Computational
algebraic number theory
tackles lattice-based cryptography

Daniel J. Bernstein
University of Illinois at Chicago &
Technische Universiteit Eindhoven

---

*Moving to the left*
*Moving to the right*
*Big generator*
*Moving through the night*
—Yes, "Big Generator", 1987

---

## The short-generator problem

Take degree-$n$ number field $K$.
i.e. field $K \subseteq \mathbf{C}$ with $\operatorname{len}_{\mathbf{Q}} K = n$.

(Weaker specification: field $K$
with $\mathbf{Q} \subseteq K$ and $\operatorname{len}_{\mathbf{Q}} K = n$.)

e.g. $n = 2$; $K = \mathbf{Q}(i) =$
$\mathbf{Q} \oplus \mathbf{Q}i \hookrightarrow\!\!\!\!\rightarrow \mathbf{Q}[x]/(x^2+1)$.
e.g. $n = 256$; $\zeta = \exp(\pi i/n)$;
$K = \mathbf{Q}(\zeta) \hookrightarrow\!\!\!\!\rightarrow \mathbf{Q}[x]/(x^n+1)$.
e.g. $n = 660$; $\zeta = \exp(2\pi i/661)$;
$K = \mathbf{Q}(\zeta) \hookrightarrow\!\!\!\!\rightarrow \mathbf{Q}[x]/(x^n+\cdots+1)$.
e.g. $K = \mathbf{Q}(\sqrt{2}, \sqrt{3}, \sqrt{5}, \ldots, \sqrt{29})$.

ational
c number theory
attice-based cryptography

. Bernstein
ty of Illinois at Chicago &
che Universiteit Eindhoven

---

*Moving to the left*

*Moving to the right*

*Big generator*

*Moving through the night*

Yes, "Big Generator", 1987

---

## The short-generator problem

Take degree-$n$ number field $K$.
i.e. field $K \subseteq \mathbf{C}$ with $\text{len}_{\mathbf{Q}} K = n$.

(Weaker specification: field $K$ with $\mathbf{Q} \subseteq K$ and $\text{len}_{\mathbf{Q}} K = n$.)

e.g. $n = 2$; $K = \mathbf{Q}(i) = \mathbf{Q} \oplus \mathbf{Q}i \hookrightarrow\!\!\!\!\rightarrow \mathbf{Q}[x]/(x^2+1)$.

e.g. $n = 256$; $\zeta = \exp(\pi i/n)$; $K = \mathbf{Q}(\zeta) \hookrightarrow\!\!\!\!\rightarrow \mathbf{Q}[x]/(x^n+1)$.

e.g. $n = 660$; $\zeta = \exp(2\pi i/661)$; $K = \mathbf{Q}(\zeta) \hookrightarrow\!\!\!\!\rightarrow \mathbf{Q}[x]/(x^n+\cdots+1)$.

e.g. $K = \mathbf{Q}(\sqrt{2}, \sqrt{3}, \sqrt{5}, \ldots, \sqrt{29})$.

Define $\mathcal{O}$
$\mathcal{O} \hookrightarrow\!\!\!\!\rightarrow \mathbf{Z}^n$

Nonzero
factor u
powers o

e.g. $K =$
$\Rightarrow \mathcal{O} =$

e.g. $\zeta =$
$\Rightarrow \mathcal{O} =$

e.g. $\zeta =$
$\Rightarrow \mathcal{O} =$

e.g. $K =$
$\mathbf{Z}[(1+\sqrt$

theory
ed cryptography

n
is at Chicago &
siteit Eindhoven

---

*Moving to the left*
*oving to the right*
*Big generator*
*hrough the night*
Generator", 1987

---

## The short-generator problem

Take degree-$n$ number field $K$.
i.e. field $K \subseteq \mathbf{C}$ with $\text{len}_{\mathbf{Q}} K = n$.

(Weaker specification: field $K$ with $\mathbf{Q} \subseteq K$ and $\text{len}_{\mathbf{Q}} K = n$.)

e.g. $n = 2$; $K = \mathbf{Q}(i) = \mathbf{Q} \oplus \mathbf{Q}i \hookrightarrow\!\!\!\!\rightarrow \mathbf{Q}[x]/(x^2+1)$.

e.g. $n = 256$; $\zeta = \exp(\pi i/n)$; $K = \mathbf{Q}(\zeta) \hookrightarrow\!\!\!\!\rightarrow \mathbf{Q}[x]/(x^n+1)$.

e.g. $n = 660$; $\zeta = \exp(2\pi i/661)$; $K = \mathbf{Q}(\zeta) \hookrightarrow\!\!\!\!\rightarrow \mathbf{Q}[x]/(x^n+\cdots+1)$.

e.g. $K = \mathbf{Q}(\sqrt{2}, \sqrt{3}, \sqrt{5}, \ldots, \sqrt{29})$.

Define $\mathcal{O} = \overline{\mathbf{Z}} \cap K$
$\mathcal{O} \hookrightarrow \mathbf{Z}^n$ as $\mathbf{Z}$-mo

Nonzero ideals of
factor uniquely as
powers of prime id

e.g. $K = \mathbf{Q}(i) \hookrightarrow\!\!\!\!\rightarrow$
$\Rightarrow \mathcal{O} = \mathbf{Z}[i] \hookrightarrow\!\!\!\!\rightarrow \mathbf{Z}$
e.g. $\zeta = \exp(\pi i/2$
$\Rightarrow \mathcal{O} = \mathbf{Z}[\zeta] \hookrightarrow\!\!\!\!\rightarrow \mathbf{Z}$
e.g. $\zeta = \exp(2\pi i/$
$\Rightarrow \mathcal{O} = \mathbf{Z}[\zeta] \hookrightarrow\!\!\!\!\rightarrow \cdots$
e.g. $K = \mathbf{Q}(\sqrt{5}) =$
$\mathbf{Z}[(1+\sqrt{5})/2] \hookrightarrow\!\!\!\!\rightarrow \mathbf{Z}$

graphy

ago &
hoven

---

*the left*
*e right*
*nerator*
*e night*
, 1987

---

## The short-generator problem

Take degree-$n$ number field $K$.
i.e. field $K \subseteq \mathbf{C}$ with $\text{len}_{\mathbf{Q}} K = n$.

(Weaker specification: field $K$
with $\mathbf{Q} \subseteq K$ and $\text{len}_{\mathbf{Q}} K = n$.)

e.g. $n = 2$; $K = \mathbf{Q}(i) =$
$\mathbf{Q} \oplus \mathbf{Q}i \hookrightarrow\!\!\!\!\rightarrow \mathbf{Q}[x]/(x^2 + 1)$.
e.g. $n = 256$; $\zeta = \exp(\pi i/n)$;
$K = \mathbf{Q}(\zeta) \hookrightarrow\!\!\!\!\rightarrow \mathbf{Q}[x]/(x^n + 1)$.
e.g. $n = 660$; $\zeta = \exp(2\pi i/661)$;
$K = \mathbf{Q}(\zeta) \hookrightarrow\!\!\!\!\rightarrow \mathbf{Q}[x]/(x^n + \cdots + 1)$.
e.g. $K = \mathbf{Q}(\sqrt{2}, \sqrt{3}, \sqrt{5}, \ldots, \sqrt{29})$.

Define $\mathcal{O} = \overline{\mathbf{Z}} \cap K$; subring
$\mathcal{O} \hookrightarrow\!\!\!\!\rightarrow \mathbf{Z}^n$ as $\mathbf{Z}$-modules.

Nonzero ideals of $\mathcal{O}$
factor uniquely as products
powers of prime ideals of $\mathcal{O}$.

e.g. $K = \mathbf{Q}(i) \hookrightarrow\!\!\!\!\rightarrow \mathbf{Q}[x]/(x^2$
$\Rightarrow \mathcal{O} = \mathbf{Z}[i] \hookrightarrow\!\!\!\!\rightarrow \mathbf{Z}[x]/(x^2 +$
e.g. $\zeta = \exp(\pi i/256)$, $K =$
$\Rightarrow \mathcal{O} = \mathbf{Z}[\zeta] \hookrightarrow\!\!\!\!\rightarrow \mathbf{Z}[x]/(x^{256}$
e.g. $\zeta = \exp(2\pi i/661)$, $K =$
$\Rightarrow \mathcal{O} = \mathbf{Z}[\zeta] \hookrightarrow\!\!\!\!\rightarrow \cdots$.
e.g. $K = \mathbf{Q}(\sqrt{5}) \Rightarrow \mathcal{O} =$
$\mathbf{Z}[(1+\sqrt{5})/2] \hookrightarrow\!\!\!\!\rightarrow \mathbf{Z}[x]/(x^2 -$

## The short-generator problem

Take degree-$n$ number field $K$.
i.e. field $K \subseteq \mathbf{C}$ with $\text{len}_{\mathbf{Q}} K = n$.

(Weaker specification: field $K$
with $\mathbf{Q} \subseteq K$ and $\text{len}_{\mathbf{Q}} K = n$.)

e.g. $n = 2$; $K = \mathbf{Q}(i) =$
$\mathbf{Q} \oplus \mathbf{Q}i \hookrightarrow\!\!\!\!\rightarrow \mathbf{Q}[x]/(x^2+1)$.
e.g. $n = 256$; $\zeta = \exp(\pi i/n)$;
$K = \mathbf{Q}(\zeta) \hookrightarrow\!\!\!\!\rightarrow \mathbf{Q}[x]/(x^n+1)$.
e.g. $n = 660$; $\zeta = \exp(2\pi i/661)$;
$K = \mathbf{Q}(\zeta) \hookrightarrow\!\!\!\!\rightarrow \mathbf{Q}[x]/(x^n+\cdots+1)$.
e.g. $K = \mathbf{Q}(\sqrt{2}, \sqrt{3}, \sqrt{5}, \ldots, \sqrt{29})$.

Define $\mathcal{O} = \overline{\mathbf{Z}} \cap K$; subring of $K$.
$\mathcal{O} \hookrightarrow\!\!\!\!\rightarrow \mathbf{Z}^n$ as $\mathbf{Z}$-modules.

Nonzero ideals of $\mathcal{O}$
factor uniquely as products of
powers of prime ideals of $\mathcal{O}$.

e.g. $K = \mathbf{Q}(i) \hookrightarrow\!\!\!\!\rightarrow \mathbf{Q}[x]/(x^2+1)$
$\Rightarrow \mathcal{O} = \mathbf{Z}[i] \hookrightarrow\!\!\!\!\rightarrow \mathbf{Z}[x]/(x^2+1)$.
e.g. $\zeta = \exp(\pi i/256)$, $K = \mathbf{Q}(\zeta)$
$\Rightarrow \mathcal{O} = \mathbf{Z}[\zeta] \hookrightarrow\!\!\!\!\rightarrow \mathbf{Z}[x]/(x^{256}+1)$.
e.g. $\zeta = \exp(2\pi i/661)$, $K = \mathbf{Q}(\zeta)$
$\Rightarrow \mathcal{O} = \mathbf{Z}[\zeta] \hookrightarrow\!\!\!\!\rightarrow \cdots$.
e.g. $K = \mathbf{Q}(\sqrt{5}) \Rightarrow \mathcal{O} =$
$\mathbf{Z}[(1+\sqrt{5})/2] \hookrightarrow\!\!\!\!\rightarrow \mathbf{Z}[x]/(x^2-x-1)$.

rt-generator problem

gree-$n$ number field $K$.
$K \subseteq \mathbf{C}$ with $\text{len}_{\mathbf{Q}} K = n$.

specification: field $K$
$\subseteq K$ and $\text{len}_{\mathbf{Q}} K = n$.)

2; $K = \mathbf{Q}(i) =$
$\twoheadrightarrow \mathbf{Q}[x]/(x^2+1)$.

256; $\zeta = \exp(\pi i/n)$;
$\zeta) \twoheadrightarrow \mathbf{Q}[x]/(x^n+1)$.

660; $\zeta = \exp(2\pi i/661)$;
$\zeta) \twoheadrightarrow \mathbf{Q}[x]/(x^n+\cdots+1)$.
$= \mathbf{Q}(\sqrt{2}, \sqrt{3}, \sqrt{5}, \ldots, \sqrt{29})$.

Define $\mathcal{O} = \overline{\mathbf{Z}} \cap K$; subring of $K$.
$\mathcal{O} \hookrightarrow \mathbf{Z}^n$ as $\mathbf{Z}$-modules.

Nonzero ideals of $\mathcal{O}$
factor uniquely as products of
powers of prime ideals of $\mathcal{O}$.

e.g. $K = \mathbf{Q}(i) \twoheadrightarrow \mathbf{Q}[x]/(x^2+1)$
$\Rightarrow \mathcal{O} = \mathbf{Z}[i] \twoheadrightarrow \mathbf{Z}[x]/(x^2+1)$.
e.g. $\zeta = \exp(\pi i/256)$, $K = \mathbf{Q}(\zeta)$
$\Rightarrow \mathcal{O} = \mathbf{Z}[\zeta] \twoheadrightarrow \mathbf{Z}[x]/(x^{256}+1)$.
e.g. $\zeta = \exp(2\pi i/661)$, $K = \mathbf{Q}(\zeta)$
$\Rightarrow \mathcal{O} = \mathbf{Z}[\zeta] \twoheadrightarrow \cdots$.
e.g. $K = \mathbf{Q}(\sqrt{5}) \Rightarrow \mathcal{O} =$
$\mathbf{Z}[(1+\sqrt{5})/2] \twoheadrightarrow \mathbf{Z}[x]/(x^2-x-1)$.

The sho
Find "sh
given th

e.g. $\zeta =$
$\mathcal{O} = \mathbf{Z}[\zeta$
The $\mathbf{Z}$-s
$201 - 23$
$935 - 10$
$979 - 11$
$718 - 82$
is an ide
Can you
such tha

## or problem

mber field $K$.
ith $\text{len}_{\mathbf{Q}} K = n$.

tion: field $K$
$\text{en}_{\mathbf{Q}} K = n$.)

$\mathbf{Q}(i) =$
$x^2+1)$.
$\exp(\pi i/n)$;
$]/(x^n+1)$.
$\exp(2\pi i/661)$;
$]/(x^n+\cdots+1)$.
$\sqrt{3}, \sqrt{5}, \ldots, \sqrt{29})$.

Define $\mathcal{O} = \overline{\mathbf{Z}} \cap K$; subring of $K$.
$\mathcal{O} \hookrightarrow \mathbf{Z}^n$ as $\mathbf{Z}$-modules.

Nonzero ideals of $\mathcal{O}$
factor uniquely as products of
powers of prime ideals of $\mathcal{O}$.

e.g. $K = \mathbf{Q}(i) \hookrightarrow \mathbf{Q}[x]/(x^2+1)$
$\Rightarrow \mathcal{O} = \mathbf{Z}[i] \hookrightarrow \mathbf{Z}[x]/(x^2+1)$.
e.g. $\zeta = \exp(\pi i/256)$, $K = \mathbf{Q}(\zeta)$
$\Rightarrow \mathcal{O} = \mathbf{Z}[\zeta] \hookrightarrow \mathbf{Z}[x]/(x^{256}+1)$.
e.g. $\zeta = \exp(2\pi i/661)$, $K = \mathbf{Q}(\zeta)$
$\Rightarrow \mathcal{O} = \mathbf{Z}[\zeta] \hookrightarrow \cdots$.
e.g. $K = \mathbf{Q}(\sqrt{5}) \Rightarrow \mathcal{O} =$
$\mathbf{Z}[(1+\sqrt{5})/2] \hookrightarrow \mathbf{Z}[x]/(x^2-x-1)$.

The short-generate
Find "short" nonz
given the principal

e.g. $\zeta = \exp(\pi i/4)$
$\mathcal{O} = \mathbf{Z}[\zeta] \hookrightarrow \mathbf{Z}[x]/$
The $\mathbf{Z}$-submodule
$201 - 233\zeta - 430$
$935 - 1063\zeta - 198$
$979 - 1119\zeta - 209$
$718 - 829\zeta - 153$
is an ideal $I$ of $\mathcal{O}$.
Can you find a sho
such that $I = g\mathcal{O}$

$\underline{n}$

$K$.

$= n$.

$K$

$n$.)

);

.

661);

$\cdot + 1)$.

$\sqrt{29})$.

Define $\mathcal{O} = \overline{\mathbf{Z}} \cap K$; subring of $K$.
$\mathcal{O} \hookrightarrow\!\!\!\!\rightarrow \mathbf{Z}^n$ as $\mathbf{Z}$-modules.

Nonzero ideals of $\mathcal{O}$
factor uniquely as products of
powers of prime ideals of $\mathcal{O}$.

e.g. $K = \mathbf{Q}(i) \hookrightarrow\!\!\!\!\rightarrow \mathbf{Q}[x]/(x^2+1)$
$\Rightarrow \mathcal{O} = \mathbf{Z}[i] \hookrightarrow\!\!\!\!\rightarrow \mathbf{Z}[x]/(x^2+1)$.
e.g. $\zeta = \exp(\pi i/256)$, $K = \mathbf{Q}(\zeta)$
$\Rightarrow \mathcal{O} = \mathbf{Z}[\zeta] \hookrightarrow\!\!\!\!\rightarrow \mathbf{Z}[x]/(x^{256}+1)$.
e.g. $\zeta = \exp(2\pi i/661)$, $K = \mathbf{Q}(\zeta)$
$\Rightarrow \mathcal{O} = \mathbf{Z}[\zeta] \hookrightarrow\!\!\!\!\rightarrow \cdots$.
e.g. $K = \mathbf{Q}(\sqrt{5}) \Rightarrow \mathcal{O} =$
$\mathbf{Z}[(1+\sqrt{5})/2] \hookrightarrow\!\!\!\!\rightarrow \mathbf{Z}[x]/(x^2-x-1)$.

The short-generator problem
Find "short" nonzero $g \in \mathcal{O}$
given the principal ideal $g\mathcal{O}$

e.g. $\zeta = \exp(\pi i/4)$; $K = \mathbf{Q}($
$\mathcal{O} = \mathbf{Z}[\zeta] \hookrightarrow\!\!\!\!\rightarrow \mathbf{Z}[x]/(x^4+1)$.
The $\mathbf{Z}$-submodule of $\mathcal{O}$ gen
$201 - 233\zeta - 430\zeta^2 - 712\zeta^3$
$935 - 1063\zeta - 1986\zeta^2 - 329$
$979 - 1119\zeta - 2092\zeta^2 - 347$
$718 - 829\zeta - 1537\zeta^2 - 254($
is an ideal $I$ of $\mathcal{O}$.
Can you find a short $g \in \mathcal{O}$
such that $I = g\mathcal{O}$?

Define $\mathcal{O} = \overline{\mathbf{Z}} \cap K$; subring of $K$.
$\mathcal{O} \hookrightarrow\!\!\!\!\rightarrow \mathbf{Z}^n$ as $\mathbf{Z}$-modules.

Nonzero ideals of $\mathcal{O}$
factor uniquely as products of
powers of prime ideals of $\mathcal{O}$.

e.g. $K = \mathbf{Q}(i) \hookrightarrow\!\!\!\!\rightarrow \mathbf{Q}[x]/(x^2+1)$
$\Rightarrow \mathcal{O} = \mathbf{Z}[i] \hookrightarrow\!\!\!\!\rightarrow \mathbf{Z}[x]/(x^2+1)$.
e.g. $\zeta = \exp(\pi i/256)$, $K = \mathbf{Q}(\zeta)$
$\Rightarrow \mathcal{O} = \mathbf{Z}[\zeta] \hookrightarrow\!\!\!\!\rightarrow \mathbf{Z}[x]/(x^{256}+1)$.
e.g. $\zeta = \exp(2\pi i/661)$, $K = \mathbf{Q}(\zeta)$
$\Rightarrow \mathcal{O} = \mathbf{Z}[\zeta] \hookrightarrow\!\!\!\!\rightarrow \cdots$.
e.g. $K = \mathbf{Q}(\sqrt{5}) \Rightarrow \mathcal{O} =$
$\mathbf{Z}[(1+\sqrt{5})/2] \hookrightarrow\!\!\!\!\rightarrow \mathbf{Z}[x]/(x^2-x-1)$.

The short-generator problem:
Find "short" nonzero $g \in \mathcal{O}$
given the principal ideal $g\mathcal{O}$.

e.g. $\zeta = \exp(\pi i/4)$; $K = \mathbf{Q}(\zeta)$;
$\mathcal{O} = \mathbf{Z}[\zeta] \hookrightarrow\!\!\!\!\rightarrow \mathbf{Z}[x]/(x^4+1)$.
The $\mathbf{Z}$-submodule of $\mathcal{O}$ gen by
$201 - 233\zeta - 430\zeta^2 - 712\zeta^3$,
$935 - 1063\zeta - 1986\zeta^2 - 3299\zeta^3$,
$979 - 1119\zeta - 2092\zeta^2 - 3470\zeta^3$,
$718 - 829\zeta - 1537\zeta^2 - 2546\zeta^3$
is an ideal $I$ of $\mathcal{O}$.
Can you find a short $g \in \mathcal{O}$
such that $I = g\mathcal{O}$?

$\mathcal{O} = \overline{\mathbf{Z}} \cap K$; subring of $K$.
$^n$ as $\mathbf{Z}$-modules.

ideals of $\mathcal{O}$
niquely as products of
of prime ideals of $\mathcal{O}$.

$= \mathbf{Q}(i) \hookrightarrow \mathbf{Q}[x]/(x^2+1)$
$\mathbf{Z}[i] \hookrightarrow \mathbf{Z}[x]/(x^2+1)$.
$\exp(\pi i/256)$, $K = \mathbf{Q}(\zeta)$
$\mathbf{Z}[\zeta] \hookrightarrow \mathbf{Z}[x]/(x^{256}+1)$.
$\exp(2\pi i/661)$, $K = \mathbf{Q}(\zeta)$
$\mathbf{Z}[\zeta] \hookrightarrow \cdots$.
$= \mathbf{Q}(\sqrt{5}) \Rightarrow \mathcal{O} =$
$\sqrt{5})/2] \hookrightarrow \mathbf{Z}[x]/(x^2-x-1)$.

The short-generator problem:
Find "short" nonzero $g \in \mathcal{O}$
given the principal ideal $g\mathcal{O}$.

e.g. $\zeta = \exp(\pi i/4)$; $K = \mathbf{Q}(\zeta)$;
$\mathcal{O} = \mathbf{Z}[\zeta] \hookrightarrow \mathbf{Z}[x]/(x^4+1)$.
The $\mathbf{Z}$-submodule of $\mathcal{O}$ gen by
$201 - 233\zeta - 430\zeta^2 - 712\zeta^3$,
$935 - 1063\zeta - 1986\zeta^2 - 3299\zeta^3$,
$979 - 1119\zeta - 2092\zeta^2 - 3470\zeta^3$,
$718 - 829\zeta - 1537\zeta^2 - 2546\zeta^3$
is an ideal $I$ of $\mathcal{O}$.
Can you find a short $g \in \mathcal{O}$
such that $I = g\mathcal{O}$?

The latt

Use LLL
short ele
$\mathbf{Z}A + \mathbf{Z}$
$A = (20$
$B = (93$
$C = (97$
$D = (71$

$\zeta$; subring of $K$.
dules.

$\mathcal{O}$
products of
leals of $\mathcal{O}$.

$\mathbf{Q}[x]/(x^2+1)$
$[x]/(x^2+1)$.
56), $K = \mathbf{Q}(\zeta)$
$[x]/(x^{256}+1)$.
661), $K = \mathbf{Q}(\zeta)$
$\ddots$
$\Rightarrow \mathcal{O} =$
$\mathbf{Z}[x]/(x^2-x-1)$.

The short-generator problem:
Find "short" nonzero $g \in \mathcal{O}$
given the principal ideal $g\mathcal{O}$.

e.g. $\zeta = \exp(\pi i/4)$; $K = \mathbf{Q}(\zeta)$;
$\mathcal{O} = \mathbf{Z}[\zeta] \twoheadrightarrow \mathbf{Z}[x]/(x^4+1)$.
The $\mathbf{Z}$-submodule of $\mathcal{O}$ gen by
$201 - 233\zeta - 430\zeta^2 - 712\zeta^3$,
$935 - 1063\zeta - 1986\zeta^2 - 3299\zeta^3$,
$979 - 1119\zeta - 2092\zeta^2 - 3470\zeta^3$,
$718 - 829\zeta - 1537\zeta^2 - 2546\zeta^3$
is an ideal $I$ of $\mathcal{O}$.
Can you find a short $g \in \mathcal{O}$
such that $I = g\mathcal{O}$?

The lattice perspe

Use LLL to quickly
short elements of
$\mathbf{Z}A + \mathbf{Z}B + \mathbf{Z}C +$
$A = (201, -233, -$
$B = (935, -1063,$
$C = (979, -1119,$
$D = (718, -829, -$

of $K$.

of

$+1)$

$1)$.

$\mathbf{Q}(\zeta)$

$+1)$.

$\mathbf{Q}(\zeta)$

$+x-1)$.

The short-generator problem:
Find "short" nonzero $g \in \mathcal{O}$
given the principal ideal $g\mathcal{O}$.

e.g. $\zeta = \exp(\pi i/4)$; $K = \mathbf{Q}(\zeta)$;
$\mathcal{O} = \mathbf{Z}[\zeta] \twoheadrightarrow \mathbf{Z}[x]/(x^4+1)$.
The $\mathbf{Z}$-submodule of $\mathcal{O}$ gen by
$201 - 233\zeta - 430\zeta^2 - 712\zeta^3$,
$935 - 1063\zeta - 1986\zeta^2 - 3299\zeta^3$,
$979 - 1119\zeta - 2092\zeta^2 - 3470\zeta^3$,
$718 - 829\zeta - 1537\zeta^2 - 2546\zeta^3$
is an ideal $I$ of $\mathcal{O}$.
Can you find a short $g \in \mathcal{O}$
such that $I = g\mathcal{O}$?

<u>The lattice perspective</u>

Use LLL to quickly find
short elements of lattice
$\mathbf{Z}A + \mathbf{Z}B + \mathbf{Z}C + \mathbf{Z}D$ where
$A = (201, -233, -430, -71$
$B = (935, -1063, -1986, -$
$C = (979, -1119, -2092, -$
$D = (718, -829, -1537, -2$

The short-generator problem:
Find "short" nonzero $g \in \mathcal{O}$
given the principal ideal $g\mathcal{O}$.

e.g. $\zeta = \exp(\pi i/4)$; $K = \mathbf{Q}(\zeta)$;
$\mathcal{O} = \mathbf{Z}[\zeta] \hookrightarrow\!\!\!\!\rightarrow \mathbf{Z}[x]/(x^4+1)$.
The $\mathbf{Z}$-submodule of $\mathcal{O}$ gen by
$201 - 233\zeta - 430\zeta^2 - 712\zeta^3$,
$935 - 1063\zeta - 1986\zeta^2 - 3299\zeta^3$,
$979 - 1119\zeta - 2092\zeta^2 - 3470\zeta^3$,
$718 - 829\zeta - 1537\zeta^2 - 2546\zeta^3$
is an ideal $I$ of $\mathcal{O}$.
Can you find a short $g \in \mathcal{O}$
such that $I = g\mathcal{O}$?

The lattice perspective

Use LLL to quickly find
short elements of lattice
$\mathbf{Z}A + \mathbf{Z}B + \mathbf{Z}C + \mathbf{Z}D$ where
$A = (201, -233, -430, -712)$,
$B = (935, -1063, -1986, -3299)$,
$C = (979, -1119, -2092, -3470)$,
$D = (718, -829, -1537, -2546)$.

The short-generator problem:
Find "short" nonzero $g \in \mathcal{O}$
given the principal ideal $g\mathcal{O}$.

e.g. $\zeta = \exp(\pi i/4)$; $K = \mathbf{Q}(\zeta)$;
$\mathcal{O} = \mathbf{Z}[\zeta] \hookrightarrow\!\!\!\rightarrow \mathbf{Z}[x]/(x^4+1)$.
The $\mathbf{Z}$-submodule of $\mathcal{O}$ gen by
$201 - 233\zeta - 430\zeta^2 - 712\zeta^3$,
$935 - 1063\zeta - 1986\zeta^2 - 3299\zeta^3$,
$979 - 1119\zeta - 2092\zeta^2 - 3470\zeta^3$,
$718 - 829\zeta - 1537\zeta^2 - 2546\zeta^3$
is an ideal $I$ of $\mathcal{O}$.
Can you find a short $g \in \mathcal{O}$
such that $I = g\mathcal{O}$?

## The lattice perspective

Use LLL to quickly find
short elements of lattice
$\mathbf{Z}A + \mathbf{Z}B + \mathbf{Z}C + \mathbf{Z}D$ where
$A = (201, -233, -430, -712)$,
$B = (935, -1063, -1986, -3299)$,
$C = (979, -1119, -2092, -3470)$,
$D = (718, -829, -1537, -2546)$.

Find $(3, 1, 4, 1)$ as
$-37A + 3B - 7C + 16D$.
This was my original $g$.

The short-generator problem:
Find "short" nonzero $g \in \mathcal{O}$
given the principal ideal $g\mathcal{O}$.

e.g. $\zeta = \exp(\pi i/4)$; $K = \mathbf{Q}(\zeta)$;
$\mathcal{O} = \mathbf{Z}[\zeta] \hookrightarrow\!\!\!\rightarrow \mathbf{Z}[x]/(x^4+1)$.
The $\mathbf{Z}$-submodule of $\mathcal{O}$ gen by
$201 - 233\zeta - 430\zeta^2 - 712\zeta^3$,
$935 - 1063\zeta - 1986\zeta^2 - 3299\zeta^3$,
$979 - 1119\zeta - 2092\zeta^2 - 3470\zeta^3$,
$718 - 829\zeta - 1537\zeta^2 - 2546\zeta^3$
is an ideal $I$ of $\mathcal{O}$.
Can you find a short $g \in \mathcal{O}$
such that $I = g\mathcal{O}$?

## The lattice perspective

Use LLL to quickly find
short elements of lattice
$\mathbf{Z}A + \mathbf{Z}B + \mathbf{Z}C + \mathbf{Z}D$ where
$A = (201, -233, -430, -712)$,
$B = (935, -1063, -1986, -3299)$,
$C = (979, -1119, -2092, -3470)$,
$D = (718, -829, -1537, -2546)$.

Find $(3, 1, 4, 1)$ as
$-37A + 3B - 7C + 16D$.
This was my original $g$.

Also find, e.g., $(-4, -1, 3, 1)$.
Multiplying by root of unity
(here $\zeta^2$) preserves shortness.

rt-generator problem:
hort" nonzero $g \in \mathcal{O}$
e principal ideal $g\mathcal{O}$.

$\exp(\pi i/4)$; $K = \mathbf{Q}(\zeta)$;
$\zeta] \twoheadrightarrow \mathbf{Z}[x]/(x^4+1)$.
ubmodule of $\mathcal{O}$ gen by
$33\zeta - 430\zeta^2 - 712\zeta^3$,
$063\zeta - 1986\zeta^2 - 3299\zeta^3$,
$119\zeta - 2092\zeta^2 - 3470\zeta^3$,
$29\zeta - 1537\zeta^2 - 2546\zeta^3$
eal $I$ of $\mathcal{O}$.
find a short $g \in \mathcal{O}$
t $I = g\mathcal{O}$?

## The lattice perspective

Use LLL to quickly find
short elements of lattice
$\mathbf{Z}A + \mathbf{Z}B + \mathbf{Z}C + \mathbf{Z}D$ where
$A = (201, -233, -430, -712)$,
$B = (935, -1063, -1986, -3299)$,
$C = (979, -1119, -2092, -3470)$,
$D = (718, -829, -1537, -2546)$.

Find $(3, 1, 4, 1)$ as
$-37A + 3B - 7C + 16D$.
This was my original $g$.

Also find, e.g., $(-4, -1, 3, 1)$.
Multiplying by root of unity
(here $\zeta^2$) preserves shortness.

For muc

LLL alm
Big gap
and size
that LLL

or problem:
ero $g \in \mathcal{O}$
ideal $g\mathcal{O}$.

); $K = \mathbf{Q}(\zeta)$;
$/(x^4 + 1)$.
of $\mathcal{O}$ gen by
$\zeta^2 - 712\zeta^3$,
$86\zeta^2 - 3299\zeta^3$,
$92\zeta^2 - 3470\zeta^3$,
$7\zeta^2 - 2546\zeta^3$

ort $g \in \mathcal{O}$
?

The lattice perspective

Use LLL to quickly find
short elements of lattice
$\mathbf{Z}A + \mathbf{Z}B + \mathbf{Z}C + \mathbf{Z}D$ where
$A = (201, -233, -430, -712)$,
$B = (935, -1063, -1986, -3299)$,
$C = (979, -1119, -2092, -3470)$,
$D = (718, -829, -1537, -2546)$.

Find $(3, 1, 4, 1)$ as
$-37A + 3B - 7C + 16D$.
This was my original $g$.

Also find, e.g., $(-4, -1, 3, 1)$.
Multiplying by root of unity
(here $\zeta^2$) preserves shortness.

For much larger $n$

LLL almost never
Big gap between s
and size of "short"
that LLL typically

n:
)
.
ζ);
by
$^3$,
$99\zeta^3$,
$70\zeta^3$,
$6\zeta^3$

## The lattice perspective

Use LLL to quickly find
short elements of lattice
$\mathbf{Z}A + \mathbf{Z}B + \mathbf{Z}C + \mathbf{Z}D$ where
$A = (201, -233, -430, -712)$,
$B = (935, -1063, -1986, -3299)$,
$C = (979, -1119, -2092, -3470)$,
$D = (718, -829, -1537, -2546)$.

Find $(3, 1, 4, 1)$ as
$-37A + 3B - 7C + 16D$.
This was my original $g$.

Also find, e.g., $(-4, -1, 3, 1)$.
Multiplying by root of unity
(here $\zeta^2$) preserves shortness.

For much larger $n$:

LLL almost never finds $g$.
Big gap between size of $g$
and size of "short" vectors
that LLL typically finds in $I$.

## The lattice perspective

Use LLL to quickly find
short elements of lattice
$\mathbf{Z}A + \mathbf{Z}B + \mathbf{Z}C + \mathbf{Z}D$ where
$A = (201, -233, -430, -712)$,
$B = (935, -1063, -1986, -3299)$,
$C = (979, -1119, -2092, -3470)$,
$D = (718, -829, -1537, -2546)$.

Find $(3, 1, 4, 1)$ as
$-37A + 3B - 7C + 16D$.
This was my original $g$.

Also find, e.g., $(-4, -1, 3, 1)$.
Multiplying by root of unity
(here $\zeta^2$) preserves shortness.

For much larger $n$:

LLL almost never finds $g$.
Big gap between size of $g$
and size of "short" vectors
that LLL typically finds in $I$.

## The lattice perspective

Use LLL to quickly find
short elements of lattice
$\mathbf{Z}A + \mathbf{Z}B + \mathbf{Z}C + \mathbf{Z}D$ where
$A = (201, -233, -430, -712)$,
$B = (935, -1063, -1986, -3299)$,
$C = (979, -1119, -2092, -3470)$,
$D = (718, -829, -1537, -2546)$.

Find $(3, 1, 4, 1)$ as
$-37A + 3B - 7C + 16D$.
This was my original $g$.

Also find, e.g., $(-4, -1, 3, 1)$.
Multiplying by root of unity
(here $\zeta^2$) preserves shortness.

For much larger $n$:

LLL almost never finds $g$.
Big gap between size of $g$
and size of "short" vectors
that LLL typically finds in $I$.

Increased BKZ block size:
reduced gap but slower.

## The lattice perspective

Use LLL to quickly find
short elements of lattice
$\mathbf{Z}A + \mathbf{Z}B + \mathbf{Z}C + \mathbf{Z}D$ where
$A = (201, -233, -430, -712)$,
$B = (935, -1063, -1986, -3299)$,
$C = (979, -1119, -2092, -3470)$,
$D = (718, -829, -1537, -2546)$.

Find $(3, 1, 4, 1)$ as
$-37A + 3B - 7C + 16D$.
This was my original $g$.

Also find, e.g., $(-4, -1, 3, 1)$.
Multiplying by root of unity
(here $\zeta^2$) preserves shortness.

For much larger $n$:

LLL almost never finds $g$.
Big gap between size of $g$
and size of "short" vectors
that LLL typically finds in $I$.

Increased BKZ block size:
reduced gap but slower.

Fancier lattice algorithms:
Under reasonable assumptions,
2015 Laarhoven–de Weger
finds $g$ in time $\approx 1.23^n$.
Big progress compared to, e.g.,
2008 Nguyen–Vidick ($\approx 1.33^n$)
but still exponential time.

ice perspective

to quickly find
ements of lattice
$B + \mathbf{Z}C + \mathbf{Z}D$ where
$01, -233, -430, -712)$,
$35, -1063, -1986, -3299)$,
$79, -1119, -2092, -3470)$,
$28, -829, -1537, -2546)$.

$1, 4, 1)$ as
$3B - 7C + 16D$.
s my original $g$.

d, e.g., $(-4, -1, 3, 1)$.
ing by root of unity
) preserves shortness.

For much larger $n$:

LLL almost never finds $g$.
Big gap between size of $g$
and size of "short" vectors
that LLL typically finds in $I$.

Increased BKZ block size:
reduced gap but slower.

Fancier lattice algorithms:
Under reasonable assumptions,
2015 Laarhoven–de Weger
finds $g$ in time $\approx 1.23^n$.
Big progress compared to, e.g.,
2008 Nguyen–Vidick ($\approx 1.33^n$)
but still exponential time.

Exploitin

Use LLL
generate
What ha

Pure lat
Work m

ctive

y find
lattice
$\mathbf{Z}D$ where
$-430, -712)$,
$-1986, -3299)$,
$-2092, -3470)$,
$-1537, -2546)$.

$+ 16D$.
nal $g$.

$4, -1, 3, 1)$.
ot of unity
s shortness.

For much larger $n$:

LLL almost never finds $g$.
Big gap between size of $g$
and size of "short" vectors
that LLL typically finds in $I$.

Increased BKZ block size:
reduced gap but slower.

Fancier lattice algorithms:
Under reasonable assumptions,
2015 Laarhoven–de Weger
finds $g$ in time $\approx 1.23^n$.
Big progress compared to, e.g.,
2008 Nguyen–Vidick ($\approx 1.33^n$)
but still exponential time.

Exploiting factoriz

Use LLL, BKZ, etc
generate rather sh
What happens if $d$

Pure lattice approa
Work much harder

e
2),
3299),
3470),
546).

).

s.

For much larger $n$:

LLL almost never finds $g$.
Big gap between size of $g$
and size of "short" vectors
that LLL typically finds in $I$.

Increased BKZ block size:
reduced gap but slower.

Fancier lattice algorithms:
Under reasonable assumptions,
2015 Laarhoven–de Weger
finds $g$ in time $\approx 1.23^n$.
Big progress compared to, e.g.,
2008 Nguyen–Vidick ($\approx 1.33^n$)
but still exponential time.

Exploiting factorization

Use LLL, BKZ, etc. to
generate rather short $\alpha \in g$
What happens if $\alpha\mathcal{O} \neq g\mathcal{O}$?

Pure lattice approach: Disca
Work much harder, find sho

For much larger $n$:

LLL almost never finds $g$.
Big gap between size of $g$
and size of "short" vectors
that LLL typically finds in $I$.

Increased BKZ block size:
reduced gap but slower.

Fancier lattice algorithms:
Under reasonable assumptions,
2015 Laarhoven–de Weger
finds $g$ in time $\approx 1.23^n$.
Big progress compared to, e.g.,
2008 Nguyen–Vidick ($\approx 1.33^n$)
but still exponential time.

## Exploiting factorization

Use LLL, BKZ, etc. to
generate rather short $\alpha \in g\mathcal{O}$.
What happens if $\alpha\mathcal{O} \neq g\mathcal{O}$?

Pure lattice approach: Discard $\alpha$.
Work much harder, find shorter $\alpha$.

For much larger $n$:

LLL almost never finds $g$.
Big gap between size of $g$
and size of "short" vectors
that LLL typically finds in $I$.

Increased BKZ block size:
reduced gap but slower.

Fancier lattice algorithms:
Under reasonable assumptions,
2015 Laarhoven–de Weger
finds $g$ in time $\approx 1.23^n$.
Big progress compared to, e.g.,
2008 Nguyen–Vidick ($\approx 1.33^n$)
but still exponential time.

## Exploiting factorization

Use LLL, BKZ, etc. to
generate rather short $\alpha \in g\mathcal{O}$.
What happens if $\alpha\mathcal{O} \neq g\mathcal{O}$?

Pure lattice approach: Discard $\alpha$.
Work much harder, find shorter $\alpha$.

Alternative: Gain information
from factorization of ideals.

For much larger $n$:

LLL almost never finds $g$.
Big gap between size of $g$
and size of "short" vectors
that LLL typically finds in $I$.

Increased BKZ block size:
reduced gap but slower.

Fancier lattice algorithms:
Under reasonable assumptions,
2015 Laarhoven–de Weger
finds $g$ in time $\approx 1.23^n$.
Big progress compared to, e.g.,
2008 Nguyen–Vidick ($\approx 1.33^n$)
but still exponential time.

## Exploiting factorization

Use LLL, BKZ, etc. to
generate rather short $\alpha \in g\mathcal{O}$.
What happens if $\alpha\mathcal{O} \neq g\mathcal{O}$?

Pure lattice approach: Discard $\alpha$.
Work much harder, find shorter $\alpha$.

Alternative: Gain information
from factorization of ideals.

e.g. If $\alpha_1\mathcal{O} = g\mathcal{O} \cdot P^2 \cdot Q^2$

For much larger $n$:

LLL almost never finds $g$.
Big gap between size of $g$
and size of "short" vectors
that LLL typically finds in $I$.

Increased BKZ block size:
reduced gap but slower.

Fancier lattice algorithms:
Under reasonable assumptions,
2015 Laarhoven–de Weger
finds $g$ in time $\approx 1.23^n$.
Big progress compared to, e.g.,
2008 Nguyen–Vidick ($\approx 1.33^n$)
but still exponential time.

## Exploiting factorization

Use LLL, BKZ, etc. to
generate rather short $\alpha \in g\mathcal{O}$.
What happens if $\alpha\mathcal{O} \neq g\mathcal{O}$?

Pure lattice approach: Discard $\alpha$.
Work much harder, find shorter $\alpha$.

Alternative: Gain information
from factorization of ideals.

e.g. If $\alpha_1\mathcal{O} = g\mathcal{O} \cdot P^2 \cdot Q^2$
and $\alpha_2\mathcal{O} = g\mathcal{O} \cdot P \cdot Q^3$

For much larger $n$:

LLL almost never finds $g$.
Big gap between size of $g$
and size of "short" vectors
that LLL typically finds in $I$.

Increased BKZ block size:
reduced gap but slower.

Fancier lattice algorithms:
Under reasonable assumptions,
2015 Laarhoven–de Weger
finds $g$ in time $\approx 1.23^n$.
Big progress compared to, e.g.,
2008 Nguyen–Vidick ($\approx 1.33^n$)
but still exponential time.

## Exploiting factorization

Use LLL, BKZ, etc. to
generate rather short $\alpha \in g\mathcal{O}$.
What happens if $\alpha\mathcal{O} \neq g\mathcal{O}$?

Pure lattice approach: Discard $\alpha$.
Work much harder, find shorter $\alpha$.

Alternative: Gain information
from factorization of ideals.

e.g. If $\alpha_1\mathcal{O} = g\mathcal{O} \cdot P^2 \cdot Q^2$
and $\alpha_2\mathcal{O} = g\mathcal{O} \cdot P \cdot Q^3$
and $\alpha_3\mathcal{O} = g\mathcal{O} \cdot P \cdot Q^2$

For much larger $n$:

LLL almost never finds $g$.
Big gap between size of $g$
and size of "short" vectors
that LLL typically finds in $I$.

Increased BKZ block size:
reduced gap but slower.

Fancier lattice algorithms:
Under reasonable assumptions,
2015 Laarhoven–de Weger
finds $g$ in time $\approx 1.23^n$.
Big progress compared to, e.g.,
2008 Nguyen–Vidick ($\approx 1.33^n$)
but still exponential time.

## Exploiting factorization

Use LLL, BKZ, etc. to
generate rather short $\alpha \in g\mathcal{O}$.
What happens if $\alpha\mathcal{O} \neq g\mathcal{O}$?

Pure lattice approach: Discard $\alpha$.
Work much harder, find shorter $\alpha$.

Alternative: Gain information
from factorization of ideals.

e.g. If $\alpha_1\mathcal{O} = g\mathcal{O} \cdot P^2 \cdot Q^2$
and $\alpha_2\mathcal{O} = g\mathcal{O} \cdot P \cdot Q^3$
and $\alpha_3\mathcal{O} = g\mathcal{O} \cdot P \cdot Q^2$ then
$P = \alpha_1\alpha_3^{-1}\mathcal{O}$ and $Q = \alpha_2\alpha_3^{-1}\mathcal{O}$
and $g\mathcal{O} = \alpha_1^{-1}\alpha_2^{-2}\alpha_3^4\mathcal{O}$.

h larger $n$:

ost never finds $g$.
between size of $g$
of "short" vectors
L typically finds in $I$.

d BKZ block size:
gap but slower.

lattice algorithms:
easonable assumptions,
arhoven–de Weger
n time $\approx 1.23^n$.
gress compared to, e.g.,
guyen–Vidick ($\approx 1.33^n$)
exponential time.

## Exploiting factorization

Use LLL, BKZ, etc. to
generate rather short $\alpha \in g\mathcal{O}$.
What happens if $\alpha\mathcal{O} \neq g\mathcal{O}$?

Pure lattice approach: Discard $\alpha$.
Work much harder, find shorter $\alpha$.

Alternative: Gain information
from factorization of ideals.

e.g. If $\alpha_1\mathcal{O} = g\mathcal{O} \cdot P^2 \cdot Q^2$
and $\alpha_2\mathcal{O} = g\mathcal{O} \cdot P \cdot Q^3$
and $\alpha_3\mathcal{O} = g\mathcal{O} \cdot P \cdot Q^2$ then
$P = \alpha_1\alpha_3^{-1}\mathcal{O}$ and $Q = \alpha_2\alpha_3^{-1}\mathcal{O}$
and $g\mathcal{O} = \alpha_1^{-1}\alpha_2^{-2}\alpha_3^4\mathcal{O}$.

General
factor $\alpha$
of some

Solve sy
to find g
as produ

:

finds $g$.
size of $g$
' vectors
finds in $I$.

ock size:
lower.

orithms:
assumptions,
e Weger
..$23^n$.
pared to, e.g.,
ck ($\approx 1.33^n$)
al time.

## Exploiting factorization

Use LLL, BKZ, etc. to
generate rather short $\alpha \in g\mathcal{O}$.
What happens if $\alpha\mathcal{O} \neq g\mathcal{O}$?

Pure lattice approach: Discard $\alpha$.
Work much harder, find shorter $\alpha$.

Alternative: Gain information
from factorization of ideals.

e.g. If $\alpha_1\mathcal{O} = g\mathcal{O} \cdot P^2 \cdot Q^2$
and $\alpha_2\mathcal{O} = g\mathcal{O} \cdot P \cdot Q^3$
and $\alpha_3\mathcal{O} = g\mathcal{O} \cdot P \cdot Q^2$ then
$P = \alpha_1\alpha_3^{-1}\mathcal{O}$ and $Q = \alpha_2\alpha_3^{-1}\mathcal{O}$
and $g\mathcal{O} = \alpha_1^{-1}\alpha_2^{-2}\alpha_3^4\mathcal{O}$.

General strategy:
factor $\alpha\mathcal{O}$ into pr
of some primes an

Solve system of e
to find generator f
as product of pow

ns,

.g.,
$3^n$)

Exploiting factorization

Use LLL, BKZ, etc. to
generate rather short $\alpha \in g\mathcal{O}$.
What happens if $\alpha\mathcal{O} \neq g\mathcal{O}$?

Pure lattice approach: Discard $\alpha$.
Work much harder, find shorter $\alpha$.

Alternative: Gain information
from factorization of ideals.

e.g. If $\alpha_1\mathcal{O} = g\mathcal{O} \cdot P^2 \cdot Q^2$
and $\alpha_2\mathcal{O} = g\mathcal{O} \cdot P \cdot Q^3$
and $\alpha_3\mathcal{O} = g\mathcal{O} \cdot P \cdot Q^2$ then
$P = \alpha_1\alpha_3^{-1}\mathcal{O}$ and $Q = \alpha_2\alpha_3^{-1}\mathcal{O}$
and $g\mathcal{O} = \alpha_1^{-1}\alpha_2^{-2}\alpha_3^4\mathcal{O}$.

General strategy: For many
factor $\alpha\mathcal{O}$ into products of
of some primes and $g\mathcal{O}$.

Solve system of equations
to find generator for $g\mathcal{O}$
as product of powers of the

## Exploiting factorization

Use LLL, BKZ, etc. to
generate rather short $\alpha \in g\mathcal{O}$.
What happens if $\alpha\mathcal{O} \neq g\mathcal{O}$?

Pure lattice approach: Discard $\alpha$.
Work much harder, find shorter $\alpha$.

Alternative: Gain information
from factorization of ideals.

e.g. If $\alpha_1\mathcal{O} = g\mathcal{O} \cdot P^2 \cdot Q^2$
and $\alpha_2\mathcal{O} = g\mathcal{O} \cdot P \cdot Q^3$
and $\alpha_3\mathcal{O} = g\mathcal{O} \cdot P \cdot Q^2$ then
$P = \alpha_1\alpha_3^{-1}\mathcal{O}$ and $Q = \alpha_2\alpha_3^{-1}\mathcal{O}$
and $g\mathcal{O} = \alpha_1^{-1}\alpha_2^{-2}\alpha_3^4\mathcal{O}$.

General strategy: For many $\alpha$'s,
factor $\alpha\mathcal{O}$ into products of powers
of some primes and $g\mathcal{O}$.

Solve system of equations
to find generator for $g\mathcal{O}$
as product of powers of the $\alpha$'s.

## Exploiting factorization

Use LLL, BKZ, etc. to
generate rather short $\alpha \in g\mathcal{O}$.
What happens if $\alpha\mathcal{O} \neq g\mathcal{O}$?

Pure lattice approach: Discard $\alpha$.
Work much harder, find shorter $\alpha$.

Alternative: Gain information
from factorization of ideals.

e.g. If $\alpha_1\mathcal{O} = g\mathcal{O} \cdot P^2 \cdot Q^2$
and $\alpha_2\mathcal{O} = g\mathcal{O} \cdot P \cdot Q^3$
and $\alpha_3\mathcal{O} = g\mathcal{O} \cdot P \cdot Q^2$ then
$P = \alpha_1\alpha_3^{-1}\mathcal{O}$ and $Q = \alpha_2\alpha_3^{-1}\mathcal{O}$
and $g\mathcal{O} = \alpha_1^{-1}\alpha_2^{-2}\alpha_3^4\mathcal{O}$.

General strategy: For many $\alpha$'s,
factor $\alpha\mathcal{O}$ into products of powers
of some primes and $g\mathcal{O}$.

Solve system of equations
to find generator for $g\mathcal{O}$
as product of powers of the $\alpha$'s.

"Can the system be solved?"

— Becomes increasingly
reasonable to expect as the
number of equations approaches
and passes the number of primes.

## Exploiting factorization

Use LLL, BKZ, etc. to
generate rather short $\alpha \in g\mathcal{O}$.
What happens if $\alpha\mathcal{O} \neq g\mathcal{O}$?

Pure lattice approach: Discard $\alpha$.
Work much harder, find shorter $\alpha$.

Alternative: Gain information
from factorization of ideals.

e.g. If $\alpha_1\mathcal{O} = g\mathcal{O} \cdot P^2 \cdot Q^2$
and $\alpha_2\mathcal{O} = g\mathcal{O} \cdot P \cdot Q^3$
and $\alpha_3\mathcal{O} = g\mathcal{O} \cdot P \cdot Q^2$ then
$P = \alpha_1\alpha_3^{-1}\mathcal{O}$ and $Q = \alpha_2\alpha_3^{-1}\mathcal{O}$
and $g\mathcal{O} = \alpha_1^{-1}\alpha_2^{-2}\alpha_3^4\mathcal{O}$.

General strategy: For many $\alpha$'s,
factor $\alpha\mathcal{O}$ into products of powers
of some primes and $g\mathcal{O}$.

Solve system of equations
to find generator for $g\mathcal{O}$
as product of powers of the $\alpha$'s.

"Can the system be solved?"

— Becomes increasingly
reasonable to expect as the
number of equations approaches
and passes the number of primes.

"But {primes} is infinite!"

ng factorization

, BKZ, etc. to
e rather short $\alpha \in g\mathcal{O}$.
appens if $\alpha\mathcal{O} \neq g\mathcal{O}$?

tice approach: Discard $\alpha$.
uch harder, find shorter $\alpha$.

ive: Gain information
torization of ideals.

$_1\mathcal{O} = g\mathcal{O} \cdot P^2 \cdot Q^2$
$\mathcal{O} = g\mathcal{O} \cdot P \cdot Q^3$
$\mathcal{O} = g\mathcal{O} \cdot P \cdot Q^2$ then
$\alpha_3^{-1}\mathcal{O}$ and $Q = \alpha_2\alpha_3^{-1}\mathcal{O}$
$= \alpha_1^{-1}\alpha_2^{-2}\alpha_3^4\mathcal{O}$.

General strategy: For many $\alpha$'s,
factor $\alpha\mathcal{O}$ into products of powers
of some primes and $g\mathcal{O}$.

Solve system of equations
to find generator for $g\mathcal{O}$
as product of powers of the $\alpha$'s.

"Can the system be solved?"

— Becomes increasingly
reasonable to expect as the
number of equations approaches
and passes the number of primes.

"But {primes} is infinite!"

— Restr
e.g., all

ation

c. to
ort $\alpha \in g\mathcal{O}$.
$\alpha\mathcal{O} \neq g\mathcal{O}$?

ach: Discard $\alpha$.
r, find shorter $\alpha$.

information
of ideals.

$\cdot P^2 \cdot Q^2$
$^2 \cdot Q^3$
$^2 \cdot Q^2$ then
$Q = \alpha_2\alpha_3^{-1}\mathcal{O}$
$^2\alpha_3^4\mathcal{O}$.

General strategy: For many $\alpha$'s, factor $\alpha\mathcal{O}$ into products of powers of some primes and $g\mathcal{O}$.

Solve system of equations to find generator for $g\mathcal{O}$ as product of powers of the $\alpha$'s.

"Can the system be solved?"

— Becomes increasingly reasonable to expect as the number of equations approaches and passes the number of primes.

"But {primes} is infinite!"

— Restrict to a "f
e.g., all primes of

$\mathcal{O}$.
?
ard $\alpha$.
rter $\alpha$.
on
n
$\frac{-1}{3}\mathcal{O}$

General strategy: For many $\alpha$'s, factor $\alpha\mathcal{O}$ into products of powers of some primes and $g\mathcal{O}$.

Solve system of equations
to find generator for $g\mathcal{O}$
as product of powers of the $\alpha$'s.

"Can the system be solved?"

— Becomes increasingly reasonable to expect as the number of equations approaches and passes the number of primes.

"But {primes} is infinite!"

— Restrict to a "factor base
e.g., all primes of norm $\leq y$.

General strategy: For many $\alpha$'s, factor $\alpha\mathcal{O}$ into products of powers of some primes and $g\mathcal{O}$.

Solve system of equations
to find generator for $g\mathcal{O}$
as product of powers of the $\alpha$'s.

"Can the system be solved?"

— Becomes increasingly reasonable to expect as the number of equations approaches and passes the number of primes.

"But {primes} is infinite!"

— Restrict to a "factor base": e.g., all primes of norm $\leq y$.

General strategy: For many $\alpha$'s, factor $\alpha\mathcal{O}$ into products of powers of some primes and $g\mathcal{O}$.

Solve system of equations to find generator for $g\mathcal{O}$ as product of powers of the $\alpha$'s.

"Can the system be solved?"

— Becomes increasingly reasonable to expect as the number of equations approaches and passes the number of primes.

"But {primes} is infinite!"

— Restrict to a "factor base": e.g., all primes of norm $\leq y$.

"But what if $\alpha\mathcal{O}$ doesn't factor into those primes?"

General strategy: For many $\alpha$'s, factor $\alpha\mathcal{O}$ into products of powers of some primes and $g\mathcal{O}$.

Solve system of equations
to find generator for $g\mathcal{O}$
as product of powers of the $\alpha$'s.

"Can the system be solved?"

— Becomes increasingly reasonable to expect as the number of equations approaches and passes the number of primes.

"But {primes} is infinite!"

— Restrict to a "factor base": e.g., all primes of norm $\leq y$.

"But what if $\alpha\mathcal{O}$ doesn't factor into those primes?"

— Then throw it away.
But often it *does* factor.

General strategy: For many $\alpha$'s,
factor $\alpha\mathcal{O}$ into products of powers
of some primes and $g\mathcal{O}$.

Solve system of equations
to find generator for $g\mathcal{O}$
as product of powers of the $\alpha$'s.

"Can the system be solved?"

— Becomes increasingly
reasonable to expect as the
number of equations approaches
and passes the number of primes.

"But {primes} is infinite!"

— Restrict to a "factor base":
e.g., all primes of norm $\leq y$.

"But what if $\alpha\mathcal{O}$ doesn't
factor into those primes?"

— Then throw it away.
But often it *does* factor.

Familiar issue from
"index calculus" DL methods,
CFRAC, LS, QS, NFS, etc.
Model the norm of $(\alpha/g)\mathcal{O}$
as "random" integer in $[1, x]$;
$y$-smoothness chance $\approx 1/y$
if $\log y \approx \sqrt{(1/2)\log x \log\log x}$.

strategy: For many $\alpha$'s,
$\mathcal{O}$ into products of powers
primes and $g\mathcal{O}$.

stem of equations
generator for $g\mathcal{O}$
uct of powers of the $\alpha$'s.

e system be solved?"

mes increasingly
ble to expect as the
of equations approaches
ses the number of primes.

rimes} is infinite!"

— Restrict to a "factor base":
e.g., all primes of norm $\leq y$.

"But what if $\alpha\mathcal{O}$ doesn't
factor into those primes?"

— Then throw it away.
But often it *does* factor.

Familiar issue from
"index calculus" DL methods,
CFRAC, LS, QS, NFS, etc.
Model the norm of $(\alpha/g)\mathcal{O}$
as "random" integer in $[1, x]$;
$y$-smoothness chance $\approx 1/y$
if $\log y \approx \sqrt{(1/2)\log x \log\log x}$.

Variation
Generate
factor $\alpha$
After en
solve sys
obtain g

For many $\alpha$'s,
oducts of powers
d $g\mathcal{O}$.

quations
for $g\mathcal{O}$
ers of the $\alpha$'s.

be solved?"

asingly
ect as the
ons approaches
mber of primes.

nfinite!"

— Restrict to a "factor base":
e.g., all primes of norm $\leq y$.

"But what if $\alpha\mathcal{O}$ doesn't
factor into those primes?"

— Then throw it away.
But often it *does* factor.

Familiar issue from
"index calculus" DL methods,
CFRAC, LS, QS, NFS, etc.
Model the norm of $(\alpha/g)\mathcal{O}$
as "random" integer in $[1, x]$;
$y$-smoothness chance $\approx 1/y$
if $\log y \approx \sqrt{(1/2)\log x \log\log x}$.

Variation: Ignore g
Generate rather sh
factor $\alpha\mathcal{O}$ into sm
After enough $\alpha$'s,
solve system of eq
obtain generator f

$\alpha$'s,
powers

$\alpha$'s.

"

ches
rimes.

— Restrict to a "factor base":
e.g., all primes of norm $\leq y$.

"But what if $\alpha\mathcal{O}$ doesn't
factor into those primes?"

— Then throw it away.
But often it *does* factor.

Familiar issue from
"index calculus" DL methods,
CFRAC, LS, QS, NFS, etc.
Model the norm of $(\alpha/g)\mathcal{O}$
as "random" integer in $[1, x]$;
$y$-smoothness chance $\approx 1/y$
if $\log y \approx \sqrt{(1/2)\log x \log\log x}$.

Variation: Ignore $g\mathcal{O}$.
Generate rather short $\alpha \in \mathcal{O}$
factor $\alpha\mathcal{O}$ into small primes
After enough $\alpha$'s,
solve system of equations;
obtain generator for each pr

— Restrict to a "factor base":
e.g., all primes of norm $\leq y$.

"But what if $\alpha\mathcal{O}$ doesn't
factor into those primes?"

— Then throw it away.
But often it *does* factor.

Familiar issue from
"index calculus" DL methods,
CFRAC, LS, QS, NFS, etc.
Model the norm of $(\alpha/g)\mathcal{O}$
as "random" integer in $[1, x]$;
$y$-smoothness chance $\approx 1/y$
if $\log y \approx \sqrt{(1/2)\log x \log\log x}$.

Variation: Ignore $g\mathcal{O}$.
Generate rather short $\alpha \in \mathcal{O}$,
factor $\alpha\mathcal{O}$ into small primes.
After enough $\alpha$'s,
solve system of equations;
obtain generator for each prime.

— Restrict to a "factor base":
e.g., all primes of norm $\leq y$.

"But what if $\alpha\mathcal{O}$ doesn't
factor into those primes?"

— Then throw it away.
But often it *does* factor.

Familiar issue from
"index calculus" DL methods,
CFRAC, LS, QS, NFS, etc.
Model the norm of $(\alpha/g)\mathcal{O}$
as "random" integer in $[1, x]$;
$y$-smoothness chance $\approx 1/y$
if $\log y \approx \sqrt{(1/2)\log x \log\log x}$.

Variation: Ignore $g\mathcal{O}$.
Generate rather short $\alpha \in \mathcal{O}$,
factor $\alpha\mathcal{O}$ into small primes.
After enough $\alpha$'s,
solve system of equations;
obtain generator for each prime.
After this precomputation,
factor *one* $\alpha\mathcal{O} \subseteq g\mathcal{O}$;
obtain generator for $g\mathcal{O}$.

— Restrict to a "factor base":
e.g., all primes of norm $\leq y$.

"But what if $\alpha\mathcal{O}$ doesn't
factor into those primes?"

— Then throw it away.
But often it *does* factor.

Familiar issue from
"index calculus" DL methods,
CFRAC, LS, QS, NFS, etc.
Model the norm of $(\alpha/g)\mathcal{O}$
as "random" integer in $[1, x]$;
$y$-smoothness chance $\approx 1/y$
if $\log y \approx \sqrt{(1/2)\log x \log\log x}$.

Variation: Ignore $g\mathcal{O}$.
Generate rather short $\alpha \in \mathcal{O}$,
factor $\alpha\mathcal{O}$ into small primes.
After enough $\alpha$'s,
solve system of equations;
obtain generator for each prime.
After this precomputation,
factor *one* $\alpha\mathcal{O} \subseteq g\mathcal{O}$;
obtain generator for $g\mathcal{O}$.

"Do all primes have generators?"

— Restrict to a "factor base":
e.g., all primes of norm $\leq y$.

"But what if $\alpha\mathcal{O}$ doesn't
factor into those primes?"

— Then throw it away.
But often it *does* factor.

Familiar issue from
"index calculus" DL methods,
CFRAC, LS, QS, NFS, etc.
Model the norm of $(\alpha/g)\mathcal{O}$
as "random" integer in $[1, x]$;
$y$-smoothness chance $\approx 1/y$
if $\log y \approx \sqrt{(1/2)\log x \log\log x}$.

Variation: Ignore $g\mathcal{O}$.
Generate rather short $\alpha \in \mathcal{O}$,
factor $\alpha\mathcal{O}$ into small primes.
After enough $\alpha$'s,
solve system of equations;
obtain generator for each prime.
After this precomputation,
factor *one* $\alpha\mathcal{O} \subseteq g\mathcal{O}$;
obtain generator for $g\mathcal{O}$.

"Do all primes have generators?"

— Standard heuristics:
For many (most?) number fields,
yes; but for big cyclotomics, no!
Modulo a few small primes, yes.

rict to a "factor base":
primes of norm $\leq y$.

hat if $\alpha\mathcal{O}$ doesn't
to those primes?"

throw it away.
n it *does* factor.

issue from
alculus" DL methods,
LS, QS, NFS, etc.
he norm of $(\alpha/g)\mathcal{O}$
lom" integer in $[1, x]$;
hness chance $\approx 1/y$
$\approx \sqrt{(1/2)\log x \log\log x}$.

Variation: Ignore $g\mathcal{O}$.
Generate rather short $\alpha \in \mathcal{O}$,
factor $\alpha\mathcal{O}$ into small primes.
After enough $\alpha$'s,
solve system of equations;
obtain generator for each prime.
After this precomputation,
factor *one* $\alpha\mathcal{O} \subseteq g\mathcal{O}$;
obtain generator for $g\mathcal{O}$.

"Do all primes have generators?"

— Standard heuristics:
For many (most?) number fields,
yes; but for big cyclotomics, no!
Modulo a few small primes, yes.

{princip
kernel o
{nonzer
$C$ is a fi
the "cla

Fundam
in algeb

factor base":
norm $\leq y$.

doesn't
primes?"

away.
factor.

n
DL methods,
NFS, etc.
f $(\alpha/g)\mathcal{O}$
ger in $[1, x]$;
nce $\approx 1/y$
$\log x \log\log x$.

Variation: Ignore $g\mathcal{O}$.
Generate rather short $\alpha \in \mathcal{O}$,
factor $\alpha\mathcal{O}$ into small primes.
After enough $\alpha$'s,
solve system of equations;
obtain generator for each prime.
After this precomputation,
factor *one* $\alpha\mathcal{O} \subseteq g\mathcal{O}$;
obtain generator for $g\mathcal{O}$.

"Do all primes have generators?"

— Standard heuristics:
For many (most?) number fields,
yes; but for big cyclotomics, no!
Modulo a few small primes, yes.

{principal nonzero
kernel of a semigr
{nonzero ideals} –
$C$ is a finite abelia
the "class group o

Fundamental obje
in algebraic numbe

e":

ls,

];

$\frac{\quad}{\log x}$.

Variation: Ignore $g\mathcal{O}$.
Generate rather short $\alpha \in \mathcal{O}$,
factor $\alpha\mathcal{O}$ into small primes.
After enough $\alpha$'s,
solve system of equations;
obtain generator for each prime.
After this precomputation,
factor *one* $\alpha\mathcal{O} \subseteq g\mathcal{O}$;
obtain generator for $g\mathcal{O}$.

"Do all primes have generators?"

— Standard heuristics:
For many (most?) number fields,
yes; but for big cyclotomics, no!
Modulo a few small primes, yes.

{principal nonzero ideals} is
kernel of a semigroup map
{nonzero ideals} $\twoheadrightarrow C$ where
$C$ is a finite abelian group,
the "class group of $K$".

Fundamental object of study
in algebraic number theory.

Variation: Ignore $g\mathcal{O}$.
Generate rather short $\alpha \in \mathcal{O}$,
factor $\alpha\mathcal{O}$ into small primes.
After enough $\alpha$'s,
solve system of equations;
obtain generator for each prime.
After this precomputation,
factor *one* $\alpha\mathcal{O} \subseteq g\mathcal{O}$;
obtain generator for $g\mathcal{O}$.

"Do all primes have generators?"

— Standard heuristics:
For many (most?) number fields,
yes; but for big cyclotomics, no!
Modulo a few small primes, yes.

{principal nonzero ideals} is
kernel of a semigroup map
{nonzero ideals} $\twoheadrightarrow C$ where
$C$ is a finite abelian group,
the "class group of $K$".

Fundamental object of study
in algebraic number theory.

Variation: Ignore $g\mathcal{O}$.
Generate rather short $\alpha \in \mathcal{O}$,
factor $\alpha\mathcal{O}$ into small primes.
After enough $\alpha$'s,
solve system of equations;
obtain generator for each prime.
After this precomputation,
factor *one* $\alpha\mathcal{O} \subseteq g\mathcal{O}$;
obtain generator for $g\mathcal{O}$.

"Do all primes have generators?"

— Standard heuristics:
For many (most?) number fields,
yes; but for big cyclotomics, no!
Modulo a few small primes, yes.

{principal nonzero ideals} is
kernel of a semigroup map
{nonzero ideals} $\twoheadrightarrow C$ where
$C$ is a finite abelian group,
the "class group of $K$".

Fundamental object of study
in algebraic number theory.

Factoring many small $\alpha\mathcal{O}$
is a standard textbook method
of computing class group
and generators of ideals.

Variation: Ignore $g\mathcal{O}$.
Generate rather short $\alpha \in \mathcal{O}$,
factor $\alpha\mathcal{O}$ into small primes.
After enough $\alpha$'s,
solve system of equations;
obtain generator for each prime.
After this precomputation,
factor *one* $\alpha\mathcal{O} \subseteq g\mathcal{O}$;
obtain generator for $g\mathcal{O}$.

"Do all primes have generators?"

— Standard heuristics:
For many (most?) number fields,
yes; but for big cyclotomics, no!
Modulo a few small primes, yes.

{principal nonzero ideals} is
kernel of a semigroup map
{nonzero ideals} $\twoheadrightarrow C$ where
$C$ is a finite abelian group,
the "class group of $K$".

Fundamental object of study
in algebraic number theory.

Factoring many small $\alpha\mathcal{O}$
is a standard textbook method
of computing class group
and generators of ideals.

Also compute unit group $\mathcal{O}^*$
via ratios of generators.

n: Ignore $g\mathcal{O}$.
e rather short $\alpha \in \mathcal{O}$,
$\mathcal{O}$ into small primes.
ough $\alpha$'s,
stem of equations;
enerator for each prime.
is precomputation,
*ne* $\alpha\mathcal{O} \subseteq g\mathcal{O}$;
enerator for $g\mathcal{O}$.

primes have generators?"

dard heuristics:
y (most?) number fields,
for big cyclotomics, no!
a few small primes, yes.

{principal nonzero ideals} is
kernel of a semigroup map
{nonzero ideals} $\twoheadrightarrow C$ where
$C$ is a finite abelian group,
the "class group of $K$".

Fundamental object of study
in algebraic number theory.

Factoring many small $\alpha\mathcal{O}$
is a standard textbook method
of computing class group
and generators of ideals.

Also compute unit group $\mathcal{O}^*$
via ratios of generators.

<u>Big gene</u>

Smart–V
this met
a genera
with larg
large, th
generato
$\theta$ may ta

Indeed, g
product
Must be
but extr

$g\mathcal{O}$.
hort $\alpha \in \mathcal{O}$,
all primes.

uations;
or each prime.
putation,
$g\mathcal{O}$;
or $g\mathcal{O}$.

ve generators?"

stics:
number fields,
clotomics, no!
all primes, yes.

{principal nonzero ideals} is kernel of a semigroup map {nonzero ideals} $\twoheadrightarrow C$ where $C$ is a finite abelian group, the "class group of $K$".

Fundamental object of study in algebraic number theory.

Factoring many small $\alpha\mathcal{O}$ is a standard textbook method of computing class group and generators of ideals.

Also compute unit group $\mathcal{O}^*$ via ratios of generators.

Big generator

Smart–Vercauteren this method is like a generator of larg with large coefficie large, that writing generator down as $\theta$ may take expone

Indeed, generator product of powers Must be $gu$ for so but extremely unli

$\mathcal{O}$,
.

ime.

ors?"

ields,
no!
yes.

{principal nonzero ideals} is
kernel of a semigroup map
{nonzero ideals} $\twoheadrightarrow C$ where
$C$ is a finite abelian group,
the "class group of $K$".

Fundamental object of study
in algebraic number theory.

Factoring many small $\alpha\mathcal{O}$
is a standard textbook method
of computing class group
and generators of ideals.

Also compute unit group $\mathcal{O}^*$
via ratios of generators.

Big generator

Smart–Vercauteren: "Howev
this method is likely to prod
a generator of large height,
with large coefficients. Indee
large, that writing the obtai
generator down as a polynor
$\theta$ may take exponential time

Indeed, generator found for
product of powers of various
Must be $gu$ for some $u \in \mathcal{O}$
but extremely unlikely to be

{principal nonzero ideals} is kernel of a semigroup map {nonzero ideals} $\twoheadrightarrow C$ where $C$ is a finite abelian group, the "class group of $K$".

Fundamental object of study in algebraic number theory.

Factoring many small $\alpha\mathcal{O}$ is a standard textbook method of computing class group and generators of ideals.

Also compute unit group $\mathcal{O}^*$ via ratios of generators.

Big generator

Smart–Vercauteren: "However this method is likely to produce a generator of large height, i.e., with large coefficients. Indeed so large, that writing the obtained generator down as a polynomial in $\theta$ may take exponential time."

Indeed, generator found for $g\mathcal{O}$ is product of powers of various $\alpha$'s. Must be $gu$ for some $u \in \mathcal{O}^*$, but extremely unlikely to be $g$.

{principal nonzero ideals} is kernel of a semigroup map {nonzero ideals} $\twoheadrightarrow C$ where $C$ is a finite abelian group, the "class group of $K$".

Fundamental object of study in algebraic number theory.

Factoring many small $\alpha\mathcal{O}$ is a standard textbook method of computing class group and generators of ideals.

Also compute unit group $\mathcal{O}^*$ via ratios of generators.

Big generator

Smart–Vercauteren: "However this method is likely to produce a generator of large height, i.e., with large coefficients. Indeed so large, that writing the obtained generator down as a polynomial in $\theta$ may take exponential time."

Indeed, generator found for $g\mathcal{O}$ is product of powers of various $\alpha$'s. Must be $gu$ for some $u \in \mathcal{O}^*$, but extremely unlikely to be $g$.

How do we find $g$ from $gu$?

al nonzero ideals} is
f a semigroup map
o ideals} $\twoheadrightarrow C$ where
nite abelian group,
ss group of $K$".

ental object of study
raic number theory.

g many small $\alpha\mathcal{O}$
dard textbook method
uting class group
erators of ideals.

mpute unit group $\mathcal{O}^*$
s of generators.

Big generator

Smart–Vercauteren: "However this method is likely to produce a generator of large height, i.e., with large coefficients. Indeed so large, that writing the obtained generator down as a polynomial in $\theta$ may take exponential time."

Indeed, generator found for $g\mathcal{O}$ is product of powers of various $\alpha$'s. Must be $gu$ for some $u \in \mathcal{O}^*$, but extremely unlikely to be $g$.

How do we find $g$ from $gu$?

There ar
ring map

ideals} is
oup map
↠ $C$ where
n group,
f $K$".

ct of study
er theory.

mall $\alpha\mathcal{O}$
book method
s group
ideals.

t group $\mathcal{O}^*$
ators.

Big generator

Smart–Vercauteren: "However this method is likely to produce a generator of large height, i.e., with large coefficients. Indeed so large, that writing the obtained generator down as a polynomial in $\theta$ may take exponential time."

Indeed, generator found for $g\mathcal{O}$ is product of powers of various $\alpha$'s. Must be $gu$ for some $u \in \mathcal{O}^*$, but extremely unlikely to be $g$.

How do we find $g$ from $gu$?

There are exactly
ring maps $\varphi_1, \ldots,$

e

y

od

*

## Big generator

Smart–Vercauteren: "However this method is likely to produce a generator of large height, i.e., with large coefficients. Indeed so large, that writing the obtained generator down as a polynomial in $\theta$ may take exponential time."

Indeed, generator found for $g\mathcal{O}$ is product of powers of various $\alpha$'s. Must be $gu$ for some $u \in \mathcal{O}^*$, but extremely unlikely to be $g$.

How do we find $g$ from $gu$?

There are exactly $n$ distinct ring maps $\varphi_1, \ldots, \varphi_n : K \to$

## Big generator

Smart–Vercauteren: "However this method is likely to produce a generator of large height, i.e., with large coefficients. Indeed so large, that writing the obtained generator down as a polynomial in $\theta$ may take exponential time."

Indeed, generator found for $g\mathcal{O}$ is product of powers of various $\alpha$'s. Must be $gu$ for some $u \in \mathcal{O}^*$, but extremely unlikely to be $g$.

How do we find $g$ from $gu$?

There are exactly $n$ distinct ring maps $\varphi_1, \ldots, \varphi_n : K \to \mathbf{C}$.

## Big generator

Smart–Vercauteren: "However this method is likely to produce a generator of large height, i.e., with large coefficients. Indeed so large, that writing the obtained generator down as a polynomial in $\theta$ may take exponential time."

Indeed, generator found for $g\mathcal{O}$ is product of powers of various $\alpha$'s. Must be $gu$ for some $u \in \mathcal{O}^*$, but extremely unlikely to be $g$.

How do we find $g$ from $gu$?

There are exactly $n$ distinct ring maps $\varphi_1, \ldots, \varphi_n : K \to \mathbf{C}$.

Define $\mathrm{Log} : K^* \to \mathbf{R}^n$ by $\mathrm{Log} = (\log|\varphi_1|, \ldots, \log|\varphi_n|)$.

## Big generator

Smart–Vercauteren: "However this method is likely to produce a generator of large height, i.e., with large coefficients. Indeed so large, that writing the obtained generator down as a polynomial in $\theta$ may take exponential time."

Indeed, generator found for $g\mathcal{O}$ is product of powers of various $\alpha$'s. Must be $gu$ for some $u \in \mathcal{O}^*$, but extremely unlikely to be $g$.

How do we find $g$ from $gu$?

There are exactly $n$ distinct ring maps $\varphi_1, \ldots, \varphi_n : K \to \mathbf{C}$.

Define $\mathrm{Log} : K^* \to \mathbf{R}^n$ by $\mathrm{Log} = (\log|\varphi_1|, \ldots, \log|\varphi_n|)$.

$\mathrm{Log}\,\mathcal{O}^*$ is a lattice of rank $r_1 + r_2 - 1$ where

$$r_1 = \#\{i : \varphi_i(K) \subseteq \mathbf{R}\},$$
$$2r_2 = \#\{i : \varphi_i(K) \not\subseteq \mathbf{R}\}.$$

## Big generator

Smart–Vercauteren: "However this method is likely to produce a generator of large height, i.e., with large coefficients. Indeed so large, that writing the obtained generator down as a polynomial in $\theta$ may take exponential time."

Indeed, generator found for $g\mathcal{O}$ is product of powers of various $\alpha$'s. Must be $gu$ for some $u \in \mathcal{O}^*$, but extremely unlikely to be $g$.

How do we find $g$ from $gu$?

There are exactly $n$ distinct ring maps $\varphi_1, \ldots, \varphi_n : K \to \mathbf{C}$.

Define $\mathrm{Log} : K^* \to \mathbf{R}^n$ by $\mathrm{Log} = (\log|\varphi_1|, \ldots, \log|\varphi_n|)$.

$\mathrm{Log}\,\mathcal{O}^*$ is a lattice of rank $r_1 + r_2 - 1$ where

$r_1 = \#\{i : \varphi_i(K) \subseteq \mathbf{R}\}$,

$2r_2 = \#\{i : \varphi_i(K) \not\subseteq \mathbf{R}\}$.

e.g. $\zeta = \exp(\pi i/256)$, $K = \mathbf{Q}(\zeta)$: images of $\zeta$ under ring maps are $\zeta, \zeta^3, \zeta^5, \ldots, \zeta^{511}$.

$r_1 = 0$; $r_2 = 128$; rank 127.

erator

Vercauteren: "However
hod is likely to produce
tor of large height, i.e.,
ge coefficients. Indeed so
at writing the obtained
or down as a polynomial in
ake exponential time."

generator found for $g\mathcal{O}$ is
of powers of various $\alpha$'s.
$gu$ for some $u \in \mathcal{O}^*$,
emely unlikely to be $g$.

we find $g$ from $gu$?

There are exactly $n$ distinct
ring maps $\varphi_1, \ldots, \varphi_n : K \to \mathbf{C}$.

Define $\mathrm{Log} : K^* \to \mathbf{R}^n$ by
$\mathrm{Log} = (\log|\varphi_1|, \ldots, \log|\varphi_n|)$.

$\mathrm{Log}\,\mathcal{O}^*$ is a lattice
of rank $r_1 + r_2 - 1$ where
$r_1 = \#\{i : \varphi_i(K) \subseteq \mathbf{R}\}$,
$2r_2 = \#\{i : \varphi_i(K) \not\subseteq \mathbf{R}\}$.

e.g. $\zeta = \exp(\pi i/256)$, $K = \mathbf{Q}(\zeta)$:
images of $\zeta$ under ring maps
are $\zeta, \zeta^3, \zeta^5, \ldots, \zeta^{511}$.
$r_1 = 0$; $r_2 = 128$; rank 127.

Compute
as sum o
for the c

n: "However
ely to produce
ge height, i.e.,
ents. Indeed so
the obtained
a polynomial in
ential time."

found for $g\mathcal{O}$ is
of various $\alpha$'s.
me $u \in \mathcal{O}^*$,
kely to be $g$.

from $gu$?

There are exactly $n$ distinct
ring maps $\varphi_1, \ldots, \varphi_n : K \to \mathbf{C}$.

Define $\mathrm{Log} : K^* \to \mathbf{R}^n$ by
$\mathrm{Log} = (\log|\varphi_1|, \ldots, \log|\varphi_n|)$.

$\mathrm{Log}\,\mathcal{O}^*$ is a lattice
of rank $r_1 + r_2 - 1$ where
$r_1 = \#\{i : \varphi_i(K) \subseteq \mathbf{R}\}$,
$2r_2 = \#\{i : \varphi_i(K) \not\subseteq \mathbf{R}\}$.

e.g. $\zeta = \exp(\pi i/256)$, $K = \mathbf{Q}(\zeta)$:
images of $\zeta$ under ring maps
are $\zeta, \zeta^3, \zeta^5, \ldots, \zeta^{511}$.
$r_1 = 0$; $r_2 = 128$; rank 127.

Compute $\mathrm{Log}\,gu$
as sum of multiple
for the original $\alpha$'s

ver
uce
i.e.,
ed so
ned
mial in
."
$g\mathcal{O}$ is
s $\alpha$'s.
$^*$,
$g$.

There are exactly $n$ distinct
ring maps $\varphi_1, \ldots, \varphi_n : K \to \mathbf{C}$.

Define $\mathrm{Log} : K^* \to \mathbf{R}^n$ by
$\mathrm{Log} = (\log|\varphi_1|, \ldots, \log|\varphi_n|)$.

$\mathrm{Log}\,\mathcal{O}^*$ is a lattice
of rank $r_1 + r_2 - 1$ where
$r_1 = \#\{i : \varphi_i(K) \subseteq \mathbf{R}\}$,
$2r_2 = \#\{i : \varphi_i(K) \not\subseteq \mathbf{R}\}$.

e.g. $\zeta = \exp(\pi i/256)$, $K = \mathbf{Q}(\zeta)$:
images of $\zeta$ under ring maps
are $\zeta, \zeta^3, \zeta^5, \ldots, \zeta^{511}$.
$r_1 = 0$; $r_2 = 128$; rank 127.

Compute $\mathrm{Log}\, gu$
as sum of multiples of $\mathrm{Log}\, o$
for the original $\alpha$'s.

There are exactly $n$ distinct
ring maps $\varphi_1, \ldots, \varphi_n : K \to \mathbf{C}$.

Define $\mathrm{Log} : K^* \to \mathbf{R}^n$ by
$\mathrm{Log} = (\log|\varphi_1|, \ldots, \log|\varphi_n|)$.

$\mathrm{Log}\,\mathcal{O}^*$ is a lattice
of rank $r_1 + r_2 - 1$ where
$r_1 = \#\{i : \varphi_i(K) \subseteq \mathbf{R}\}$,
$2r_2 = \#\{i : \varphi_i(K) \not\subseteq \mathbf{R}\}$.

e.g. $\zeta = \exp(\pi i/256)$, $K = \mathbf{Q}(\zeta)$:
images of $\zeta$ under ring maps
are $\zeta, \zeta^3, \zeta^5, \ldots, \zeta^{511}$.
$r_1 = 0$; $r_2 = 128$; rank 127.

Compute $\mathrm{Log}\, gu$
as sum of multiples of $\mathrm{Log}\,\alpha$
for the original $\alpha$'s.

There are exactly $n$ distinct
ring maps $\varphi_1, \ldots, \varphi_n : K \to \mathbf{C}$.

Define $\operatorname{Log} : K^* \to \mathbf{R}^n$ by
$\operatorname{Log} = (\log|\varphi_1|, \ldots, \log|\varphi_n|)$.

$\operatorname{Log} \mathcal{O}^*$ is a lattice
of rank $r_1 + r_2 - 1$ where
$r_1 = \#\{i : \varphi_i(K) \subseteq \mathbf{R}\}$,
$2r_2 = \#\{i : \varphi_i(K) \not\subseteq \mathbf{R}\}$.

e.g. $\zeta = \exp(\pi i/256)$, $K = \mathbf{Q}(\zeta)$:
images of $\zeta$ under ring maps
are $\zeta, \zeta^3, \zeta^5, \ldots, \zeta^{511}$.
$r_1 = 0$; $r_2 = 128$; rank 127.

Compute $\operatorname{Log} gu$
as sum of multiples of $\operatorname{Log} \alpha$
for the original $\alpha$'s.

Find elements of $\operatorname{Log} \mathcal{O}^*$
close to $\operatorname{Log} gu$.

There are exactly $n$ distinct
ring maps $\varphi_1, \ldots, \varphi_n : K \to \mathbf{C}$.

Define $\operatorname{Log} : K^* \to \mathbf{R}^n$ by
$\operatorname{Log} = (\log|\varphi_1|, \ldots, \log|\varphi_n|)$.

$\operatorname{Log} \mathcal{O}^*$ is a lattice
of rank $r_1 + r_2 - 1$ where
$r_1 = \#\{i : \varphi_i(K) \subseteq \mathbf{R}\}$,
$2r_2 = \#\{i : \varphi_i(K) \not\subseteq \mathbf{R}\}$.

e.g. $\zeta = \exp(\pi i/256)$, $K = \mathbf{Q}(\zeta)$:
images of $\zeta$ under ring maps
are $\zeta, \zeta^3, \zeta^5, \ldots, \zeta^{511}$.
$r_1 = 0$; $r_2 = 128$; rank 127.

Compute $\operatorname{Log} gu$
as sum of multiples of $\operatorname{Log} \alpha$
for the original $\alpha$'s.

Find elements of $\operatorname{Log} \mathcal{O}^*$
close to $\operatorname{Log} gu$.

This is a close-vector problem
("bounded-distance decoding").
"Embedding" heuristic:
CVP as fast as SVP.

There are exactly $n$ distinct
ring maps $\varphi_1, \ldots, \varphi_n : K \to \mathbf{C}$.

Define $\mathrm{Log} : K^* \to \mathbf{R}^n$ by
$\mathrm{Log} = (\log|\varphi_1|, \ldots, \log|\varphi_n|)$.

$\mathrm{Log}\,\mathcal{O}^*$ is a lattice
of rank $r_1 + r_2 - 1$ where
$r_1 = \#\{i : \varphi_i(K) \subseteq \mathbf{R}\}$,
$2r_2 = \#\{i : \varphi_i(K) \not\subseteq \mathbf{R}\}$.

e.g. $\zeta = \exp(\pi i/256)$, $K = \mathbf{Q}(\zeta)$:
images of $\zeta$ under ring maps
are $\zeta, \zeta^3, \zeta^5, \ldots, \zeta^{511}$.
$r_1 = 0$; $r_2 = 128$; rank 127.

Compute $\mathrm{Log}\, gu$
as sum of multiples of $\mathrm{Log}\,\alpha$
for the original $\alpha$'s.

Find elements of $\mathrm{Log}\,\mathcal{O}^*$
close to $\mathrm{Log}\, gu$.

This is a close-vector problem
("bounded-distance decoding").
"Embedding" heuristic:
CVP as fast as SVP.

This finds $\mathrm{Log}\, u$.
Easily reconstruct $g$
up to a root of unity.
$\#\{\text{roots of unity}\}$ is small.

re exactly $n$ distinct
ps $\varphi_1, \ldots, \varphi_n : K \to \mathbf{C}$.

Log : $K^* \to \mathbf{R}^n$ by
$\log|\varphi_1|, \ldots, \log|\varphi_n|)$.

is a lattice
$r_1 + r_2 - 1$ where
$\{i : \varphi_i(K) \subseteq \mathbf{R}\}$,
$\{i : \varphi_i(K) \not\subseteq \mathbf{R}\}$.

$\exp(\pi i/256)$, $K = \mathbf{Q}(\zeta)$:
of $\zeta$ under ring maps
$, \zeta^5, \ldots, \zeta^{511}$.
$r_2 = 128$; rank 127.

Compute $\operatorname{Log} gu$
as sum of multiples of $\operatorname{Log} \alpha$
for the original $\alpha$'s.

Find elements of $\operatorname{Log} \mathcal{O}^*$
close to $\operatorname{Log} gu$.

This is a close-vector problem
("bounded-distance decoding").
"Embedding" heuristic:
CVP as fast as SVP.

This finds $\operatorname{Log} u$.
Easily reconstruct $g$
up to a root of unity.
#{roots of unity} is small.

A subfie

(2014.02

Say we k
for a pro

$n$ distinct
$\varphi_n : K \to \mathbf{C}$.

$\to \mathbf{R}^n$ by
$\ldots, \log|\varphi_n|)$.

e

1 where
$) \subseteq \mathbf{R}\}$,
$) \not\subseteq \mathbf{R}\}$.

56), $K = \mathbf{Q}(\zeta)$:
ring maps
$^{511}$.

rank 127.

Compute $\operatorname{Log} gu$
as sum of multiples of $\operatorname{Log} \alpha$
for the original $\alpha$'s.

Find elements of $\operatorname{Log} \mathcal{O}^*$
close to $\operatorname{Log} gu$.

This is a close-vector problem
("bounded-distance decoding").
"Embedding" heuristic:
CVP as fast as SVP.

This finds $\operatorname{Log} u$.
Easily reconstruct $g$
up to a root of unity.
#{roots of unity} is small.

A subfield-logarith

(2014.02 Bernstei

Say we know Log
for a proper subfie

**C**.

).

$\mathbf{Q}(\zeta)$:

s

Compute $\operatorname{Log} gu$
as sum of multiples of $\operatorname{Log} \alpha$
for the original $\alpha$'s.

Find elements of $\operatorname{Log} \mathcal{O}^*$
close to $\operatorname{Log} gu$.

This is a close-vector problem
("bounded-distance decoding").
"Embedding" heuristic:
CVP as fast as SVP.

This finds $\operatorname{Log} u$.
Easily reconstruct $g$
up to a root of unity.
#{roots of unity} is small.

A subfield-logarithm attack

(2014.02 Bernstein)

Say we know $\operatorname{Log} \operatorname{norm}_{K:F} g$
for a proper subfield $F \subset K$.

Compute $\operatorname{Log} gu$
as sum of multiples of $\operatorname{Log} \alpha$
for the original $\alpha$'s.

Find elements of $\operatorname{Log} \mathcal{O}^*$
close to $\operatorname{Log} gu$.

This is a close-vector problem
("bounded-distance decoding").
"Embedding" heuristic:
CVP as fast as SVP.

This finds $\operatorname{Log} u$.
Easily reconstruct $g$
up to a root of unity.
#{roots of unity} is small.

## A subfield-logarithm attack

(2014.02 Bernstein)

Say we know $\operatorname{Log} \operatorname{norm}_{K:F} g$
for a proper subfield $F \subset K$.

Compute $\operatorname{Log} gu$
as sum of multiples of $\operatorname{Log} \alpha$
for the original $\alpha$'s.

Find elements of $\operatorname{Log} \mathcal{O}^*$
close to $\operatorname{Log} gu$.

This is a close-vector problem
("bounded-distance decoding").
"Embedding" heuristic:
CVP as fast as SVP.

This finds $\operatorname{Log} u$.
Easily reconstruct $g$
up to a root of unity.
#{roots of unity} is small.

## A subfield-logarithm attack

(2014.02 Bernstein)

Say we know $\operatorname{Log} \operatorname{norm}_{K:F} g$
for a proper subfield $F \subset K$.

We also know $\operatorname{Log} \operatorname{norm}_{K:F} gu$,
so we know $\operatorname{Log} \operatorname{norm}_{K:F} u$.

Compute $\operatorname{Log} gu$
as sum of multiples of $\operatorname{Log} \alpha$
for the original $\alpha$'s.

Find elements of $\operatorname{Log} \mathcal{O}^*$
close to $\operatorname{Log} gu$.

This is a close-vector problem
("bounded-distance decoding").
"Embedding" heuristic:
CVP as fast as SVP.

This finds $\operatorname{Log} u$.
Easily reconstruct $g$
up to a root of unity.
#{roots of unity} is small.

## A subfield-logarithm attack

(2014.02 Bernstein)

Say we know $\operatorname{Log} \operatorname{norm}_{K:F} g$
for a proper subfield $F \subset K$.

We also know $\operatorname{Log} \operatorname{norm}_{K:F} gu$,
so we know $\operatorname{Log} \operatorname{norm}_{K:F} u$.

This linearly constrains $\operatorname{Log} u$
to a shifted sublattice of $\operatorname{Log} \mathcal{O}^*$.
Number of independent
constraints: unit rank for $F$.

Compute $\operatorname{Log} gu$
as sum of multiples of $\operatorname{Log} \alpha$
for the original $\alpha$'s.

Find elements of $\operatorname{Log} \mathcal{O}^*$
close to $\operatorname{Log} gu$.

This is a close-vector problem
("bounded-distance decoding").
"Embedding" heuristic:
CVP as fast as SVP.

This finds $\operatorname{Log} u$.
Easily reconstruct $g$
up to a root of unity.
#{roots of unity} is small.

## A subfield-logarithm attack

(2014.02 Bernstein)

Say we know $\operatorname{Log} \operatorname{norm}_{K:F} g$
for a proper subfield $F \subset K$.

We also know $\operatorname{Log} \operatorname{norm}_{K:F} gu$,
so we know $\operatorname{Log} \operatorname{norm}_{K:F} u$.

This linearly constrains $\operatorname{Log} u$
to a shifted sublattice of $\operatorname{Log} \mathcal{O}^*$.
Number of independent
constraints: unit rank for $F$.

Find elements close to $\operatorname{Log} gu$.
Lower-dimension lattice problem,
if unit rank of $F$ is positive.

e Log $gu$
of multiples of Log $\alpha$
original $\alpha$'s.

ments of Log $\mathcal{O}^*$
Log $gu$.

a close-vector problem
ed-distance decoding").
ding" heuristic:
fast as SVP.

ds Log $u$.
econstruct $g$
root of unity.
of unity} is small.

## A subfield-logarithm attack

(2014.02 Bernstein)

Say we know Log norm$_{K:F}$ $g$
for a proper subfield $F \subset K$.

We also know Log norm$_{K:F}$ $gu$,
so we know Log norm$_{K:F}$ $u$.

This linearly constrains Log $u$
to a shifted sublattice of Log $\mathcal{O}^*$.
Number of independent
constraints: unit rank for $F$.

Find elements close to Log $gu$.
Lower-dimension lattice problem,
if unit rank of $F$ is positive.

Start by
Log norm
for *each*

Various
dependir

es of $\operatorname{Log}\alpha$
s.

$\operatorname{Log}\mathcal{O}^*$

tor problem
ce decoding").
ristic:
VP.

$g$
ity.
is small.

## A subfield-logarithm attack

(2014.02 Bernstein)

Say we know $\operatorname{Log}\operatorname{norm}_{K:F} g$
for a proper subfield $F \subset K$.

We also know $\operatorname{Log}\operatorname{norm}_{K:F} gu$,
so we know $\operatorname{Log}\operatorname{norm}_{K:F} u$.

This linearly constrains $\operatorname{Log} u$
to a shifted sublattice of $\operatorname{Log}\mathcal{O}^*$.
Number of independent
constraints: unit rank for $F$.

Find elements close to $\operatorname{Log} gu$.
Lower-dimension lattice problem,
if unit rank of $F$ is positive.

Start by recursivel
$\operatorname{Log}\operatorname{norm}_{K:F} g$ via
for *each* $F \subset K$.

Various constraint
depending on subf

α

m

g" ).

## A subfield-logarithm attack

(2014.02 Bernstein)

Say we know $\operatorname{Log} \operatorname{norm}_{K:F} g$
for a proper subfield $F \subset K$.

We also know $\operatorname{Log} \operatorname{norm}_{K:F} gu$,
so we know $\operatorname{Log} \operatorname{norm}_{K:F} u$.

This linearly constrains $\operatorname{Log} u$
to a shifted sublattice of $\operatorname{Log} \mathcal{O}^*$.
Number of independent
constraints: unit rank for $F$.

Find elements close to $\operatorname{Log} gu$.
Lower-dimension lattice problem,
if unit rank of $F$ is positive.

Start by recursively computi
$\operatorname{Log} \operatorname{norm}_{K:F} g$ via norm of $g$
for *each* $F \subset K$.

Various constraints on $\operatorname{Log} u$
depending on subfield struct

## A subfield-logarithm attack

(2014.02 Bernstein)

Say we know $\operatorname{Log} \operatorname{norm}_{K:F} g$ for a proper subfield $F \subset K$.

We also know $\operatorname{Log} \operatorname{norm}_{K:F} gu$, so we know $\operatorname{Log} \operatorname{norm}_{K:F} u$.

This linearly constrains $\operatorname{Log} u$ to a shifted sublattice of $\operatorname{Log} \mathcal{O}^*$. Number of independent constraints: unit rank for $F$.

Find elements close to $\operatorname{Log} gu$. Lower-dimension lattice problem, if unit rank of $F$ is positive.

Start by recursively computing $\operatorname{Log} \operatorname{norm}_{K:F} g$ via norm of $g\mathcal{O}$ for *each* $F \subset K$.

Various constraints on $\operatorname{Log} u$, depending on subfield structure.

## A subfield-logarithm attack

(2014.02 Bernstein)

Say we know $\operatorname{Log} \operatorname{norm}_{K:F} g$ for a proper subfield $F \subset K$.

We also know $\operatorname{Log} \operatorname{norm}_{K:F} gu$, so we know $\operatorname{Log} \operatorname{norm}_{K:F} u$.

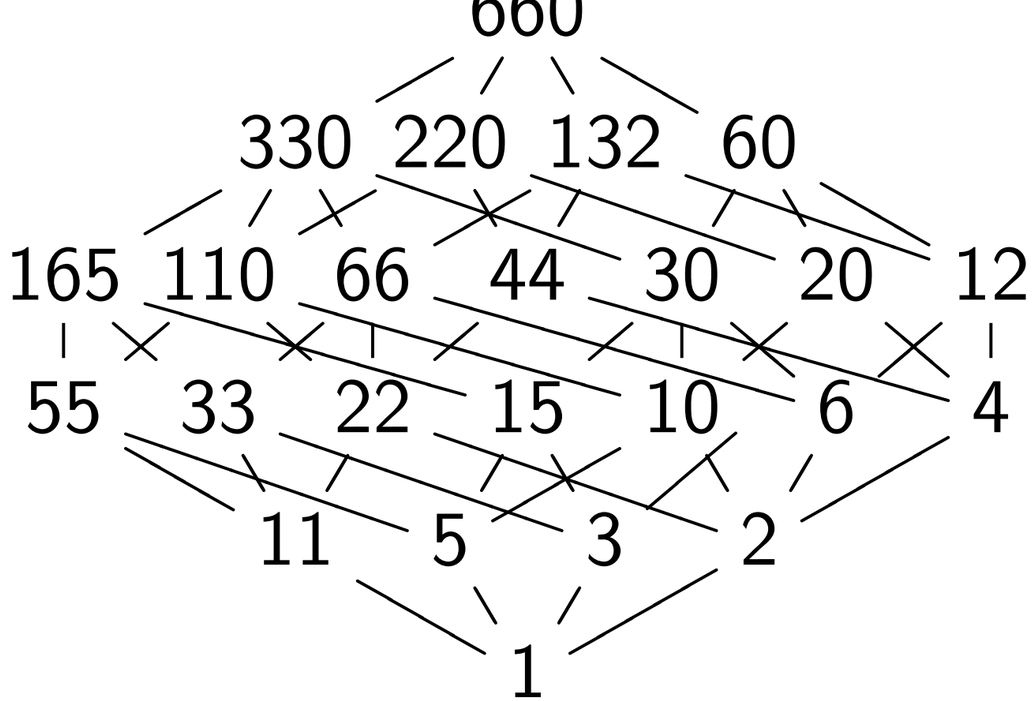
This linearly constrains $\operatorname{Log} u$ to a shifted sublattice of $\operatorname{Log} \mathcal{O}^*$. Number of independent constraints: unit rank for $F$.

Find elements close to $\operatorname{Log} gu$. Lower-dimension lattice problem, if unit rank of $F$ is positive.

Start by recursively computing $\operatorname{Log} \operatorname{norm}_{K:F} g$ via norm of $g\mathcal{O}$ for *each* $F \subset K$.

Various constraints on $\operatorname{Log} u$, depending on subfield structure.

e.g. $\zeta = \exp(2\pi i/661)$, $K = \mathbf{Q}(\zeta)$. Degrees of subfields of $K$:



## ld-logarithm attack

2 Bernstein)

know $\operatorname{Log}\operatorname{norm}_{K:F} g$
oper subfield $F \subset K$.

know $\operatorname{Log}\operatorname{norm}_{K:F} gu$,
now $\operatorname{Log}\operatorname{norm}_{K:F} u$.

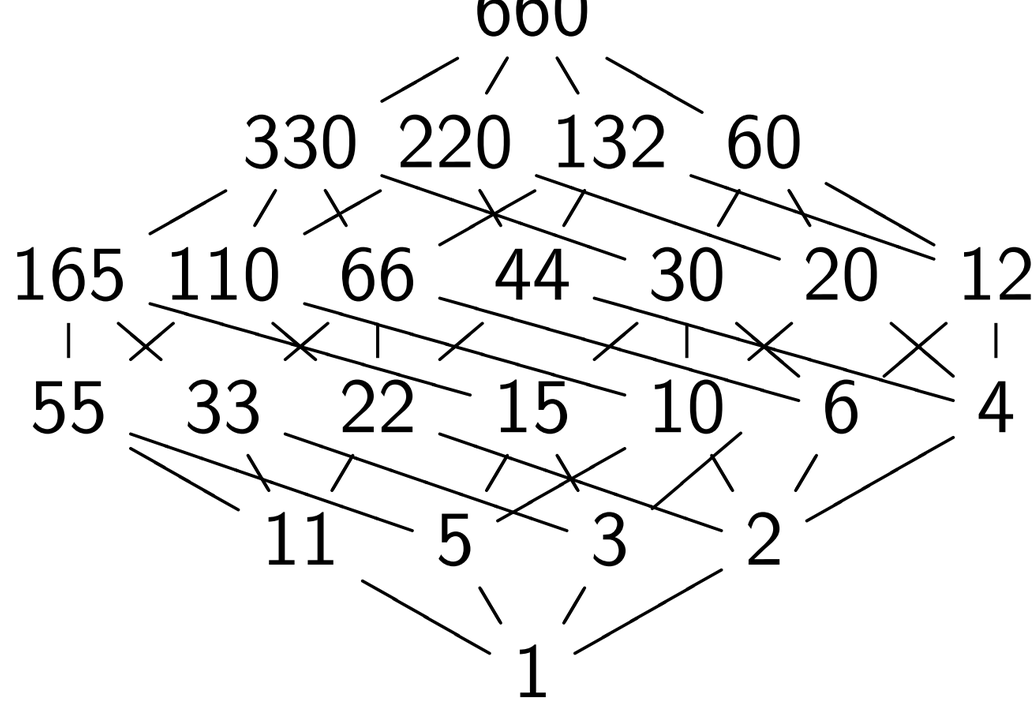
early constrains $\operatorname{Log} u$
fted sublattice of $\operatorname{Log} \mathcal{O}^*$.
of independent
nts: unit rank for $F$.

ments close to $\operatorname{Log} gu$.
imension lattice problem,
ank of $F$ is positive.

Start by recursively computing
$\operatorname{Log}\operatorname{norm}_{K:F} g$ via norm of $g\mathcal{O}$
for *each* $F \subset K$.

Various constraints on $\operatorname{Log} u$,
depending on subfield structure.

e.g. $\zeta = \exp(2\pi i/661)$, $K = \mathbf{Q}(\zeta)$.
Degrees of subfields of $K$:




Most ex
Compos
$K = \mathbf{Q}(\sqrt$
CVP be

m attack

n)

norm$_{K:F}\, g$
eld $F \subset K$.

$g$ norm$_{K:F}\, gu$,
orm$_{K:F}\, u$.

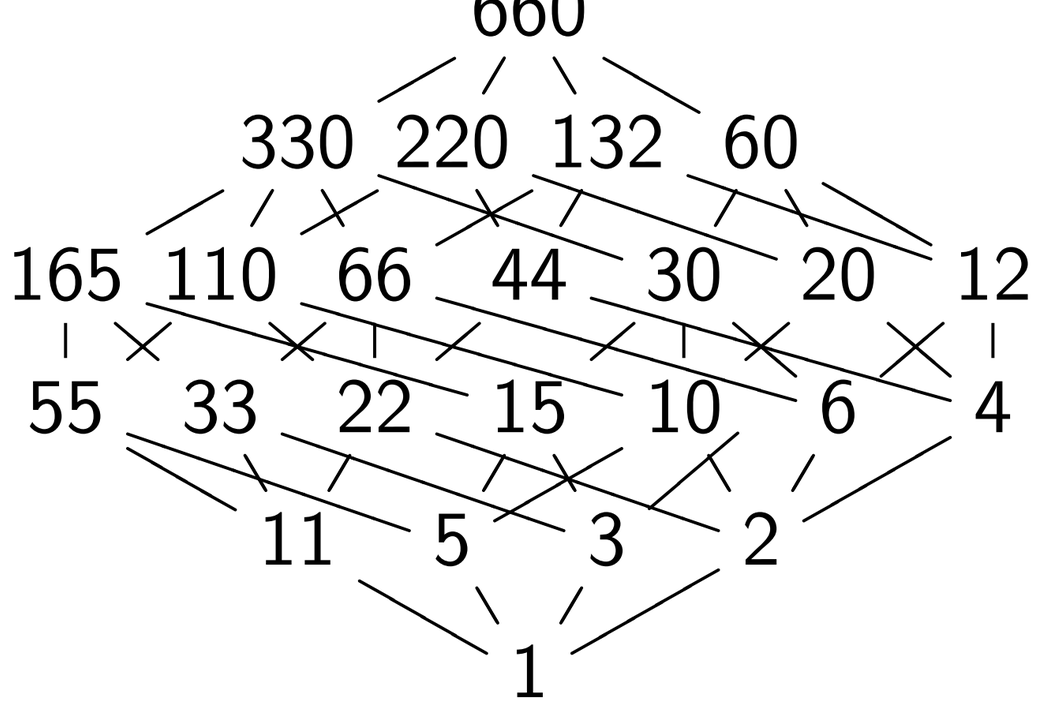
rains Log $u$
tice of Log $\mathcal{O}^*$.
ndent
ank for $F$.

se to Log $gu$.
attice problem,
s positive.

Start by recursively computing Log norm$_{K:F}\, g$ via norm of $g\mathcal{O}$ for *each* $F \subset K$.

Various constraints on Log $u$, depending on subfield structure.

e.g. $\zeta = \exp(2\pi i/661)$, $K = \mathbf{Q}(\zeta)$.
Degrees of subfields of $K$:




Most extreme case
Composite of quad
$K = \mathbf{Q}(\sqrt{2}, \sqrt{3}, \sqrt{}$
CVP becomes triv

$g\,u$,

$u$

$g\,\mathcal{O}^*$.

.

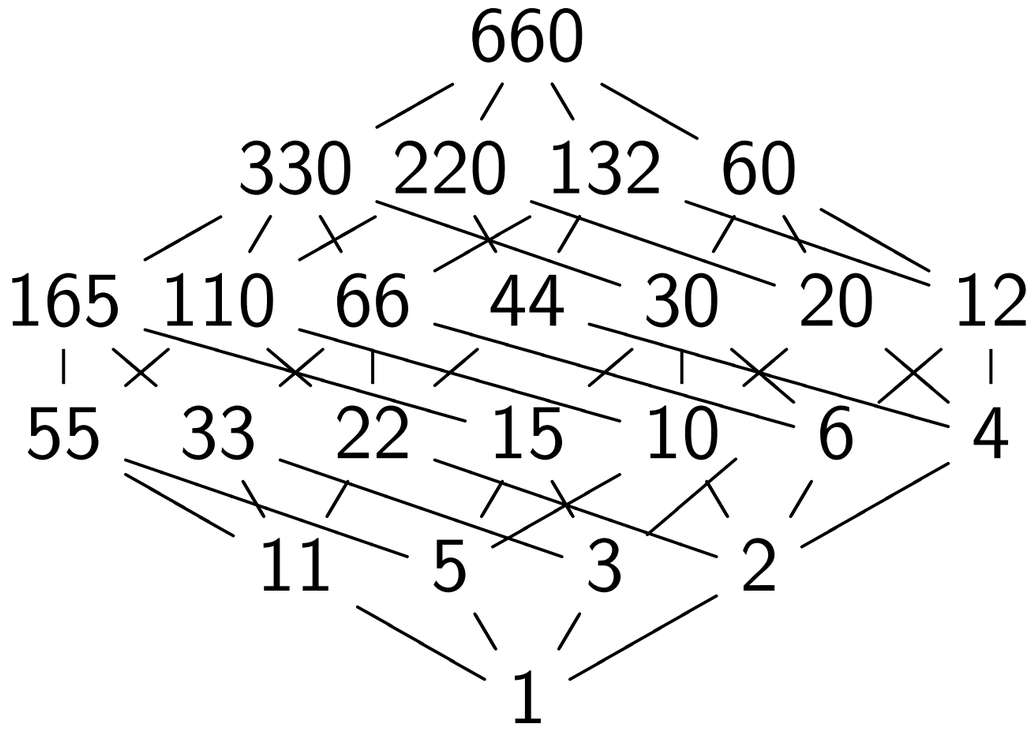
$g\,u$.

blem,

Start by recursively computing
$\operatorname{Log}\operatorname{norm}_{K:F} g$ via norm of $g\mathcal{O}$
for *each* $F \subset K$.

Various constraints on $\operatorname{Log} u$,
depending on subfield structure.

e.g. $\zeta = \exp(2\pi i/661)$, $K = \mathbf{Q}(\zeta)$.
Degrees of subfields of $K$:

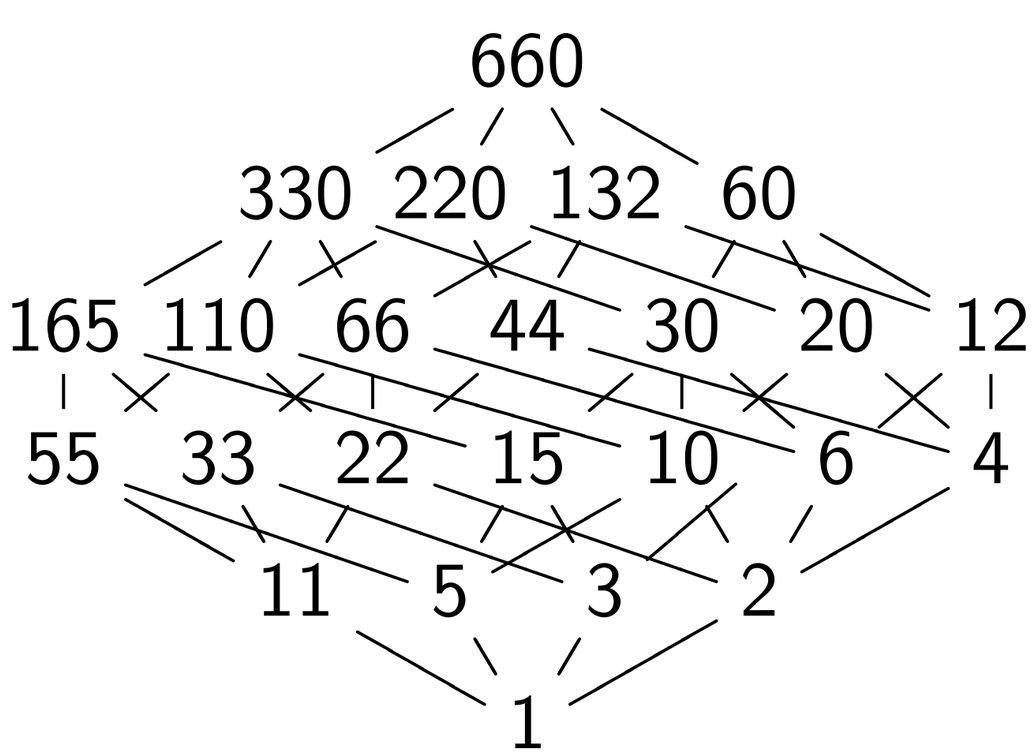



Most extreme case:
Composite of quadratics, su
$K = \mathbf{Q}(\sqrt{2}, \sqrt{3}, \sqrt{5}, \ldots, \sqrt{29}$
CVP becomes trivial!

Start by recursively computing $\operatorname{Log}\operatorname{norm}_{K:F} g$ via norm of $g\mathcal{O}$ for *each* $F \subset K$.

Various constraints on $\operatorname{Log} u$, depending on subfield structure.

e.g. $\zeta = \exp(2\pi i/661)$, $K = \mathbf{Q}(\zeta)$.
Degrees of subfields of $K$:




Most extreme case:

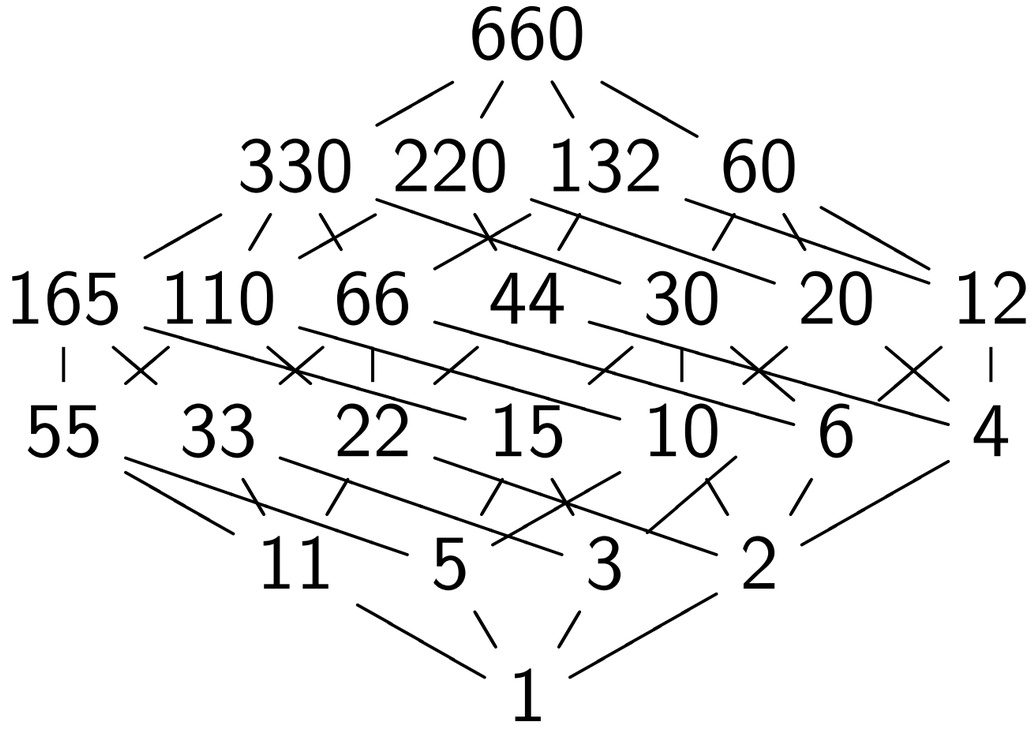
Composite of quadratics, such as $K = \mathbf{Q}(\sqrt{2}, \sqrt{3}, \sqrt{5}, \ldots, \sqrt{29})$.

CVP becomes trivial!

Start by recursively computing $\operatorname{Log} \operatorname{norm}_{K:F} g$ via norm of $g\mathcal{O}$ for *each* $F \subset K$.

Various constraints on $\operatorname{Log} u$, depending on subfield structure.

e.g. $\zeta = \exp(2\pi i/661)$, $K = \mathbf{Q}(\zeta)$.
Degrees of subfields of $K$:




Most extreme case:
Composite of quadratics, such as $K = \mathbf{Q}(\sqrt{2}, \sqrt{3}, \sqrt{5}, \ldots, \sqrt{29})$.
CVP becomes trivial!

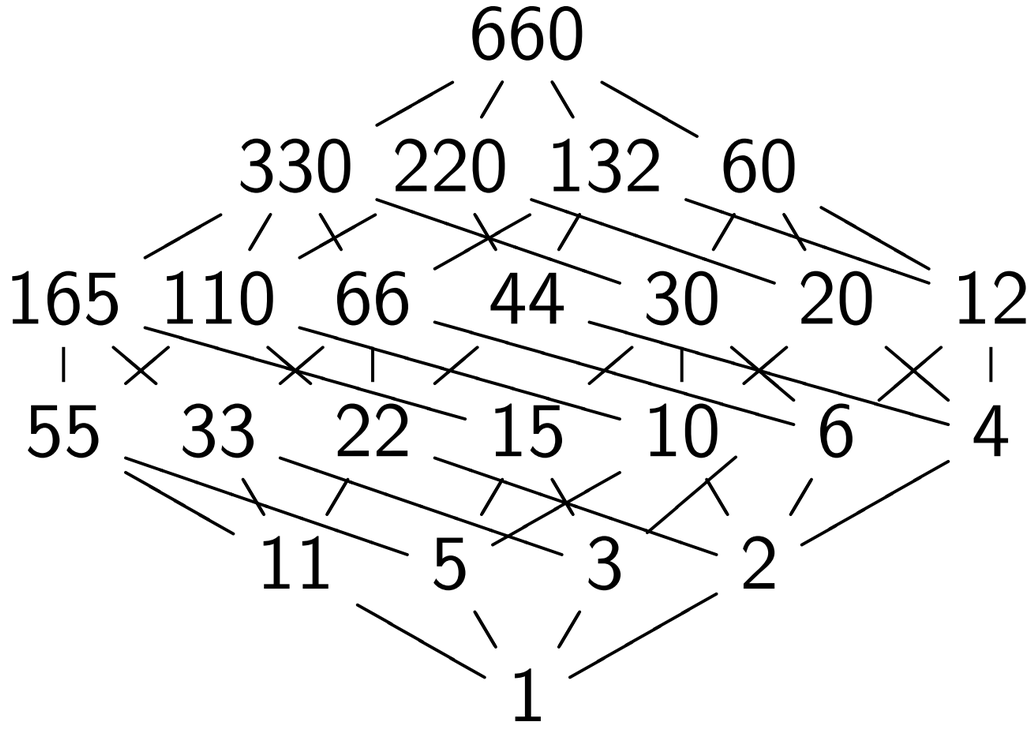
Opposite extreme: prime degree; the only proper subfield is $\mathbf{Q}$.
My recommendation: big Galois group; e.g., $\mathbf{Q}[x]/(x^p - x - 1)$.

Many intermediate cases.

Start by recursively computing $\operatorname{Log} \operatorname{norm}_{K:F} g$ via norm of $g\mathcal{O}$ for *each* $F \subset K$.

Various constraints on $\operatorname{Log} u$, depending on subfield structure.

e.g. $\zeta = \exp(2\pi i/661)$, $K = \mathbf{Q}(\zeta)$.
Degrees of subfields of $K$:




Most extreme case:
Composite of quadratics, such as $K = \mathbf{Q}(\sqrt{2}, \sqrt{3}, \sqrt{5}, \ldots, \sqrt{29})$.
CVP becomes trivial!

Opposite extreme: prime degree; the only proper subfield is $\mathbf{Q}$.
My recommendation: big Galois group; e.g., $\mathbf{Q}[x]/(x^p - x - 1)$.

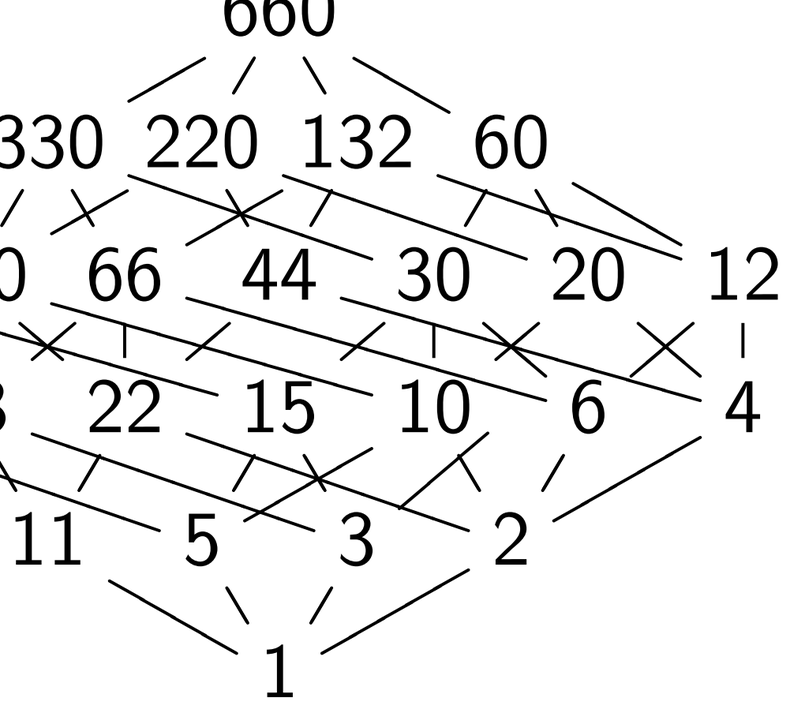
Many intermediate cases.

Confused summary by Cramer–Ducas–Peikert–Regev: method "may yield slightly subexponential runtimes in *cyclotomic* rings of *highly smooth* index".

recursively computing
$\mathrm{m}_{K:F}\, g$ via norm of $g\mathcal{O}$
$F \subset K$.

constraints on $\mathrm{Log}\, u$,
ng on subfield structure.

$\exp(2\pi i/661)$, $K = \mathbf{Q}(\zeta)$.
of subfields of $K$:




Most extreme case:

Composite of quadratics, such as
$K = \mathbf{Q}(\sqrt{2}, \sqrt{3}, \sqrt{5}, \ldots, \sqrt{29})$.
CVP becomes trivial!

Opposite extreme: prime degree;
the only proper subfield is $\mathbf{Q}$.
My recommendation: big Galois
group; e.g., $\mathbf{Q}[x]/(x^p - x - 1)$.

Many intermediate cases.

Confused summary by Cramer–
Ducas–Peikert–Regev: method
"may yield slightly subexponential
runtimes in *cyclotomic* rings of
*highly smooth* index".

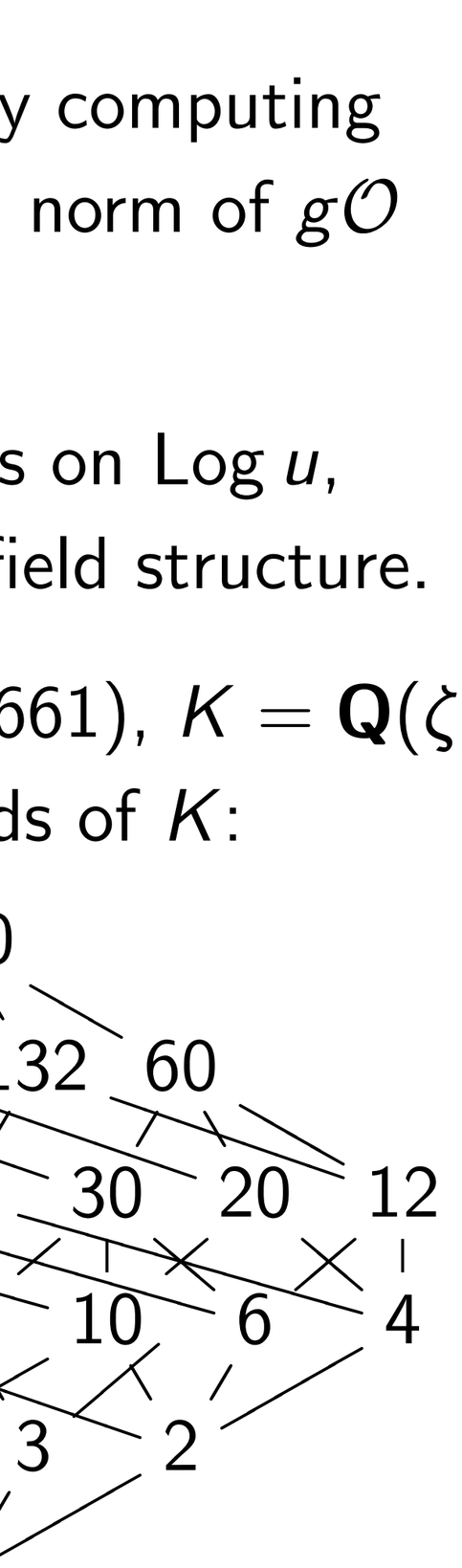
Further

(1) 2014
Shepher
for **cyclo**
(good) b

y computing
norm of $g\mathcal{O}$

s on Log $u$,
ield structure.

661), $K = \mathbf{Q}(\zeta)$.
ds of $K$:




Most extreme case:
Composite of quadratics, such as $K = \mathbf{Q}(\sqrt{2}, \sqrt{3}, \sqrt{5}, \ldots, \sqrt{29})$.
CVP becomes trivial!

Opposite extreme: prime degree; the only proper subfield is $\mathbf{Q}$.
My recommendation: big Galois group; e.g., $\mathbf{Q}[x]/(x^p - x - 1)$.

Many intermediate cases.

Confused summary by Cramer–Ducas–Peikert–Regev: method "may yield slightly subexponential runtimes in *cyclotomic* rings of *highly smooth* index".

Further improveme

(1) 2014.10 Camp
Shepherd: Quickly
for **cyclotomics** u
(good) basis for cy

ng
$g\mathcal{O}$

$\iota$,

ture.

$= \mathbf{Q}(\zeta)$.

Most extreme case:

Composite of quadratics, such as $K = \mathbf{Q}(\sqrt{2}, \sqrt{3}, \sqrt{5}, \ldots, \sqrt{29})$.

CVP becomes trivial!

Opposite extreme: prime degree; the only proper subfield is $\mathbf{Q}$.

My recommendation: big Galois group; e.g., $\mathbf{Q}[x]/(x^p - x - 1)$.

Many intermediate cases.

Confused summary by Cramer–Ducas–Peikert–Regev: method "may yield slightly subexponential runtimes in *cyclotomic* rings of *highly smooth* index".

Further improvements: (1),

(1) 2014.10 Campbell–Grov
Shepherd: Quickly solve CV
for **cyclotomics** using know
(good) basis for cyclotomic

Most extreme case:

Composite of quadratics, such as $K = \mathbf{Q}(\sqrt{2}, \sqrt{3}, \sqrt{5}, \ldots, \sqrt{29})$.
CVP becomes trivial!

Opposite extreme: prime degree; the only proper subfield is $\mathbf{Q}$.
My recommendation: big Galois group; e.g., $\mathbf{Q}[x]/(x^p - x - 1)$.

Many intermediate cases.

Confused summary by Cramer–Ducas–Peikert–Regev: method "may yield slightly subexponential runtimes in *cyclotomic* rings of *highly smooth* index".

Further improvements: (1), (2)

(1) 2014.10 Campbell–Groves–Shepherd: Quickly solve CVP for **cyclotomics** using known (good) basis for cyclotomic units.

Most extreme case:

Composite of quadratics, such as $K = \mathbf{Q}(\sqrt{2}, \sqrt{3}, \sqrt{5}, \ldots, \sqrt{29})$.
CVP becomes trivial!

Opposite extreme: prime degree; the only proper subfield is $\mathbf{Q}$.
My recommendation: big Galois group; e.g., $\mathbf{Q}[x]/(x^p - x - 1)$.

Many intermediate cases.

Confused summary by Cramer–Ducas–Peikert–Regev: method "may yield slightly subexponential runtimes in *cyclotomic* rings of *highly smooth* index".

Further improvements: (1), (2)

(1) 2014.10 Campbell–Groves–Shepherd: Quickly solve CVP for **cyclotomics** using known (good) basis for cyclotomic units.

Analysis in paper is bogus, but algorithm is very fast.

Most extreme case:

Composite of quadratics, such as $K = \mathbf{Q}(\sqrt{2}, \sqrt{3}, \sqrt{5}, \ldots, \sqrt{29})$.
CVP becomes trivial!

Opposite extreme: prime degree; the only proper subfield is $\mathbf{Q}$.
My recommendation: big Galois group; e.g., $\mathbf{Q}[x]/(x^p - x - 1)$.

Many intermediate cases.

Confused summary by Cramer–Ducas–Peikert–Regev: method "may yield slightly subexponential runtimes in *cyclotomic* rings of *highly smooth* index".

Further improvements: (1), (2)

(1) 2014.10 Campbell–Groves–Shepherd: Quickly solve CVP for **cyclotomics** using known (good) basis for cyclotomic units.

Analysis in paper is bogus, but algorithm is very fast.

Plagiarized and properly analyzed by Cramer–Ducas–Peikert–Regev.

Most extreme case:

Composite of quadratics, such as $K = \mathbf{Q}(\sqrt{2}, \sqrt{3}, \sqrt{5}, \ldots, \sqrt{29})$.
CVP becomes trivial!

Opposite extreme: prime degree;
the only proper subfield is $\mathbf{Q}$.
My recommendation: big Galois group; e.g., $\mathbf{Q}[x]/(x^p - x - 1)$.

Many intermediate cases.

Confused summary by Cramer–Ducas–Peikert–Regev: method "may yield slightly subexponential runtimes in *cyclotomic* rings of *highly smooth* index".

Further improvements: (1), (2)

(1) 2014.10 Campbell–Groves–Shepherd: Quickly solve CVP for **cyclotomics** using known (good) basis for cyclotomic units.

Analysis in paper is bogus,
but algorithm is very fast.

Plagiarized and properly analyzed by Cramer–Ducas–Peikert–Regev.

(2) 2015.01 Song announcement: Fast **quantum** algorithm for $gu$.
"PIP ... solved [BiasseSong'14]".
But paper not available yet.