# What is an Explicit Construction?

Bill Gasarch- U. of MD-College Park gasarch@cs.umd.edu

# Point of This Talk

1. A probabilistic proof shows that something exists but does not show how to find it.
2. Such proofs are often informally called non-explicit.
3. We formalize the notion of explicit.
4. We show that, under HARDness assumptions, many such proofs can be made explicit.
5. These ideas are somewhat folklore.

# Simplifying Assumptions for This Talk

1. We only deal with 2-colorings, though we have results for $c$-colorings.
2. We assume $k$ is large and (if need be) even.
3. We ignore additive constants.

# PART I: A Test Case: Lower Bounds on $W(k)$

**Theorem**

*For every $k$ there exists $W$ such that, for every 2-coloring of $[W]$, there exists a monochromatic arithmetic progression of length $k$ (mono $k$-AP). The least such $W$ is denoted $W(k)$.*

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|---|---|---|---|---|---|---|---|---|
| $R$ | $R$ | $B$ | $B$ | $R$ | $R$ | $B$ | $B$ | $R$ |

No mono 3-AP in coloring of [8], but $1, 5, 9$ is mono 3-AP.

Notes: Known upper bounds on $W(k)$ are Huge!

[GRS,Gow,She,VDW].

# Non-Explicit Lower Bounds on $W(k)$

**Theorem**

$W(k) \geq 2^{k/2}$.

**Proof.**

Let $n = 2^{k/2}$.

$$\text{Prob}(\text{2-coloring of } [n] \text{ has no mono } k\text{-AP}) \geq 1 - \frac{1}{2k} > 0.$$

Hence exists a 2-coloring of $[n]$ with no mono $k$-AP. □

# Rand Alg: Always Fast, Usually Right

We give a Rand. Algorithm to find a proper coloring of $[2^{k/2}]$.
COLOR ALGORITHM

1. $n = 2^{k/2}$.
2. Pick a random 2-coloring $COL$ of $[n]$.
3. If the random 2-coloring is proper then output($COL$). Else output(I AM A FAILURE!!!!).

GOOD NEWS: COLOR runs in $O(n^2)$ time.
BAD NEWS: COLOR sometimes does not return anything.
GOOD NEWS: COLOR is honest about its failure.
GOOD NEWS: Prob(COLOR returns proper col)$\geq 1 - 1/2k$.

# Making Explicit Explicit

**Definition**

An explicit proof that $W(k) \geq f(k)$ is an algorithm that will produce a 2-coloring of $[f(k)]$ that has no mono $k$-AP's, in time poly$(f(k))$.

# PART II: Explicit Lower Bounds

1. Circuit means a family of fanin-2 AND,OR,NOT circuits.
2. There is a circuit for each input size $n$.
3. The size of a circuit is the number of gates it has.
4. Since a circuit is a circuit family, the size is a function $s(n)$.

# Derandomization

## Definition

$s$ poly, $\alpha$ constant. $G : \{0,1\}^* \to \{0,1\}^*$. For all $n$ $G$ restricted to $\{0,1\}^{\alpha \log n}$ has range $\{0,1\}^n$. $G$ is $(s,\alpha)$-*pseudorandom* if for every $s(n)$-sized circuit $C$

$$|\Pr(C(y) = 1 : y \in \{0,1\}^n) - \Pr(C(G(t)) = 1 : t \in \{0,1\}^{\alpha \log n})| < \frac{1}{4}$$

(No $s(n)$-sized circuit can tell the two sets apart, up to $\frac{1}{4}$.)

# HARDness Assumption

**Definition**

If $f : \{0,1\}^* \rightarrow \{0,1\}$ then $s(f)$ is the size of the smallest circuit that computes $f$. $s$ is a function of $n$.

**Definition**

**HARD** is the following assumption:

there exists $f$ computable in time $2^{O(n)}$ such that $s(f) = 2^{\Omega(n)}$.

# HARD $\Longrightarrow$ Exists Pseudorandom

**Lemma**

*Assume* **HARD**. *For all polynomials s there exists an* $\alpha$ *and an* $(s, \alpha)$*-pseudorandom generator G such that*

1. *G restricted to* $\{0,1\}^{\alpha \log n}$ *has range* $\{0,1\}^n$.
2. *G on inputs of length* $\alpha \log n$ *runs in poly(n) (poly in length of output).*

Note: Due to Impagliazzo and Wigderson [IW].

# HARD $\Longrightarrow$ Explicit Lower Bounds (Statement)

**Theorem**

*Assume* **HARD**. $W(k) \geq n = 2^{k/2}$ *Explicitly.*
*(There is an algorithm that will find a proper 2-colorings of* $[n]$ *in time poly(n).)*

# HARD $\Longrightarrow$ Explicit Lower Bounds (Algorithm)

Proof: By Lemma $(\forall s)(\exists \alpha, G)$. $G$ is $(s, \alpha)$-pseudorandom $G : \{0,1\}^{\alpha \log n} \rightarrow \{0,1\}^n$. We pick $s(n)$ later.

COLOR ALGORITHM

1. $n = 2^{k/2}$.
2. For all $t \in \{0,1\}^{\alpha \log n}$ compute $G(t)$. If $G(t)$ is a proper 2-coloring then output$(G(t))$ and HALT. If not then try next one. (Note- the number of $t$ is $O(2^{\alpha \log n}) = O(n^{\alpha})$, a poly.)

Need to show that one of the $t$ works.

# HARD $\Longrightarrow$ Explicit Lower Bounds (Proof)

KEY POINT: There exists $n^2$ sized circuit $C$ that checks if colorings are proper. By Lemma there exists $\alpha$ and $(n^2, \alpha)$-Pseudorandom $G$.

$$|\Pr(C(y) = 1 : y \in \{0,1\}^n) - \Pr(C(G(t)) = 1 : t \in \{0,1\}^{\alpha \log n})| < \frac{1}{4}$$

# HARD $\Longrightarrow$ Explicit Lower Bounds (Proof)

KEY POINT: There exists $n^2$ sized circuit $C$ that checks if colorings are proper. By Lemma there exists $\alpha$ and $(n^2, \alpha)$-Pseudorandom $G$.

$$|\Pr(C(y) = 1 : y \in \{0,1\}^n) - \Pr(C(G(t)) = 1 : t \in \{0,1\}^{\alpha \log n})| < \frac{1}{4}$$

$1 - \frac{1}{k} \geq \frac{3}{4}$ of all colorings of $[n]$ are proper, so

$$\Pr(C(y) = 1 : y \in \{0,1\}^n) \geq 3/4$$

# HARD $\Longrightarrow$ Explicit Lower Bounds (Proof)

KEY POINT: There exists $n^2$ sized circuit $C$ that checks if colorings are proper. By Lemma there exists $\alpha$ and $(n^2, \alpha)$-Pseudorandom $G$.

$$|\Pr(C(y) = 1 : y \in \{0,1\}^n) - \Pr(C(G(t)) = 1 : t \in \{0,1\}^{\alpha \log n})| < \frac{1}{4}$$

$1 - \frac{1}{k} \geq \frac{3}{4}$ of all colorings of $[n]$ are proper, so

$$\Pr(C(y) = 1 : y \in \{0,1\}^n) \geq 3/4$$

Hence

$$\Pr(C(G(t)) = 1 : z \in \{0,1\}^{\alpha \log n}) \geq 3/4 - 1/4 = 1/2 > 0.$$

Hence

$$(\exists t \in \{0,1\}^{\alpha \log n})[C(G(t)) = 1].$$

# Explicit Lower Bound on $W(k)$

Theorem

1. *There is a randomized algorithm that will produce a 2-coloring of $[n]$ (where $n = 2^{k-4}/k$) without any mono k-AP's. The algorithm runs in time poly(n).*
2. **HARD** $\Longrightarrow$ *$W(k) \geq 2^{k-4}/k$ Explicitly.*

Note: Best known lower bounds:
$(\forall \epsilon > 0)(\exists k_0)(\forall k \geq k_0)[W(k) \geq \frac{2^k}{k^\epsilon}]$.Szabo [Sz]. Not Explicit.
Does not generalize to $c$ colors.

# Proof of (1)- The Randomized Algorithm

We give Randomized Algorithm to color $[2^{k-4}/k]$.

COLOR ALGORITHM

1. Color $[n]$ with $n$ random bits.
2. **For** $E \in k\text{-}AP$ if $E$ is mono FIX($E$).

FIX($E$)

1. Recolor $E$ with $k$ random bits.
2. While ($\exists$ mono $D \in k\text{-}AP$ with $D \cap E \neq \emptyset$), FIX($D$).

# If COLOR Halts then it Works!

**Lemma**

*For all k-AP's $E$, after main* **For** *loop looks at $E$, whenever any subsequent call to* **FIX** *from the main* **For** *loop returns, $E$ is not mono.*

**Proof.**

If $E$ is ever made mono, FIX cleans up its own mess, so $E$ will be non-mono when FIX returns. □

**Lemma**

*If COLOR finishes then it has output a proper coloring.*

# Why Does COLOR Usually Work?

PLAN:

1. Let $s$ be a poly in $n$ to be chosen later.
2. View $z \in \{0,1\}^{n+ks}$ as $z_0 z_1 \ldots z_s$, $|z_0| = n$, for $1 \leq i \leq s$, $|z_i| = k$.
3. Can run COLOR using the bits of $z$ if it calls FIX $\leq s$ times.
4. We show that for at least $\frac{3}{4}$ of all $z$, COLOR uses $\leq s$ calls to FIX.

# Most $z$ Work!

Fix $z$ of length $n + ks$ ($s$ later). Run COLOR using $z$. We show the following:

1. If COLOR called FIX $\geq s$ times then from recursion FIX forest, the mono colors, and the final assignment, one can recover $z$.
2. Recursion FIX forest, mono colors, and final assignment can be coded with $n + (3 + \lg(kn))s$ bits.
3. If $s = n^2 + 1$ and $z$ is Kolmogorov Random then FIX is called $\leq n^2$ time (else get short description for $z$).

# Recovering $z$: An Example with $n = 14$, $k = 4$, $s = 2$ (I)

What we know before we see first call to FIX:

| $i$ | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | $z_0^1$ | $z_0^2$ | $z_0^3$ | $z_0^4$ | $z_0^5$ | $z_0^6$ | $z_0^7$ | $z_0^8$ | $z_0^9$ | $z_0^{10}$ | $z_0^{11}$ | $z_0^{12}$ | $z_0^{13}$ | $z_0^{14}$ |

# Recovering $z$: An Example with $n = 14$, $k = 4$, $s = 2$ (I)

What we know before we see first call to FIX:

| $i$ | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | $z_0^1$ | $z_0^2$ | $z_0^3$ | $z_0^4$ | $z_0^5$ | $z_0^6$ | $z_0^7$ | $z_0^8$ | $z_0^9$ | $z_0^{10}$ | $z_0^{11}$ | $z_0^{12}$ | $z_0^{13}$ | $z_0^{14}$ |

First call to FIX is on $(2, 5, 8, 11)$ because they are all $R$.
NOW KNOW: $z_0^2$, $z_0^5$, $z_0^8$, $z_0^{11}$ are all $R$.

| $i$ | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | $z_0^1$ | $z_0^2$ | $z_0^3$ | $z_0^4$ | $z_0^5$ | $z_0^6$ | $z_0^7$ | $z_0^8$ | $z_0^9$ | $z_0^{10}$ | $z_0^{11}$ | $z_0^{12}$ | $z_0^{13}$ | $z_0^{14}$ |

# Recovering $z$: An Example with $n = 14$, $k = 4$, $s = 2$ (I)

What we know before we see first call to FIX:

| $i$ | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | $z_0^1$ | $z_0^2$ | $z_0^3$ | $z_0^4$ | $z_0^5$ | $z_0^6$ | $z_0^7$ | $z_0^8$ | $z_0^9$ | $z_0^{10}$ | $z_0^{11}$ | $z_0^{12}$ | $z_0^{13}$ | $z_0^{14}$ |

First call to FIX is on $(2, 5, 8, 11)$ because they are all $R$.
NOW KNOW: $z_0^2$, $z_0^5$, $z_0^8$, $z_0^{11}$ are all $R$.

| $i$ | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | $z_0^1$ | $z_0^2$ | $z_0^3$ | $z_0^4$ | $z_0^5$ | $z_0^6$ | $z_0^7$ | $z_0^8$ | $z_0^9$ | $z_0^{10}$ | $z_0^{11}$ | $z_0^{12}$ | $z_0^{13}$ | $z_0^{14}$ |

NOW KNOW: $z_1^1$, $z_1^2$, $z_1^3$, $z_1^4$ were used to recolor $(2, 5, 8, 11)$.

| $i$ | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | $z_0^1$ | $z_0^2$ | $z_0^3$ | $z_0^4$ | $z_0^5$ | $z_0^6$ | $z_0^7$ | $z_0^8$ | $z_0^9$ | $z_0^{10}$ | $z_0^{11}$ | $z_0^{12}$ | $z_0^{13}$ | $z_0^{14}$ |
| 1 | $z_0^1$ | $z_1^1$ | $z_0^3$ | $z_0^4$ | $z_1^2$ | $z_0^6$ | $z_0^7$ | $z_1^3$ | $z_0^9$ | $z_0^{10}$ | $z_1^4$ | $z_0^{12}$ | $z_0^{13}$ | $z_0^{14}$ |

# Recovering $z$: An Example with $n = 14$, $k = 4$, $s = 2$ (II)

| $i$ | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | $z_0^1$ | $z_0^2$ | $z_0^3$ | $z_0^4$ | $z_0^5$ | $z_0^6$ | $z_0^7$ | $z_0^8$ | $z_0^9$ | $z_0^{10}$ | $z_0^{11}$ | $z_0^{12}$ | $z_0^{13}$ | $z_0^{14}$ |
| 1 | $z_0^1$ | $z_1^1$ | $z_0^3$ | $z_0^4$ | $z_1^2$ | $z_0^6$ | $z_0^7$ | $z_1^3$ | $z_0^9$ | $z_0^{10}$ | $z_1^4$ | $z_0^{12}$ | $z_0^{13}$ | $z_0^{14}$ |

# Recovering $z$: An Example with $n = 14$, $k = 4$, $s = 2$ (II)

| $i$ | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | $z_0^1$ | $z_0^2$ | $z_0^3$ | $z_0^4$ | $z_0^5$ | $z_0^6$ | $z_0^7$ | $z_0^8$ | $z_0^9$ | $z_0^{10}$ | $z_0^{11}$ | $z_0^{12}$ | $z_0^{13}$ | $z_0^{14}$ |
| 1 | $z_0^1$ | $z_1^1$ | $z_0^3$ | $z_0^4$ | $z_1^2$ | $z_0^6$ | $z_0^7$ | $z_1^3$ | $z_0^9$ | $z_0^{10}$ | $z_1^4$ | $z_0^{12}$ | $z_0^{13}$ | $z_0^{14}$ |

Second call to FIX is on $(2, 3, 4, 5)$ because they are all BLUE.
NOW KNOW: $z_1^1$, $z_0^3$, $z_0^4$, $z_1^2$ are all BLUE.

# Recovering $z$: An Example with $n = 14$, $k = 4$, $s = 2$ (II)

| $i$ | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | $z_0^1$ | $z_0^2$ | $z_0^3$ | $z_0^4$ | $z_0^5$ | $z_0^6$ | $z_0^7$ | $z_0^8$ | $z_0^9$ | $z_0^{10}$ | $z_0^{11}$ | $z_0^{12}$ | $z_0^{13}$ | $z_0^{14}$ |
| 1 | $z_0^1$ | $z_1^1$ | $z_0^3$ | $z_0^4$ | $z_1^2$ | $z_0^6$ | $z_0^7$ | $z_1^3$ | $z_0^9$ | $z_0^{10}$ | $z_1^4$ | $z_0^{12}$ | $z_0^{13}$ | $z_0^{14}$ |

Second call to FIX is on $(2, 3, 4, 5)$ because they are all BLUE.
NOW KNOW: $z_1^1$, $z_0^3$, $z_0^4$, $z_1^2$ are all BLUE.
NOW KNOW: $z_2^1$, $z_2^2$, $z_2^3$, $z_2^4$ were used to recolor $(2, 3, 4, 5)$.

| $i$ | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | $z_0^1$ | $z_0^2$ | $z_0^3$ | $z_0^4$ | $z_0^5$ | $z_0^6$ | $z_0^7$ | $z_0^8$ | $z_0^9$ | $z_0^{10}$ | $z_0^{11}$ | $z_0^{12}$ | $z_0^{13}$ | $z_0^{14}$ |
| 1 | $z_0^1$ | $z_1^1$ | $z_0^3$ | $z_0^4$ | $z_1^2$ | $z_0^6$ | $z_0^7$ | $z_1^3$ | $z_0^9$ | $z_0^{10}$ | $z_1^4$ | $z_0^{12}$ | $z_0^{13}$ | $z_0^{14}$ |
| 2 | $z_0^1$ | $z_2^1$ | $z_2^2$ | $z_2^3$ | $z_2^4$ | $z_0^6$ | $z_0^7$ | $z_1^3$ | $z_0^9$ | $z_0^{10}$ | $z_1^4$ | $z_0^{12}$ | $z_0^{13}$ | $z_0^{14}$ |

KEY: Each call to FIX revealed four bits.

# Recovering $z$: An Example with $n = 14$, $k = 4$, $s = 2$ (III)

| $i$ | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | $z_0^1$ | $z_0^2$ | $z_0^3$ | $z_0^4$ | $z_0^5$ | $z_0^6$ | $z_0^7$ | $z_0^8$ | $z_0^9$ | $z_0^{10}$ | $z_0^{11}$ | $z_0^{12}$ | $z_0^{13}$ | $z_0^{14}$ |
| 1 | $z_0^1$ | $z_1^1$ | $z_0^3$ | $z_0^4$ | $z_1^2$ | $z_0^6$ | $z_0^7$ | $z_1^3$ | $z_0^9$ | $z_0^{10}$ | $z_1^4$ | $z_0^{12}$ | $z_0^{13}$ | $z_0^{14}$ |
| 2 | $z_0^1$ | $z_2^1$ | $z_2^2$ | $z_2^3$ | $z_2^4$ | $z_0^6$ | $z_0^7$ | $z_1^3$ | $z_0^9$ | $z_0^{10}$ | $z_1^4$ | $z_0^{12}$ | $z_0^{13}$ | $z_0^{14}$ |

Final assignment is

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $B$ | $B$ | $B$ | $R$ | $R$ | $B$ | $R$ | $R$ | $B$ | $B$ | $R$ | $R$ | $B$ | $R$ |

NOW KNOW:

| $i$ | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | $z_0^1$ | $z_0^2$ | $z_0^3$ | $z_0^4$ | $z_0^5$ | $z_0^6$ | $z_0^7$ | $z_0^8$ | $z_0^9$ | $z_0^{10}$ | $z_0^{11}$ | $z_0^{12}$ | $z_0^{13}$ | $z_0^{14}$ |
| 1 | $z_0^1$ | $z_1^1$ | $z_0^3$ | $z_0^4$ | $z_1^2$ | $z_0^6$ | $z_0^7$ | $z_1^3$ | $z_0^9$ | $z_0^{10}$ | $z_1^4$ | $z_0^{12}$ | $z_0^{13}$ | $z_0^{14}$ |
| 2 | $z_0^1$ | $z_2^1$ | $z_2^2$ | $z_2^3$ | $z_2^4$ | $z_0^6$ | $z_0^7$ | $z_1^3$ | $z_0^9$ | $z_0^{10}$ | $z_1^4$ | $z_0^{12}$ | $z_0^{13}$ | $z_0^{14}$ |

# Can Code Recursion FIX Forest, Colors, Final Assignment

1. If $E$ is a $k$-AP of $[n]$ then $|\{F : F \cap E \neq \emptyset\}| \leq kn$.
2. Hence all of the labels of the non-roots of the recursion FIX forest can be represented with $\lg(kn)$ bits.
3. Can code rec. forest, colors, and final assignment with $n + n^2 + (3 + \lg(kn))s$ bits.

# If $s = n^2 + 1$ Then Most $z$ Work

If COLOR with $z$ makes $\geq s$ calls to FIX then can describe $z$ with

$$n + n^2 + (3 + \lg(kn))s \text{ bits.}$$

Take $z$ to be Kolm Rand of length $n + ks$.

$$\begin{aligned} n + n^2 + (3 + \lg(kn))s &\geq n + ks \\ n^2 &\geq s \text{ use } n = 2^{k-4}/k \end{aligned}$$

If COLOR is run with Kolmogorov Random $z$ then $\leq n^2$ calls to FIX are made. Most $z$ are Kolm. Rand. Hence for most $z \in \{0,1\}^{n+n^2+1}$ if run COLOR with $z$ will use $\leq n^2$ calls to FIX and thus halt.

# Proof of (2): **HARD** $\Longrightarrow$ $W(k) \geq 2^{k-4}/k$ Explicitly

## EXPLICIT COLORING

By modification of Lemma $(\forall s)(\exists \alpha, G)$. $G$ is $(s, \alpha)$-pseudorandom $G : \{0,1\}^{\alpha \log n} \rightarrow \{0,1\}^{n+km}$. We pick $s(n)$ later.

1. $n = 2^{k-4}/k$.
2. For all $t \in \{0,1\}^{\alpha \log n}$ compute $G(t) \in \{0,1\}^{n+km}$. Run COLOR using $G(t)$ for the random bits. Check if answer is proper coloring. If not, try next $t$. (Note that number of $t$ is $O(2^{\alpha \log n}) = O(n^{\alpha})$.)

Show that one of the $t$ works- similar to proof of **HARD** $\Longrightarrow$ $W(k) \geq 2^{k/2}$ Explicitly! $\square$

# Past and Future History

1. Best unconditional explicit lower bounds on $W(k)$ is $W(k) \geq k2^k$ for $k$ prime due to Berlekamp [Be].
2. Rand Alg based on Moser's STOC 2009 talk [MosTa].
3. Pascal Schweitzer's obtained lower bounds on VDW numbers using Kolm Complexity, but not constructive [Sch].
4. Can be improved to from $2^k/16k$ to $2^k/ek$ by Moser-Tardos [MT] or Beigel (Private communication).
5. For the weaker result, $W(k) \geq 2^{k/2}$ there may be TRUE HARDness assumptions that we can apply such that this result will now be explicit.

# Generalization-Main Theorem

Key to last proof was that two $k$-AP's do not intersect much: If $E$ is a $k$-AP of $[n]$ then $|\{D : D \cap E \neq \emptyset\}| \leq kn$

**Theorem**

*Let $m, k, n \in \mathbb{N}$. Let $H$ be a $k$-uniform hypergraph on $n$ vertices with $m$ edges $E_1, \ldots, E_m$. Assume also that, for all edges $E$,*

$$|\{E_i : E \cap E_i \neq \emptyset\}| \leq 2^{k-4}.$$

*Assume* **HARD**. *Then $H$ can be 2-colored and there is an algorithm to find a proper 2-coloring of $H$ in poly($m$) time.*

# PART IV: Applications: Ramsey Theory and $k$-CNF SAT

We will get (assuming **HARD**)

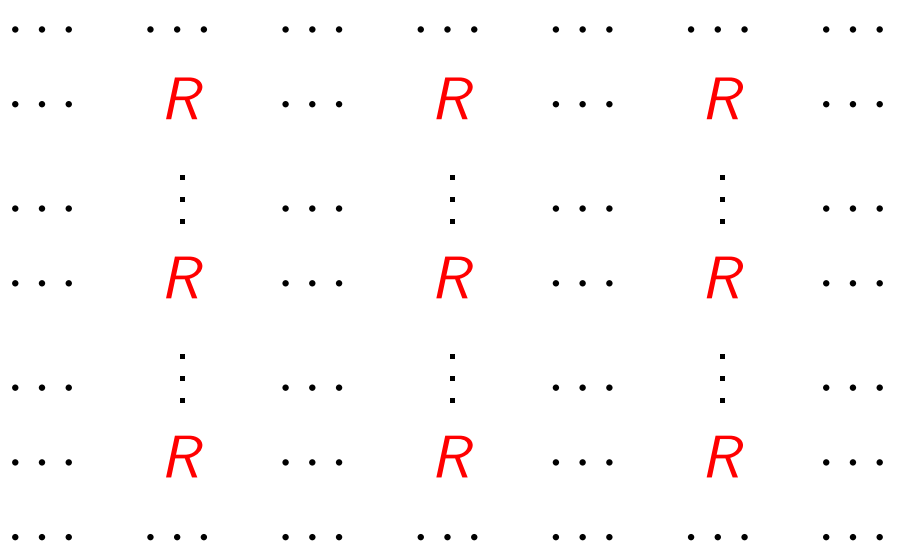
1. Explicit lower bounds on Multi-Dim VDW numbers (from Gallai-Witt Theorem).
2. Explicit lower bounds on Polynomial VDW numbers.
3. A Deterministic Alg for a subcase of $k$-CNF SAT.

# Gallai-Witt Theorem (Subcase)

**Theorem**

*For all $k$ there exists $G = G(k)$ such that for every 2-coloring of $G \times G$ there exists a mono 2-dim grid that is $k \times k$.*

(Example with $k = 3$.)




Notes: Holds in any number of dimensions. Proven by Gallai (reported by Rado [R1,R2]) and Witt [Wi]. Bounds on $G$ are HUGE!

# Explicit Lower Bounds on $G(k)$

**Theorem**

*Let* $k \in \mathbb{N}$. *Assume* **HARD**. $G(k) \geq \frac{2^{k^2-4}}{k^4}$ *Explicitly.*

# Polynomial Van Der Waerden Theorem (subcase)

**Theorem**

*For all $k, g$ there exists $W = (k, g)$ such that any 2-coloring of $[W]$, there exists $a, d \in \mathbb{N}^+$, such that $a, a+d, a+d^2, \ldots, a+d^g$ are all the same color.*

Note: Bounds HUGE. Holds for other sets of polynomials. First proven by Bergelson and Leibman [BL]. See Walters [Wa] for purely combinatorial proof.

# Explicit Lower Bounds on $W(k, g)$

**Theorem**

*Let* $k, g \in \mathbb{N}$. *Assume* **HARD**.

$$W(k, g) \geq \left(2^{k-4}/k\right)^{(1+g)/g}.$$

*Explicitly*

# Explicit Satisfying Assignments

**Theorem**

*Let $\phi = E_1 \wedge \cdots \wedge E_m$ be a $k$-CNF formula such that, for all clauses $E$ of $\phi$, $|\{i : E_i \cap E \neq \emptyset\}| \leq 2^{k-4}$. Assume* **HARD**. *There exists a poly time algorithm that finds a satisfying assignment of $\phi$.*

**Proof.**

Make hypergraph out of $\phi$, clauses are $k$-edges, and $\{x, \neg x\}$ is 2-edge. Use variant of main theorem. □

Note: This was first proven by Moser [MosTa] and improved by Moser-Tardos[MosTa,MT] to $2^k/e$.

# PART V: What About Ramsey Numbers?

The following is folklore.

## Definition

**HARD2** is the following statement: *There is an* $\epsilon > 0$ *and a set* $A \in DTIME(2^{O(n)})$ *such that, every algorithm that decides* $A$ *uses space* $\geq 2^{\epsilon n}$

## Theorem

*Assume* **HARD2**. *Then there is an algorithm that will, given* $n = 2^{k/2}$, *output a set of* $p(n)$ *graphs* ($p$ *some polynomial*) *on* $n$ *vertices such that at least 3/4 of them have neither an ind set of size* $k$ *or a clique or size* $k$.

# PART VI: OPEN QUESTIONS

1. Obtain Explicit lower bounds without HARDness assumption (long standing open problem).
2. Find reasonable HARDness assumption that yields Explicit lower bounds on Ramsey Number $R(k)$.
3. Re-examine other Prob constructions and see if HARDness assumptions makes them explicit. (Especially of interest- Expander Graphs.)

# Bibliography


BL V. Bergelson and A. Leibman. Polynomial extensions of van der Waerden's and Szemerédi's theorems. *Journal of the American Mathematical Society*, pages 725–753, 1996. `http://www.math.ohio-state.edu/~vitaly/` or `http://www.cs.umd.edu/~gasarch/vdw/vdw.html`.

Be E. Berlekamp. A construction for partitions which avoids long arithmetic progressions. *CMB*, 11:409–414, 1968. See `www.cs.umd.edu/~gasarch//vdw/berlekampvdw.pdf`.

# Bibliography (cont)


Gow W. Gowers. A new proof of Szemerédi's theorem. *Geometric and Functional Analysis*, 11:465–588, 2001. `http://www.dpmms.cam.ac.uk/~wtg10/papers/html` or `http://www.springerlink.com`.

GRS R. Graham, B. Rothchild, and J. Spencer. *Ramsey Theory*. Wiley, 1990.

IW R. Impagliazzo and A. Wigderson. Randomness vs time: derandomization under a uniform assumption. *Journal of Computer and System Sciences*, 65:672–694, 2002. Prior version in CCC01. Full Version at `http://www.math.ias.edu/~avi/PUBLICATIONS/`.

MosTa R. Moser. A constructive proof of the general Lovasz local lemma, 2009. This is slides for his talk at STOC 2009, which differs from his paper. `http://www.robinmoser.ch/cplll09stocpr.pdf`

# Bibliography (cont)


MT R. Moser and G. Tardos. A constructive proof of the general Lovasz local lemma, 2009. http://arxiv.org/abs/0903.0544.

R1 R. Rado. Studien zur kombinatorik. *Mathematische Zeitschrift*, pages 424–480, 1933. http://www.cs.umd.edu/~gasarch/vdw/vdw.html.

R2 R. Rado. Notes on combinatorial analysis. *Proceedings of the London Mathematical Society*, pages 122–160, 1943. http://www.cs.umd.edu/~gasarch/vdw/vdw.html.

# Bibliography (cont)


Sch P. Schweitzer. Using the incompressibility method to obtain local lemma results for Ramsey-type problems. *Information Processing Letters*, 109, 2009.

Sh S. Shelah. Primitive recursive bounds for van der Waerden numbers. *Journal of the American Mathematical Society*, pages 683–697, 1988. `http://www.jstor.org/view/08940347/di963031/96p0024f/0`.

Sz Z. Szabo. An application of Lovasz's local lemma— a new lower bound on the van der Waerden numbers. *Random Structures and Algorithms*, 1, 1990. Available at `http://www.cs.umd.edu/~gasarch/vdw/vdw.html`.

VDW B. van der Waerden. Beweis einer Baudetschen Vermutung. *Nieuw Arch. Wisk.*, 15:212–216, 1927.

# Bibliography (cont)


Wa M. Walters. Combinatorial proofs of the polynomial van der Waerden theorem and the polynomial Hales-Jewett theorem. *Journal of the London Mathematical Society*, 61:1–12, 2000. `http://jlms.oxfordjournals.org/cgi/reprint/61/1/1` or `http://jlms.oxfordjournals.org/` or or `http://www.cs.umd.edu/~gasarch/vdw/vdw.html`.

Wi Witt. Ein kombinatorischer satz de elementargeometrie. *Mathematische Nachrichten*, pages 261–262, 1951. `http://www.cs.umd.edu/~gasarch/vdw/vdw.html`.