TUOV
Triangular Unbalanced Oil and Vinegar

Boru Gong

tuovsig@gmail.com, Jintai.Ding@gmail.com

September 4, 2023
Multivariate Signature Scheme

- **Public Key:**
  \[ \mathcal{P}(x_1, \cdots, x_n) = (p_1(x_1, \cdots, x_n), \cdots, p_m(x_1, \cdots, x_n)) , \]

  where each \( p_i \) is a multivariate polynomial over \( \mathbb{F}_q \).

- **Private Key:** a way to compute \( \mathcal{P}^{-1} \).

- **Signing a hash of a document:**
  \[ (x_1, \cdots, x_n) \in \mathcal{P}^{-1}(y_1, \cdots, y_m). \]

- **Verifying:**
  \[ (y_1, \cdots, y_m) \overset{?}{=} \mathcal{P}(x_1, \cdots, x_n). \]
Multivariate Signature Schemes

- The public key $\mathcal{P}(x_1, \cdots, x_n)$ should be *almost surjective*;
  - $n \geq m$ is necessary.

- The signing and verification should be *efficient*;

- Key sizes should be as *small* as possible.
Theoretical Foundation

Direct attack is to solve the set of equations:

$$\mathcal{P}(x_1, \ldots, x_n) = (y'_1, \ldots, y'_m).$$
Efficiency considerations lead to mainly quadratic constructions.

\[ p_\ell(x_1, \ldots, x_n) = \sum_{i,j} \alpha_{ij} x_i x_j + \sum_i \beta_i x_i + \gamma_\ell. \]

Mathematical structure consideration: any set of higher degree polynomial equations can be reduced to a set of quadratic equations.

For instance, \( x_1 x_2 x_3 = 5 \) is equivalent to

\[
\begin{align*}
x_1 x_2 - y &= 0 \\
y x_3 &= 5.
\end{align*}
\]
What is MQ Problem

**MQ Problem**

**Given**: a multivariate quadratic equation system with $m$ equations and $n$ variables over $\mathbb{F}_q$.

**Find**: a solution over $\mathbb{F}_q$. 

Boru Gong

TUOV Triangular Unbalanced Oil and Vinegar
Hardness of MQ Problem

Theorem

1. MQ problem is NP-complete. [Garey-Johnson 1979]
2. MQ problem can be solved in polynomial time, when: $m \leq \frac{n^2}{2}$ [Shamir 1999], when: $n \geq \frac{m^2}{2}$ and $\text{char}(\mathbb{F}_q) = 2$ [Miura-Hashimoto-Takagi 2013].

Moreover, it is believed $\exists 0 < \varepsilon < \frac{1}{2}$ s.t. MQ problem is hard when $n = \alpha m^2$ and $\varepsilon < \alpha < \frac{1}{2}$. 
The original *balanced* version was introduced by Jacques Patarin in 1997.

- Inspired by **linearization attack** to Matsumoto-Imai cryptosystem.
- \( n = 2m \).

Kipnis and Shamir proposed an attack which breaks this balanced OV scheme in 1998.

Kipnis, Patarin and Goubin proposed a modified scheme called *Unbalanced* Oil and Vinegar (UOV) signature scheme in 1999.

- \( n > 2m \).
Let $\mathbb{F} = \mathbb{F}_q$ be a finite field with $q$ elements;

- $v$: the number of \textit{vinegar} variables;
- $m = o$: the number of \textit{oil} variables;
- $n := v + m$: the number of variables.

$V = \{1, \ldots, v\}$, $O = \{v + 1, \ldots, n\}$.
We denote the variables $x_i$ ($i \in V$) as \textit{Vinegar variables},
$x_{v+1}, \ldots, x_n$ as \textit{Oil variables}. 

(Unbalanced) Oil and Vinegar Scheme
(Unbalanced) Oil and Vinegar Scheme

**OV-polynomial**

An \((n, m)\)-**OV-polynomial** \(f\) over \(\mathbb{F}\) is defined as

\[
\sum_{i=1}^{n-m} \sum_{j=1}^{n} \alpha_{i,j} \cdot x_i x_j + \sum_{i=1}^{n} \beta_i \cdot x_i + \gamma
\]

The homogeneous quadratic part of \(f\) can be uniquely represented in a upper-triangular quadratic form:

\[
\begin{bmatrix}
A^{(1)} & A^{(2)} \\
0_{m \times (n-m)} & 0_{m \times m}
\end{bmatrix}
\]

Note that there is no "Oil \(\times\) Oil" part in an OV-polynomial.
Key Generation

UOV Central Map

\[ \mathcal{F} = (f_1, \ldots, f_m), \text{ where each } f_k \text{ is an} \]

\[(n, m)\text{-OV-polynomial, } k = 1, \ldots, m.\]

- **Private Key** is \( (\mathcal{T}, \mathcal{F}) \), where
  - \( \mathcal{F} : \mathbb{F}^n \rightarrow \mathbb{F}^m \) is a UOV central map.
  - \( \mathcal{F} = (f_1, \ldots, f_m) \).
  - \( \mathcal{T} : \mathbb{F}^n \rightarrow \mathbb{F}^n \) is an affine map;

- **Public Key** is \( \mathcal{P} = \mathcal{F} \circ \mathcal{T} \).
Signature Generation

\( z \leftarrow \text{Sign}(\mu): \)

1. Use a hash function \( \mathcal{H} : \{0, 1\}^* \rightarrow \mathbb{F}^m \) to compute a digest \( w = \mathcal{H}(\mu) \).

2. Find a pre-image \( y \in \mathbb{F}^n \) of \( w \) under the central map \( \mathcal{F} \).

3. Compute the signature \( z \in \mathbb{F}^n \) by \( z = \mathcal{T}^{-1}(y) \).
How to find $\mathcal{F}^{-1}$

Inversion of $\mathcal{F}^{-1}$ is efficient indeed:

1. Fix values for vinegar variables $x'_1, \cdots, x'_v$.

2. $f_k = \sum \alpha_{i,j}^{(k)} x_i x'_j + \sum \alpha_{i,j}^{(k)} x'_i x'_j + \sum \beta_i^{(k)} x_i + \sum \beta_i^{(k)} x'_i + \gamma^{(k)}$

3. $\mathcal{F}(x'_1, \cdots, x'_v, \cdots) : \mathbb{F}_q^m \rightarrow \mathbb{F}_q^m$ is a linear system in oil variables $x_{v+1}, \cdots, x_n$. 

Signature verification is fast:

- Check whether $\mathcal{H}(d) = \mathcal{P}(z)$. If so, then the signature $z$ is accepted, otherwise rejected.
Attacks on UOV

- Kipnis-Shamir attack (i.e., UOV attack)
- Reconciliation attack
- Collision attack
- Direct attack
- Intersection attack
- MinRank attack
- Quantum attack
| NIST S.L. | $n$ | $m$ | $q$ | $|\text{epk}|$ (bytes) | $|\text{esk}|$ (bytes) | $|\text{cpk}|$ (bytes) | $|\text{csk}|$ (bytes) | signature (bytes) |
|----------|----|----|----|----------------|----------------|----------------|----------------|-----------------|
| uov-Ip   | 1  | 112| 44 | 256           | 278 432       | 43 576         | 48             | 128             |
| uov-Ia   | 1  | 160| 64 | 16            | 412 160       | 66 576         | 48             | 96              |
| uov-III  | 3  | 184| 72 | 256           | 1 225 440     | 189 232        | 48             | 200             |
| uov-V    | 5  | 244| 96 | 256           | 2 869 440     | 446 992        | 48             | 260             |

**Figure:** Recommended parameter sets and the corresponding key/signature sizes for UOV.
**UOV: Performances**

<table>
<thead>
<tr>
<th></th>
<th>KeyGen</th>
<th>Haswell Sign</th>
<th>Verify</th>
<th>KeyGen</th>
<th>Skylake Sign</th>
<th>Verify</th>
</tr>
</thead>
<tbody>
<tr>
<td>uov-Ip-classic</td>
<td>3311188</td>
<td>116624</td>
<td>82668</td>
<td>2903434</td>
<td>105324</td>
<td>90336</td>
</tr>
<tr>
<td>uov-Ip-pkc</td>
<td>3393872</td>
<td></td>
<td>311720</td>
<td>2858724</td>
<td></td>
<td></td>
</tr>
<tr>
<td>uov-Ip-pkc+skc</td>
<td>3287336</td>
<td>2251440</td>
<td></td>
<td>2848774</td>
<td>1876442</td>
<td></td>
</tr>
<tr>
<td>uov-Ip-classic</td>
<td>4945376</td>
<td>123376</td>
<td>60832</td>
<td>4332050</td>
<td>109314</td>
<td>58274</td>
</tr>
<tr>
<td>uov-Ip-pkc</td>
<td>5002756</td>
<td></td>
<td>398596</td>
<td>4376338</td>
<td></td>
<td></td>
</tr>
<tr>
<td>uov-Ip-pkc+skc</td>
<td>5448272</td>
<td>3042756</td>
<td></td>
<td>4450838</td>
<td>2473254</td>
<td></td>
</tr>
<tr>
<td>uov-II-classic</td>
<td>22046680</td>
<td>346424</td>
<td>275216</td>
<td>17603360</td>
<td>299316</td>
<td>241588</td>
</tr>
<tr>
<td>uov-II-pkc</td>
<td>22389144</td>
<td>1280160</td>
<td></td>
<td>17534058</td>
<td></td>
<td>917402</td>
</tr>
<tr>
<td>uov-II-pkc+skc</td>
<td>21779704</td>
<td>11381092</td>
<td></td>
<td>17157802</td>
<td>9965110</td>
<td></td>
</tr>
<tr>
<td>uov-V-classic</td>
<td>58162124</td>
<td>690752</td>
<td>514100</td>
<td>48480444</td>
<td>591812</td>
<td>470886</td>
</tr>
<tr>
<td>uov-V-pkc</td>
<td>57315504</td>
<td>2842416</td>
<td></td>
<td>46656796</td>
<td></td>
<td>2032992</td>
</tr>
<tr>
<td>uov-V-pkc+skc</td>
<td>57306980</td>
<td>26021784</td>
<td></td>
<td>45492216</td>
<td>22992816</td>
<td></td>
</tr>
<tr>
<td>Dilithium 2†</td>
<td>97621*</td>
<td>281078*</td>
<td>108711*</td>
<td>70548</td>
<td>194892</td>
<td>72633</td>
</tr>
<tr>
<td>Falcon-512‡</td>
<td>19189801*</td>
<td>792360*</td>
<td>103281*</td>
<td>26604000</td>
<td>948132</td>
<td>81036</td>
</tr>
<tr>
<td>SPHINCS+§</td>
<td>1334220</td>
<td>33651546</td>
<td>2150290</td>
<td>1510712*</td>
<td>50084397*</td>
<td>2254495*</td>
</tr>
</tbody>
</table>

† Security level II. ‡ Sphincs+-SHA2-128f-simple. * Data from SUPERCOP [20].

**Figure:** Benchmarking results of AVX2 implementations. Numbers are the median CPU cycles of 1000 executions each.
Why we do TUOV?

- The hardness of the UOV scheme relies on the UOV assumption, i.e., it is hard to find a pre-image of $\mathcal{P}$.

- On the one hand, it is know that

  \[
  \text{find a pre-image of } \mathcal{P} \leq \text{solve MQ problem}.
  \]

- On the other hand, it is not know whether

  \[
  \text{find a pre-image of } \mathcal{P} \overset{?}{=} \text{solve MQ problem}.
  \]

- A large part of coefficients in $\mathcal{F}$ are zeroes, which makes $\mathcal{P} = \mathcal{F} \circ \mathcal{T}$ failed to be proved as random as MQ map.
Why TUOV?

Our solution:
- add some nonzero parts in $F$; and
- keep $F$ efficiently invertible.

Our result: the *Triangular* Unbalanced Oil and Vinegar (TUOV) scheme.
Definition

The **Triangular** in the name TUOV refers to a triangular map (or, *de Jonquiére* map) as

\[
\mathcal{J} : \mathbb{F}_q^n \rightarrow \mathbb{F}_q^n,
\]

\[
x \mapsto (x_1, x_2 + g_2(x_1), \ldots, x_n + g_n(x_1, \ldots, x_{n-1}))
\]

where \( g_i \) is a polynomial over \( \mathbb{F}_q \).

The triangular map \( \mathcal{J} \) is efficiently invertible.
parameters: let $m, v, o_1$ be integers and the number of variables is given by $n = v + o_1 + (m - o_1)$.

index sets:
- $V = \{1, \ldots, v\}$,
- $O_1 = \{v + 1, \ldots, v + o_1\}$,
- $O_2 = \{v + o_1 + 1, \ldots, n\}$.

variables: denote $x_i$ ($i \in V$) as Vinegar variables, and $x_{v+1}, \ldots, x_n$ ($O_1 \cup O_2$) as Oil variables.
**Definition**

For \( d \geq 1 \), an \((n, m, d)\)-**TOV-polynomial** \( f \) over \( \mathbb{F} = \mathbb{F}_q \) is defined as

\[
\sum_{i=n-m+1}^{n-m+d} \sum_{j=n-m+1}^{n-m+d} \alpha_{i,j} \cdot x_i x_j + \sum_{i=1}^{n-m} \sum_{j=1}^{n} \alpha_{i,j} \cdot x_i x_j + \sum_{i=1}^{n} \beta_i \cdot x_i + \gamma.
\]

The homogeneous quadratic part of \( f \):

\[
\begin{bmatrix}
A^{(1)} & A^{(2)} & A^{(3)} \\
0 & A^{(5)} & 0 \\
0 & 0 & 0
\end{bmatrix}
\]

\((n, m, d)\)-TOV-polynomial adds an "\( O_1 \times O_1 (o_1 = d) \)" part.
TUOV Central Map

Definition

A **TUOV central map** with parameters \((n, m, m_1, m_2, q)\) is \(\mathcal{F} = (f_1, \ldots, f_m)\), where \(f_k\) is

\[
\begin{cases}
(n, m)\text{-OV-polynomial}, & k = 1, \ldots, m_1 \\
(n, m, k - m_1)\text{-TOV-polynomial}, & k = m_1 + 1, \ldots, m_2 \\
(n, m - m_2 + m_1 - 1)\text{-OV-polynomial}, & k = m_2 + 1, \ldots, m.
\end{cases}
\]
TUOV Central Map

$m_1$ OV-polynomials

$(m_2-m_1)$ TOV-polynomials

$(m-m_2)$ OV-polynomials

"O_1 x O_1" part

no "O_2 x O_2" part
Key Generation

- **Private Key**: \((S, \mathcal{F}, \mathcal{T})\), where
  - \(S : \mathbb{F}^m \rightarrow \mathbb{F}^m, \mathcal{T} : \mathbb{F}^n \rightarrow \mathbb{F}^n\) are affine maps;
  - \(\mathcal{F} : \mathbb{F}^n \rightarrow \mathbb{F}^m\) is a TUOV central map.

- **Public Key**: \(P = S \circ \mathcal{F} \circ \mathcal{T}\).
Signature Generation

\[ z \leftarrow \text{Sign}(\mu): \]

1. Use a hash function \( \mathcal{H} : \{0, 1\}^* \rightarrow \mathbb{F}^m \) to compute a digest \( w = \mathcal{H}(\mu) \).
2. Compute \( x = S^{-1}(w) \in \mathbb{F}^m \).
3. Find a pre-image \( y \in \mathbb{F}^n \) of \( x \) under the central map \( \mathcal{F} \).
4. Compute the signature \( z \in \mathbb{F}^n \) by \( z = T^{-1}(y) \).
How to find the pre-image under $\mathcal{F}$ (1/2)

**Step 1:** Use one $(n, m)$-OV-polynomial and all TOV-polynomials in $\mathcal{F}$:

\[
\sum_{j=v+1}^{n} L_{1,j}(x_1, \ldots, x_v)x_j + Q_1(x_1, \ldots, x_v) \\
g(x_{v+1}) + \sum_{j=v+1}^{n} L_{2,j}(x_1, \ldots, x_v)x_j + Q_2(x_1, \ldots, x_v) \\
\ldots \\
g(x_{v+1}, \ldots, x_{v+o_1-1}) + \sum_{j=v+1}^{n} L_{o_1,j}(x_1, \ldots, x_v)x_j + Q_{o_1}(x_1, \ldots, x_v)
\]

Solve linear equations of $x_i$ ($i \in V$), so that the **red parts** = 0 and **blue parts** = 1. Substitute the solution, then we can solve $x_i$ ($i \in O_1$) efficiently – by triangular map.
Step 2: Use remaining OV-polynomials in $\mathcal{F}$:
Substitute $x_i$ ($i \in V \cup O_1$), we can get $x_i$ ($i \in O_2$) by solving linear system.

Note
Note that a $(n, m, d)$-TOV-polynomial adds a $(n, m)$-polynomial is still a $(n, m, d)$-TOV-polynomial, thus we can randomly add $(n, m)$-polynomials to $(n, m, d)$-TOV-polynomial to get random $x_i$ ($i \in V$) at the beginning.
Signature Verification

- Check whether $\mathcal{H}(d) = \mathcal{P}(z)$. If so, then the signature $z$ is accepted, otherwise rejected.
Definition: Hardness of MQ problem

The MQ problem parameterized by \((n, m, q)\) is called \((t, \varepsilon)\)-hard, if there exists no algorithm that, given a random MQ-map \(\mathcal{M} : \mathbb{F}^n \to \mathbb{F}^m\), on input \(y := \mathcal{M}(w)\) with \(w \leftarrow \mathbb{F}^n\), outputs \(w'\) such that \(\mathcal{M}(w') = y\) with probability no less than \(\varepsilon\) in processing time \(t\).
TUOV map

TUOV map is \( P = S \circ F \circ T : \mathbb{F}^n \rightarrow \mathbb{F}^m \) where \( S : \mathbb{F}^m \rightarrow \mathbb{F}^m \) and \( T : \mathbb{F}^n \rightarrow \mathbb{F}^n \) are invertible affine transformations and \( F : \mathbb{F}^n \rightarrow \mathbb{F}^m \) is a TUOV central map.

Definition: Hardness of TUOV problem

TUOV problem with params \((n, m, m_1, m_2, q)\) is \((t, \varepsilon)\)-hard if there exists no algorithm that, given a random TUOV map \( P : \mathbb{F}^n \rightarrow \mathbb{F}^m \), on input \( z = P(w) \) with \( w \overset{\$}{\leftarrow} \mathbb{F}^n \), outputs \( w' \) such that \( P(w') = z \) with probability no less than \( \varepsilon \) in processing time \( t \).
Security Proof

**Theorem**

If MQ problem with params $\left(n = \frac{11}{24} \cdot m^2, m, q\right)$ is $(t, \varepsilon)$-hard, then TUOV problem with params $\left(n = \frac{1}{2} \cdot m^2, m, m_1 = \frac{1}{2} \cdot m, m_2 = \frac{3}{4} \cdot m, q\right)$ is $(t, \varepsilon)$-hard.
Our goal is to prove: it is highly probable that for a random MQ-map $\mathcal{M}$, there exists an invertible affine map $Q$ and a TUOV central map $F$ such that $\mathcal{M} = F \circ Q$.

Hence, if there exists an algorithm $A$ that solves TUOV problem, we can construct an algorithm $B^A$ that solves MQ-problem.

Specifically, when taking MQ input $(\mathcal{M}, y)$, $B^A$ can randomly sample an $S$, make query $(P = S \circ \mathcal{M}, z = S(y))$ on $A$ and output the $w'$ returned by $A$. 
Without loss of generality, we consider quadratic polynomials only with their quadratic part.

- Set the matrix representation of $Q^{-1}$ to be

$$Q = \begin{bmatrix} I_{n-m} & Q^{(2)} \\ 0_{m 	imes (n-m)} & I_m \end{bmatrix}. $$

- There are $(n - m) \times m$ unknown variables.
- Recall our goal is to find $\mathcal{F}$ and $Q$ such that $\mathcal{M} = \mathcal{F} \circ Q$.
- Hence we want $\mathcal{M} \circ Q^{-1}$ to be some TUOV central map, i.e. it satisfies some equations.
Proof Sketch

The total number of equations to solve is

\[ \frac{m_1}{2} (m + 1)m + \frac{m_2 - m_1}{6} \left(3m(m + 1) - (m_2 - m_1)^2 - 3(m_2 - m_1) - 2\right) + \frac{m - m_2}{2} (m - m_2 + m_1 - 1)(m - m_2 + m_1). \]

If we pick \( m_1 = \frac{1}{2}m \) and \( m_2 = \frac{3}{4}m \), i.e., \( m_2 - m_1 = \frac{1}{4}m \), then we have roughly \( \frac{11}{24}m^3 \) equations. As long as the number of variables is no less than equations, i.e., \( n \geq \frac{11}{24} \cdot m^2 \), there exists invertible linear transformation \( Q \) such that \( \mathcal{M} \circ Q^{-1} \) is TUOV central map with high probability.
To our knowledge, attacks against UOV is applicable to TUOV, and vice versa.
For efficient implementation, we choose \( m_1 = m_2 \).

| NIST Security Level \((n, m, m_1, q)\) | \(|\text{upk}|\) (bytes) | \(|\text{usk}|\) (bytes) | \(|\text{cpk}|\) (bytes) | \(|\text{csk}|\) (bytes) | \(|\sigma|\) (bytes) |
|-------------------------------|-----------------|-----------------|-----------------|-----------------|-----------------|
| tuov-Ip \((112, 44, 22, 256)\) | 278432          | 239391          | 42608           | 48              | 112             |
| tuov-IIs \((160, 64, 32, 16)\) | 412160          | 350272          | 65552           | 48              | 80              |
| tuov-III \((184, 72, 36, 256)\) | 1225440         | 1048279         | 186640          | 48              | 184             |
| tuov-V \((244, 96, 48, 256)\)  | 2869440         | 2443711         | 442384          | 48              | 244             |

**Figure:** Recommended parameter sets and the corresponding key/signature sizes for TUOV.
## TUOV: Performances

<table>
<thead>
<tr>
<th>Schemes</th>
<th>Optimized Implementations (AVX2)</th>
<th>KeyGen</th>
<th>Sign</th>
<th>Verify</th>
</tr>
</thead>
<tbody>
<tr>
<td>tuov-Ip</td>
<td></td>
<td>10,682,834</td>
<td>220,792</td>
<td>127,722</td>
</tr>
<tr>
<td>tuov-Ip-pkc</td>
<td></td>
<td>10,682,834</td>
<td>220,792</td>
<td>127,722</td>
</tr>
<tr>
<td>tuov-Ip-pkc+skc</td>
<td></td>
<td>6,617,102</td>
<td>6,698,588</td>
<td>491,120</td>
</tr>
<tr>
<td>tuov-Ie</td>
<td></td>
<td>32,007,930</td>
<td>272,394</td>
<td>103,746</td>
</tr>
<tr>
<td>tuov-Ie-pkc</td>
<td></td>
<td>32,007,930</td>
<td>272,394</td>
<td>103,746</td>
</tr>
<tr>
<td>tuov-Ie-pkc+skc</td>
<td></td>
<td>15,635,380</td>
<td>21,534,990</td>
<td>570,194</td>
</tr>
<tr>
<td>Dilithium-II</td>
<td></td>
<td>113,316</td>
<td>272,332</td>
<td>123,916</td>
</tr>
<tr>
<td>tuov-III</td>
<td></td>
<td>57,322,074</td>
<td>608,604</td>
<td>442,770</td>
</tr>
<tr>
<td>tuov-III-pkc</td>
<td></td>
<td>57,322,074</td>
<td>608,604</td>
<td>442,770</td>
</tr>
<tr>
<td>tuov-III-pkc+skc</td>
<td></td>
<td>33,336,974</td>
<td>33,409,538</td>
<td>1,914,056</td>
</tr>
<tr>
<td>Dilithium-III</td>
<td></td>
<td>197,026</td>
<td>448,172</td>
<td>199,656</td>
</tr>
<tr>
<td>tuov-V</td>
<td></td>
<td>139,948,218</td>
<td>1,133,958</td>
<td>786,450</td>
</tr>
<tr>
<td>tuov-V-pkc</td>
<td></td>
<td>139,948,218</td>
<td>1,133,958</td>
<td>786,450</td>
</tr>
<tr>
<td>tuov-V-pkc+skc</td>
<td></td>
<td>85,778,546</td>
<td>74,923,822</td>
<td>4,520,748</td>
</tr>
<tr>
<td>Dilithium-V</td>
<td></td>
<td>303,434</td>
<td>551,760</td>
<td>313,096</td>
</tr>
</tbody>
</table>

**Figure:** Benchmarking results of AVX2 implementations. Numbers are the median CPU cycles of 10000 executions each.
UOV is competitive with the new NIST standards by most measures, except for public key size.

With the triangular structure, TUOV has slower signing speed than UOV, while it gains a reduction to hard problem (MQ-problem when \( n = \alpha m^2 \) where \( \varepsilon < \alpha < 1/2 \)).
Thanks and Any Questions?

tuovsig@gmail.com, Jintai.Ding@gmail.com