On the Parameterized Complexity of Grid Contraction

Saket Saurabh, Uverton Dos Souza, and
Prafullkumar Tale

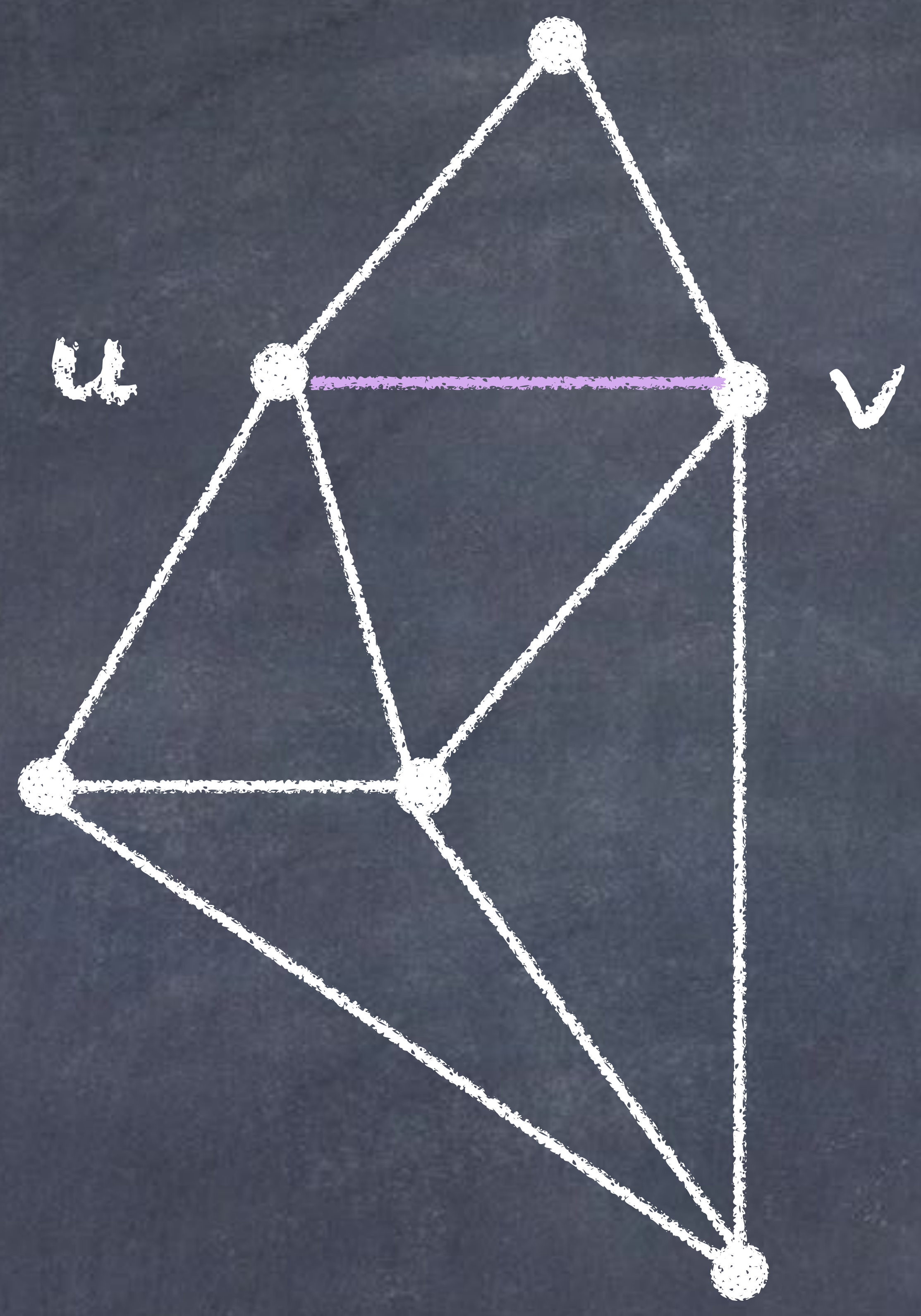
Graph Modification Problems

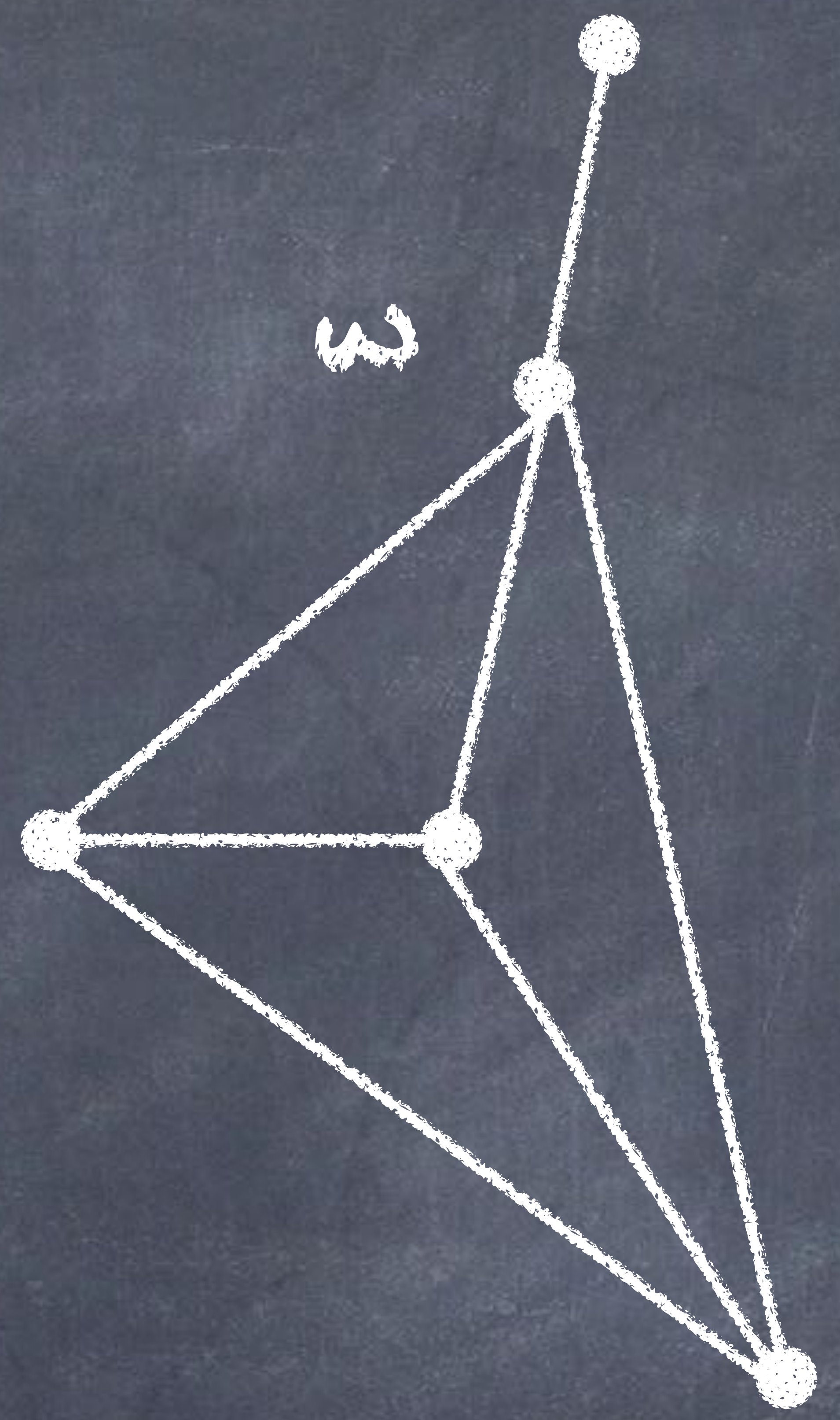
$\mathcal{G}$ – Graph class (poly time recognisable)

Input: Graph G, int k

Question: Can we make at most k modifications in G s.t. resulting graph is in $\mathcal{G}$?

Modification : (1) vertex deletion (2) edge deletion (3) edge addition (4) edge contraction





Edge contraction

No self-loop
No parallel edges

$\mathcal{G}$ – Contraction

Input: Graph G, int k

Question: Can we contract at most k edges in G s.t. resulting graph is in $\mathcal{G}$?

Parameterized Complexity and Known Results

Our Results

Overview of FPT Algorithm

Open Questions

Parameterized Complexity

- Instance + relevant secondary measure (i.e. parameter)

  Ex. For $\mathcal{G}$-Contraction Problem
  Instance: (G, k)
  Parameter: k

- Objective : Find an algorithm f(k) poly(n)

  f(k) – function depending only on k
  k – collection of parameters

- Fixed Parameter Tractable (FPT): Class of problems that can be solved in $f(k)\,\mathrm{poly}(n)$.

  n – size of input
  k – parameter

- Such algorithm may not exist for every problem + selected parameter W[i]-Hardness

Redefining FPT : Kernelization

Redefining FPT : Kernelization

Problem admits a kernel of size $g(k)$ if there exists an poly-time algorithm that

– on input (G, k) produces output (G', k') s.t. $|G'| + k' <= g(k)$

– (G, k) is a Yes instance if and only if (G', k') is.

$\boxed{\text{Theorem : Problem is in FPT iff it admits a kernel.}}$

Q. If a problem is in FPT, does it admit a polynomial kernel?

$\mathcal{G}$ – Contraction: FPT

Heggernes + H.L.L.P. (2011):	Path | Tree	exp(k)
Golovach + K.P.T. (2011):	min-deg >= 14	exp(k)
Heggernes + H.L.P. (2011):	Bipartite	exp(exp(k))
Golovach + H.P. (2012):	Planar	f(k)
Guillemont & Marx (2013):	Bipartite	exp(k^2)
Belmonte + G.H.P.(2013):	max-deg <= 2	exp(k log(k))
Guo & Cai (2015):	Clique	exp(k log k)

$\mathcal{G}$ – Contraction: Kernel

Heggernes + H.L.L.P. (2011):	Path	k^2
	Tree	no-poly
	min-deg >= 14	??
	Bipartite	??
	Planar	??
Belmonte + G.H.P.(2013):	max-deg <= 2	k^2
Guo & Cai (2015):	Clique	no-poly

$\mathcal{G}$ – Contraction: W[i]-Hardness

Golovach + K.P.T.(2011):	min-deg >= d	W[1] (only k)
Belmonte + G.H.P.(2013):	max-deg <= d	W[2] (only k)
Cai & Gua(2013) + Lokshtanov + M.S.(2013):	Chordal	W[2]
Agrawal + L.S.Z.(2017):	Split	W[1]

Comparing Graph Modification problems.

(1) vertex deletion
(2) edge deletion
(3) edge addition
(4) edge contraction

Edge Contraction problems are harder than corresponding vertex/edge deletion/addition problems.

Edge Contraction problems are harder than corresponding vertex/edge deletion/addition problems.

Ex. (1) Target Graph Class: {P_4}

Q1 : Delete vertices in G to get P_4.

(Find induced P_4) Polynomial time

Q2 : Contract edges in G to get P_4.

NP-Hard

Edge Contraction problems are harder than corresponding vertex/edge deletion/addition problems.

Ex. (2) Target Graph Class: Chordal Graphs

Q1 : Add at most k edges in G to get a chordal graph.

(Minimum Fill-In) FPT

Q2 : Contract at most k edges in G to get a chordal graph.

W[2]-Hard

Edge Contraction problems are harder than corresponding vertex/edge deletion/addition problems.

Ex. (3) Target Graph Class: Acyclic Graphs

Q1 : Delete at most k vertices to make G acyclic.

(Feedback Vertex Set) k^2 kernel

Q2 : Contract at most k edges to make G acyclic.

No-poly kernel

$\boxed{\text{Parameterized Complexity and Known Results}}$

Our Results

Overview of FPT Algorithm

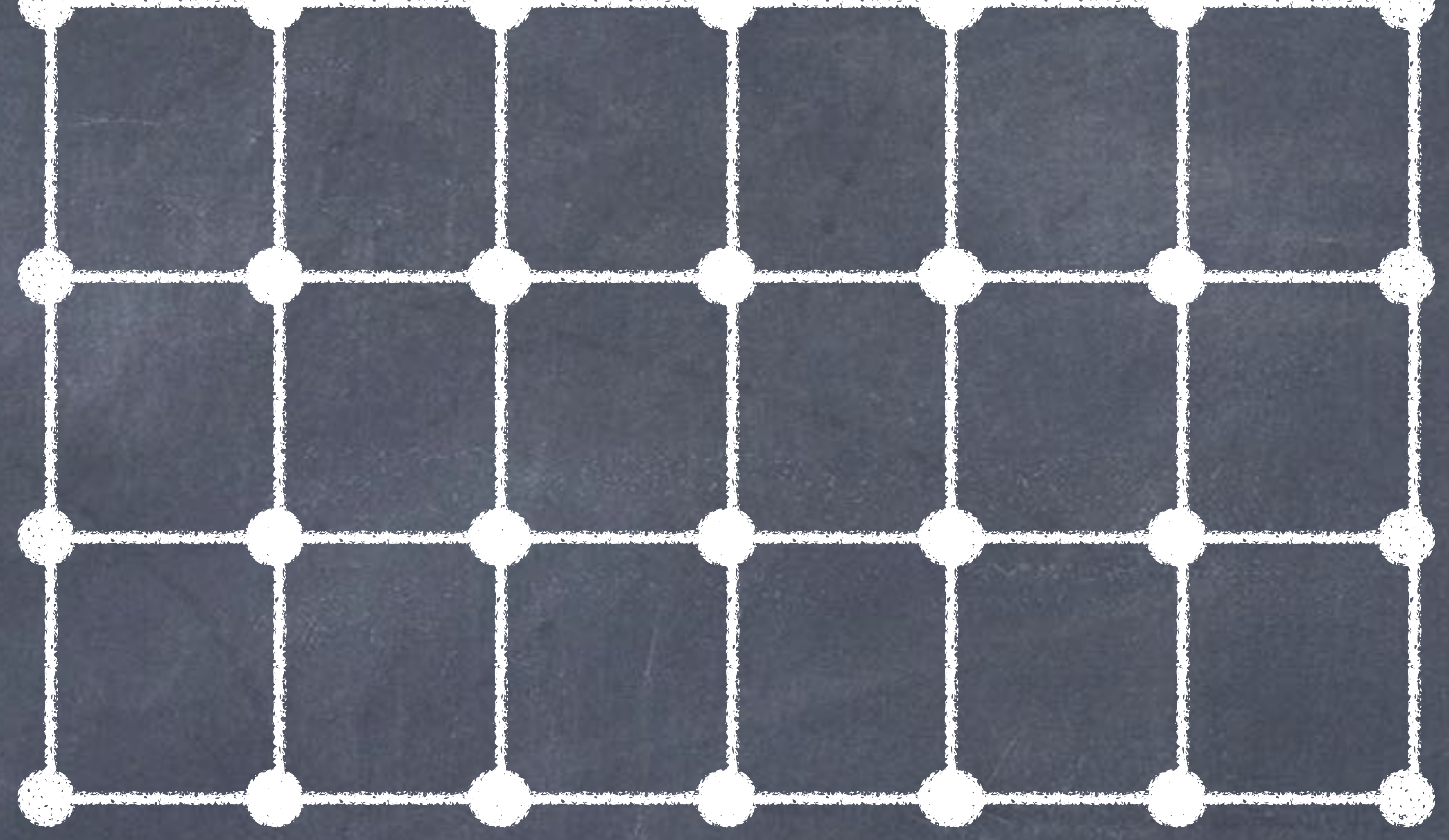
Open Questions

(r x q)Grid

- Graph on r * q vertices

- Vertex v := [i, j]

  1 <= i <= r; 1 <= j <= q



(4 x 7) - Grid

- Edge (v_1, v_2) iff |i1 - i2| + |j1 - j2| = 1

Grid Contraction

Input : Graph G, integer k

Parameter : k

Question : Does there exist a set F of at most k edges s.t. G/F is a grid?

Our Results: Grid Contraction

(R1) is NP-Hard

- does not admit an algo running in time 2^{o(k)} under ETH

(R2) admits an FPT algo running in time 4^{6k} * n^c

(R3) admits a kernel with O(k^4) vertices and edges.

Parameterized Complexity and Known Results

$\boxed{\text{Our Results}}$

Overview of FPT Algorithm

Open Questions

Grid Contraction

Input : Graph G, integers k

Parameter : k

Question : Does there exist a set F of at most k edges s.t. G/F is a grid ?

Bounded Grid Contraction

Input : Graph G, integers k, r

Parameter : $k + r$

Question : Does there exist a set F of at most k edges s.t. G/F is a grid with r rows ?

(G, k) – Yes inst. of Grid Contr.

iff (G, k, r) – Yes inst. of Bounded Grid Contr.

for r in $\{1, 2, \ldots, |V(G)|\}$

FPT Algorithms: Two Phases

Phase (I):

Bounded Grid Contraction is FPT
when parameterised by $k + r$.

Phase (II):

(G, k) – Yes inst. of Grid Contr.

iff (G, k, r) – Yes inst. of Bounded Grid Contr.

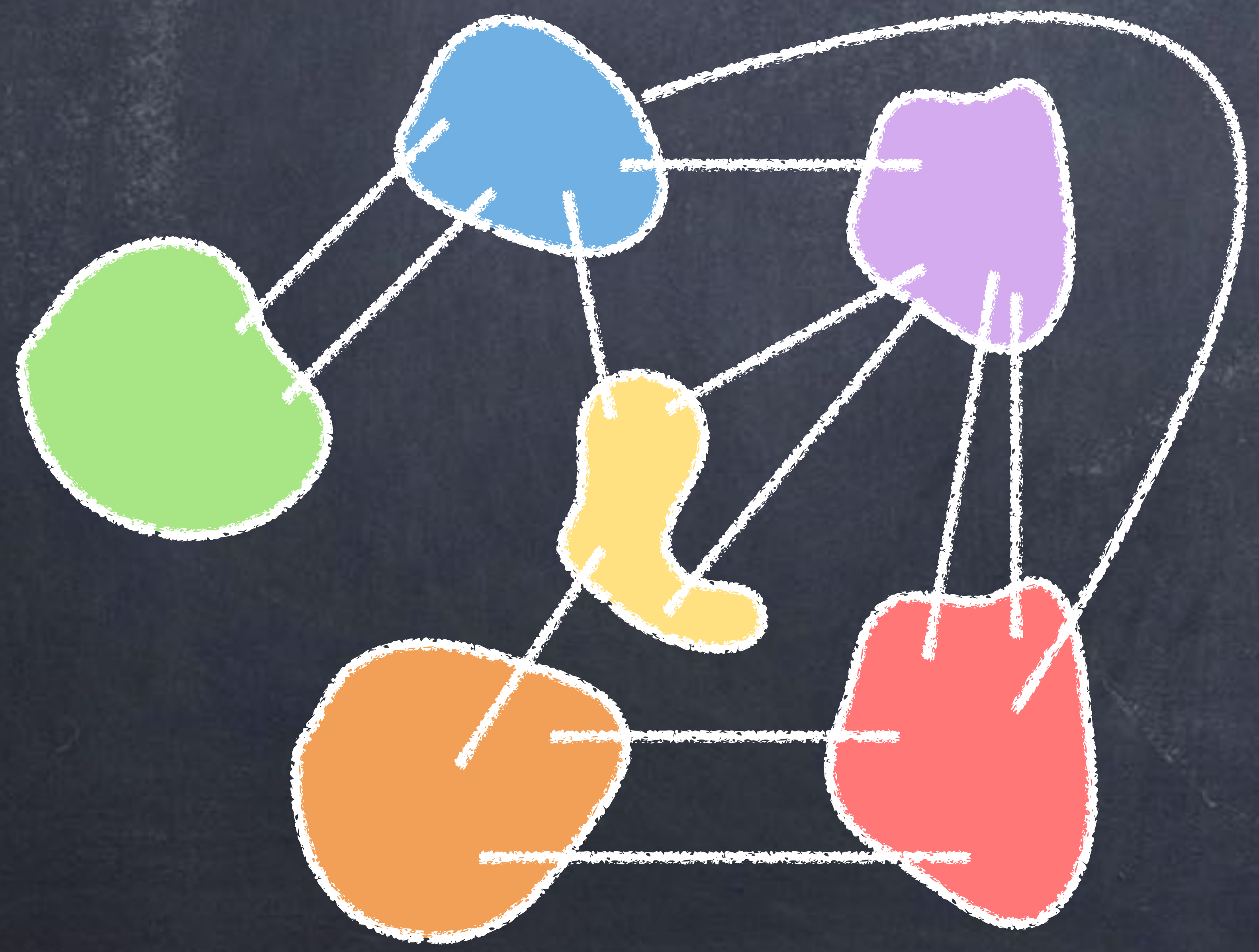
for r in $\{1, 2, \ldots, 2k + 5\}$

Edge contraction :: Graph Partitioning

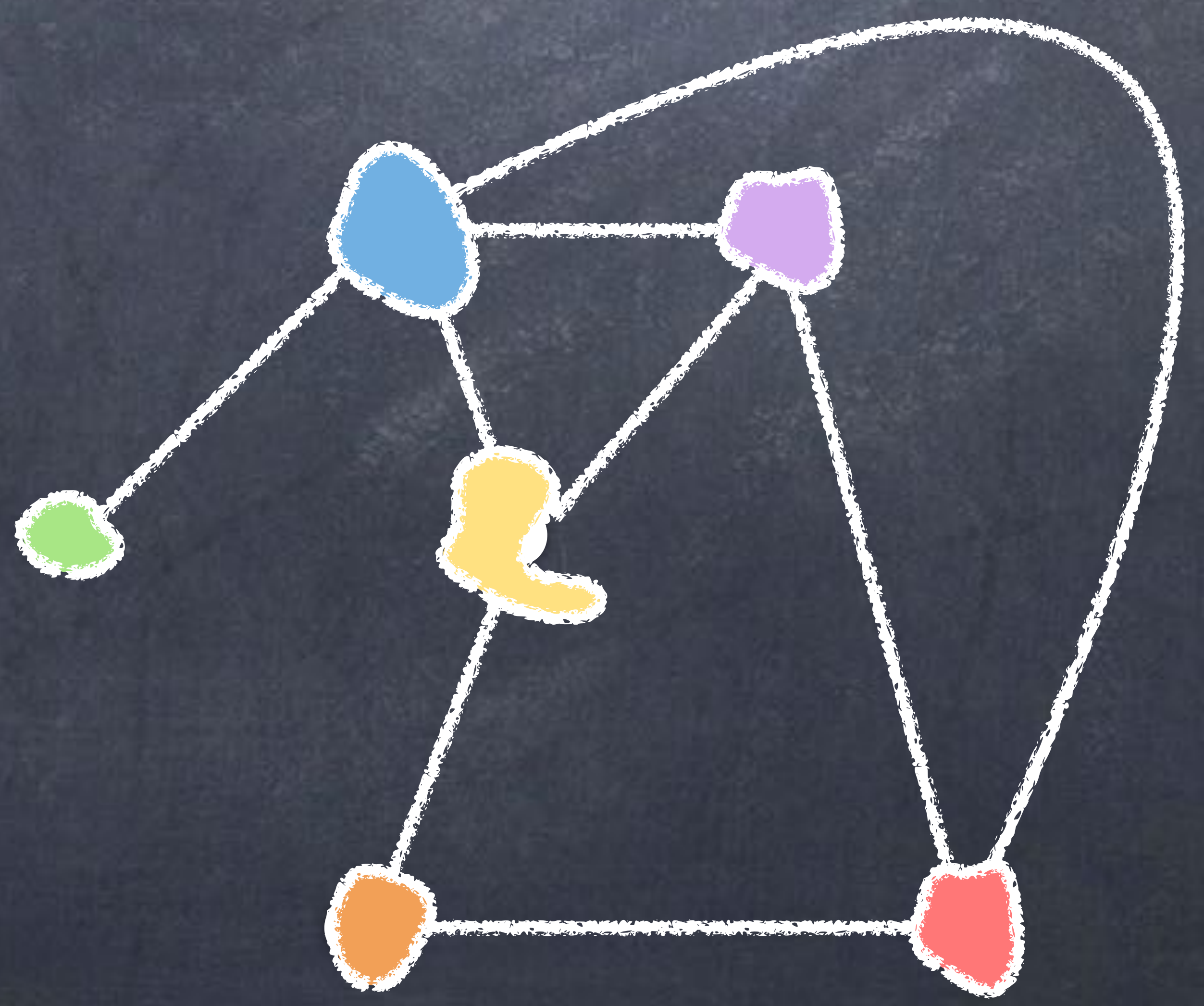
G can be contracted to H :: V(G) can be partitioned into |V(H)| many parts s.t.

a. Each part is connected

b. Each part is mapped to some vertex is H s.t. V1, V2 adjacent iff h1, h2 are adjacent.

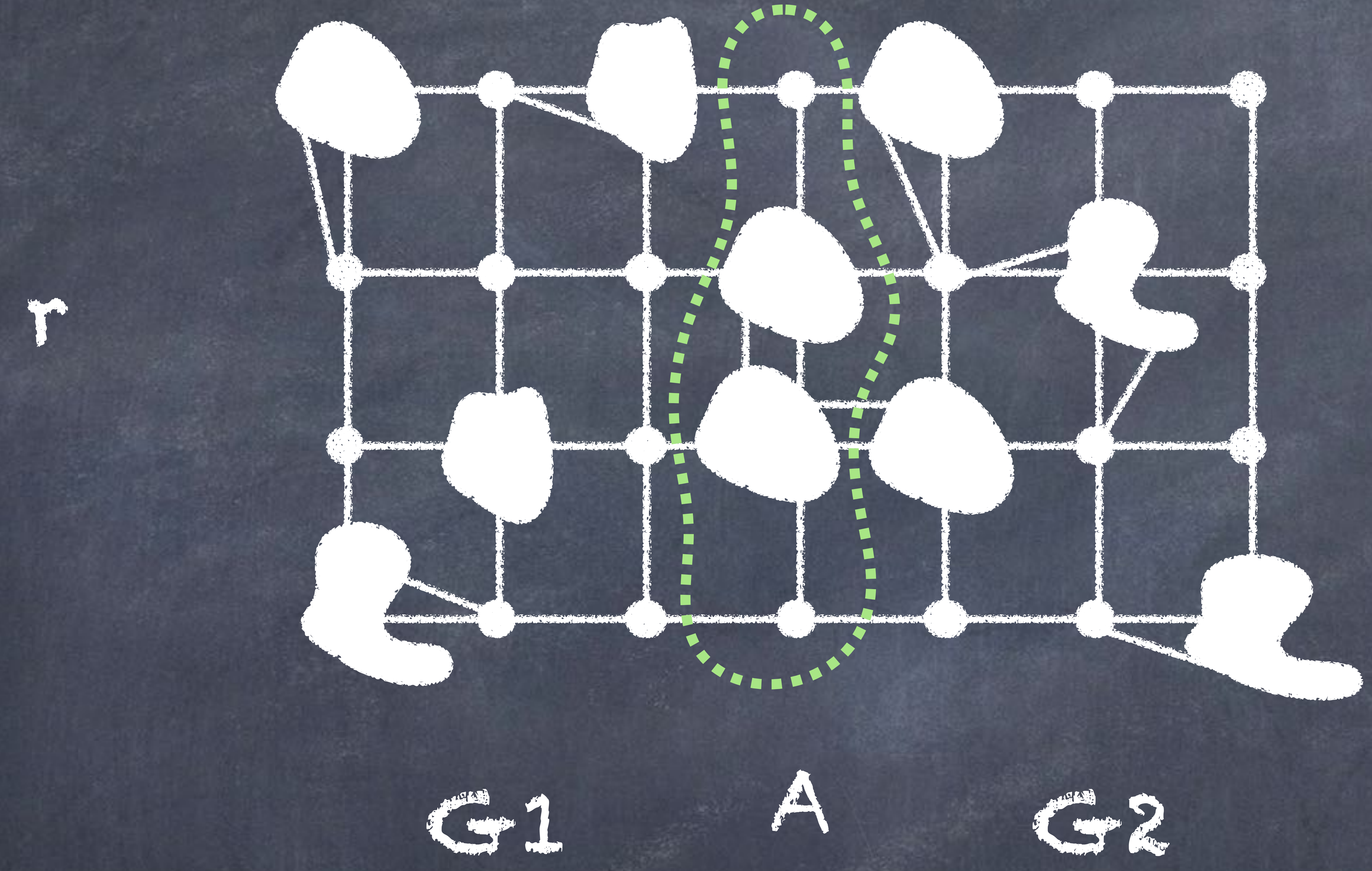


G



H

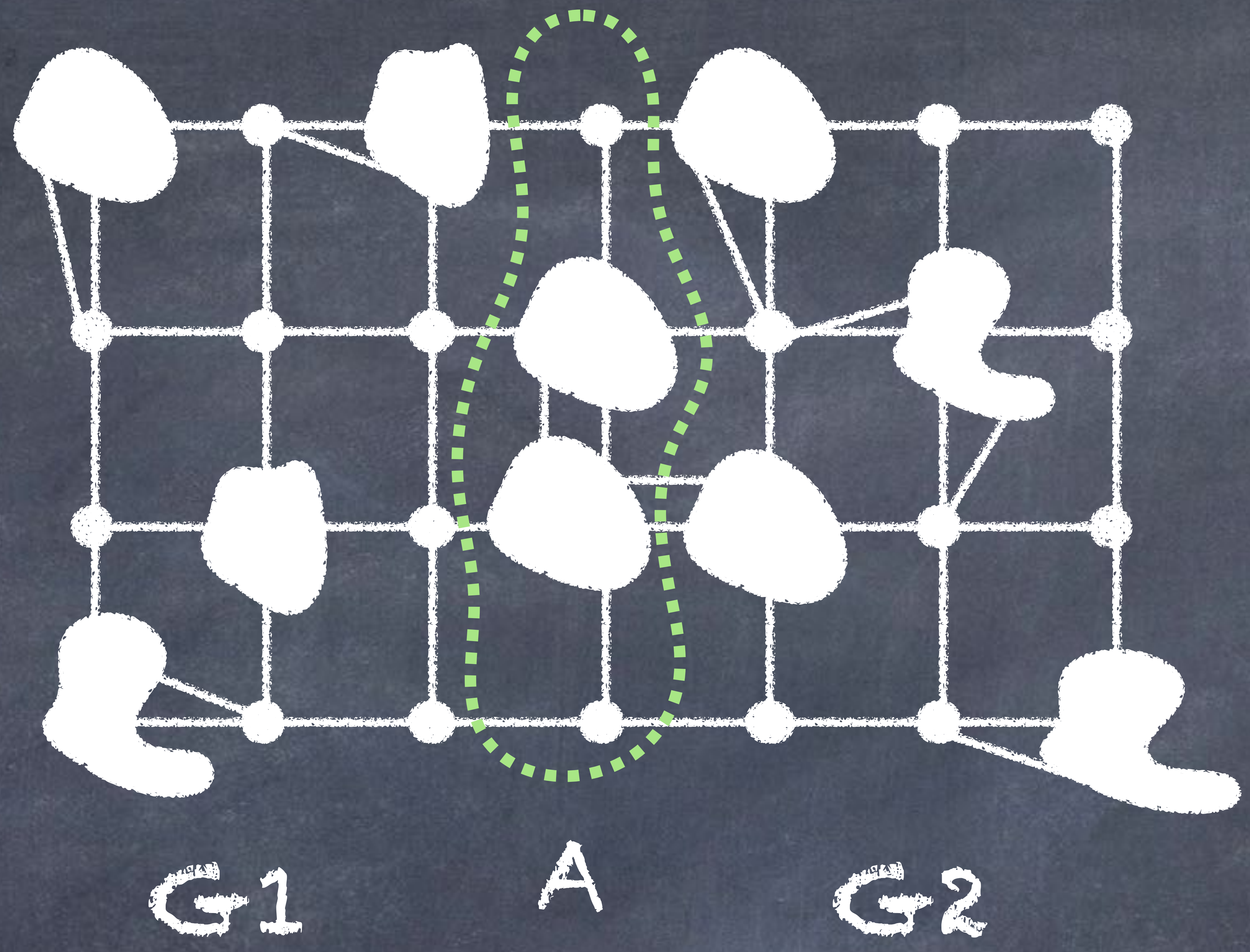
If G is k-contractible to a (r x q) grid

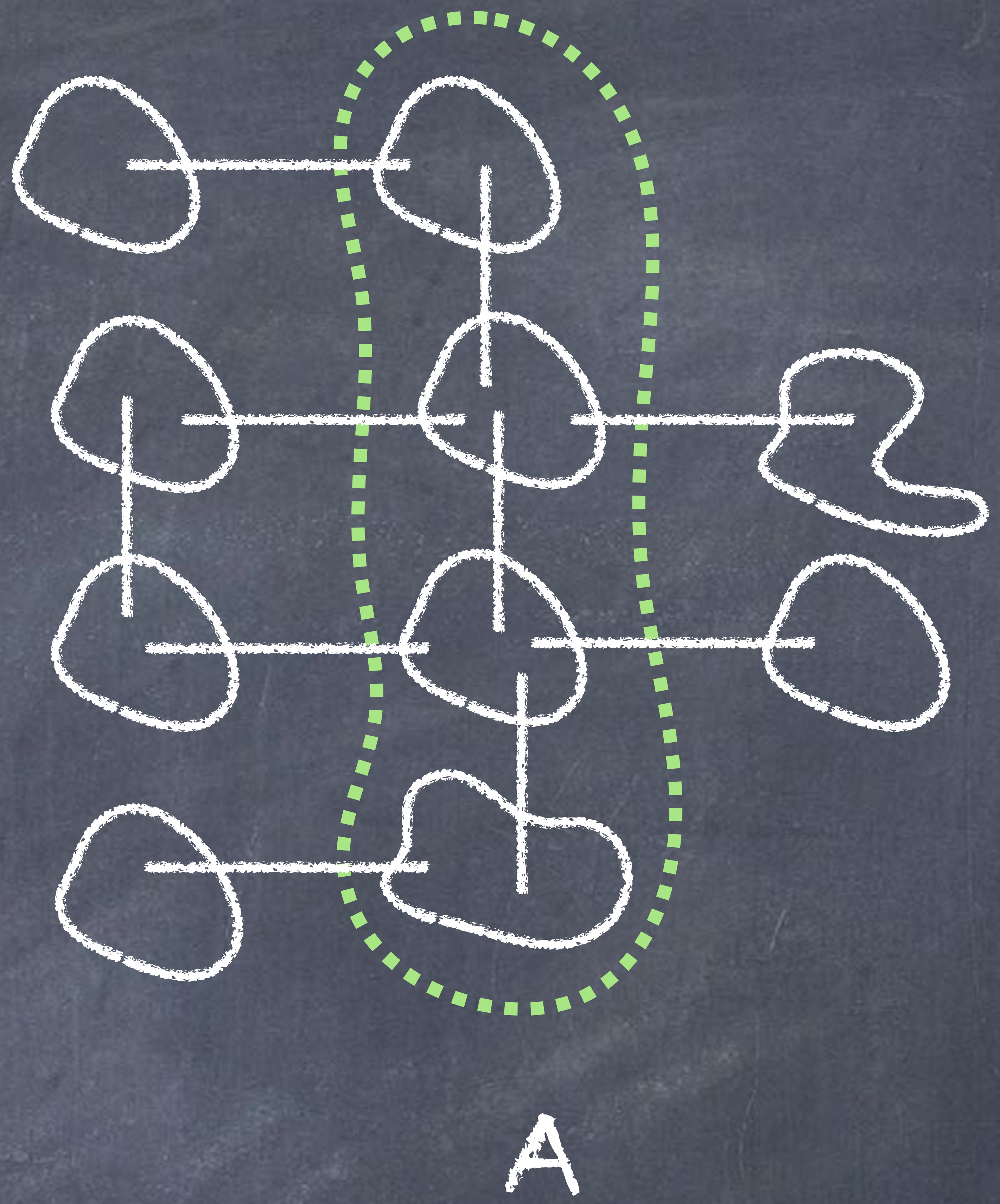


Contraction of G1 and G2 to grids affects only A

Dynamic Programming

r



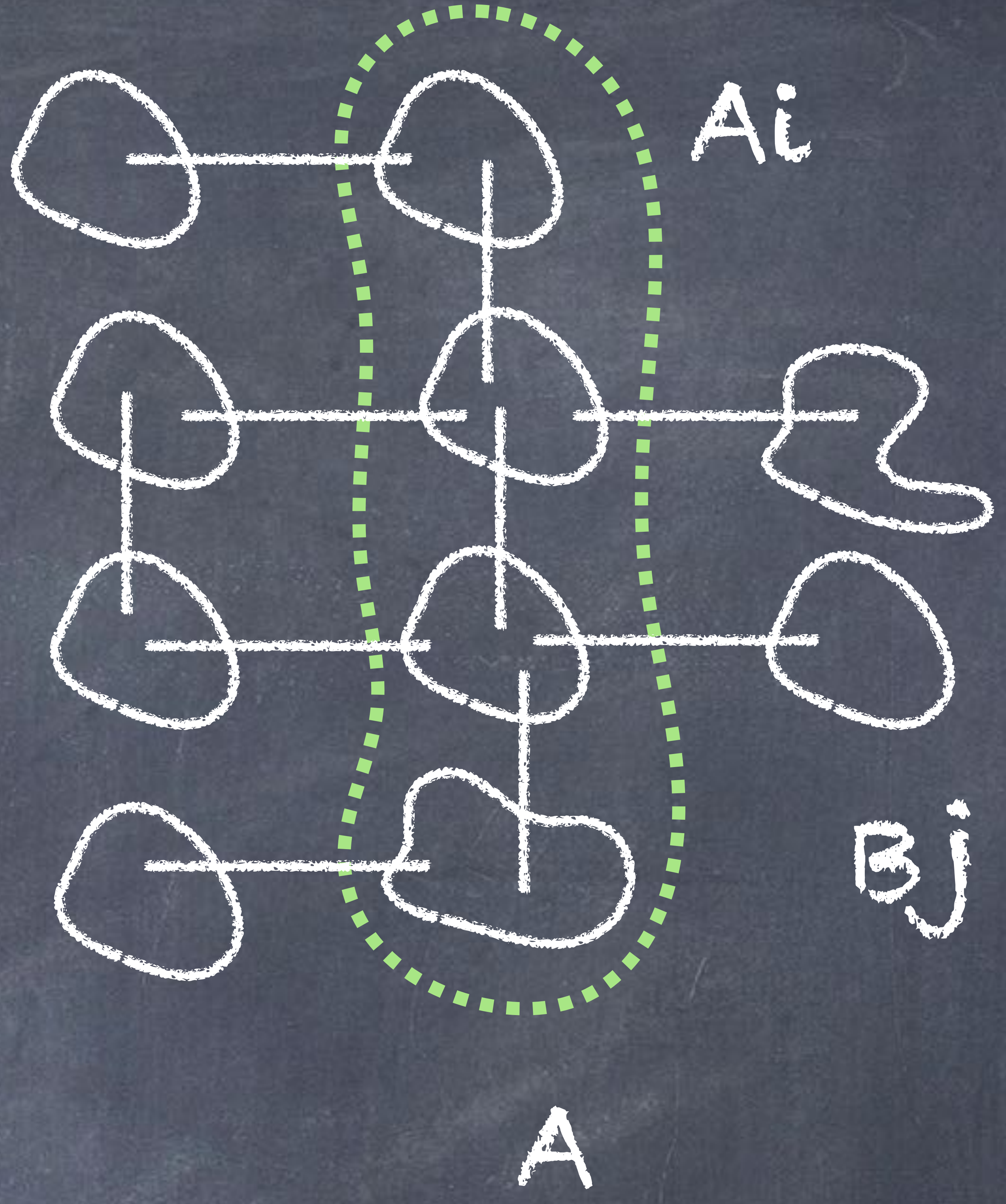


Separator in G:

- connected
- bounded closed nbr.
- 'well partitioned'

Separator in G:

- connected
- bounded closed nbr.
- 'well partitioned'



Def: [r-slab] An ordered r-partition <A1, A2, ... , Ar> of set A, such that

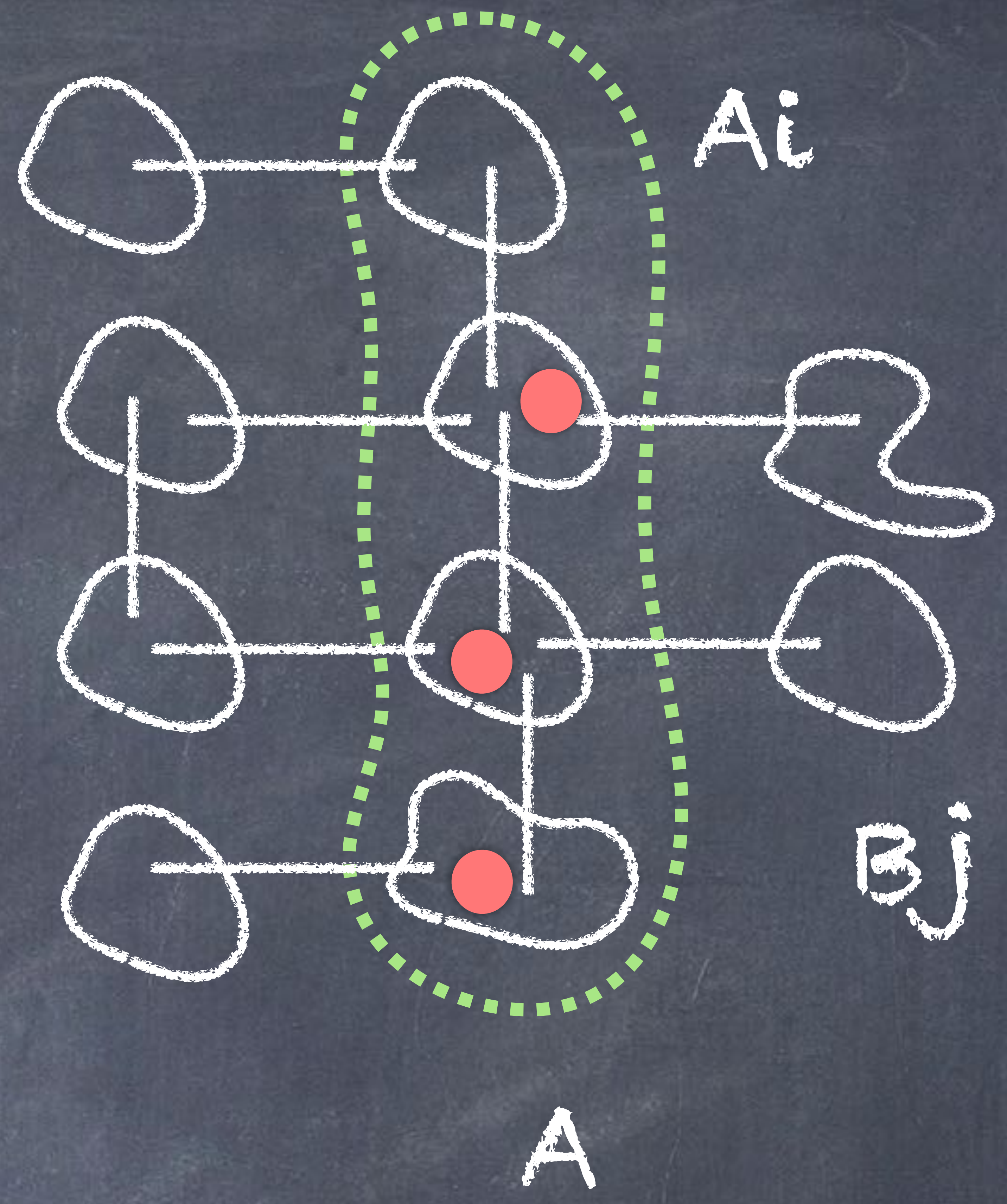
(i) Ai is non-empty and connected

(ii) Ai, Aj are adjacent iff $|i - j| = 1$

(iii) $Bi := N(Ai) \setminus A$. If Bi, Bj are adj. then $|i - j| = 1$

Def: [r-slab] An ordered r-partition <A1, A2, ... , Ar> of set A, such that

(i), (ii), and (iii)



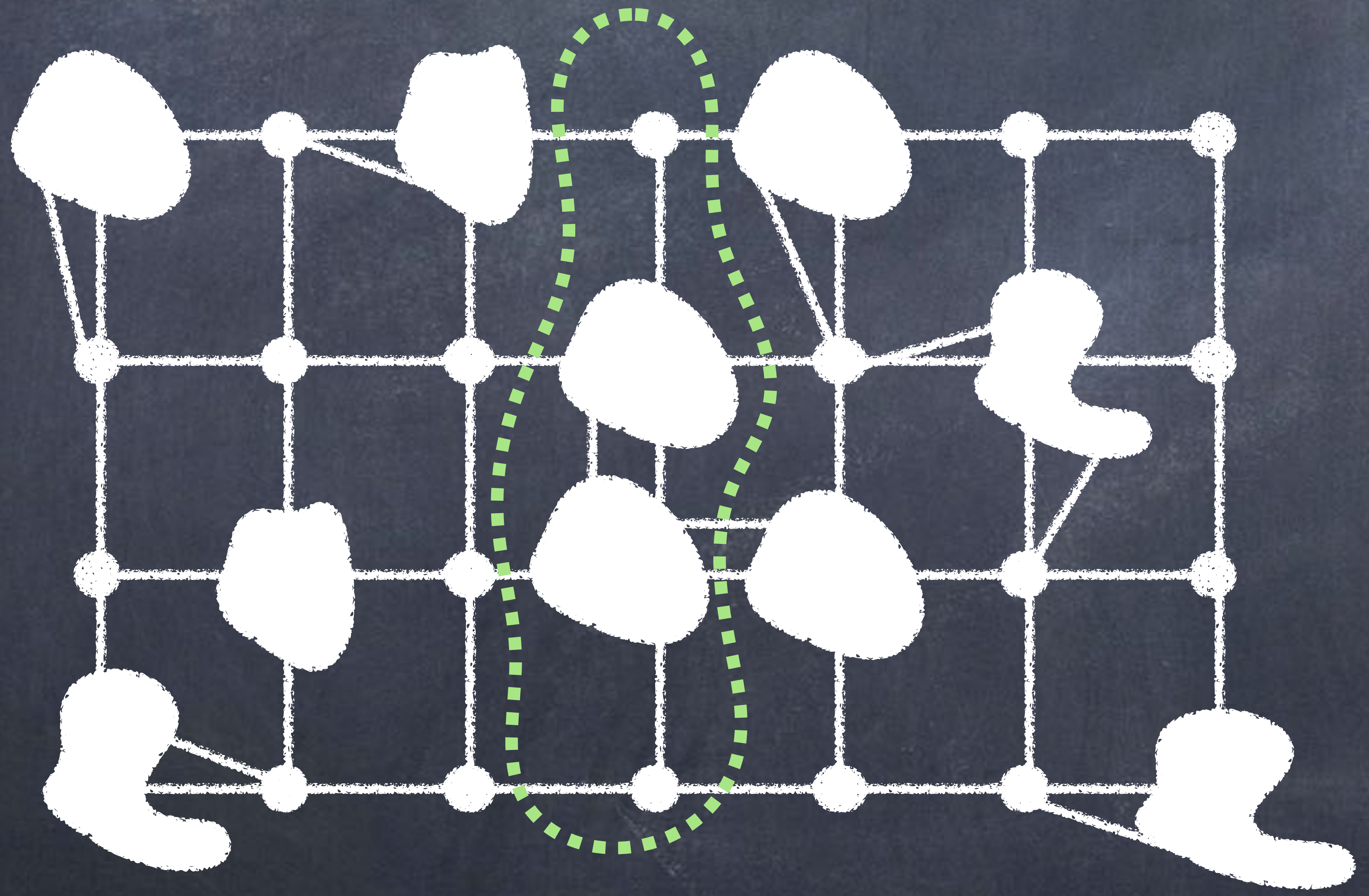
Def:[{ (a, b) } r-slab] r-slab A s.t. |A| <= a, |N(A)| = |B| <= b.

Def: [{ (Q1, Q2, ..., Qr) } r-slab]: r-slab A s.t. Qi contains in Ai.

Def: { (Q1, Q2, ..., Qr); (a, b) } r-slab

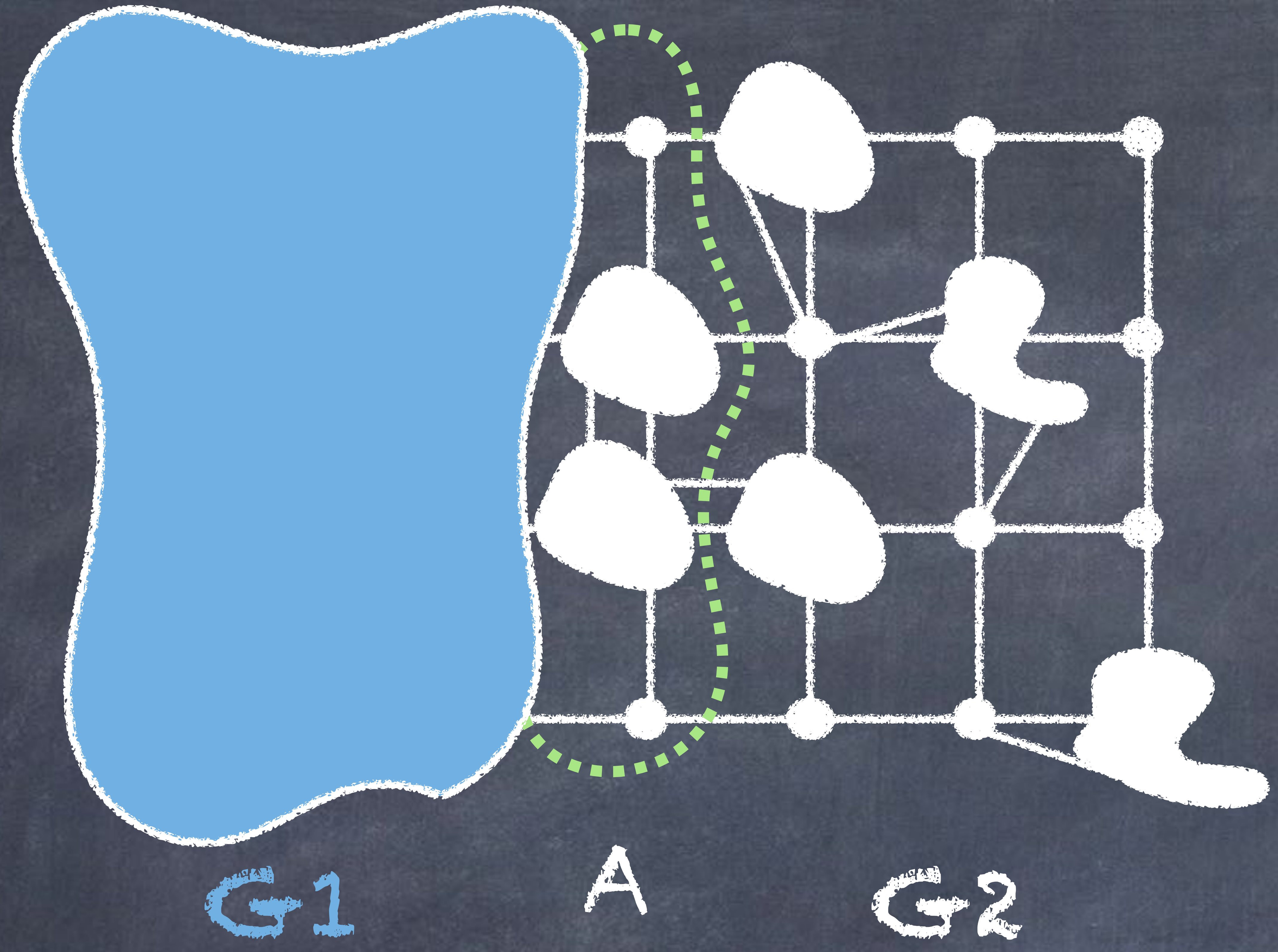
Def: { (Q1, Q2, ..., Qr); (a, b) } r-slab

Lemma: The number of these types of r-slabs are at most exp(a + b - |Q|).



G1 A G2

- a+b <= k + 3r
- Neg. |Q| term to get better running time

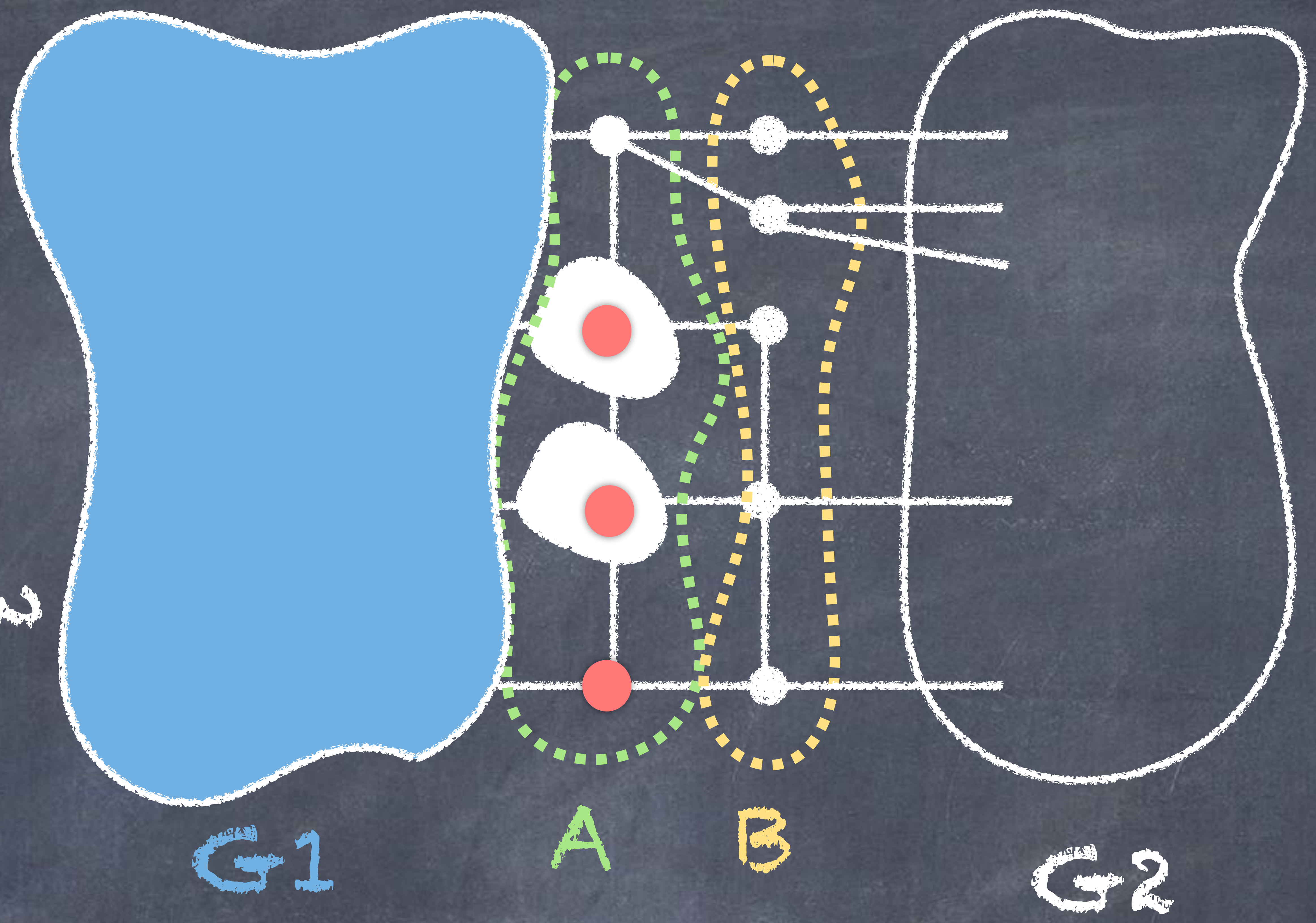


Def: [nice -subsets] Connected subsets that are separated by {(a, b)} r-slabs.

Lemma: The number of nice-subsets is at most exp(a + b).

Dynamic Programming

T[G1, P(Q)] – Min nr of edges to contract G1 into grid s.t. vertices in last row are 'adj' to P(Q).



If T[G1, P(Q)] is true

Construct { (Q1, Q2, ..., Qr); (a, b) } r-slab and update T[G1 + A; P(B)]

Lemma: Nr. Of entries in the table is $\exp(k + r)$.

Lemma: Time spent at each entry is $\exp(k + r)$.

Theorem: Bounded Grid Contraction is FPT when parameterised by $k + r$.

Corr.: Tailored algorithm for $r = 1$ gives algo for Path Contraction running in time 2^k.

Improvement over $2^{k + k/\log k}$.

FPT Algorithms: Two Phases

Phase (I):

Bounded Grid Contraction is FPT when parameterised by $k + r$.

Phase (II):

(G, k) – Yes inst. of Grid Contr.

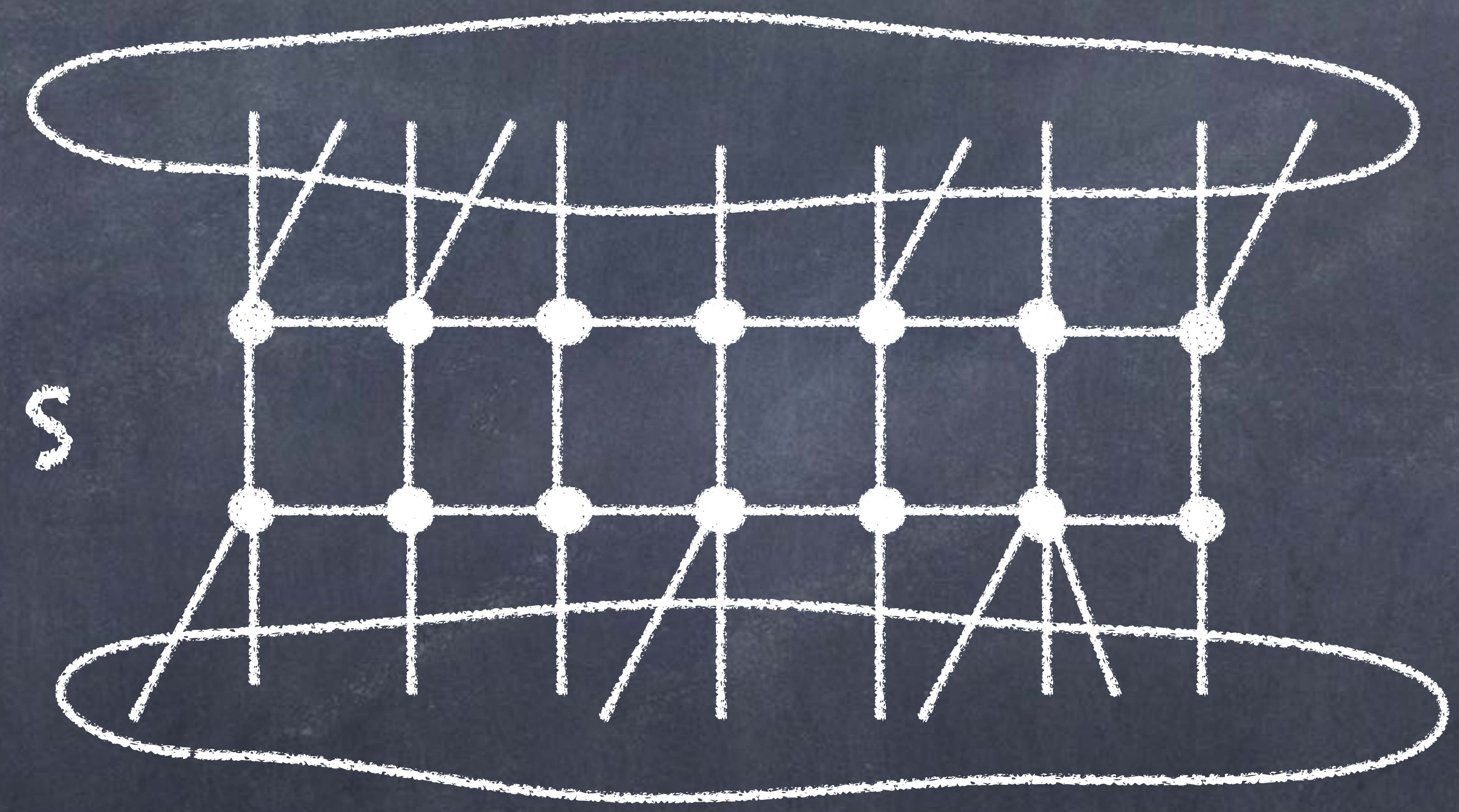
iff (G, k, r) – Yes inst. of Bounded Grid Contr.

for r in $\{1, 2, \ldots, 2k + 5\}$

Phase (II):

If G is k-contractible to (r x q) grid with r > 2k + 5

then there is a (2 x q) separator S.



It is safe to contract all vertical edges in S

If G is k-contractible to $(r \times q)$ grid with $r > 2k + 5$

then there is a $(2 \times q)$ separator S.

It is safe to contract all vertical edges in S

$$(G, k, r) \rightarrow (G', k, r - 1)$$

Parameter r is reduced.

Such a separator can be found in poly time.

FPT Algorithms: Two Phases

Phase (I):

Bounded Grid Contraction is FPT
when parameterised by $k + r$.

Phase (II):

(G, k) – Yes inst. of Grid Contr.

iff (G, k, r) – Yes inst. of Bounded Grid Contr.

for r in $\{1, 2, \ldots, 2k + 5\}$

Parameterized Complexity and Known Results

Our Results

Overview of FPT Algorithm

Open Questions

Q1. Can we improve running time from 4^{6k} to 2^k?

1-slab :: connected comp.

Better bounds for (a,b)-connected comp.

Q2. Can we improve kernel?

kernel – k^4
Turing Compression – k^2

$\mathcal{G}$ – Contraction: Kernel

Grid Contraction ???

Path Max Deg <= 2

Bounded Tree/Cactus

Poly kernel

Bounded path width

Q3. Any graph class $\mathcal{G}$, s.t $\mathcal{G}$–Contraction admits a poly kernel and some width parameter is unbounded?

Thank you.