Dimension-Preserving Reductions Between Lattice Problems

Noah Stephens-Davidowitz*

Courant Institute of Mathematical Sciences,
New York University.
noahsd@cs.nyu.edu

Last updated September 6, 2016.

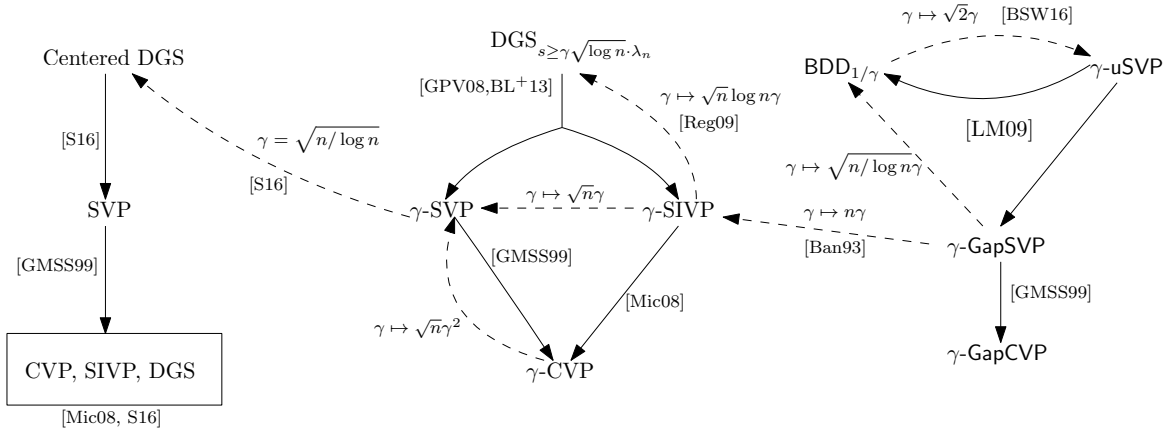
Abstract

Computational problems on lattices have found a remarkable number of applications in computer science. In particular, over the past twenty years, many strong cryptographic primitives have been constructed with their security based on the (worst-case) hardness of various lattice problems.

Due to their importance, there has been much work towards understanding the relationship between these problems. For the parameters that typically interest us, the fastest known algorithms for lattice problems run in time that is exponential in the dimension of the lattice. Therefore, we are typically interested in reductions that *preserve* this dimension. (We actually relax this notion slightly and consider a reduction to be "dimension-preserving" if it increases the dimension by at most an additive constant.)

We summarize many of the known results in this area.

1 The reductions



From left to right, we have the exact lattice search problems (and close relatives), their approximate variants, and their decisional variants (and close relatives). All arrows represent efficient dimension-preserving reductions. Dashed arrows represent reductions that change the approximation factor. We omit the trivial reductions from decision problems to the corresponding search problems and from approximation problems to the corresponding exact problems, and we have left out many constants.

*Supported by the National Science Foundation (NSF) under Grant No. CCF-1320188. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the NSF.

All reductions between BDD, uSVP and GapSVP are originally due to [LM09], though [BSW16] improved the constant in the reduction from BDD to uSVP. The reductions from uSVP to BDD and from uSVP to GapSVP only work for polynomially bounded approximation factors γ. The equivalence of CVP and SIVP is due to [Mic08].

The reduction from $(n\gamma)$-GapSVP to γ-SIVP is an immediate consequence of Banaszczyk's famous transference theorem [Ban93] (which says that $1 \leq \lambda_1(\mathcal{L})\lambda_n(\mathcal{L}^*) \leq O(n)$).

The reduction from $(\sqrt{n}\gamma)$-SIVP to γ-SVP is known in folklore [Mic15], but I don't know of any published source. So, I've include a short proof in Section 4.

Similarly, I know of no source for the reduction from $(\sqrt{n}\gamma^2)$-CVP to γ-SVP (though many works use similar ideas, such as [Kan87, LLS90, MG02, DRS14]). So, I've included a proof in Section 5. The reduction relies crucially on the reduction from BDD to GapSVP in [LM09], and it likely would have appeared in one of [Kan87, LLS90, MG02] if the authors had known about this reduction from BDD to GapSVP when they wrote their respective papers.

2 Some open problems

A lot remains open in this area. Indeed, in nearly all cases, we do not know whether the approximation factors that we obtain are optimal. I will briefly discuss two interesting open questions below, with the disclaimer that this is far from a complete list.

First, does a dimension-preserving reduction exist from γ-SVP to γ'-SIVP with $\gamma/\gamma' \leq \text{poly}(n)$ for all $\gamma = \text{poly}(n)$ [Mic15]? Such a reduction would show that the three most important approximate lattice search problems are equivalent up to polynomial factors. Note that SIVP is *equivalent* to CVP in its exact version. Perhaps this reduction can be extended further?

Second, what is the relationship between the search and decision problems under dimension-preserving reductions? Of course, there are trivial decision-to-search reductions, but what about reductions in the other direction? In fact, it is easy to see that in their exact versions, search and decision lattice problems are equivalent (i.e., SVP reduces to GapSVP and CVP reduces to GapCVP). And, nearly all of our techniques for solving lattice decision problems actually work by solving the corresponding search problem. So, one might hope that the problems are equivalent. Alternatively, one might hope to conclusively separate the problems, or at least find techniques for decision problems that seem "inherently decisional." Recently, [S15] showed that there exist search-to-decision reductions for approximation factors slightly greater than one that lose quite a bit in the approximation factor, but this is still far from a complete answer.

3 Definitions

A lattice $\mathcal{L} \subset \mathbb{Q}^n$ is the set of all integer linear combinations of linearly independent basis vectors $\mathbf{b}_1, \ldots, \mathbf{b}_n \in \mathbb{Q}^n$. $\mathbf{B} = (\mathbf{b}_1, \ldots, \mathbf{b}_n)$ is called a basis for the lattice, and it is not unique.

We write $\lambda_1(\mathcal{L})$ for the length of a shortest non-zero vector in the lattice $\mathcal{L} \subset \mathbb{Q}^n$. Similarly, $\lambda_2(\mathcal{L})$ is the length of the shortest vector that is linearly independent from a vector of length $\lambda_1(\mathcal{L})$, and $\lambda_n(\mathcal{L})$ is the smallest radius r such that there exist n linearly independent lattice vectors of length r. For a shift vector $\mathbf{t} \in \mathbb{Q}^n$, parameter $s > 0$, lattice $\mathcal{L} \subset \mathbb{Q}^n$, the discrete Gaussian distribution $D_{\mathcal{L}-\mathbf{t},s}$ is the probability distribution over the shifted lattice $\mathcal{L} - \mathbf{t}$ that assigns probability proportional to $e^{-\pi\|\mathbf{x}\|^2/s^2}$ to each $\mathbf{x} \in \mathcal{L} - \mathbf{t}$.

3.1 Exact problems (and close relatives)

Definition. *SVP (the Shortest Vector Problem) is the search problem defined as follows: The input is a basis* $\mathbf{B}$ *for a lattice* $\mathcal{L} \subset \mathbb{Q}^n$*. The goal is to output a lattice vector* $\mathbf{x}$ *with* $\|\mathbf{x}\| = \lambda_1(\mathcal{L})$.

Definition. *CVP (the Closest Vector Problem) is the search problem defined as follows: The input is a basis* $\mathbf{B}$ *for a lattice* $\mathcal{L} \subset \mathbb{Q}^n$ *and a target vector* $\mathbf{t} \in \mathbb{Q}^n$*. The goal is to output a lattice vector* $\mathbf{x}$ *with* $\|\mathbf{x} - \mathbf{t}\| = \mathrm{dist}(\mathbf{t}, \mathcal{L})$*.*

Definition. *SIVP (the Shortest Independent Vectors Problem) is the search problem defined as follows: The input is a basis* $\mathbf{B}$ *for a lattice* $\mathcal{L} \subset \mathbb{Q}^n$*. The goal is to output* n *linearly independent lattice vectors of length at most* $\lambda_n(\mathcal{L})$*.*

Definition. *DGS (the Discrete Gaussian Sampling problem) is the sampling problem defined as follows: The input is a basis* $\mathbf{B}$ *for a lattice* $\mathcal{L} \subset \mathbb{Q}^n$*, a shift* $\mathbf{t} \in \mathbb{Q}^n$*, and a parameter* $s > 0$*. The goal is to output a vector whose distribution is statistically close to* $D_{\mathcal{L}-\mathbf{t},s}$*.*

Definition. *Centered DGS is the sampling defined as follows: The input is a basis* $\mathbf{B}$ *for a lattice* $\mathcal{L} \subset \mathbb{Q}^n$ *and a parameter* $s > 0$*. The goal is to output a vector whose distribution is statistically close to* $D_{\mathcal{L},s}$*.*

3.2 Approximation problems

Definition. *For any parameter* $\gamma \geq 1$*,* γ*-SVP (the Shortest Vector Problem) is the search problem defined as follows: The input is a basis* $\mathbf{B}$ *for a lattice* $\mathcal{L} \subset \mathbb{Q}^n$*. The goal is to output a lattice vector* $\mathbf{x}$ *with* $0 < \|\mathbf{x}\| \leq \gamma \cdot \lambda_1(\mathcal{L})$*.*

Definition. *For any parameter* $\gamma \geq 1$*,* γ*-CVP (the Closest Vector Problem) is the search problem defined as follows: The input is a basis* $\mathbf{B}$ *for a lattice* $\mathcal{L} \subset \mathbb{Q}^n$ *and a target vector* $\mathbf{t} \in \mathbb{Q}^n$*. The goal is to output a lattice vector* $\mathbf{x}$ *with* $\|\mathbf{x} - \mathbf{t}\| \leq \gamma \cdot \mathrm{dist}(\mathbf{t}, \mathcal{L})$*.*

Definition. *For any parameter* $\gamma \geq 1$*,* γ*-SIVP (the Shortest Independent Vectors Problem) is the search problem defined as follows: The input is a basis* $\mathbf{B}$ *for a lattice* $\mathcal{L} \subset \mathbb{Q}^n$*. The goal is to output* n *linearly independent lattice vectors of length at most* $\gamma \cdot \lambda_n(\mathcal{L})$*.*

Definition. *For any function* σ *that maps lattices to positive real numbers,* $DGS_{s \geq \sigma(\mathcal{L})}$ *(the Discrete Gaussian Sampling problem) is the sampling problem defined as follows: The input is a basis* $\mathbf{B}$ *for a lattice* $\mathcal{L} \subset \mathbb{Q}^n$*, a shift* $\mathbf{t} \in \mathbb{Q}^n$*, and a parameter* $s \geq \sigma(\mathcal{L})$*. The goal is to output a vector whose distribution is statistically close to* $D_{\mathcal{L}-\mathbf{t},s}$*.*

3.3 Decision problems (and close relatives)

Definition. *For any parameter* $\gamma \geq 1$*,* γ*-GapSVP (the Shortest Vector Problem) is the decision problem defined as follows: The input is a basis* $\mathbf{B}$ *for a lattice* $\mathcal{L} \subset \mathbb{Q}^n$ *and a length* $d > 0$*. It is a YES instance if* $\lambda_1(\mathcal{L}) \leq d$ *and a NO instance if* $\lambda_1(\mathcal{L}) > \gamma \cdot d$*.*

Definition. *For any parameter* $\gamma \geq 1$*,* γ*-GapCVP (the Closest Vector Problem) is the decision problem defined as follows: The input is a basis* $\mathbf{B}$ *for a lattice* $\mathcal{L} \subset \mathbb{Q}^n$ *and a target vector* $\mathbf{t} \in \mathbb{Q}^n$*. It is a YES instance if* $\mathrm{dist}(\mathbf{t}, \mathcal{L}) \leq d$ *and a NO instance if* $\mathrm{dist}(\mathbf{t}, \mathcal{L}) > \gamma \cdot d$*.*

Definition. *For any parameter* $\alpha > 0$*,* BDD_α *(the Bounded Distance Decoding Problem) is the search problem defined as follows: The input is a basis* $\mathbf{B}$ *for a lattice* $\mathcal{L} \subset \mathbb{Q}^n$ *and a target vector* $\mathbf{t} \in \mathbb{Q}^n$ *with* $\mathrm{dist}(\mathbf{t}, \mathcal{L}) < \alpha \cdot \lambda_1(\mathcal{L})$*. The goal is to output a lattice vector* $\mathbf{x}$ *with* $\|\mathbf{x} - \mathbf{t}\| = \mathrm{dist}(\mathbf{t}, \mathcal{L})$*.*

Definition. *For any parameter* $\gamma \geq 1$*,* γ*-uSVP (the unique Shortest Vector Problem) is the search problem defined as follows: The input is a basis* $\mathbf{B}$ *for a lattice* $\mathcal{L} \subset \mathbb{Q}^n$ *with* $\lambda_2(\mathcal{L}) \geq \gamma \cdot \lambda_1(\mathcal{L})$*. The goal is to output a lattice vector* $\mathbf{x}$ *with* $\|\mathbf{x}\| = \lambda_1(\mathcal{L})$*.*

4 $\sqrt{n}\gamma$-SIVP to γ-SVP

Given a basis, $\mathbf{B} = (\mathbf{b}_1, \ldots, \mathbf{b}_n)$, we define its Gram-Schmidt orthogonalization $(\widetilde{\mathbf{b}}_1, \ldots, \widetilde{\mathbf{b}}_n)$ by

$$\widetilde{\mathbf{b}}_i = \pi_{\{\mathbf{b}_1,\ldots,\mathbf{b}_{i-1}\}^\perp}(\mathbf{b}_i) \ ,$$

and the Gram-Schmidt coefficients $\mu_{i,j}$ by

$$\mu_{i,j} = \frac{\langle \mathbf{b}_i, \widetilde{\mathbf{b}}_j \rangle}{\|\widetilde{\mathbf{b}}_j\|^2} \ .$$

Here, π_A is the orthogonal projection on the subspace A and $\{\mathbf{b}_1, \ldots, \mathbf{b}_{i-1}\}^\perp$ denotes the subspace orthogonal to $\mathbf{b}_1, \ldots, \mathbf{b}_{i-1}$.

A basis $\mathbf{B} = (\mathbf{b}_1, \ldots, \mathbf{b}_n)$ of $\mathcal{L}$ is a γ-Hermite-Korkin-Zolotarev (γ-HKZ) basis if

1. $\|\mathbf{b}_1\| \leq \gamma \cdot \lambda_1(\mathcal{L})$;
2. the Gram-Schmidt coefficients of $\mathbf{B}$ satisfy $|\mu_{i,j}| \leq \frac{1}{2}$ for all $j < i$; and
3. $\pi_{\{\mathbf{b}_1\}^\perp}(\mathbf{b}_2), \ldots, \pi_{\{\mathbf{b}_1\}^\perp}(\mathbf{b}_n)$ is a γ-HKZ basis of $\pi_{\{\mathbf{b}_1\}^\perp}(\mathcal{L})$.

In particular, note that $\widetilde{\mathbf{b}}_i$ is a γ-approximate shortest non-zero vector in the projected lattice $\pi_{\{\mathbf{b}_1,\ldots,\mathbf{b}_{i-1}\}^\perp}(\mathcal{L})$.

Observe that a γ-SVP oracle can be used to compute a γ-HKZ basis (in a dimension-preserving way). (Just find a γ-approximate shortest vector, project onto the space orthogonal to that vector, and repeat.) The result then follows from the following basic lemma.

Lemma 4.1. *If $\mathbf{B} = (\mathbf{b}_1, \ldots, \mathbf{b}_n)$ is a γ-HKZ basis of $\mathcal{L}$, then $\|\mathbf{b}_i\| \leq \sqrt{n+3} \cdot \gamma \cdot \lambda_n(\mathcal{L})/2$ for all i.*

Proof. By Item 2 above, we have

$$\|\mathbf{b}_i\|^2 \leq \|\widetilde{\mathbf{b}}_i\|^2 + \frac{1}{4} \cdot \sum_{j<i} \|\widetilde{\mathbf{b}}_j\|^2 \leq \frac{n+3}{4} \cdot \max_j \|\widetilde{\mathbf{b}}_j\|^2 \ .$$

It therefore suffices to prove that $\|\widetilde{\mathbf{b}}_j\| \leq \gamma \cdot \lambda_n(\mathcal{L})$ for all j.

Suppose $\lambda_n(\mathcal{L}) < \|\widetilde{\mathbf{b}}_j\|/\gamma$ for some j, and let $\mathbf{v}_1, \ldots, \mathbf{v}_j \in \mathcal{L}$ be linearly independent lattice vectors with $\|\mathbf{v}_i\| < \|\widetilde{\mathbf{b}}_j\|/\gamma$. Since they are linearly independent, there exists some i such that $\pi_{\{\mathbf{b}_1,\ldots,\mathbf{b}_{j-1}\}^\perp}(\mathbf{v}_i) \neq \mathbf{0}$. Therefore, $\lambda_1(\pi_{\{\mathbf{b}_1,\ldots,\mathbf{b}_{j-1}\}^\perp}(\mathcal{L})) \leq \|\mathbf{v}_i\| < \|\widetilde{\mathbf{b}}_j\|/\gamma$. This contradicts the assumption that $\mathbf{B}$ is a γ-HKZ basis.

The result follows. □

Corollary 4.2. *For any $\gamma = \gamma(n) \geq 1$, there is an efficient dimension-preserving reduction from γ'-SIVP to γ-SVP where*

$$\gamma' := \frac{\sqrt{n+3}}{2} \cdot \gamma \ .$$

5 $\sqrt{n}\gamma^2$-CVP to γ-SVP

We will need two results. The first is the reduction of [LM09] from BDD to uSVP. The second is Babai's famous nearest-plane algorithm. (The reduction also uses a γ-HKZ basis, which is defined in the previous section.)

Theorem 5.1 ([LM09])**.** *For any $\gamma = \gamma(n) \geq 1$, there is an efficient dimension-preserving reduction from $1/(2\gamma)$-BDD to γ-uSVP.*

Theorem 5.2 (Babai's algorithm [Bab86])**.** *There is an efficient algorithm that takes as input a lattice basis* $(\mathbf{b}_1, \ldots, \mathbf{b}_d) \in \mathbb{Q}^n$ *and a target vector* $\mathbf{t} \in \mathrm{span}(\mathbf{b}_1, \ldots, \mathbf{b}_d)$ *and outputs a lattice vector* $\mathbf{y} \in \mathcal{L}(\mathbf{b}_1, \ldots, \mathbf{b}_d)$ *with* $\|\mathbf{y} - \mathbf{t}\|^2 \leq \frac{1}{4} \cdot \sum_i \|\widetilde{\mathbf{b}_i}\|^2$.

The idea behind the reduction is to "split" the CVP instance into a "GDD part" and a "BDD part." (Guaranteed Distance Decoding, or GDD, asks us to find a lattice point that is some fixed distance away from the target vector, regardless of how close the target is to the lattice. E.g., Theorem 5.2 gives a solution to GDD with distance $\frac{1}{2} \cdot \sqrt{\sum_i \|\widetilde{\mathbf{b}_i}\|^2}$.) To do this, we use our SVP oracle to find a good basis $\mathbf{B} := (\mathbf{b}_1, \ldots, \mathbf{b}_n)$, and we guess an index k such that $\|\widetilde{\mathbf{b}_i}\| \lesssim \mathrm{dist}(\mathbf{t}, \mathcal{L})$ for $i \leq k$ and $\|\widetilde{\mathbf{b}}_{k+1}\| \gtrsim \mathrm{dist}(\mathbf{t}, \mathcal{L})$. We then use our SVP oracle and Theorem 5.1 to solve the BDD instance given by $\pi_{(\mathbf{b}_1,\ldots,\mathbf{b}_k)^\perp}(\mathbf{t})$ and $\pi_{(\mathbf{b}_1,\ldots,\mathbf{b}_k)^\perp}(\mathcal{L})$, and we use our good basis and Babai's algorithm to solve a related GDD instance on $\mathcal{L}(\mathbf{b}_1, \ldots, \mathbf{b}_k)$. Combining the results appropriately gives an approximate closest vector to $\mathbf{t}$.

Theorem 5.3. *For any* $\gamma = \gamma(n) \geq 1$*, there is a dimension-preserving reduction from* $(\sqrt{n}\gamma^2)$*-CVP to* γ*-SVP.*

Proof. On input a lattice $\mathcal{L} \subset \mathbb{Q}^n$ and a target $\mathbf{t} \in \mathbb{Q}^n$, the reduction first uses its SVP oracle to compute a γ-HKZ basis of $\mathcal{L}$, $\mathbf{B} = (\mathbf{b}_1, \ldots, \mathbf{b}_n)$. Let $\pi_i := \pi_{\{\mathbf{b}_1,\ldots,\mathbf{b}_i\}^\perp}$ represent projection onto the subspace orthogonal to the first i basis vectors, $\hat{\pi}_i := \pi_{\mathrm{span}(\mathbf{b}_1,\ldots,\mathbf{b}_i)}$, and $\mathcal{L}_i := \pi_i(\mathcal{L})$. For $i = 0, \ldots, n$, the reductions runs the procedure from Theorem 5.1 on input $\pi_i(\mathbf{t})$ and $\mathcal{L}_i$, receiving as output $\mathbf{w}_i = \sum_{j=i+1}^n a_j \pi_i(\mathbf{b}_j)$. Let $\mathbf{x}_i := \sum_{j=i+1}^n a_j \mathbf{b}_j$ be the "lift" of $\mathbf{w}_i$ into $\mathcal{L}$. The reduction then uses Babai's algorithm on input $(\mathbf{b}_1, \ldots, \mathbf{b}_i)$ and $\hat{\pi}_i(\mathbf{t} - \mathbf{x}_i)$, receiving as output $\mathbf{z}_i \in \mathcal{L}$. Finally, it returns the closest lattice vector amongst the $\mathbf{y}_i := \mathbf{z}_i + \mathbf{x}_i$.

The running time is clear. By embedding all relevant lattices in $\mathbb{Q}^n$, we may assume without loss of generality that $\gamma = \gamma(i) = \gamma(n)$ for all $1 \leq i \leq n$. Let $0 \leq k \leq n$ be maximal such that $\|\widetilde{\mathbf{b}_i}\| \leq 2\gamma^2 \cdot \mathrm{dist}(\mathbf{t}, \mathcal{L})$ for all $0 < i \leq k$. If $k = n$, then the result follows immediately from Theorem 5.2, since $\mathbf{y}_n = \mathrm{Babai}(\mathbf{B}, \mathbf{t})$. So, we assume $k < n$.

Note that $2\gamma^2 \,\mathrm{dist}(\mathbf{t}, \mathcal{L}) < \|\widetilde{\mathbf{b}}_{k+1}\| \leq \gamma\lambda_1(\mathcal{L}_k)$, so that $\mathrm{dist}(\pi_k(\mathbf{t}), \mathcal{L}_k) \leq \mathrm{dist}(\mathbf{t}, \mathcal{L}) < \lambda_1(\mathcal{L}_k)/(2\gamma)$. I.e., $(\pi_k(\mathbf{t}), \mathcal{L}_k)$ is a valid $1/(2\gamma)$-BDD instance. Then, by Theorem 5.1, $\mathbf{w}_k \in \mathcal{L}_k$ will be the unique closest vector in $\mathcal{L}_k$ to $\pi_k(\mathbf{t})$. In particular,

$$\|\mathbf{w}_k - \pi_k(\mathbf{t})\| = \mathrm{dist}(\pi_k(\mathbf{t}), \mathcal{L}_k) \leq \mathrm{dist}(\mathbf{t}, \mathcal{L}) \,.$$

And, by Theorem 5.2,

$$\|\mathbf{z}_k - \hat{\pi}_i(\mathbf{t} - \mathbf{x}_k)\|^2 \leq \frac{1}{4} \cdot \sum_{j=1}^{k} \|\mathbf{b}_j\|^2 \leq (n-1) \cdot \gamma^4 \cdot \mathrm{dist}(\mathbf{t}, \mathcal{L})^2 \,.$$

Therefore, we have

$$\begin{aligned}
\|\mathbf{y}_k - \mathbf{t}\|^2 &= \|\pi_k(\mathbf{y}_k - \mathbf{t})\|^2 + \|\hat{\pi}_k(\mathbf{y}_k - \mathbf{t})\|^2 \\
&= \|\mathbf{w}_k - \pi_k(\mathbf{t})\|^2 + \|\mathbf{z}_k - \hat{\pi}_k(\mathbf{t} - \mathbf{x}_k)\|^2 \\
&\leq \mathrm{dist}(\mathbf{t}, \mathcal{L})^2 + (n-1) \cdot \gamma^4 \cdot \mathrm{dist}(\mathbf{t}, \mathcal{L})^2 \\
&\leq n\gamma^4 \,\mathrm{dist}(\mathbf{t}, \mathcal{L})^2 \,,
\end{aligned}$$

as needed. □

References

[Bab86] L. Babai. On Lovász' lattice reduction and the nearest lattice point problem. *Combinatorica*, 6(1):1–13, 1986.

[Ban93] W. Banaszczyk. New bounds in some transference theorems in the geometry of numbers. *Mathematische Annalen*, 296(4):625–635, 1993.

[BL+13] Zvika Brakerski, Adeline Langlois, Chris Peikert, Oded Regev, and Damien Stehlé. Classical hardness of learning with errors. In *STOC*, pages 575–584, 2013.

[BSW16] Shi Bai, Damien Stehlé, and Weiqiang Wen. Improved reduction from the Bounded Distance Decoding problem to the Unique Shortest Vector Problem in lattices. In *ICALP*, 2016.

[DRS14] Daniel Dadush, Oded Regev, and Noah Stephens-Davidowitz. On the closest vector problem with a distance guarantee. In *IEEE 29th Conference on Computational Complexity*, pages 98–109, 2014. Full version available at `http://arxiv.org/abs/1409.8063`.

[GMSS99] O. Goldreich, D. Micciancio, S. Safra, and J.-P. Seifert. Approximating shortest lattice vectors is not harder than approximating closest lattice vectors. *Information Processing Letters*, 71(2):55 – 61, 1999.

[GPV08] Craig Gentry, Chris Peikert, and Vinod Vaikuntanathan. Trapdoors for hard lattices and new cryptographic constructions. In *STOC*, pages 197–206, 2008.

[Kan87] Ravi Kannan. Minkowski's convex body theorem and integer programming. *Math. Oper. Res.*, 12(3):415–440, 1987.

[LLS90] J. C. Lagarias, H. W. Lenstra, Jr., and C.-P. Schnorr. Korkin-Zolotarev bases and successive minima of a lattice and its reciprocal lattice. *Combinatorica*, 10(4):333–348, 1990.

[LM09] Vadim Lyubashevsky and Daniele Micciancio. On bounded distance decoding, unique shortest vectors, and the minimum distance problem. In *Advances in Cryptology-CRYPTO 2009*, pages 577–594. Springer, 2009.

[Mic08] Daniele Micciancio. Efficient reductions among lattice problems. In *Proceedings of the Nineteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 84–93. ACM, New York, 2008.

[Mic15] Daniele Micciancio. Private communication, 2015.

[MG02] D. Micciancio and S. Goldwasser. *Complexity of lattice Problems: a cryptographic perspective*. Kluwer Academic Publishers, Boston, Massachusetts, March 2002.

[Reg09] Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. *Journal of the ACM*, 56(6):Art. 34, 40, 2009.

[S15] Noah Stephens-Davidowitz. Search-to-decision reductions for lattice problems with approximation factors (slightly) greater than one. `http://arxiv.org/abs/1512.04138`. 2015.

[S16] Noah Stephens-Davidowitz. Discrete Gaussian sampling reduces to CVP and SVP. In *SODA*, 2016.