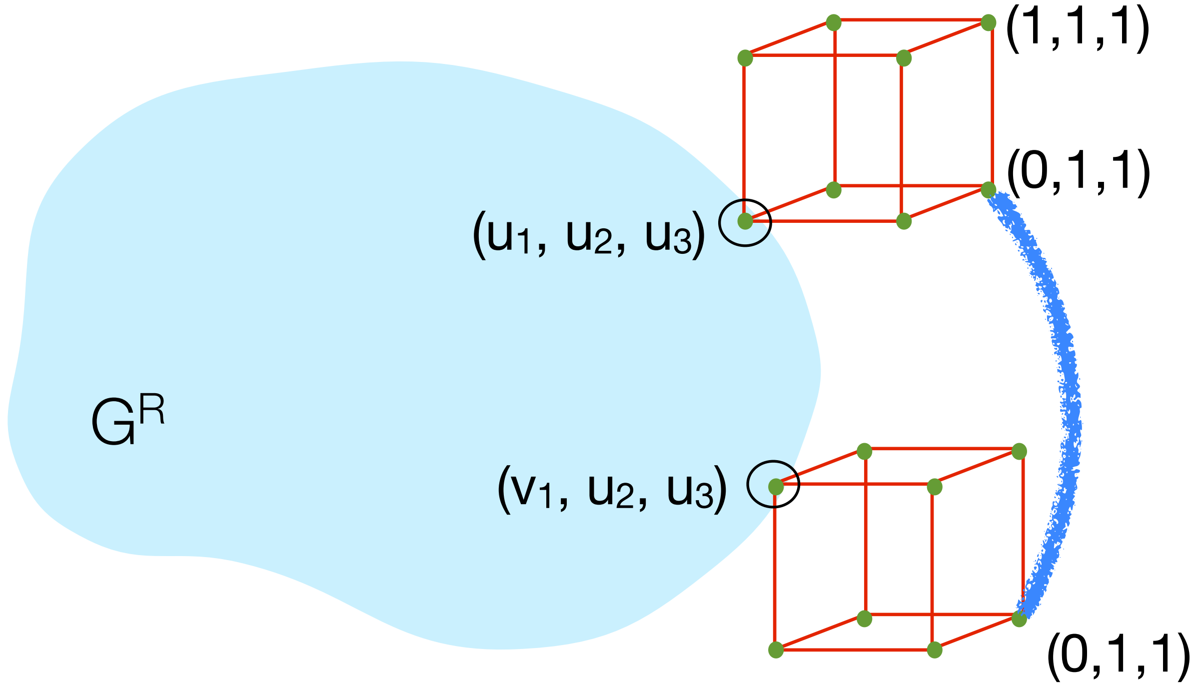

- $((u_1, u_2, u_3), (0,1,1))$ and $((v_1, u_2, u_3), (0,1,1))$ both become $((\perp, u_2, u_3), (0,1,1))$. Not so for (1,1,1).

# The resulting graph

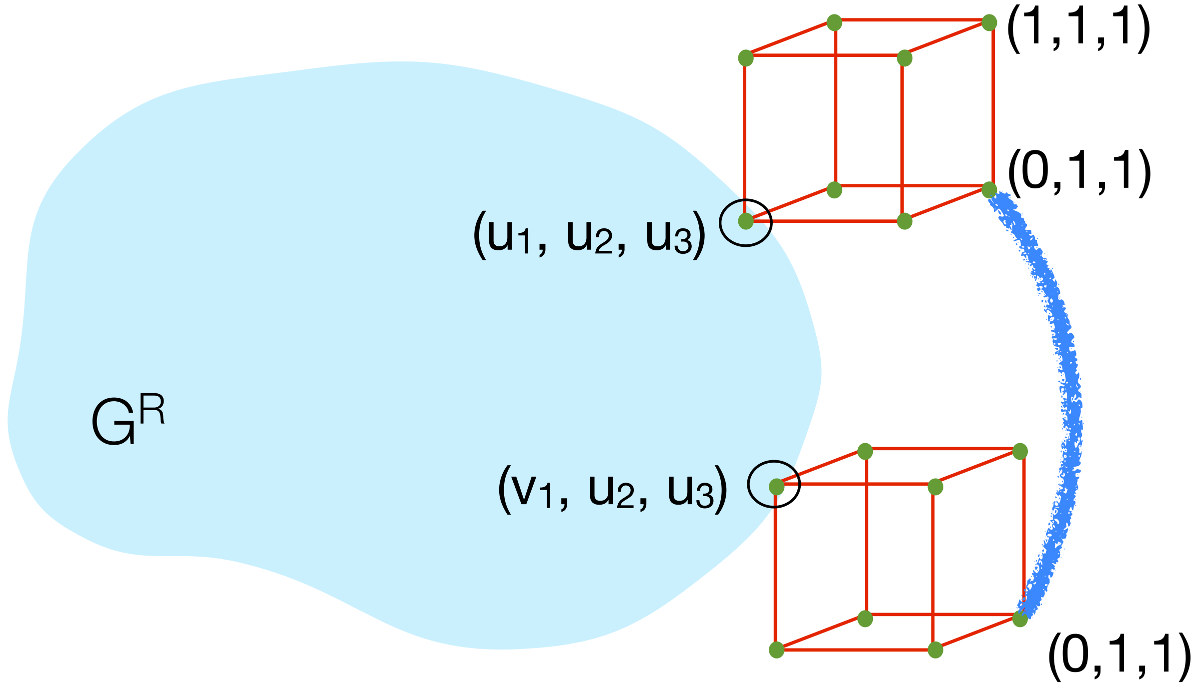



- $((u_1, u_2, u_3), (0,1,1))$ and $((v_1, u_2, u_3), (0,1,1))$ both become $((\perp, u_2, u_3), (0,1,1))$. Not so for (1,1,1).
- Completely destroys the gadget-based nature of the reduction.

# Why does it work?

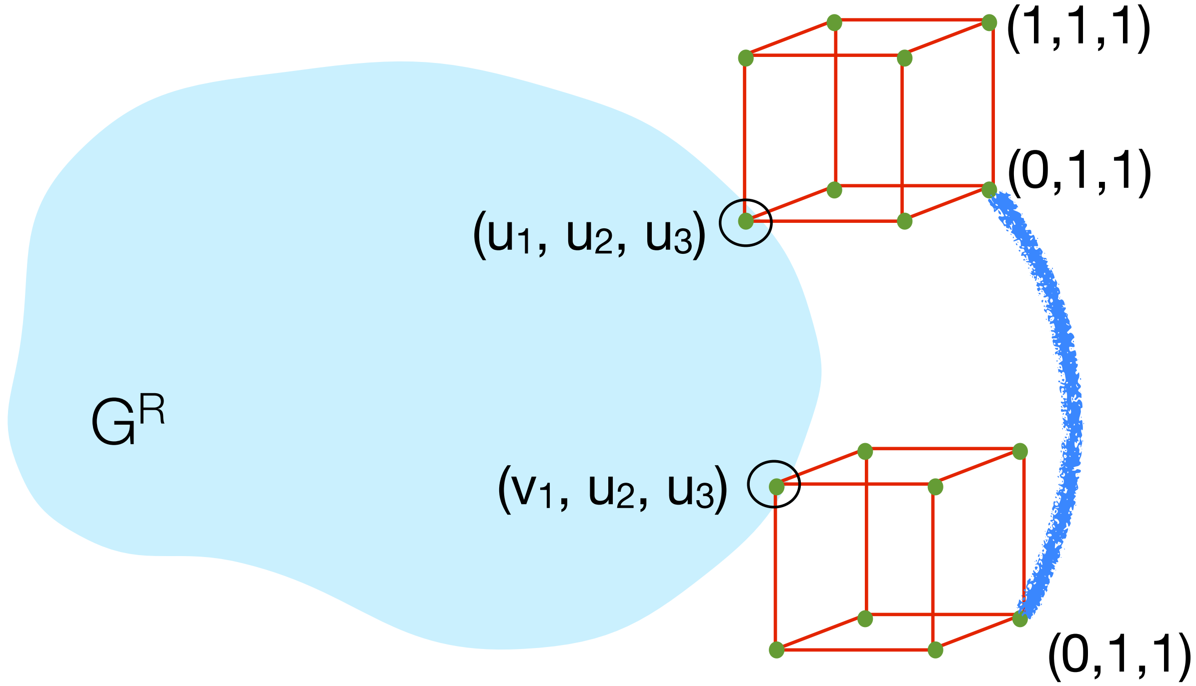


# Why does it work?




- Cuts in the new graph correspond to cuts in gadget-based graph, which are invariant under folding.

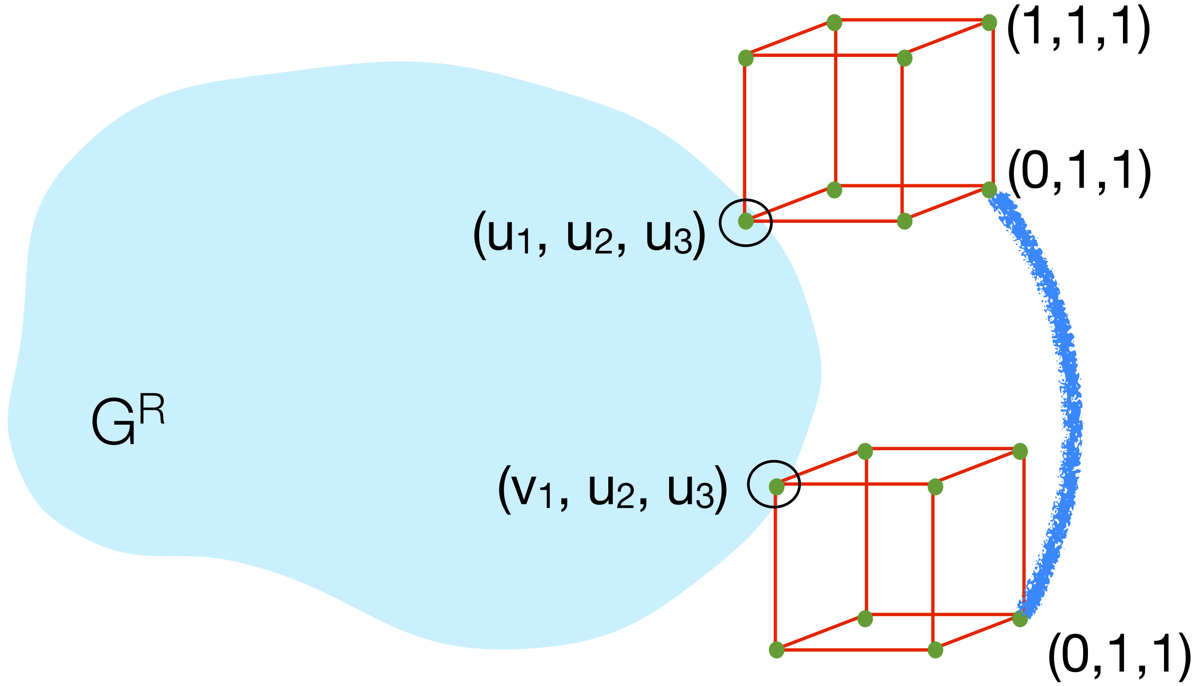
# Why does it work?




- Cuts in the new graph correspond to cuts in gadget-based graph, which are invariant under folding.

- Show that any such cut must have large variance in most cubes i.e. cannot be a cut of the underlying graph.

# Why does it work?




- Cuts in the new graph correspond to cuts in gadget-based graph, which are invariant under folding.

- Show that any such cut must have large variance in most cubes i.e. cannot be a cut of the underlying graph.

- Obtain full result by using q-ary cubes and more general folding.

# Conclusions

# Conclusions

- Something fishy. The fact that underlying problem is about expansion is used very indirectly (in correctness of UG reduction). The calculation that cuts must have large variance in cubes doesn't use it at all.

# Conclusions

- Something fishy. The fact that underlying problem is about expansion is used very indirectly (in correctness of UG reduction). The calculation that cuts must have large variance in cubes doesn't use it at all.

- Can the folding operation be generalized to other classes of Unique Games where deleting part of the question still preserves a Unique Game.

# Conclusions

- Something fishy. The fact that underlying problem is about expansion is used very indirectly (in correctness of UG reduction). The calculation that cuts must have large variance in cubes doesn't use it at all.

- Can the folding operation be generalized to other classes of Unique Games where deleting part of the question still preserves a Unique Game.

- Reduction between scales in the other direction (Balanced Separator to SSE)?

Thank You

Questions?