Electronic Colloquium on Computational Complexity, Revision 1 of Report No. 52 (1998)

ECCC

INCREMENTAL ALGORITHMS FOR LATTICE PROBLEMS

BORIS HEMKEMEIER AND FRANK VALLENTIN

ABSTRACT. In this short note we give incremental algorithms for the following lattice problems: finding a basis of a lattice, computing the successive minima, and determining the orthogonal decomposition. We prove an upper bound for the number of update steps for every insertion order. For the determination of the orthogonal decomposition we efficiently implement an argument due to Kneser.

1. INTRODUCTION

Many problems in computational geometry permit a natural computation by an incremental algorithm. Incremental algorithms process only one object at a time and insert it into a data structure. Most incremental algorithms follow an abstract framework: After processing a new object it is inserted into a data structure. It is first located where the data structure has to be changed (*localization step*). Then the data structure has to be updated locally (*update step*) in order to perform the insertion of a new object.

Here we apply the incremental construction paradigm to the design of lattice algorithms. Let $v_1, \ldots, v_m$ be vectors which span a Euclidean space E and let $L = \mathbb{Z}v_1 + \cdots + \mathbb{Z}v_m$ be the lattice which is generated by these vectors. Suppose that we want to compute a property of L. First, we compute the property of the lattice $L_1 = \mathbb{Z}v_1$. Then we check whether $v_2 \in L_1$ (localization step). If $v_2 \in L_1$, then nothing has to be done. If $v_2 \notin L_1$, then we compute the property of the lattice $L_2 = L_1 + \mathbb{Z}v_2$ (update step), etc. In every update step we compute a lattice basis for the new lattice L_i which is computationally more expensive than the localization step. Hence, this algorithmic framework is attractive if the number of update steps is small.

After fixing notation in Section 2 we state an upper bound for the number of update steps for every insertion order in Section 3. In Section 4 we give algorithms for the following lattice problems: an algorithm which finds a basis of a lattice given by a set of generators, an algorithm for the computation of the successive minima of a lattice given by a complete set of generators (a generating set S of a lattice L is called *complete* if S contains every vector $v \in L \setminus \{0\}$ with $\|v\| \leq \max_{w \in S} \|w\|$), and an algorithm for determining the orthogonal decomposition of a lattice given by a complete set of generators.

These considerations result in a simple meta algorithm with practical impact. It offers a significant performance benefit compared with straightforward implementations for classical algorithms. For experimental results see the technical report [6]. This note is a concise version of this report where we in particular emphasize the incremental algorithmic framework.

The second author was supported by the Netherlands Organization for Scientific Research under grant NWO 639.032.203 and by the Deutsche Forschungsgemeinschaft (DFG) under grant SCHU 1503/4-1.

ISSN 1433-8092

2. Notation

Let E be a d-dimensional Euclidean space. Its inner product is denoted by $(\cdot,\cdot)$ and the associated norm by $\|\cdot\| = \sqrt{(\cdot,\cdot)}$. The d-dimensional unit ball is denoted by B_d. A point set $L \subseteq E$ is called a *lattice* if there exist linearly independent vectors $b_1,\dots,b_n \in E$ such that $L = \mathbb{Z}b_1 + \cdots + \mathbb{Z}b_n$. Then, $(b_1,\dots,b_n)$ is called a *basis* of L and n is called the *rank* of L. The *volume* of L is given by $\operatorname{vol} L = |\det(b_1,\dots,b_n)|$. A lattice $L' \subseteq E$ is called a *sublattice* of L if $L' \subseteq L$. If the rank of L' and L is d, then the index of L' in L is $[L:L'] = \operatorname{vol} L' / \operatorname{vol} L$. The *$k$-th successive minima* $\lambda_k(L)$ is the minimum value λ such that λB_d contains at least k linearly independent lattice points of L. We will need the following theorem of Minkowski (see e.g. [5]).

Theorem 2.1. Let $L \subseteq E$ be a lattice of rank d. Then

$$\frac{2^d}{d!} \operatorname{vol} L \le \lambda_1(L)\lambda_2(L)\cdots\lambda_d(L) \operatorname{vol} B_d \le 2^d \operatorname{vol} L.$$

3. Chains of Sublattices

We want to construct a lattice L, which is generated by the vectors $v_1,\dots,v_m$, incrementally. Update steps are necessary if $v_i \notin \mathbb{Z}v_1 + \cdots + \mathbb{Z}v_{i-1}$, where $i = 1,\dots,m$. The next theorem gives an upper bound for the number of update steps.

Theorem 3.1. Let $v_1,\dots,v_m \in E$ be vectors which span E and which generate the lattice L. Define $B = \max_{i=1,\dots,m} \|v_i\|$. Consider the chain of lattices

$$\mathbb{Z}v_1 \subseteq \mathbb{Z}v_1 + \mathbb{Z}v_2 \subseteq \dots \subseteq \mathbb{Z}v_1 + \mathbb{Z}v_2 + \cdots + \mathbb{Z}v_m. \tag{1}$$

Then, in (1) inequality holds at most $d + \log_2(d!(B/\lambda_1(L))^d)$ times.

Proof. First we transform (1) into a new chain of lattices which are all of full rank d. Choose indices $1 \le i_1 < i_2 < \dots < i_d \le m$ such that for all $j \in \{1,\dots,d\}$ the rank of $\mathbb{Z}v_{i_1} + \cdots + \mathbb{Z}v_{i_j}$ is j and i_j is minimal with this property. We define the lattice $L' = \mathbb{Z}v_{i_1} + \cdots + \mathbb{Z}v_{i_d}$ and consider the transformed chain

$$L' = \mathbb{Z}v_1 + L' \subseteq \mathbb{Z}v_1 + \mathbb{Z}v_2 + L' \subseteq \dots \subseteq \mathbb{Z}v_1 + \mathbb{Z}v_2 + \cdots + \mathbb{Z}v_m + L' = L. \tag{2}$$

The number of inequalities in (1) is at most d plus the number of inequalities in the chain (2). Define $L'_i = \mathbb{Z}v_1 + \cdots + \mathbb{Z}v_i + L'$, where $i = 1,\dots,m$. Since we have

$$\operatorname{vol} L' / \operatorname{vol} L = [L:L'] = \prod_{i=2}^{m} [L'_i : L'_{i-1}],$$

the number of inequalities in the chain (2) is at most the number of prime factors of $\operatorname{vol} L' / \operatorname{vol} L$ which is at most $\log_2(\operatorname{vol} L' / \operatorname{vol} L)$. To finish the proof we apply Theorem 2.1 to the quotient $\operatorname{vol} L' / \operatorname{vol} L$ and use the fact $\lambda_d(L') \le B$. □

An immediate consequence of Theorem 3.1 is an upper bound for the size of a minimal generating set.

Corollary 3.2. Let $L \subseteq E$ be a lattice of full rank d. Let $S \subseteq L$ be a finite generating set of L. Then there exists a subset $S' \subseteq S$ which generates L of size at most $d + \log_2(d!(B/\lambda_1(L))^d)$ where $B = \max_{v\in S} \|v\|$.

Note that having long vectors in a lattice basis is not avoidable in general: Conway and Sloane [3] constructed a lattice in dimension 11 which is generated by its 24 shortest vectors but in which no set of 11 shortest vectors forms a basis.

4. ALGORITHMS

In this section we propose algorithms for lattice problems which take advantage of the incremental construction. The first two algorithms for computing a lattice basis and for computing the successive minima are straightforward. For the computation of the unique orthogonal decomposition we develop new ideas based on an argument of Kneser.

4.1. **Lattice Basis.** For computing a lattice basis from a *large* set of generators we use an algorithm for computing a lattice basis from a *small* set of generators as a subroutine. Such an algorithm is the LLL algorithm for linearly dependent vectors of Pohst (see e.g. [2], Chapter 2.6.4). Buchmann and Pohst ([1]) showed for (a variant of) this algorithm that the number of needed arithmetic operations is $O(d+m)^4 \log B)$. For the incremental algorithm the number of arithmetic operations is linear in m.

Algorithm 4.1 Lattice Basis

Input: Generating system $v_1, \ldots, v_m \in E$ of the lattice L.
Output: Basis $b_1, \ldots, b_n$ of L.
  $n \leftarrow 0$, $L \leftarrow \{0\}$.
  for $i = 1$ to m **do**
    if $v_i \notin L$ **then**
      Use a subroutine to get n and a basis $b_1, \ldots, b_n$ of $L + \mathbb{Z}v_i$.
      $L \leftarrow \mathbb{Z}b_1 + \cdots + \mathbb{Z}b_n$.
    end if
  end for

4.2. **Successive Minima.** For computing the successive minima of a lattice our algorithm is similar to Algorithm 4.1. However there are a few important differences: We need a complete generating system (see Section 1), the insertion order is no longer arbitrary, and in every update step it is enough to compute a basis of a subspace (instead of a lattice). Hence the number of update steps equals the rank of the lattice.

Algorithm 4.2 Successive Minima

Input: Complete generating system $S = \{v \in L \backslash \{0\} : \|v\| \leq B\}$ of L.
Output: Successive minima $\lambda_1(L), \ldots, \lambda_n(L)$ of L.
  Choose $v \in S$ with minimal norm, $S \leftarrow S \backslash \{v\}$.
  $n \leftarrow 1$, $U \leftarrow \mathbb{R}v$, $\lambda_n(L) \leftarrow \|v\|$.
  while $S \neq \emptyset$ **do**
    Choose $v \in S$ with minimal norm, $S \leftarrow S \backslash \{v\}$.
    if $v \notin L$ **then**
      $U \leftarrow U + \mathbb{R}v$.
      if $\dim U > n$ **then**
        $n \leftarrow n + 1$, $\lambda_n(L) \leftarrow \|v\|$.
      end if
    end if
  end while

4.3. **Orthogonal decomposition.** A lattice is called *decomposable* if it can be written as an orthogonal direct sum of two non trivial sublattices. Eichler ([4]) proved that every lattice can be decomposed into indecomposable sublattices which are pairwise orthogonal and that the decomposition is unique up to order of summands. In [7] Kneser gave a

constructive and much simpler proof. In this section we show how one can efficiently implement Kneser's argument.

We are given a basis $b_1, \ldots, b_n \in E$ of the lattice L, a constant B, and a complete generating system $S = \{v \in L \backslash \{0\} : \|v\| \leq B\}$. We want to find the number of indecomposable sublattices r, indices $i_1 = 1 \leq i_2 < \ldots < i_r \leq n < n+1 = i_{r+1}$ and a basis $b'_1, \ldots, b'_n$ of L such that for every $j \in \{1, \ldots, r\}$ the vectors $b'_{i_j}, \ldots, b'_{i_{j+1}-1}$ form a basis of an indecomposable sublattice.

Now we give Kneser's argument.

Definition 4.3. A vector $v \in L \backslash \{0\}$ is called *orthogonal decomposable* if there exist $x, y \in L \backslash \{0\}$ with $v = x + y$ and $(x, y) = 0$.

The orthogonal indecomposable vectors of S form the verticex set of an undirected graph $G = (V, E)$. In G two vertices $v, w \in V$ are adjacent whenever $(v, w) \neq 0$. We decompose V into vertex sets $V_1, \ldots, V_r$ of connected components of G. Then, the orthogonal decomposition of L is $L = L_1 \perp \ldots \perp L_r$ where L_i is the lattice generated by V_i.

Using standard algorithms from graph theory one can compute the connected components in time linear in $O(|V| + |E|)$. In the following we show that in this case it is possible to compute the connected components in time linear in $O(|V|)$.

O'Meara observed in [8] that for the procedure above it is not necessary to determine all orthogonal indecomposable lattice vectors in S. The length decomposable lattice vectors are enough:

Definition 4.4. A vector $v \in L \backslash \{0\}$ is called *length decomposable* if there exist $x, y \in L \backslash \{0\}$ with $v = x + y$ and $\|x\| \leq \|v\|$ and $\|y\| \leq \|v\|$.

On basis of this observation we propose the following algorithm. Its correctness follows from Proposition 4.6. In what follows we denote by π_i the orthogonal projection of E onto the subspace spanned by L_i.

Algorithm 4.5 Orthogonal Decomposition of a Lattice

Input: Complete generating system $S = \{v \in L \backslash \{0\} : \|v\| \leq B\}$ of L.
Output: Indecomposable sublattices L_i with $L = L_1 \perp \ldots \perp L_r$.

```
Choose v ∈ S with minimal norm, S ← S\{v}.
r ← 1, L_r ← Zv.
while S ≠ ∅ do
    Choose v ∈ S with minimal norm, S ← S\{v}.
    if v ∉ Σ_{i=1}^{r} L_i then
        J ← {j ∈ {1,...,r} : π_j(v) ≠ 0}.
        M ← Zv + Σ_{i∈J} L_i.
        {L_1,...,L_{r-|J|}} ← {L_i : i ∉ J}, L_{r-|J|+1} ← M, r ← r - |J| + 1.
    end if
end while
```

Proposition 4.6. At the end of each iteration the computed sublattices are indecomposable and pairwise orthogonal.

Proof. By induction the sublattices $L_1, \ldots, L_r$ are indecomposable and pairwise orthogonal. Let v be a shortest vector in S. If $v \notin \sum_{i=1}^{r} L_i$, then v is not length decomposable. In particular we have either $\pi_i(v) = 0$ or $\pi_i(v) \notin L_i$ where $i = 1, \ldots, r$. Define

$J = \{j \in \{1, \ldots, r\} : \pi_j(v) \neq 0\}$. One can choose vectors $v_j \in L_j$, where $j \in J$, which are not length decomposable and which are not orthogonal to v. In the graph G these vectors are all adjacent to v. Hence, $\mathbb{Z}v + \sum_{j \in J} L_j$ is indecomposable and we get $\sum_{i \in I \setminus J} L_i \perp (\mathbb{Z}v + \sum_{j \in J} L_j)$ because $\pi_i(v) = 0$ for $i \in I \setminus J$. □

5. Acknowledgements

We thank Martin Kneser for pointing out the reference to [8].

References

[1] J. Buchmann and M. Pohst, *Computing a lattice basis from a system of generating vectors*, Lecture Notes in Comput. Sci. 378, 54–63, Springer-Verlag, 1989.
[2] H. Cohen, *A course in computational algebraic number theory*, Springer-Verlag, 1993.
[3] J. H. Conway and N.J.A. Sloane, *A lattice without a basis of minimal vectors*. *Mathematika* **42** (1995), 175–177.
[4] M. Eichler, *Note zur Theorie der Kristallgitter*, *Math. Ann.* **125** (1952), 51–55.
[5] P.M. Gruber, C.G. Lekkerkerker, *Geometry of numbers*, North-Holland, 1987.
[6] B. Hemkemeier and F. Vallentin, *On the decompositon of lattices*, *Electronic Colloquium on Computation and Complexity* **TR98-52**, 1998.
[7] M. Kneser, *Zur Theorie der Kristallgitter*, *Math. Ann.* **127** (1954), 105–106.
[8] O.T. O'Meara, *On indecomposable quadratic forms*, *J. Reine Angew. Math.* **317** (1980), 120-156.

Universität Dortmund, Fakultät für Mathematik, 44221 Dortmund, Germany
E-mail address: bhemkemeier@gmail.com

Centrum voor Wiskunde en Informatica (CWI), Kruislaan 413, 1098 SJ Amsterdam, The Netherlands
E-mail address: f.vallentin@cwi.nl

ECCC ISSN 1433-8092
http://eccc.hpi-web.de/