Measuring, simulating and exploiting the head concavity phenomenon in BKZ

Shi Bai[1] Damien Stehlé[2] Weiqiang Wen[3]

[1]Florida Atlantic University. USA.
[2]École Normale Supérieure de Lyon. France.
[3]IRISA, Université Rennes 1. France.

ASIACRYPT 2018, BRISBANE, AUSTRALIA.

European Research Council PROMETHEUS UMR IRISA ENS DE LYON

Outline

The Blockwise-Korkine-Zolotarev (BKZ) lattice reduction algorithm is central in cryptanalysis for lattice-based cryptography.

Outline

The Blockwise-Korkine-Zolotarev (BKZ) lattice reduction algorithm is central in cryptanalysis for lattice-based cryptography.

1. Explain and quantify the shorter-than-expected phenomenon in the head region in BKZ.

Outline

The Blockwise-Korkine-Zolotarev (BKZ) lattice reduction algorithm is central in cryptanalysis for lattice-based cryptography.

1. Explain and quantify the shorter-than-expected phenomenon in the head region in BKZ.

2. A more accurate simulator for BKZ.

Outline

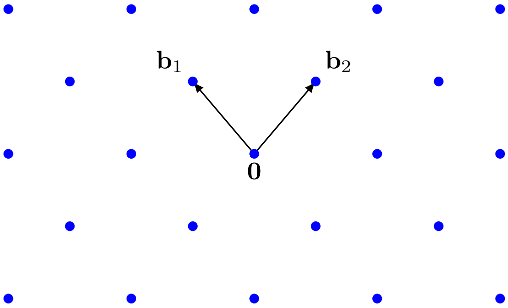
The Blockwise-Korkine-Zolotarev (BKZ) lattice reduction algorithm is central in cryptanalysis for lattice-based cryptography.

1. Explain and quantify the shorter-than-expected phenomenon in the head region in BKZ.

2. A more accurate simulator for BKZ.

3. A new BKZ variant that exploits the shorter-than-expected phenomenon.

Lattice



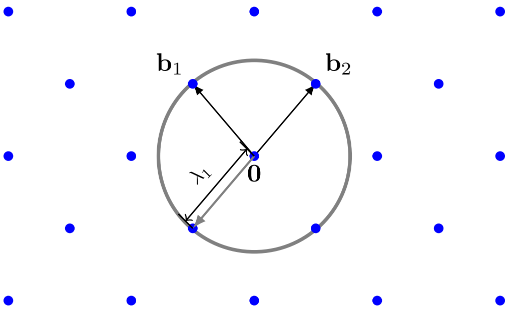
Definition

Given a set of linearly independent vectors $\{\mathbf{b}_1, \cdots, \mathbf{b}_n\} \subseteq \mathbb{Q}^m$, the lattice $\mathcal{L}$ spanned by the $\mathbf{b}_i$'s is

$$\mathcal{L}(\{\mathbf{b}_1, \cdots, \mathbf{b}_n\}) = \left\{ \sum_{i=1}^{n} z_i \mathbf{b}_i \mid z_i \in \mathbb{Z} \right\}.$$

Let $\mathbf{B}$ be the column matrix of $\{\mathbf{b}_1, \cdots, \mathbf{b}_n\}$ and denote the lattice by $\mathcal{L}(\mathbf{B})$.

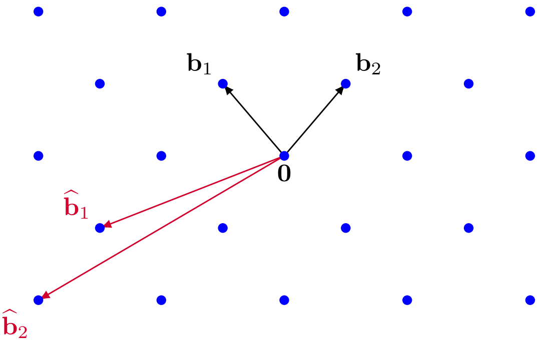
Lattice



Lattice minimum

Given a lattice $\mathcal{L}$, the minimum $\lambda_1(\mathcal{L})$ is the norm of a shortest non-zero vector in $\mathcal{L}$.

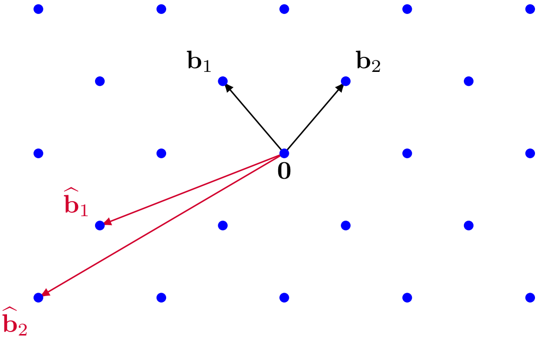
Lattice



Bases of a lattice

Given $\mathbf{B}_1, \mathbf{B}_2 \in \mathbb{Q}^{m\times n}$, then $\mathcal{L}(\mathbf{B}_1) = \mathcal{L}(\mathbf{B}_2)$ iff $\mathbf{B}_2 = \mathbf{B}_1\mathbf{U}$ for some unimodular matrix $\mathbf{U} \in \mathbb{Z}^{n\times n}$.

Lattice

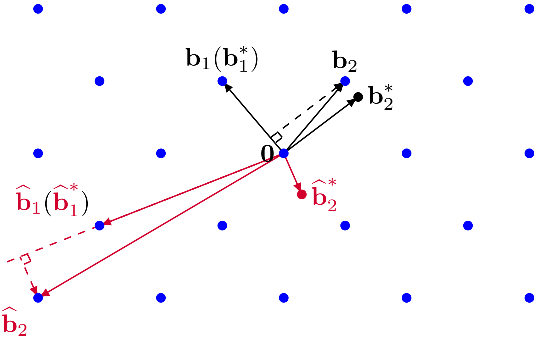


The BKZ lattice reduction algorithm helps to find bases like ($\mathbf{b}_1$, $\mathbf{b}_2$).

Bases of a lattice

Given $\mathbf{B}_1, \mathbf{B}_2 \in \mathbb{Q}^{m \times n}$, then $\mathcal{L}(\mathbf{B}_1) = \mathcal{L}(\mathbf{B}_2)$ iff $\mathbf{B}_2 = \mathbf{B}_1\mathbf{U}$ for some unimodular matrix $\mathbf{U} \in \mathbb{Z}^{n \times n}$.

Lattice



Gram-Schmidt orthogonalization

Let $\mathbf{B}^* = (\mathbf{b}_1^*, \cdots, \mathbf{b}_n^*)$ denote the Gram–Schmidt orthogonalization of $\mathbf{B}$.
The determinant of a lattice $\mathcal{L}$ is $\det(\mathcal{L}) = \prod_i \|\mathbf{b}_i^*\|$.

BKZ-β reduced

Given $\mathbf{B} = (\mathbf{b}_1, \cdots, \mathbf{b}_n)$, let $\mathbf{b}_i^{(j)}$ denote the orthogonal projection of $\mathbf{b}_i$ onto the subspace $(\mathbf{b}_1, \cdots, \mathbf{b}_{j-1})^{\perp}$.

For $i < j \leq n$, let $\mathbf{B}_{[i,j]}$ denote the (matrix) local block $(\mathbf{b}_i^{(i)}, \cdots, \mathbf{b}_j^{(i)})$ and $\mathcal{L}_{[i,j]}$ denote the lattice generated by $\mathbf{B}_{[i,j]}$.

Definition

A basis $\mathbf{B}$ is BKZ-β reduced for block size $\beta \geq 2$ if it is size-reduced* and satisfies:

$$\|\mathbf{b}_i^*\| = \lambda_1(\mathcal{L}_{[i,\min(i+\beta-1,n)]}), \ \forall i \leq n.$$

* A basis $\mathbf{B}$ is size-reduced, if it satisfies $|\mu_{i,j}| \leq 1/2$ for $j < i \leq n$ where $\mu_{i,j} = \frac{\langle \mathbf{b}_i, \mathbf{b}_j^* \rangle}{\|\mathbf{b}_j^*\|^2}$.

The BKZ algorithm

The algorithm attempts to make all local blocks satisfy above the minimality condition simultaneously.

Algorithm 1 BKZ algorithm (Schnorr and Euchner)

Input: A basis $\mathbf{B} = (\mathbf{b}_1, \cdots, \mathbf{b}_n)$, a block size β.
Output: A BKZ-β reduced basis of $\mathcal{L}(\mathbf{B})$.

```
repeat
    for i = 1 to n − 1 do
        SVP_β: find b such that ‖b^(i)‖ = λ_1(L(b_i^(i), ···, b_min(n,i+β−1)^(i))).
        if ‖b_i^*‖ > λ_1(L(b_i^(i), ···, b_min(n,i+β−1)^(i))) then
            LLL-reduce(b_1, ···, b_{i−1}, b, b_i, ···, b_min(n,i+β)).
        else
            LLL-reduce(b_1, ···, b_min(n,i+β)).
        end if
    end for
until no change occurs.
```

C. P. Schnorr and M. Euchner. Lattice basis reduction: Improved practical algorithms and solving subset sum problems. In FCT'91.

The BKZ algorithm

The algorithm attempts to make all local blocks satisfy above the minimality condition simultaneously.

Algorithm 1 BKZ algorithm (Schnorr and Euchner)

Input: A basis $\mathbf{B} = (\mathbf{b}_1, \cdots, \mathbf{b}_n)$, a block size β.
Output: A BKZ-β reduced basis of $\mathcal{L}(\mathbf{B})$.

1: **repeat**
2:     **for** $i = 1$ `to` $n-1$ **do**
3:         `SVP`$_\beta$`: find` $\mathbf{b}$ `such that` $\|\mathbf{b}^{(i)}\| = \lambda_1(\mathcal{L}(\mathbf{b}_i^{(i)}, \cdots, \mathbf{b}^{(i)}_{\min(n,i+\beta-1)}))$.
4:         **if** $\|\mathbf{b}_i^*\| > \lambda_1(\mathcal{L}(\mathbf{b}_i^{(i)}, \cdots, \mathbf{b}^{(i)}_{\min(n,i+\beta-1)}))$ **then**
5:             `LLL-reduce(`$\mathbf{b}_1, \cdots, \mathbf{b}_{i-1}, \mathbf{b}, \mathbf{b}_i, \cdots, \mathbf{b}_{\min(n,i+\beta)}$`).`
6:         **else**
7:             `LLL-reduce(`$\mathbf{b}_1, \cdots, \mathbf{b}_{\min(n,i+\beta)}$`).`
8:         **end if**
9:     **end for**
10: **until** no change occurs.

- [Line 3] In practice, SVP solver can be pruned enumeration or sieving.

SVP Challenge. https://www.latticechallenge.org/svp-challenge/.

Quality of BKZ-β reduced basis

A concrete cryptanalysis relies on the BKZ simulator of Chen and Nguyen (ASIACRYPT'11).

It uses the Gaussian heuristic on local blocks, with a modification for the tail blocks.

Gaussian heuristic

For any random n-dimensional lattice $\mathcal{L}$, we have

$$\lambda_1(\mathcal{L}) \approx \mathrm{GH}(\mathcal{L}) = \frac{1}{v_n^{1/n}} \cdot \det(\mathcal{L})^{1/n}$$

where v_n is the volume of a unit n-ball.

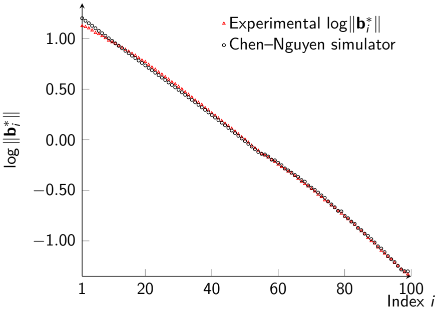
Y. Chen and P.Q. Nguyen. BKZ 2.0: Better lattice security estimates. In ASIACRYPT'11.

Algorithm 2 (Simplified) Chen-Nguyen simulator.

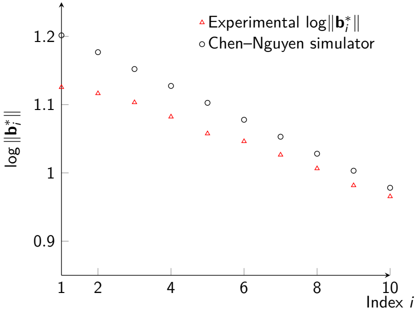
Input: G-S norms $(\|\mathbf{b}_1^*\|, \cdots, \|\mathbf{b}_n^*\|)$, a block size β.
Output: Simulated G-S norms of BKZ$_\beta$-reduced basis of $\mathcal{L}(\mathbf{B})$.

```
repeat
    for i = 1 to n − 1 do
        ~~SVP_β: find b such that ‖b^(i)‖ = λ_1(L(b_i^(i), ···, b_min(n,i+β−1)^(i))).~~
        if ‖b_i^*‖ > GH(L((b_i^(i), ···, b_min(n,i+β)^(i)))) then
            Update ‖b_i^*‖ = GH(L((b_i^(i), ···, b_min(n,i+β)^(i)))).
        else
            Keep ‖b_i^*‖ unchanged.
        end if
    end for
until no change occurs.
```

Practical behavior of Chen-Nguyen's simulator



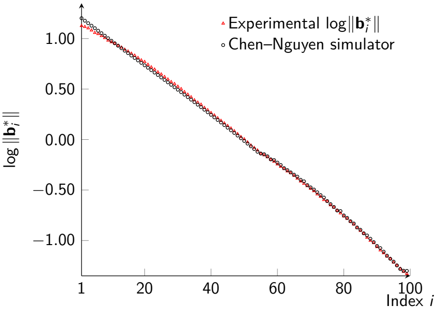
Gram–S. log-norms of BKZ_{45} at tour 50.



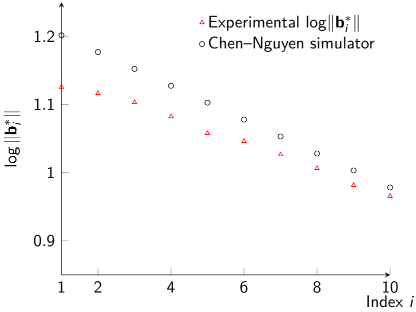
Same as left hand side, but zoomed in.

Such "head concavity" phenomenon has been reported in

Practical behavior of Chen-Nguyen's simulator



Gram–S. log-norms of BKZ_{45} at tour 50.

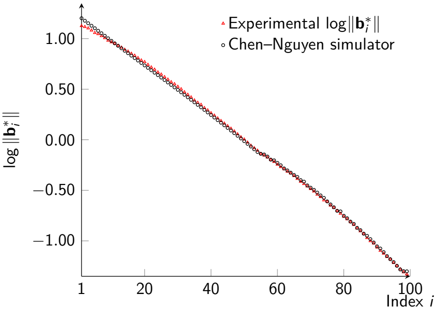


Same as left hand side, but zoomed in.

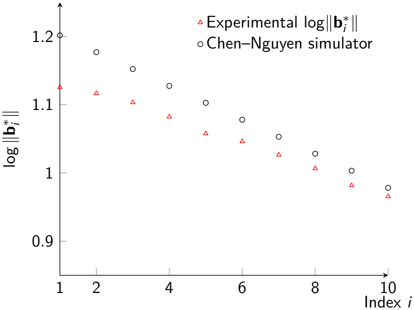
Such "head concavity" phenomenon has been reported in

- experiments of BKZ 2.0 (Chen and Nguyen, ASIACRYPT'11);

Practical behavior of Chen-Nguyen's simulator



Gram–S. log-norms of BKZ_{45} at tour 50.



Same as left hand side, but zoomed in.

Such "head concavity" phenomenon has been reported in

- experiments of BKZ 2.0 (Chen and Nguyen, ASIACRYPT'11);
- and modeled by Yu and Ducas (SAC'17).

Y. Yu and L. Ducas. Second Order Statistical Behavior of LLL and BKZ. In SAC'17.

A better simulator using the distribution of λ_1 in random lattices.

Tools

Let $\Gamma_n = \{\mathcal{L} \in \mathbb{R}^n \mid \mathrm{vol}(\mathcal{L}) = 1\}$ be the set of all full rank-n lattices with unit volume.

Chen [Cor. 3.1.4] and Södergren [Thm. 1]:

Distribution of minimum in random lattices

Sample $\mathcal{L}$ uniformly in Γ_n. The distribution of $v_n \cdot \lambda_1(\mathcal{L})^n$ converges in distribution to $\mathrm{Expo}(1/2)$ as $n \to \infty$.

Take $\lambda_1(\mathcal{L})$ as a random variable Y, then $Y = X^{1/n} \cdot \mathrm{GH}(\mathcal{L})$ for X sampled from $\mathrm{Expo}(1/2)$.

Y. Chen. Réduction de réseau et sécurité concrète du chiffrement complètement homomorphe. PhD thesis, Université Paris Diderot, 2013.

A. Södergren. On the poisson distribution of lengths of lattice vectors in a random lattice. Mathematische Zeitschrift, 2011.

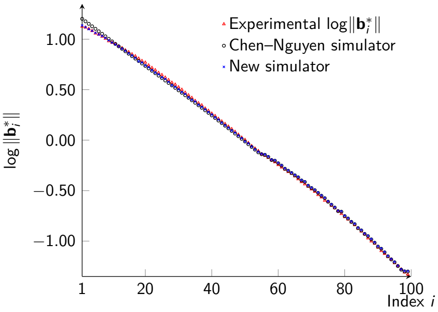
A probabilistic BKZ simulator

Algorithm 3 The new BKZ simulator (simplified)

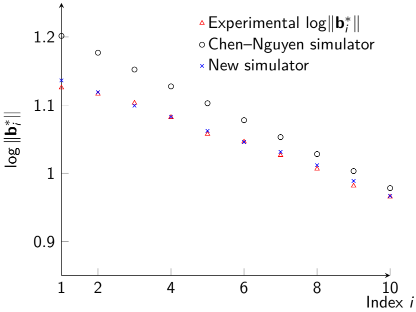
Input: G-S norms $(\|\mathbf{b}_1^*\|, \cdots, \|\mathbf{b}_n^*\|)$, a block size β.
Output: Simulated G-S norms of BKZ-β-reduced basis of $\mathcal{L}(\mathbf{B})$.

repeat
    for $i = 1$ `to` $n - 1$ **do**
        `Sample` X `from Expo[1/2].`
        if $\|\mathbf{b}_i^*\| > X^{1/\beta} \cdot \mathrm{GH}(\mathcal{L}(\mathbf{b}_i^{(i)}, \cdots, \mathbf{b}_{\min(n, i+\beta-1)}^{(i)}))$ **then**
            `Update` $\|\mathbf{b}_i^*\| = X^{1/\beta} \cdot \mathrm{GH}(\mathcal{L}(\mathbf{b}_i^{(i)}, \cdots, \mathbf{b}_{\min(n, i+\beta)}^{(i)}))$.
        else
            `Keep` $\|\mathbf{b}_i^*\|$ `unchanged.`
        end if
    end for
until `no change occurs.`

Quality of our simulator

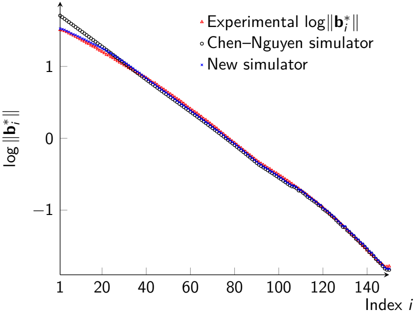


Gram–S. log-norms of BKZ_{45} at tour 50.

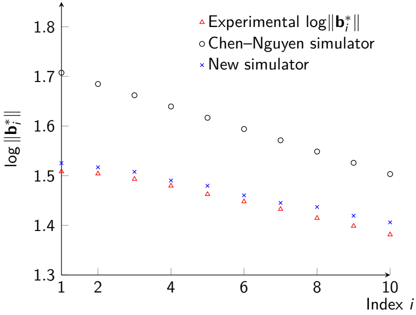


Same as left hand side, but zoomed in.

Quality of our simulator (more)

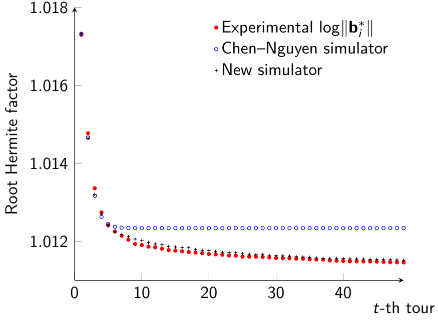


Gram–S. log-norms of BKZ_{60} at tour 20000.

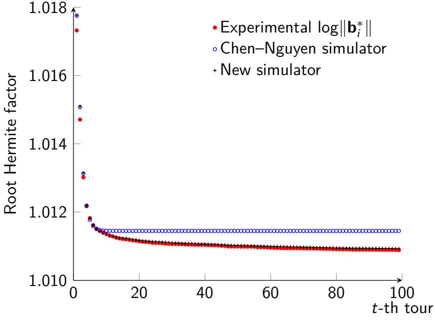


Same as left hand side, but zoomed in.

Quality of our simulator (RHF)



Evolution of RHF during BKZ_{45} (no pruned enumeration) on SVP-100.

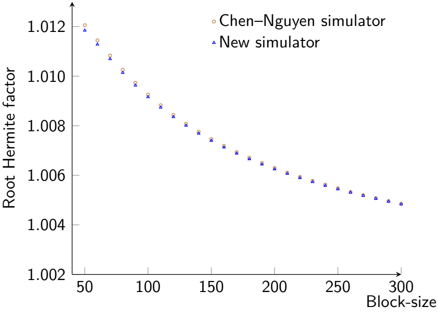


Evolution of RHF during BKZ_{60} (pruned enumeration) on SVP-150.

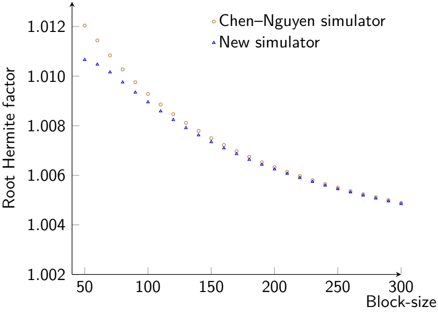
Given a lattice $\mathcal{L}(\mathbf{B})$ of rank n, the Root Hermite Factor of $\mathbf{B}$ is

$$\mathrm{RHF}(\mathbf{B}) = \left(\|\mathbf{b}_1\|/\det(\mathcal{L})^{1/n}\right)^{1/n}.$$

Limit of the head concavity



Simulated RHF for $\beta \in \{50, 60, \cdots, 300\}$.
Here the dimension is $\geq 4\beta$.



Simulated RHF for $\beta \in \{50, 60, \cdots, 300\}$.
Here the dimension is 3β.

For large block sizes, the discrepancy vanishes: both simulators converge to the same root Hermite factors.

Exploit the head concavity phenomenon!

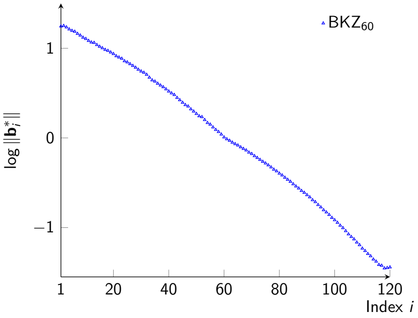
A new BKZ variant: "Pressed BKZ"

Algorithm 4 The pressed-BKZ algorithm

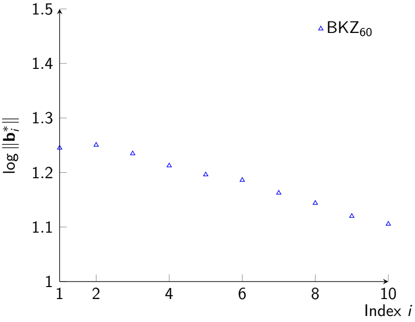
Input: A basis $\mathbf{B} = (\mathbf{b}_1, \cdots, \mathbf{b}_n)$, a block size β.
Output: A pressed-BKZ-β reduced basis of $\mathcal{L}(\mathbf{B})$.

1: **for** start = 1 to $n - \beta + 1$ **do**
2: Re-randomize $\mathcal{L}(\mathbf{b}_{\text{start}}^{(\text{start})}, \cdots, \mathbf{b}_n^{(\text{start})})$.
3: BKZ-β on the block from start to n.
4: **end for**

Experiments: BKZ-60

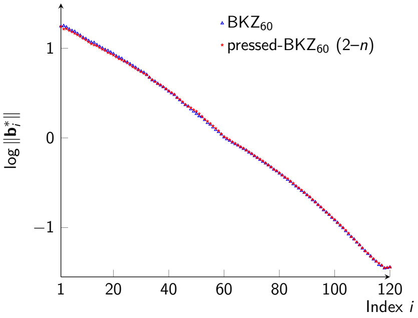


Gram–Schmidt log-norms of BKZ$_{60}$.

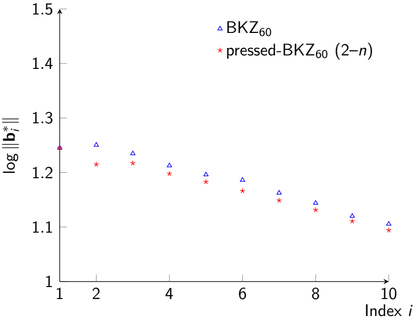


Same as left hand side, but zoomed in.

Experiments: Pressed-BKZ-60 $(2 - n)$

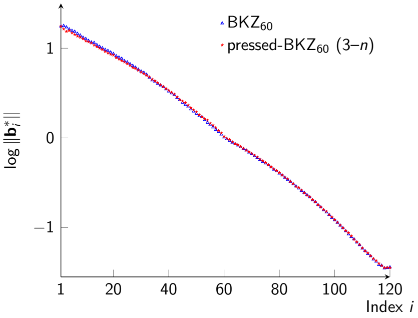


Gram–Schmidt log-norms of (Pressed-)BKZ$_{60}$.

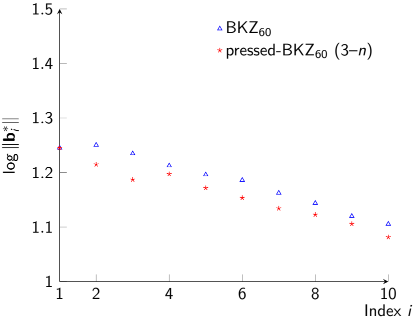


Same as left hand side, but zoomed in.

Experiments: Pressed-BKZ-60 $(3-n)$

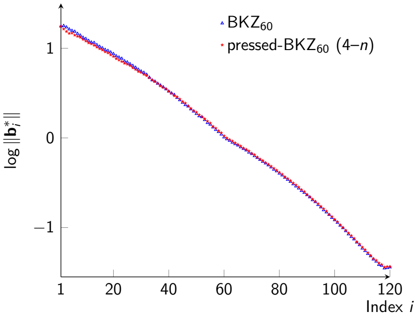


Gram–Schmidt log-norms of (Pressed-)BKZ$_{60}$.

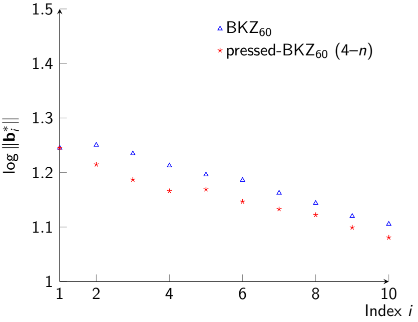


Same as left hand side, but zoomed in.

Experiments: Pressed-BKZ-60 $(4 - n)$

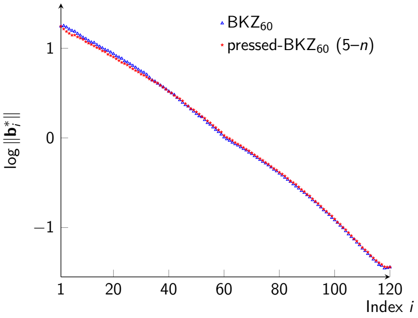


Gram–Schmidt log-norms of (Pressed-)BKZ$_{60}$.

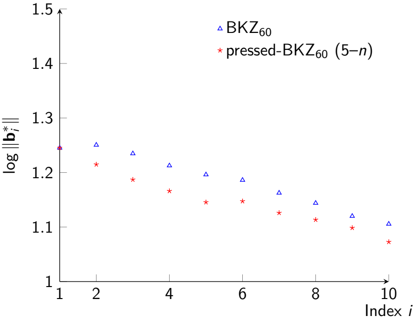


Same as left hand side, but zoomed in.

Experiments: Pressed-BKZ-60 $(5 - n)$

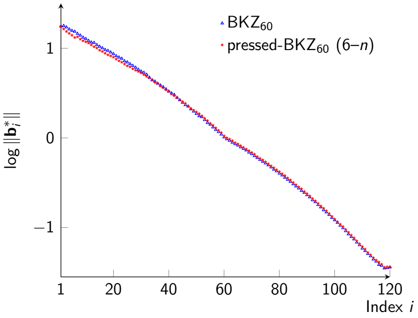


Gram–Schmidt log-norms of (Pressed-)BKZ$_{60}$.

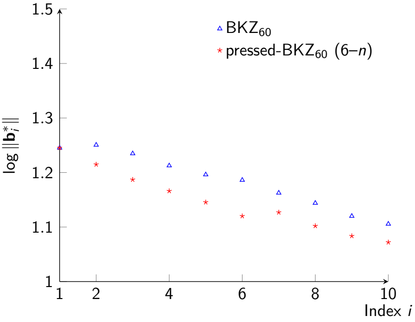


Same as left hand side, but zoomed in.

Experiments: Pressed-BKZ-60 $(6 - n)$

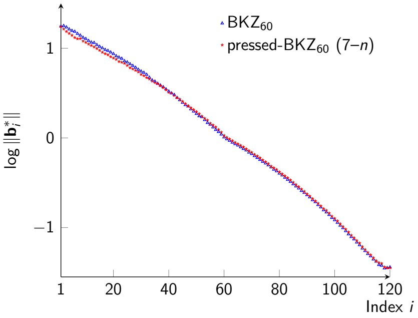


Gram–Schmidt log-norms of (Pressed-)BKZ_{60}.

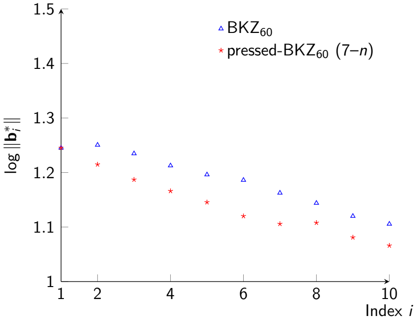


Same as left hand side, but zoomed in.

Experiments: Pressed-BKZ-60 $(7 - n)$

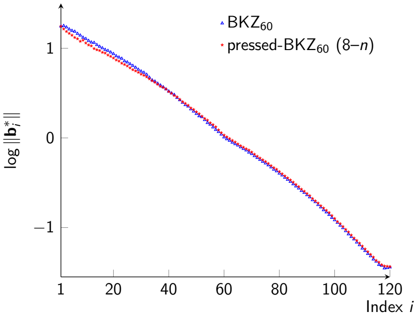


Gram–Schmidt log-norms of (Pressed-)BKZ$_{60}$.

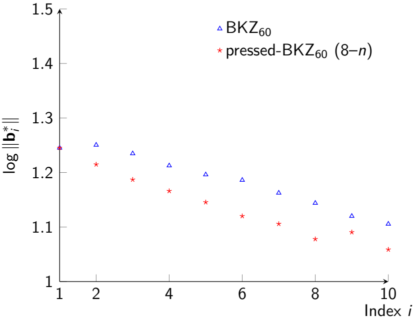


Same as left hand side, but zoomed in.

Experiments: Pressed-BKZ-60 ($8 - n$)

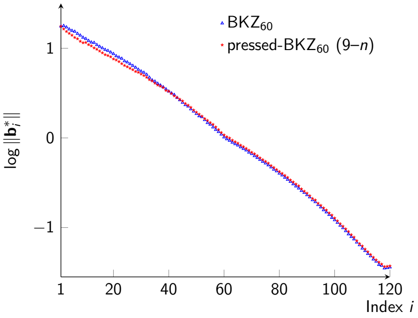


Gram–Schmidt log-norms of (Pressed-)BKZ$_{60}$.

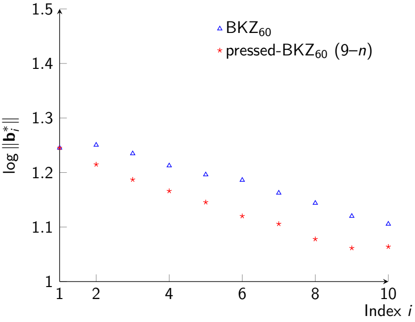


Same as left hand side, but zoomed in.

Experiments: Pressed-BKZ-60 $(9 - n)$

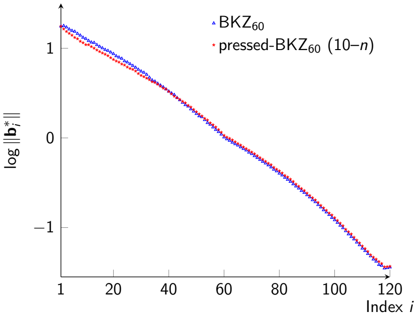


Gram–Schmidt log-norms of (Pressed-)BKZ$_{60}$.

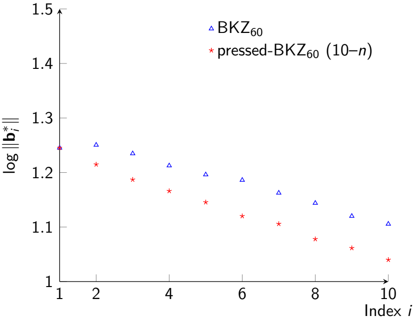


Same as left hand side, but zoomed in.

Experiments: Pressed-BKZ-60 $(10 - n)$



Gram–Schmidt log-norms of (Pressed-)BKZ$_{60}$.



Same as left hand side, but zoomed in.

Comparison with standard BKZ (in preprocessing)

Input: a SVP-120 challenge

- Quality of pressed-BKZ-60 $\approx$ BKZ-80 $\sim$ 90 (after certain #tours). Pressed-BKZ-60 takes less time;

- Solving SVP-120 using the preprocessed pressed-BKZ-60 and a variant of progressive-BKZ in the bkz2_sweet_spot branch of *fplll*. Faster (in experiments) than the lower-bound estimates in the Progressive BKZ (Aono et al. EUROCRYPT'16).

Limitation: strategy is not guaranteed to be optimal.

Y. Aono, Y. Wang, T. Hayashi, and T. Takagi. Improved progressive BKZ algorithms and their precise cost estimation by sharp simulator. EUROCRYPT'16.

`https://github.com/fplll/fpylll/tree/bkz2_sweet_spot`

Conclusion

Impacts:

Conclusion

Impacts:

- Better estimate for *concrete* cryptanalysis;

Conclusion

Impacts:

- Better estimate for *concrete* cryptanalysis;
- No impact for NIST security parameters.

Conclusion

Impacts:

- Better estimate for *concrete* cryptanalysis;
- No impact for NIST security parameters.
- Pressed-BKZ improves quality for *limited* block-sizes;

Conclusion

Impacts:

- Better estimate for *concrete* cryptanalysis;
- No impact for NIST security parameters.
- Pressed-BKZ improves quality for *limited* block-sizes;

Future work:

Conclusion

Impacts:

- Better estimate for *concrete* cryptanalysis;
- No impact for NIST security parameters.
- Pressed-BKZ improves quality for *limited* block-sizes;

Future work:

- Better *strategies* for Pressed-BKZ?

Conclusion

Impacts:

- Better estimate for *concrete* cryptanalysis;
- No impact for NIST security parameters.
- Pressed-BKZ improves quality for *limited* block-sizes;

Future work:

- Better *strategies* for Pressed-BKZ?
- Impact of Pressed-BKZ for larger blocks?

Conclusion

Impacts:

- Better estimate for *concrete* cryptanalysis;
- No impact for NIST security parameters.
- Pressed-BKZ improves quality for *limited* block-sizes;

Future work:

- Better *strategies* for Pressed-BKZ?
- Impact of Pressed-BKZ for larger blocks?
- Rigorous (or less heuristic) analysis of practical behavior of BKZ?

Thank you!