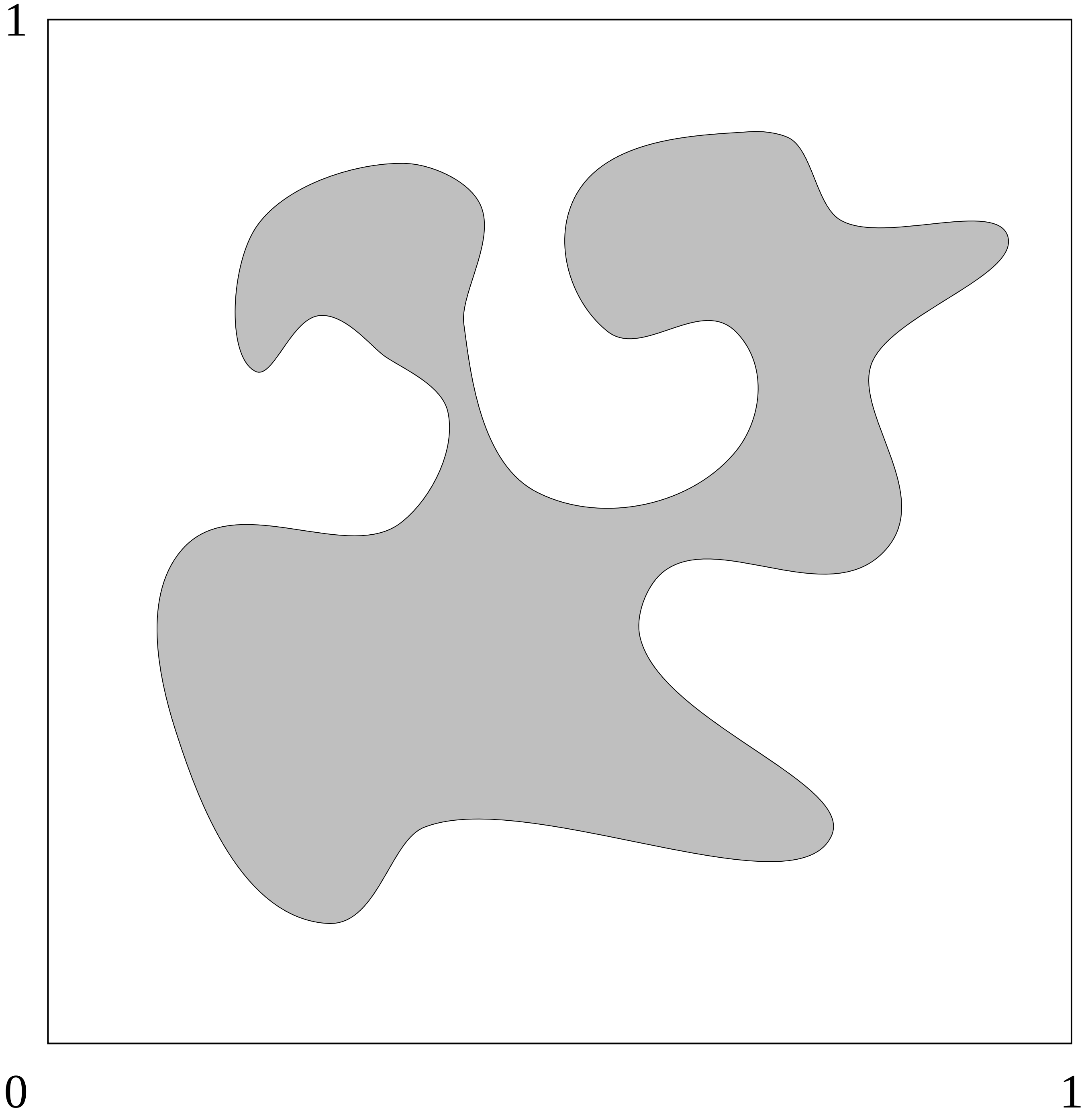
Pseudorandom Generators for Combinatorial Rectangles

Chi-Jen Lu
Institute of Information Science
Academia Sinica
Taipei, Taiwan, ROC.
Email: `cjlu@iis.sinica.edu.tw`

Area?



Rectangles of Dimension 2:

Combinatorial Rectangles $\mathcal{R}(m,d)$

- $\mathcal{R}(m,d)$ contains those

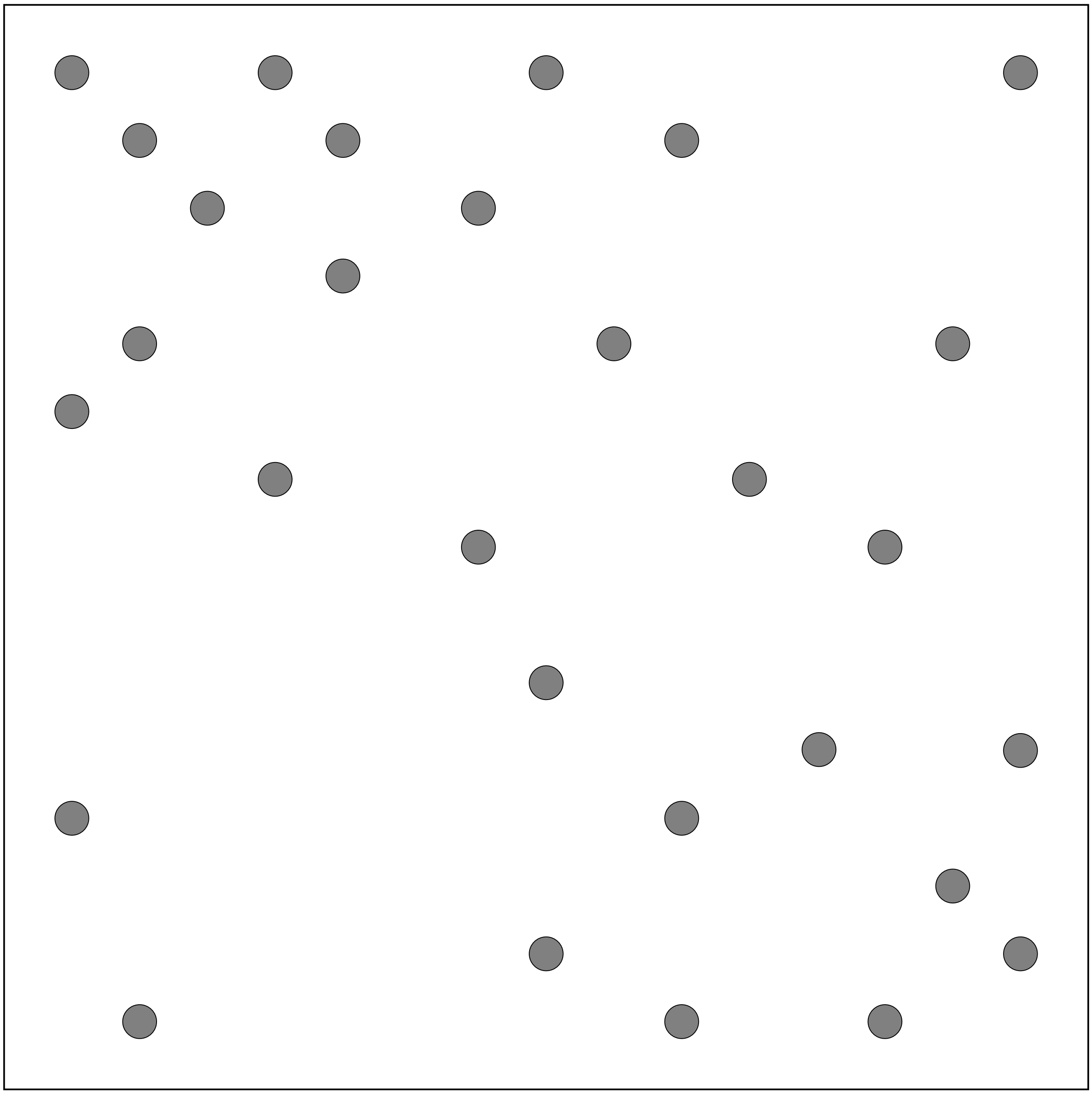
$$R_1 \times \cdots \times R_d = \{(x_1,\ldots,x_d) : x_1 \in R_1,\ldots,x_d \in R_d\},$$

  where $\forall i, R_i \subseteq [m] = \{1,\ldots,m\}$.

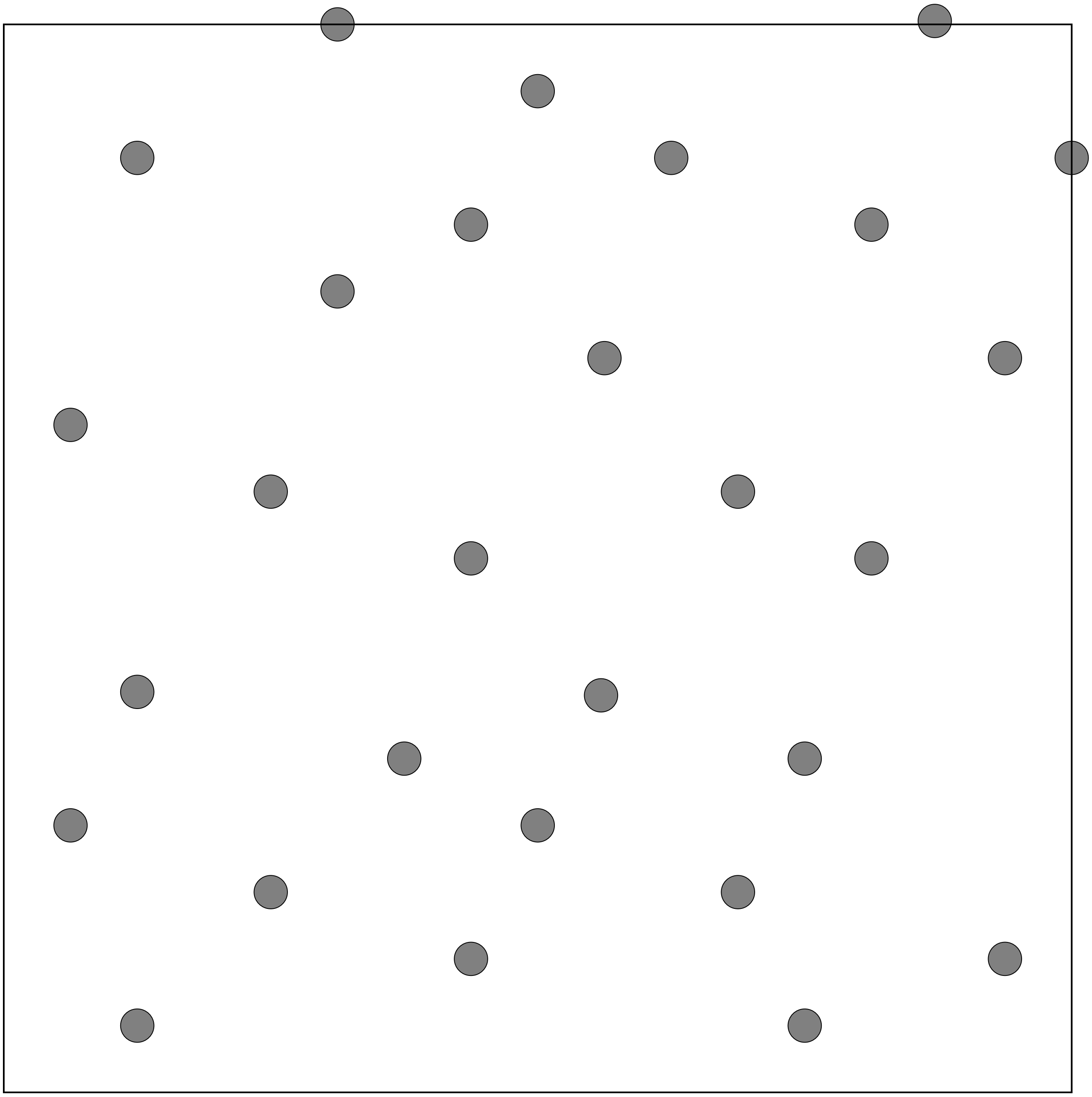
- $vol(R) = P_{x\in[m]^d}[x \in R] = \prod_{i=1}^{d} \frac{|R_i|}{m}$, for $R \in \mathcal{R}(m,d)$.
- Need $\log m^d = d\log m$ bits to sample.
- Fewer random bits? Smaller sample space?
- $g : \{0,1\}^r \to [m]^d$ is an ε-PRG for $\mathcal{R}(m,d)$, if $\forall R \in \mathcal{R}(m,d)$,

$$|vol(R) - P_z[g(z) \in R]| \leq \varepsilon.$$

A Bad Sample Set:



A Good Sample Set:

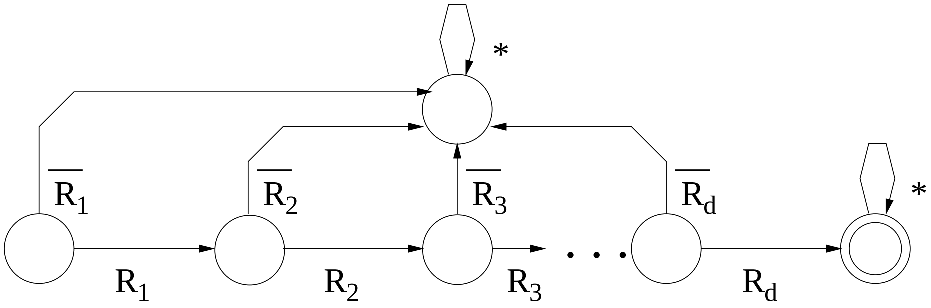


RL vs. L

- $\mathbf{L} \subseteq \mathbf{SL} \subseteq \mathbf{RL}$.
- PRGs for **RL**, using $O(\log^2 n)$ bits [N 90].
- $\mathbf{RL} \subseteq \mathbf{SC}^2 = \mathbf{DSPACE,TIME}(\log^2, n^{O(1)})$ [N 92].
- $\mathbf{RL} \subseteq \mathbf{L}^{3/2} = \mathbf{DSPACE}(\log^{3/2} n)$ [SZ 95].
- $\mathbf{SL} \subseteq \mathbf{L}^{4/3} = \mathbf{DSPACE}(\log^{4/3} n)$ [ATWZ 97].

A Special Case: Rectangles

$$R = R_1 \times R_2 \times R_3 \times \cdots \times R_d.$$



With 2 branching program.

Other Related Problems

- Approximating generalized multivalued independent random variables.
- Approximate DNF counting DNF.

ε-PRGs for $\mathcal{R}(m, d)$

Number of Random Bits	$m, d, \frac{1}{\varepsilon} = n^{\Theta(1)}$	*Reference*
$O(\log d \log \frac{md}{\varepsilon}))$	$O(\log^2 n)$	[N 90]
$O(\log \frac{1}{\varepsilon} \log \frac{md}{\varepsilon}))$	$O(\log^2 n)$	[EGLNV 92]
$O(\log d \log \frac{d}{\varepsilon} + \log m)$	$O(\log^2 n)$	[INW 94]
$O(\log^2 \frac{1}{\varepsilon} + \log(md))$	$O(\log^2 n)$	[ASWZ 96]
$O(\log^{3/2} \frac{1}{\varepsilon} + \log(md))$	$O(\log^{3/2} n)$	[Lu 98]

- $d \downarrow \quad m \uparrow$ INW.

Saving Random Bits

- k-wise independence.
- Composing PRGs:

$$\boxed{l_1 \text{ bits}} \overset{g_1}{\to} \boxed{\quad l_2 \text{ bits} \quad} \overset{g_2}{\to} \boxed{\qquad l_3 \text{ bits} \qquad}$$

k-wise Independent Space $\mathcal{H}_k(d,m)$

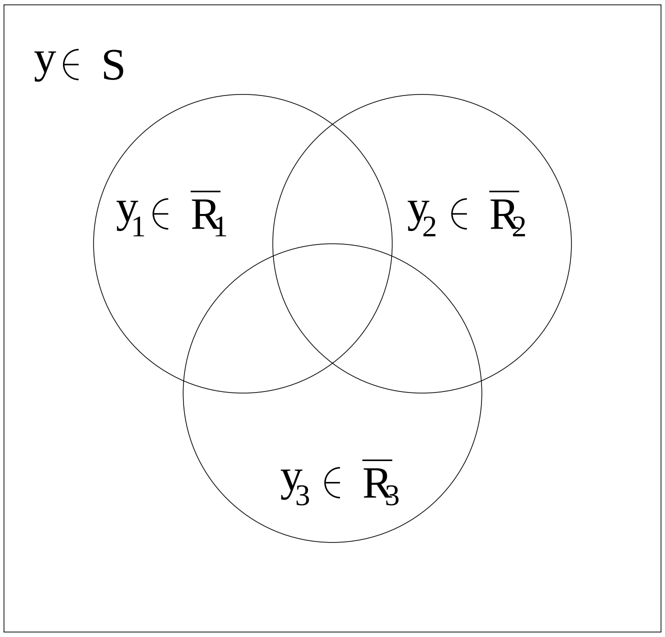
- $G \equiv \mathcal{H}_k(d,m) \subseteq [m]^d$.
- Each dim is uniform. Any k dim are independent.
- $\forall J = \{j_1, \ldots, j_k\} \subseteq [d]$ and $\forall B_{j_1}, \ldots, B_{j_k} \subseteq [m]$,

$$P_{y \in G}[\bigwedge_{j \in J} (y_j \in B_j)] = P_{y \in [m]^d}[\bigwedge_{j \in J} (y_j \in B_j)].$$

- Need $O(k \log \max(d,m))$ bits, instead of $d \log m$ bits.

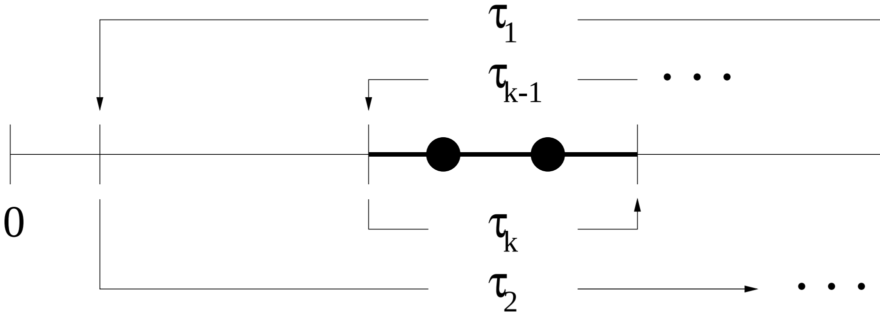
G fools $\mathcal{R}(m,d)$

- Principle of Inclusion-Exclusion: For any $S \subseteq [m]^d$,



$$\begin{aligned} P_{y\in S}[y \in R] &= 1 \quad - P_{y\in S}[\bigvee_{j\in[d]} (y_j \in \overline{R}_j)] \\ &= 1 \quad - \sum_{j\in[d]} P_{y\in S}[y_j \in \overline{R}_j] \\ &\qquad + \sum_{J\subseteq[d],|J|=2} P_{y\in S}[\bigwedge_{j\in J} (y_j \in \overline{R}_j)] \\ &\qquad - \cdots \\ &\qquad + (-1)^k \sum_{J\subseteq[d],|J|=k} P_{y\in S}[\bigwedge_{j\in J} (y_j \in \overline{R}_j)] \\ &\qquad + \cdots \end{aligned}$$

- $P_{y\in[m]^d}[y \in R]$ and $P_{y\in G}[y \in R]$ agree up to the kth terms.



- $|P_{y\in[m]^d}[y \in R] - P_{y\in G}[y \in R]| \leq \tau_k$.
- $k = \Omega(\log \frac{1}{\varepsilon})$ for error $\varepsilon \Rightarrow$ $\Omega(\log \frac{1}{\varepsilon} \log \max(m, d))$ random bits.
- Smaller k?

Divide & Conquer

$$h : [d] \to [d'].\ \ g_h : G^{d'} \to [m]^d.$$

$$\begin{tikzcd}
\underbrace{R_1 \times R_2 \times R_3} & \times & \underbrace{R_4 \times R_5 \times R_6} & \times & \underbrace{R_7 \times R_8 \times R_d} \\
R_{h^{-1}(1)} & \times & R_{h^{-1}(2)} & \times & R_{h^{-1}(d')} \\
G \arrow[u] & \times & G \arrow[u] & \times & G \arrow[u]
\end{tikzcd}$$

$$\begin{aligned}
& |vol(R) - P_{y \in G^{d'}}[g_h(y) \in R]| \\
\le\ & |\prod_{j \in [d']} vol(R_{h^{-1}(j)}) - \prod_{j \in [d']} P_{z_j \in G}[z_j \in R_{h^{-1}(j)}]| \\
\le\ & \sum_{j \in [d']} |vol(R_{h^{-1}(j)}) - P_{z_j \in G}[z_j \in R_{h^{-1}(j)}]| \\
\le\ & \sum_{j \in [d']} \tau_k(h^{-1}(j)).
\end{aligned}$$

- Note: $\tau_j(I) = \sum_{J \subseteq I, |J| = j} \prod_{i \in J} \frac{|\overline{R}|}{m}$, for $I \subseteq [d]$.

Dividing Randomly

Let $d' = 2^{O(\frac{1}{k}\log\frac{1}{\varepsilon})}$ and $H \equiv \mathcal{H}_k(d, d')$.

Lemma: $|vol(R) - P_{h\in H, y\in G^{d'}}[g_h(y) \in R]| \leq \varepsilon.$

Proof:

$$\begin{aligned}
& |vol(R) - E_{h\in H} P_{y\in G_h}[y \in R]| \\
\leq\ & E_{h\in H} |vol(R) - P_{y\in G_h}[y \in R]| \\
\leq\ & E_{h\in H} \sum_{j\in[d']} \tau_k(h^{-1}(j)) \\
=\ & \sum_{j\in[d']} E_{h\in H} \sum_{I\subseteq h^{-1}(j), |I|=k} \prod_{i\in I} \frac{|\overline{R}_i|}{m} \\
=\ & \sum_{j\in[d']} E_{h\in H} \sum_{I\subseteq [d], |I|=k} \delta[I \subseteq h^{-1}(j)] \prod_{i\in I} \frac{|\overline{R}_i|}{m} \\
=\ & \sum_{j\in[d']} \sum_{I\subseteq [d], |I|=k} P_{h\in H}[\forall i \in I, h(i) = j] \prod_{i\in I} \frac{|\overline{R}_i|}{m} \\
=\ & d' (\frac{1}{d'})^k \sum_{I\subseteq [d], |I|=k} \prod_{i\in I} \frac{|\overline{R}_i|}{m} \\
\leq\ & \varepsilon^{\Omega(1)} \tau_k([d]).
\end{aligned}$$

- *Either* $\tau_k([d])$ is small,
  or both $vol(R)$ and $P_{h,y}[g_h(y) \in R]$ are small.

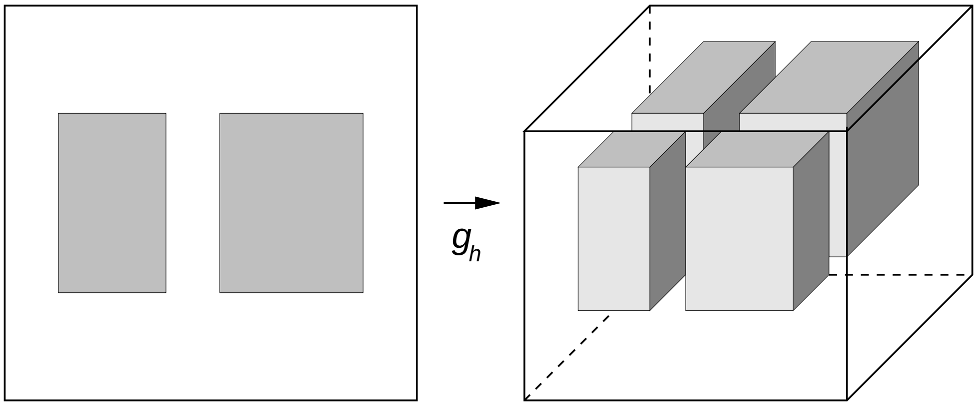
A Generator

$$g : H \times \mathbf{G^{d'}} \to [m]^d, \text{ where } g(h, \cdot) = g_h(\cdot).$$

Composing Generators

Let $m' = |G|$. Then

$$R \in \mathcal{R}(m, d) \;\; \Rightarrow \;\; g_h^{-1}(R) \in \mathcal{R}(m', d').$$



Lemma: $f : \{0,1\}^r \to [m']^{d'}$ is an ε-PRG for $\mathcal{R}(m', d')$

$\Rightarrow$ $g(\cdot, f(\cdot))$ is a 2ε-PRG for $\mathcal{R}(m, d)$.

Dimension Reduction

$\mathcal{R}(m,d) \Rightarrow \mathcal{R}(m',d')$.

- $m' = |G| = |\mathcal{H}_k(m,d)| = (m+d)^{O(k)}$.
- $d' = 2^{O(\frac{1}{k}\log\frac{1}{\varepsilon})}$.
- Cost$=\log_2 |H| = O(k \log d)$.

INW-generator for $\mathcal{R}(m',d')$

Number of random bits

$$\begin{aligned} &= O(\log m' + \log d'(\log \frac{1}{\varepsilon} + \log d')) \\ &= O(k\log(m+d) + \frac{1}{k}\log\frac{1}{\varepsilon}\log\frac{1}{\varepsilon}) \\ &= O((\log^{1/2}\frac{1}{\varepsilon})(\log m + \log d + \log\frac{1}{\varepsilon})), \end{aligned}$$

for $k = \log^{1/2}\frac{1}{\varepsilon}$.

Two More Reductions

Dimension Reduction I [ASWZ 96]:

$$\mathcal{R}(m,d) \Rightarrow \mathcal{R}(m_1,d_1): \quad \begin{aligned} d_1 &= (\tfrac{1}{\varepsilon})^{O(1)}, \\ m_1 &= (md)^{O(1)}, \\ c_1 &= O(\log d). \end{aligned}$$

Range Reduction:

Use extractors.

$$\mathcal{R}(m_1,d_1) \Rightarrow \mathcal{R}(m_2,d_2): \quad \begin{aligned} d_2 &= d_1, \\ m_2 &= (\tfrac{1}{\varepsilon})^{O(1)}, \\ c_2 &= O(\log \tfrac{md}{\varepsilon}). \end{aligned}$$

Theorem

$\exists$ ε-PRG for $\mathcal{R}(m,d)$ using $O(\log^{3/2}\frac{1}{\varepsilon} + \log m + \log d)$ random bits.

A Potential Improvement

Observation:

Okay if for small δ,

$$|P_{y\in[m]^d}[\bigwedge_{j\in J}(y_j \in \overline{R}_j)] - P_{y\in[m]^d}[\bigwedge_{j\in J}(y_j \in \overline{R}_j)]| \leq \delta.$$

Definition:

$\mathcal{R}(m,d,t)$: Rectangles from $\mathcal{R}(m,d)$ with at most t nontrivial ($\neq [m]$) dimensions.

Theorem:

$\exists$ ε-PRG for $\mathcal{R}(m,d,t)$ using $O(t + \log d + \log m + \log \frac{1}{\varepsilon})$ bits $\Rightarrow$
$\exists$ ε-PRG for $\mathcal{R}(m,d)$ using $O(\log d + \log m + \log \frac{1}{\varepsilon} \log\log \frac{1}{\varepsilon})$ bits.

Combinatorial Rectangles $\mathcal{B}(m,n,t)$

$\mathcal{B}(m,d,t)$ contains those $R = I_1 \times \cdots \times I_d$ s.t.

- $\forall i, I_i = \{a_i, a_i + 1, \ldots, a_i + b_i\}$, an interval.
- At most t nontrivial dimensions.

ε-PRGs for $\mathcal{B}(m,d,t)$

Number of Random Bits	*Reference*
$O(\log d + \log \frac{1}{\varepsilon} + t)$	[EGLNV 92]
$O(\log\log d + \log \frac{1}{\varepsilon} \log \frac{t}{\log 1/\varepsilon})$, for $t \geq \log 1/\varepsilon$	[CRS 94]
$O(\log\log d + \log \frac{1}{\varepsilon} \sqrt{\log \frac{t}{\log 1/\varepsilon}})$, for $t \geq \log 1/\varepsilon$	[Lu 98]

Improved Pseudorandom Generators for Combinatorial Rectangles

Chi-Jen Lu
Institute of Information Science
Academia Sinica
Taipei, Taiwan, ROC
Email: `cjlu@iis.sinica.edu.tw`

Abstract

We construct a pseudorandom generator which uses $O(\log m + \log d + \log^{3/2} 1/\epsilon)$ bits and approximates the volume of any combinatorial rectangle in $\{1, \ldots, m\}^d$ to within ϵ error. This improves on the previous construction by Armoni, Saks, Wigderson, and Zhou [3] using $O(\log m + \log d + \log^2 1/\epsilon)$ bits. For a subclass of rectangles with at most $t \geq \log 1/\epsilon$ nontrivial dimensions and each dimension being an interval, we also give a pseudorandom generator using $O(\log\log d + \log 1/\epsilon \log^{1/2} \frac{t}{\log 1/\epsilon})$ bits, which again improves the previous upper bound $O(\log\log d + \log 1/\epsilon \log \frac{t}{\log 1/\epsilon})$ by Chari, Rohatgi, and Srinivasan [4].

1 Introduction

Pseudorandom generators for combinatorial rectangles have been actively studied recently, because they are closely related to some fundamental problems in theoretical computer science, such as derandomizing RL, DNF approximate counting, and approximating the distributions of independent multivalued random variables.

Let V be a finite set with the uniform distribution. The *volume* of a set $A \subseteq V$ is defined as

$$vol(A) = P_{x \in V}[x \in A].$$

Let $\mathcal{A}$ be a family of subsets from V. We want to sample from a much smaller space, instead of from V, and still be able to approximate the volume of any subset $A \in \mathcal{A}$. We call a function $g : \{0,1\}^l \to V$ an *ϵ-generator* using l bits for $\mathcal{A}$, if for all $A \in \mathcal{A}$,

$$|P_{y \in \{0,1\}^l}[g(y) \in A] - vol(A)| \leq \epsilon.$$

For positive integers m and d, a *combinatorial rectangle* of type (m, d) is a subset of $[m]^d = \{1, \ldots, m\}^d$ of the form $R_1 \times \cdots \times R_d$, where $R_i \subseteq [m]$ for all $i \in [d]$. Let $\mathcal{R}(m, d)$ denote the family of all such rectangles. The volume of a rectangle $R \in \mathcal{R}(m, d)$ is now $\prod_{i \in [d]} \frac{|R_i|}{m}$. Our goal is to find explicit ϵ-generators with small l for $\mathcal{R}(m, d)$ and some particular subclasses of it.

As observed by Even, Goldreich, Luby, Nisan, and Veličković [5], this is a special case of constructing pseudorandom generators for RL. Nisan's generator for RL [10] is currently the best, using $O(\log^2 n)$ bits. Because it has many important applications and no improvement has been made for several years, one might hope that solving this special case could shed some light on the general problem.

It's easy to show that a random function mapping from $O(\log m + \log d + \log 1/\epsilon)$ bits to $[m]^d$ is very likely to be an ϵ-generator for $\mathcal{R}(m, d)$. However, the efficient construction of an explicit one still remains open. Even *et al.* [5] gave two ϵ-generators. One uses $O((\log m+\log d+\log 1/\epsilon)\log 1/\epsilon)$ bits based on k-wise independence, and the other uses $O((\log m+\log d+\log 1/\epsilon)\log d)$ bits based on Nisan's generator for RL. Armoni *at al.* [3] observed that the generator of Impagliazzo, Nisan, and Wigderson [7] for communication networks also gives an ϵ-generator for $\mathcal{R}(m,d)$ using $O(\log m+(\log d+\log 1/\epsilon)\log d)$ bits, which is good when d is small. They then reduced the original problem to the case when d is small (a formal definition of reductions will be given in the next section), and used the INW-generator to get an ϵ-generator for $\mathcal{R}(m,d)$ using $O(\log m+\log d+\log^2 1/\epsilon)$ bits.

When m, d, and $1/\epsilon$ are polynomially related, say all $n^{\Theta(1)}$, all these previous generators still use $\Theta(\log^2 n)$ bits, which is the current barrier for its generalized problem — constructing generators for RL. We break this barrier for the first time, and give an ϵ-generator for $\mathcal{R}(m,d)$ using $O(\log m+\log d+\log^{3/2} 1/\epsilon)$ bits. Our construction is based on that of Armoni *at al.* [3], and uses two more reductions to further reduce the dimension before applying the INW-generator. The overall construction can be seen as a composition of several generators for rectangles. Independently, Radhakrishnan and Ta-shma [13] have a slightly weaker result using a very similar idea.

We also observe that further improvements can be made if one can do better for a special case. Let $\mathcal{R}(m,d,k)$ be the set of rectangles from $\mathcal{R}(m,d)$ with at most k nontrivial dimensions (those not equal to $[m]$). We show that if an explicit ϵ-generator using $O(k+\log m+\log d+\log 1/\epsilon)$ bits for $\mathcal{R}(m,d,k)$ exists, we can construct an explicit ϵ-generator using $O(\log m+\log d+\log 1/\epsilon \log\log 1/\epsilon)$ bits for $\mathcal{R}(m,d)$. Unfortunately we still don't know how to construct such a generator for $\mathcal{R}(m,d,k)$.

Another interesting special case is for rectangles where each dimension is an interval. Let $\mathcal{B}(m,d,k)$ be the set of rectangles from $\mathcal{R}(m,d,k)$ with each dimension being an interval. Even *et al.* [5] observed that the problem of approximating the distribution of independent multivalued random variables can be reduced to this case. They gave a generator using $O(k+\log d+\log 1/\epsilon)$ bits. This is good when $k=O(\log 1/\epsilon)$. For the case $k\geq \log 1/\epsilon$, Chari *et al.* [4] gave a generator using $O(\log\log d+\log 1/\epsilon \log \frac{k}{\log 1/\epsilon})$ bits. Here, we improve this again to $O(\log\log d+\log 1/\epsilon \log^{1/2} \frac{k}{\log 1/\epsilon})$.

We will not emphasize the efficiency of our generators, but one can easily check that all the generators can be computed in simultaneous $(md/\epsilon)^{O(1)}$ time and $O(\log m+\log d+\log 1/\epsilon)$ space.

It's worth mentioning that the hitting version of our problem has already been settled. Linial, Luby, Saks, and Zuckerman [8] gave an explicit generator using $O(\log m+\log\log d+\log 1/\epsilon)$ bits that can hit any rectangle in $\mathcal{R}(m,d)$ of volume at least ϵ. In fact the work of Armoni *at al.* [3] followed this result closely, and so does ours.

2 Preliminaries

2.1 Notations

For a set V, we let 2^V denote the family of subsets of V. For a rectangle $R\in\mathcal{R}(m,d)$ and a set of indices $I\subseteq[d]$, we let R_I denote the subrectangle of R restricted to those dimensions in I. Similarly, for a vector $v\in[m]^d$ and $I\subseteq[d]$, we define v_I to be the subvector of v restricted to those dimensions in I.

Let $\mathcal{H}_k(n_1,n_2)$ denote the standard family of k-wise independent hash functions from $[n_1]$ to $[n_2]$. It can be identified with $[|\mathcal{H}_k(n_1,n_2)|]$ in the sense that there is a one-to-one mapping from $[|\mathcal{H}_k(n_1,n_2)|]$ onto $\mathcal{H}_k(n_1,n_2)$ that can be efficiently computed. Whenever we can identify a class $\mathcal{F}$ of functions with $[|\mathcal{F}|]$, we can use numbers in $[|\mathcal{F}|]$ to represent functions in $\mathcal{F}$.

There is a natural correspondence between functions from A to B and vectors in $B^{|A|}$. So $\mathcal{H}_k(n_1,n_2)$ can be seen as $|\mathcal{H}_k(n_1,n_2)|$ vectors in $[n_2]^{n_1}$. For a function $f: A\rightarrow[m]^d$, an element $x\in A$, and an index $y\in[d]$, we will use $f(x)(y)$ to denote the yth dimension in the vector

$f(x) \in [m]^d$.

When we sample from a finite set, the default distribution is the uniform distribution over that set. All the logarithms throughout this paper will have base 2.

2.2 Reductions

We adopt the notion of reduction introduced by Armoni *at al.* [3]. It enables us to reduce a harder problem to an easier one, and then focus our attention on solving the easier problem. A class $\mathcal{F}$ of functions from a set V_2 to a set V_1 defines a reduction from V_1 to V_2. Let $\mathcal{A}_1 \subseteq 2^{V_1}$ and $\mathcal{A}_2 \subseteq 2^{V_2}$. $\mathcal{F}$ is said to be $(\mathcal{A}_1, \mathcal{A}_2, \epsilon)$-good, if for each $R \in \mathcal{A}_1$ the following hold:

1. $\forall f \in \mathcal{F}, f^{-1}(R) \in \mathcal{A}_2$, and
2. $|E_{f\in\mathcal{F}}[vol(f^{-1}(R))] - vol(R)| \leq \epsilon$.

Suppose now that $\mathcal{F}$ is $(\mathcal{A}_1, \mathcal{A}_2, \epsilon_1)$ good and $g : \{0,1\}^s \to V_2$ is an ϵ_2-generator for $\mathcal{A}_2$. Armoni *at al.* [3] showed that the function $g' : \{0,1\}^s \times \mathcal{F} \to V_1$, defined as $g'(y, f) = (f \circ g)(y)$, is an $(\epsilon_1 + \epsilon_2)$-generator for $\mathcal{A}_1$. The *reduction cost* of $\mathcal{F}$ is $\log |\mathcal{F}|$, which is the number of extra bits needed for the new generator. The following lemma follows immediately.

Lemma 2.1 *For each i, $0 \leq i \leq l$, let V_i be a set and $\mathcal{A}_i \subseteq 2^{V_i}$. Suppose that $\mathcal{F}_i$ is $(\mathcal{A}_{i-1}, \mathcal{A}_i, \epsilon_{i-1})$-good for $1 \leq i \leq l$, and $g : \{0,1\}^s \to V_l$ is an ϵ_l-generator for $\mathcal{A}_l$. Then the function g' : $\{0,1\}^s \times \mathcal{F}_1 \times \cdots \times \mathcal{F}_l \to V_0$ defined as $g'(x, f_1, \ldots, f_l) = (f_1 \circ \cdots \circ f_l \circ g)(x)$, is a $(\sum_{i=0}^{l} \epsilon_i)$-genertor for $\mathcal{A}_0$.*

So to construct a generator for $\mathcal{A}_0$, it suffices to find a series of reductions from $\mathcal{A}_0$ to $\mathcal{A}_l$, and then find a generator for $\mathcal{A}_l$.

Notice that an $(\mathcal{A}_1, \mathcal{A}_2, \epsilon)$-good reduction $\mathcal{F}$ actually corresponds to a special kind of ϵ-generator for $\mathcal{A}_1$. Let $h : V_2 \times \mathcal{F} \to V_1$ be defined as $h(y, f) = f(y)$. Then the second condition guarantees that for all $R \in \mathcal{A}_1$,

$$\begin{aligned} |P_{(y,f)\in V_2\times\mathcal{F}}[h(y,f) \in R] - vol(R)| &= |E_{f\in\mathcal{F}}[P_{y\in V_2}[f(y) \in R]] - vol(R)| \\ &= |E_{f\in\mathcal{F}}[vol(f^{-1}(R))] - vol(R)| \\ &\leq \epsilon. \end{aligned}$$

The first condition guarantees that one part of h's input, V_2, can come from the output of a generator for $\mathcal{A}_2$, and makes the composition of generators possible. So one way of finding a reduction is to use some generator that might use many bits but can be composed with other generators.

3 The Pseudorandom Generator for $\mathcal{R}(m, d)$

3.1 The Overview of the Construction

The INW-generator uses $O(\log m + (\log d + \log 1/\epsilon) \log d)$ bits, which is good when d is small. The idea of Armoni *at al.* [3] is to reduce the dimension of rectangles first to $d' = (1/\epsilon)^{O(1)}$ before applying the INW-generator.

In addition to that, we also reduce m to $m' = (1/\epsilon)^{O(1)}$. The INW-generator for $\mathcal{R}(m', d')$ needs $O(\log m' + (\log d' + \log 1/\epsilon) \log d') = O(\log 1/\epsilon + \log^2 1/\epsilon)$ bits. Observe that we do not lose by letting m' increase a little. By allowing m' to grow to $m'' = (1/\epsilon)^{O(\sqrt{\log 1/\epsilon})}$, we are able to reduce

d' to $d'' = 2^{O(\sqrt{\log 1/\epsilon})}$. The INW-generator now uses $O(\log^{3/2} 1/\epsilon)$ bits for $R(m'', d'')$. The total reduction cost is $O(\log m + \log d + \log^{3/2} 1/\epsilon)$, and we have the desired generator for $\mathcal{R}(m, d)$.

More precisely, we will use the following three reductions.

- $\mathcal{F}_1$ is called the first dimension reduction family, and is used to reduce d to $d_1 = (1/\epsilon)^{O(1)}$. It is $(\mathcal{R}(m, d), \mathcal{R}(m_1, d_1), \epsilon/4)$-good, where $m_1 = (md)^{O(1)}$. The reduction cost is $O(\log d)$.

- $\mathcal{F}_2$ is called the range reduction family, and is used to reduce m_1 to $m_2 = (1/\epsilon)^{O(1)}$. It is $(\mathcal{R}(m_1, d_1), \mathcal{R}(m_2, d_2), \epsilon/4)$-good, where $d_2 = d_1$. The reduction cost is $O(\log m + \log d + \log 1/\epsilon)$.

- $\mathcal{F}_3$ is called the second dimension reduction family, and is used to reduce d_2 to $d_3 = 2^{(3 \log \frac{12}{\epsilon})/(k-1)}$, where k is a parameter to be chosen to optimize our construction. It is $(\mathcal{R}(m_2, d_2), \mathcal{R}(m_3, d_3), \epsilon/4)$-good, where $m_3 = |\mathcal{H}_k(d_2, m_2)| = (1/\epsilon)^{O(k)}$. The reduction cost is $\log |\mathcal{H}_k(d_2, d_3)| = O(k \log 1/\epsilon)$.

Together with the $\epsilon/4$-generator for $\mathcal{R}(d_3, m_3)$ from the INW-generator, we have an ϵ-generator for $\mathcal{R}(m, d)$. The number of bits used depends on k, and choosing $k = \log^{1/2} 1/\epsilon$ results in the minimum $O(\log m + \log d + \log^{3/2} 1/\epsilon)$.

3.2 The First Dimension Reduction Function Family

Let $\mathcal{F}_1$ be the reduction family used by Armoni *at al.* [3], which is the composition of three reduction families. $\mathcal{F}_1$ is $(\mathcal{R}(m, d), \mathcal{R}(m_1, d_1), \epsilon/4)$-good, where $m_1 = (dm)^{O(1)}$, $d_1 = (1/\epsilon)^{O(1)}$, and each is a power of 2. $|\mathcal{F}_1| = d^{O(1)}$. Let $V_1 = [m_1]^{d_1}$.

3.3 The Range Reduction Function Family

We will use a generator of Nisan and Zuckerman [12], based on an extractor of Goldreich and Wigderson [6]. The idea of using extractors for range reduction was inspired by that of Radhakrishnan and Ta-Shma [13]. We choose a more appropriate extractor and get a better reduction.

Definition 3.1 *A function $E : \{0,1\}^s \times \{0,1\}^t \to \{0,1\}^l$ is an (s, r, t, l, δ)-extractor if for x chosen from a distribution over $\{0,1\}^s$ with min-entropy[1] r, and y chosen from the uniform distribution over $\{0,1\}^t$, the distribution $E(x, y)$ has distance[2] at most δ to the uniform distribution over $\{0,1\}^l$.*

Extractors are used to extract randomness from weakly random sources, and have many other applications. For more details, please refer to an excellent survey by Nisan [11]. We use an extractor due to Goldreich and Wigderson [6].

Lemma 3.1 *There are constants c_1 and c_2 such that for any s, γ, and δ with $s > \gamma$, $s - \gamma > l$, and $\delta > 2^{-(s-l-c_1\gamma)/c_2}$, an explicit $(s, s-\gamma, O(\gamma + \log \frac{1}{\delta}), l, \delta)$-extractor exists.*

Choose $\delta = \epsilon/(4d_1) = \epsilon^{O(1)}$, $\gamma = \lceil \log 1/\delta \rceil$, $t = O(\gamma)$, and $l = \log m_1$. Choose $s = l + c\gamma = O(\log(md) + \log 1/\epsilon) = O(\log(md/\epsilon))$ for some constant c, such that $2^{-(s-l-c_1\gamma)/c_2} = 2^{-((c-c_1)/c_2)\log 1/\delta} < 2^{\log \delta} = \delta$. We have the following extractor for this setting.

Corollary 3.1 *There exists an explicit $(s, s-\gamma, t, l, \delta)$-extractor A.*

---

[1] The min-entropy of a distribution D on a set S is $\min_{a \in S} \log \frac{1}{D(a)}$.

[2] The distance of two distributions D_1 and D_2 over a set S is defined as $\max_{A \subseteq S} |D_1(A) - D_2(A)|$

The building block of Nisan and Zuckerman's generator for space bounded Turing machines [12], when using the extractor A, has the form $G : \{0,1\}^s \times \{0,1\}^{td_1} \rightarrow [m_1]^{d_1}$, where

$$G(x, y_1, \ldots, y_{d_1}) = (A(x, y_1), \ldots, A(x, y_{d_1})).$$

For $R = R_1 \times \cdots \times R_{d_1} \in \mathcal{R}(m_1, d_1)$, we first show that

$$|P_{x,y_1,\ldots,y_{d_1}}[(A(x,y_1),\ldots,A(x,y_{d_1})) \in R] - vol(R)| \leq d_1\delta = \frac{\epsilon}{4}.$$

For $i \in [d_1]$, let $D_i(x, y_1, \ldots, y_i)$ denote the event that $(A(x, y_1), \ldots, A(x, y_i))$ is in $R_1 \times \cdots \times R_i$. Also let $p_i = P_{x,y_1,\ldots,y_i}[D_i(x, y_1, \ldots, y_i)]$, $q_i = \prod_{j=1}^{i} \frac{|R_j|}{m_1}$, and $r_{i+1} = P_{x,y_1,\ldots,y_{i+1}}[A(x, y_{i+1}) \in R_{i+1} | D_i(x, y_1, \ldots, y_i)]$. The following is a simplified version of a lemma due to Nisan and Zuckerman [12], based on our choice of parameters.

Lemma 3.2 *For any $i \in [d_1]$, $|p_i - q_i| \leq i\delta$.*

Proof: Use induction on i. It's true for $i = 1$ from the definition of extractors. Assuming that $|p_i - q_i| \leq i\delta$, we'll show that $|p_{i+1} - q_{i+1}| \leq (i+1)\delta$.

$$\begin{aligned} |p_{i+1} - q_{i+1}| &= |p_i r_{i+1} - p_i \frac{|R_{i+1}|}{m_1} + p_i \frac{|R_{i+1}|}{m_1} - q_i \frac{|R_{i+1}|}{m_1}| \\ &\leq p_i |r_{i+1} - \frac{|R_{i+1}|}{m_1}| + |p_i - q_i|. \end{aligned}$$

There are two cases.

- $p_i \leq \delta$:

  $|p_{i+1} - q_{i+1}| \leq \delta + i\delta = (i+1)\delta.$

- $p_i > \delta$:

  The distribution of x conditioned on $D_i(x, y_1, \ldots, y_i)$ has min-entropy at least $s - \log 1/\delta \geq s - \gamma$. From Corollary 3.1, $|P_{x,y_1,\ldots,y_i}[A(x, y_{i+1}) \in R_{i+1} | D_i(x, y_1, \ldots, y_i)] - \frac{|R_{i+1}|}{m_1}| \leq \delta$, and $|p_{i+1} - q_{i+1}| \leq \delta + i\delta = (i+1)\delta$. ∎

Now let $m_2 = 2^t = (1/\delta)^{O(1)} = (d_1/\epsilon)^{O(1)} = (1/\epsilon)^{O(1)}$, $d_2 = d_1$, and $V_2 = [m_2]^{d_2}$. Consider the reduction $\mathcal{F}_2 = \{f_x \mid x \in \{0,1\}^s\}$, where $f_x : V_2 \rightarrow V_1$ is defined as follows

$$f_x(y_1, \ldots, y_{d_2}) = G(x, y_1, \ldots, y_{d_1}).$$

Then $f_x^{-1}(R) = R'_1 \times \cdots \times R'_{d_2} \in \mathcal{R}(m_2, d_2)$, where $R'_i = \{y_i \mid A(x, y_i) \in R_i\}$. Also,

$$\begin{aligned} |E_x[vol(f_x^{-1}(R))] - vol(R)| &= |P_{x,y_1,\ldots,y_{d_1}}[G(x, y_1, \ldots, y_{d_1}) \in R] - vol(R)| \\ &\leq \epsilon/4. \end{aligned}$$

So we have the following lemma.

Lemma 3.3 *$\mathcal{F}_2$ is $(\mathcal{R}(m_1, d_1), \mathcal{R}(m_2, d_2), \epsilon/4)$-good.*

3.4 The Second Dimension Reduction Function Family

Let $R = R_1 \times \cdots \times R_{d_2} \in \mathcal{R}(m_2, d_2)$. We want to partition the d_2 dimensions of R into d_3 parts using some function $h : [d_2] \to [d_3]$ in the natural way. For $q \in [d_3]$, those dimensions of R that are mapped to q form a subrectangle $R_{h^{-1}(q)} = \prod_{i \in h^{-1}(q)} R_i$. Based on the idea of Even *et al.* [5], its volume can be approximated by sampling from the k-wise independent space $G = \mathcal{H}_k(d_2, m_2)$. We use d_3 copies of G, one for each subrectangle. The corresponding rectangle $R^{(h)} = R'_1 \times \cdots \times R'_{d_3}$, where $R'_q = \{p \in G : p_{h^{-1}(q)} \in R_{h^{-1}(q)}\}$, should have a volume close to that of R. The error depends on the choice of k and h. We will show that for $k = O(\log^{1/2} 1/\epsilon)$ and h chosen uniformly from $H = \mathcal{H}(d_2, d_3)$, the expected error is at most $\epsilon/4$.

More formally, let $d_3 = 2^{(3 \log \frac{12}{\epsilon})/(k-1)}$, $m_3 = |G|$, $V_3 = [m_3]^{d_3}$, and $\mathcal{F}_3 = \{f_h : h \in H\}$, where $f_h : V_3 \to V_2$ is defined as follows

$$f_h(p_1, \ldots, p_{d_3}) = (p_{h(1)}(1), \ldots, p_{h(d_2)}(d_2)).$$

Then for $R \in \mathcal{R}(m_2, d_2)$ and $f_h \in \mathcal{F}_3$, $f_h^{-1}(R) = R^{(h)} \in \mathcal{R}(m_3, d_3)$.

We also need the following notation for the proofs below. For $R = R_1 \times \cdots \times R_{d_2} \in \mathcal{R}(m_2, d_2)$, let $\tilde{R}$ denote the rectangle $\overline{R}_1 \times \cdots \times \overline{R}_{d_2} \in \mathcal{R}(m_2, d_2)$, where $\overline{R}_i = [m_2] \setminus R_i$. For $i, j \in [d_2]$, and $I \subseteq [d_2]$, denote

$$\begin{aligned} \delta_i &= \frac{|\overline{R}_i|}{m_2}, \\ \pi(I) &= vol(\tilde{R}_I) = \prod_{i \in I} \delta_i, \\ \gamma(I) &= P_{p \in G}[p_I \in \tilde{R}_I], \text{ and} \\ \tau_j(I) &= \sum_{J \subseteq I, |J| = j} \pi(I). \end{aligned}$$

The approximation error of each subrectangle can be bounded in the following way.

Proposition 3.1 $\forall I \subseteq [d_2], |P_{p \in G}[p_I \in R_I] - vol(R_I)| \leq \tau_k(I)$

Proof: Because G is a k-wise independent space, for $J \subseteq I$ with $|J| \leq k$, $\pi(J) = \gamma(J)$. From the principle of inclusion and exclusion, we have the following.

$$vol(R_I) = \prod_{i \in I} (1 - \delta_i) = \sum_{J \subseteq I} (-1)^{|J|} \pi(J) = \sum_{j=0}^{k} (-1)^j \tau_j(I) + \sum_{j=k+1}^{|I|} (-1)^j \tau_j(I).$$

$$P_{p \in G}[p_I \in R_I] = \sum_{J \subseteq I} (-1)^{|J|} \gamma(J) = \sum_{j=0}^{k} (-1)^j \tau_j(I) + \sum_{j=k+1}^{|I|} (-1)^j \sum_{J \subseteq I, |J| = j} \gamma(J).$$

Now the proposition follows because both $vol(R_I)$ and $P_{p \in G}[p_I \in R_I]$ fall between $\sum_{j=0}^{k-1} (-1)^j \tau_j(I)$ and $\sum_{j=0}^{k} (-1)^j \tau_j(I)$. ∎

The approximation error of any partition can be bounded by the following.

Lemma 3.4 $\forall h : [d_2] \to [d_3], |vol(R^{(h)}) - vol(R)| \leq \sum_{q \in [d_3]} \tau_k(h^{-1}(q))$

Proof:

$$\begin{aligned} vol(R) &= \prod_{q\in[d_3]} vol(R_{h^{-1}(q)}). \\ vol(R^{(h)}) &= \prod_{q\in[d_3]} P_{p\in G}[p_{h^{-1}(q)} \in R_{h^{-1}(q)}]. \end{aligned}$$

This lemma follows from the previous proposition and the known fact that $|\prod_{i=1}^{l} x_i - \prod_{i=1}^{l} y_i| \le \sum_{i=1}^{l} |x_i - y_i|$ when $0 \le x_i, y_i \le 1$ for all $i \in [l]$. ■

Finally, we can bound the expected approximation error.

Lemma 3.5 *For* $R \in \mathcal{R}(m_2, d_2)$, $|E_{h\in H}[vol(R^{(h)})] - vol(R)| \le \frac{\epsilon}{4}$.

Proof:

$$\begin{aligned} |E_{h\in H}[vol(R^{(h)})] - vol(R)| &\le E_{h\in H}[|vol(R^{(h)}) - vol(R)|] \\ &\le E_{h\in H}[\sum_{q\in[d_3]} \tau_k(h^{-1}(q))] \\ &= \sum_{q\in[d_3]} E_{h\in H}[\sum_{I\subseteq h^{-1}(q),|I|=k} \pi(I)] \\ &= \sum_{q\in[d_3]} \sum_{I\subseteq[d_2],|I|=k} P_{h\in H}[\forall i \in I\ h(i) = q]\pi(I) \\ &= \sum_{q\in[d_3]} \sum_{I\subseteq[d_2],|I|=k} (1/d_3)^k \pi(I) \\ &= (1/d_3)^{k-1}\tau_k([d_2]). \end{aligned}$$

Let $\alpha = \sum_{i\in[d_2]} \delta_i$. There are two cases depending on the value of α.

- $\alpha \le \log \frac{12}{\epsilon}$:

$\tau_k([d_2])$ gets it maximum value when $\delta_i = \frac{\alpha}{d_2}$ for all $i \in [d_2]$. So $\tau_k([d_2]) \le (\frac{ed_2}{k})^k (\frac{\alpha}{d_2})^k \le (\frac{e\log\frac{12}{\epsilon}}{k})^k$, which is again maximized when $k = \log\frac{12}{\epsilon}$. So for $d_3 = 2^{(3\log\frac{12}{\epsilon})/(k-1)}$, we have

$$\begin{aligned} |E_{h\in H}[vol(R^{(h)})] - vol(R)| &\le 2^{-3\log\frac{12}{\epsilon}} e^{\log\frac{12}{\epsilon}} \\ &= 2^{-(3-\log e)\log\frac{12}{\epsilon}} \\ &= (\frac{\epsilon}{12})^{3-\log e} \\ &\le \frac{\epsilon}{12}. \end{aligned}$$

- $\alpha > \log\frac{12}{\epsilon}$:

In this case, both $E_{h\in H}[vol(R^{(h)})]$ and $vol(R)$ are small, so their difference is small. First, $vol(R) = \prod_{i\in[d_2]}(1-\delta_i) \le 2^{-\sum_{i\in[d_2]}\delta_i} < \frac{\epsilon}{12}$.

Next, we show that $E_{h\in H}[vol(R^{(h)})] \le \frac{\epsilon}{3}$. Let d' be the smallest integer such that $\log\frac{12}{\epsilon} - 1 < \sum_{i\in[d']}\delta_i \le \log\frac{12}{\epsilon}$. Let $R' = R_{[d']} \times [m_2]^{d_2-d'}$. From the previous case $E_{h\in H}[vol(R'^{(h)})] \le vol(R') +$

$\frac{\epsilon}{12}$. So

$$\begin{aligned} E_{h\in H}[vol(R^{(h)})] &\leq E_{h\in H}[vol(R'^{(h)})] \\ &\leq vol(R') + \frac{\epsilon}{12} \\ &\leq 2^{-\sum_{i\in[d']}\delta_i} + \frac{\epsilon}{12} \\ &\leq 2^{-\log\frac{12}{\epsilon}+1} + \frac{\epsilon}{12} \\ &= \frac{\epsilon}{4}. \end{aligned}$$

$0 \leq E_{h\in H}[vol(R^{(h)})], vol(R) \leq \frac{\epsilon}{4}$. So $|E_{h\in H}[vol(R^{(h)})] - vol(R)| \leq \frac{\epsilon}{4}$. ∎

So we have the following lemma.

Lemma 3.6 *$\mathcal{F}_3$ is $(\mathcal{R}(m_2, d_2), \mathcal{R}(m_3, d_3), \frac{\epsilon}{4})$-good.*

Theorem 3.1 *There is an explicit ϵ-generator for $\mathcal{R}(m, d)$, using $O(\log m + \log d + \log^{3/2} 1/\epsilon)$ bits.*

Proof: $|\mathcal{F}_1| = d^{O(1)}$, $|\mathcal{F}_2| = 2^s = (md/\epsilon)^{O(1)}$, and $|\mathcal{F}_3| = |\mathcal{H}_k(d_2, d_3)| \leq d_2^k = (1/\epsilon)^{O(k)}$. The INW-generator gives us an $\frac{\epsilon}{4}$-generator for $\mathcal{R}(m_3, d_3)$ using $O(\log m_3 + (\log d_3 + \log 1/\epsilon)\log d_3) = O(k \log 1/\epsilon + 1/k \log^2 1/\epsilon)$ bits. From Lemma 2.1, we have an ϵ-generator for $\mathcal{R}(m, d)$ using $O(\log m + \log d + k \log 1/\epsilon + 1/k \log^2 1/\epsilon)$ bits. When $k = O(\log^{1/2} 1/\epsilon)$, the number of bits used gets its minimum value $O(\log m + \log d + \log^{3/2} 1/\epsilon)$. ∎

4 A Potential Improvement

For $\mathcal{F}_3$, we can replace the k-wise independent space H by an approximate k-wise independent space H', over $[d_3]^{d_2}$, such that for any $I \subseteq [d_2]$ with $|I| \leq k$ and for any $y \in [d_3]^{d_2}$,

$$|P_{x\in H'}[x_I = y_I] - (\frac{1}{d_3})^{|I|}| \leq O(\frac{\epsilon}{d_3}).$$

A simple generalization from the constructions of Alon, Goldreich, Hastad, and Peralta [1], or Naor and Naor [9] gives us $|H'| = (\frac{k \log d_2}{\epsilon})^{O(1)} = (\frac{1}{\epsilon})^{O(1)}$. H' can also be identified efficiently with $|H'|$. One can easily verify that only an additional $O(\epsilon)$ error is introduced in Lemma 3.5, and now the reduction cost for $\mathcal{F}_3$ is $O(\log 1/\epsilon)$. From now on we will use H' instead of H in $\mathcal{F}_3$.

Recall from the previous section that $m_3 = |\mathcal{H}_k(d_2, m_2)| = (1/\epsilon)^{O(k)}$ and $d_3 = 2^{(3\log 1/\epsilon)/(k-1)}$. Larger k implies smaller d_3 but larger m_3. The optimum is attained at $k = \Theta(\log^{1/2} 1/\epsilon)$. If we can replace $\mathcal{H}_k(d_2, m_2)$ by a smaller space, we might be able to choose a larger k and get a smaller d_3. Remeber that d_3 copies of $\mathcal{H}_k(d_2, m_2)$ are used to approximate the volumes of the d_3 subrectangles of R partitioned by a function $h : [d_2] \to [d_3]$. The approximation is guaranteed by the fact that for $R \in \mathcal{R}(m_2, d_2)$ and for $J \subseteq [d_2]$ with $|J| \leq k$,

$$|\pi(J) - \gamma(J)| = |vol(\tilde{R}_J) - P_{p\in G}[p_J \in \tilde{R}_J]| = 0.$$

We want to use a smaller space by allowing a small error ϵ' instead of 0 above. The approximate k-wise independent space does not help here, because it needs $\Omega(k \log m_2 + \log 1/\epsilon')$ bits to achieve an error ϵ' here, no better than G. However, observe that what we need here is to approximate the volume of a rectangle with at most k nontrivial dimensions. This turns out to be a special case of our original problem — constructing a pseudorandom generator for $\mathcal{R}(m,d,k)$.

Suppose that $g : \{0,1\}^s \rightarrow [m_2]^{d_2}$ is an ϵ'-generator for $\mathcal{R}(m,d,k)$. In $\mathcal{F}_3$, we replace $\mathcal{H}_k(d_2,m_2)$ by the space generated by g. Let $m_3 = 2^s$. For $h \in H'$, let $f_h : [m_3]^{d_3} \rightarrow [m_2]^{d_2}$ be defined as follows

$$f_h(x_1,\ldots,x_{d_3}) = (g(x_{h(1)})(1),\ldots,g(x_{h(d_2)})(d_2)).$$

For $R = R_1 \times \cdots \times R_{d_2} \in \mathcal{R}(m_2,d_2)$, $f_h^{-1}(R) = R^{(h)} = R'_1 \times \cdots \times R'_{d_3} \in \mathcal{R}(m_3,d_3)$, where $R'_q = \{x \in [m_3] : g(x)_{h^{-1}(q)} \in R_{h^{-1}(q)}\}$. Then for $J \subseteq h^{-1}(q)$ with $|J| \leq k$, $|P_{x\in[m_3]}[g(x)_J \in \tilde{R}_J] - vol(\tilde{R}_J)| \leq \epsilon'$. So, for any $h : [d_2] \rightarrow [d_3]$,

$$\begin{aligned}
|vol(R) - vol(R^{(h)})| &\leq \sum_{q\in[d_3]} \Big(\Big(\sum_{J\subseteq h^{-1}(q),|J|\leq k} \epsilon'\Big) + \tau_k(h^{-1}(q))\Big) \\
&\leq d_2^{k+2}\epsilon' + \sum_{q\in[d_3]} \tau_k(h^{-1}(q)),
\end{aligned}$$

and $|E_{h\in H'}[vol(R^{(h)})] - vol(R)| \leq d_2^{k+2}\epsilon' + \frac{\epsilon}{4} = \epsilon/2$, for $\epsilon' = \frac{\epsilon}{4d_2^{k+2}}$.

So if we have a better ϵ'-generator for $\mathcal{R}(m_2,d_2,k)$, we can choose a larger k and thus a smaller d_3.

Theorem 4.1 *If there exists an explicit ϵ-generator for $\mathcal{R}(m,d,k)$ using $O(k+\log d+\log m+\log\frac{1}{\epsilon})$ bits, then there exists an explicit ϵ-generator for $\mathcal{R}(m,d)$ using $O(\log d+\log m+\log\frac{1}{\epsilon}\log\log\frac{1}{\epsilon})$ bits.*

Proof: Using the ϵ'-generator for $\mathcal{R}(m_2,d_2,k)$ in $\mathcal{F}_3$ gives $m_3 = (\frac{2^k d_2 m_2}{\epsilon/d_2^k})^{O(1)} = (\frac{d_2^k}{\epsilon})^{O(1)}$. We want to repeatedly reduce the dimensions of rectangles. Notice that each time the dimension is reduced, we can choose a larger next k.

For $3 \leq i \leq l = \log\log\frac{12}{\epsilon} - \log\log\log\frac{12}{\epsilon}$, let

$$\begin{aligned}
k_i &= 2^i, \\
d_i &= 2^{O((\log\frac{1}{\epsilon})/k_i)} = 2^{O((\log\frac{1}{\epsilon})/2^i)}, \\
\epsilon_i &= \frac{\epsilon}{4d_{i-1}^{k_i+2}} = \epsilon^{O(1)}, \text{ and} \\
m_i &= \Big(\frac{2^{k_i}m_{i-1}d_{i-1}}{\epsilon_i}\Big)^{O(1)} = \Big(\frac{1}{\epsilon}\Big)^{O(i)}
\end{aligned}$$

For $3 \leq i \leq l$, let $\mathcal{F}_i$ be the dimension reduction discussed before, using the assumed ϵ_i-generator for $\mathcal{R}(m_{i-1},d_{i-1},k_i)$. One can check that each $\mathcal{F}_i$ is $(\mathcal{R}(m_{i-1},d_{i-1}),\mathcal{R}(m_i,d_i),\epsilon)$-good. Using the INW-generator as an ϵ-generator for $\mathcal{R}(m_l,d_l)$, we have an $O(\epsilon\log\log\frac{1}{\epsilon})$-generator for $\mathcal{R}(m,d)$. The total number of bits used is

$$\begin{aligned}
& O(\log m + \log d + \log\tfrac{1}{\epsilon} + \log\tfrac{1}{\epsilon}\log\log\tfrac{1}{\epsilon} + \log m_l + (\log d_l + \log\tfrac{1}{\epsilon})\log d_l) \\
= \;& O(\log m + \log d + \log\tfrac{1}{\epsilon}\log\log\tfrac{1}{\epsilon}).
\end{aligned}$$

Replacing $\epsilon\log\log\frac{1}{\epsilon}$ by ϵ, we have an ϵ-generator for $\mathcal{R}(m,d)$ using $O(\log m+\log d+\log\frac{1}{\epsilon}\log\log\frac{1}{\epsilon})$ bits. ■

We don't know yet how to construct such an explicit ϵ-generator for $\mathcal{R}(m,d,k)$ using $O(\log k + \log d + \log m + \log 1/\epsilon)$ bits. Using an idea of Auer, Long, and Srinivasan [2], we can derive one using $O(\log k + \log m + \log^{3/2} 1/\epsilon)$ bits, which improves their upper bound, but does not serve our purpose here.

5 The Pseudorandom Generator for $\mathcal{B}(m,d,t)$

Recall that $\mathcal{B}(m,d,t)$ denotes the class of rectangles from $\mathcal{R}(m,d)$ with at most t nontrivial dimensions and each dimension being an interval. For $\mathcal{B}(m,d,t)$, Even *et al.* [5] have an ϵ-generator using $O(t+\log\log d+\log 1/\epsilon)$ bits. Unfortunately, we cannot apply the iterative procedure in the previous section to $\mathcal{B}(m,d,d)$ because after applying the dimension reduction once, each dimension is no longer an interval.

For $t \geq \log 1/\epsilon$, Chari *et al.* [4] had an ϵ-generator for $\mathcal{B}(m,d,t)$ using $O(\log\log d + \log 1/\epsilon \log \frac{t}{\log 1/\epsilon})$ bits, a significant improvement in the dependence on t. This is improved again by the following theorem.

Theorem 5.1 *For $t \geq \log 1/\epsilon$, there is an explicit ϵ-generator for $\mathcal{B}(m,d,t)$, using $O(\log\log d + \log 1/\epsilon \log^{1/2} \frac{t}{\log 1/\epsilon})$ bits.*

Proof: Here we use only one reduction, the modified dimension reduction discussed in the previous section. Let $k \leq t$ be a parameter to be chosen later. For $\epsilon' = \epsilon(\frac{k}{t})^{O(k)}$, let $g : \{0,1\}^s \to [m]^d$ be the ϵ'-generator of Even *et al.* for $\mathcal{B}(m,d,k)$. Let $m' = 2^s$ and $d' = 2^{(3\log\frac{12}{\epsilon})/(k-1)}$. Given $R = R_1 \times \cdots \times R_d \in \mathcal{B}(m,d,t)$, assume w.l.o.g. that the first t dimensions are nontrivial. For $h \in H'$ let $R^{(h)} = f_h^{-1}(R) = R'_1 \times \cdots \times R'_d$, where $R'_q = \{x \in [m'] : g(x)_{h^{-1}(q)} \in R_{h^{-1}(q)}\}$.

For $J \not\subseteq [t]$,

$$|vol(\tilde{R}_J) - P_{x\in[m']}[g(x)_J \in \tilde{R}_J]| = 0,$$

because $\overline{R}_j = \emptyset$ for $j \notin [t]$. For $J \subseteq [t]$ with $|J| \leq k$,

$$|vol(\tilde{R}_J) - P_{x\in[m']}[g(x)_J \in \tilde{R}_J]| \leq 2^{|J|}\epsilon',$$

as each $\tilde{R}_J$ is the union of at most $2^{|J|}$ rectangles from $\mathcal{B}(m,|J|,|J|)$. Then

$$\begin{aligned}
|E_{h\in H'}[vol(R^{(h)})] - vol(R)| &\leq \sum_{j=0}^{k} \sum_{J\subseteq[t],|J|=j} 2^j\epsilon' + E_{h\in H'}[\sum_{q\in[d']} \tau_k(h^{-1}(q))] \\
&\leq (\frac{t}{k})^{O(k)}\epsilon' + O(\epsilon) \\
&\leq O(\epsilon).
\end{aligned}$$

Combined with the INW-generator, we get an ϵ-generator using $O(\log\frac{k\log d}{\epsilon} + k\log\frac{t}{k} + \log\frac{1}{\epsilon}\frac{\log 1/\epsilon}{k}) = O(\log k + \log\log d + \log 1/\epsilon + k\log\frac{t}{k} + \frac{\log^2 1/\epsilon}{k})$ bits. Choosing $k = \frac{\log 1/\epsilon}{\log^{1/2}\frac{t}{\log 1/\epsilon}}$ results in $O(\log\log d + \log 1/\epsilon \log^{1/2}\frac{t}{\log 1/\epsilon})$. ∎

6 Acknowledgements

We would like to thank David Barrington for correcting some mistakes and making useful suggestions. We would like to thank Shiyu Zhou for telling us the result in [13] and for some helpful comments. We would also like to thank Amnon Ta-Shma, Jaikumar Radhakrishnan, and Avi Wigderson for reading this paper.

References

[1] N. Alon, O. Goldreich, J. Hastad, and R. Peralta, Simple constructions of almost k-wise independent random variables, *Random Structures and Algorithms*, 3(3), pages 289-303, 1992.

[2] P. Auer, P. Long, and A. Srinivasan, Approximating hyper-rectangles: learning and pseudorandom sets, In *Proceedings of the 29th Annual ACM Symposium on Theory of Computing*, 1997.

[3] R. Armoni, M. Saks, A. Wigderson, and S. Zhou, Discrepancy sets and pseudorandom generators for combinatorial rectangles, In *Proceedings of the 37th Annual IEEE Symposium on Foundations of Computer Science*, pages 412-421, 1996.

[4] S. Chari, P. Rohatgi, and A. Srinivasan, Improved algorithms via approximations of probability distributions, In *Proceedings of the 26th Annual ACM Symposium on Theory of Computing*, pages 584-592, 1994.

[5] G. Even, O. Goldreich, M. Luby, N. Nisan, and B. Velicković, Approximations of general independent distributions. In *Proceedings of the 24th Annual ACM Symposium on Theory of Computing*, pages 10-16, 1992.

[6] O. Goldreich and A. Wigderson, Tiny families of functions with random properties: a quality-size trade-off for hashing, In *Proceedings of the 26th Annual ACM Symposium on Theory of Computing*, pages 574-583, 1994.

[7] R. Impagliazzo, N. Nisan, and A. Wigderson, Pseudorandomness for network algorithms, In *Proceedings of the 26th Annual ACM Symposium on Theory of Computing*, pages 356-364, 1994.

[8] N. Linial, M. Luby, M. Saks, and D. Zuckerman, Efficient construction of a small hitting set for combinatorial rectangles in high dimension, *Combinatorica*, 17, pages 215-234, 1997.

[9] J. Naor and M. Naor, Small-bias probability spaces: efficient constructions and applications, *SIAM Journal on Computing*, 22(4), pages 838-856, 1990.

[10] N. Nisan, Pseudorandom generators for space-bounded computation, *Combinatorica*, 12, pages 449-461, 1992.

[11] N. Nisan, Extracting randomness: how and why - a survey, In *Proceedings of the 11th Annual IEEE Conference on Computational Complexity*, pages 44-58, 1996.

[12] N. Nisan and D. Zuckerman, Randomness is linear in space, *Journal of Computer and System Sciences*, 52(1), pages 43-52, 1996.

[13] J. Radhakrishnan and A. Ta-shma, Private communication.