FEASIBLE ALGORITHMS FOR LATTICE AND DIRECTED SUBSPACES

JENNIFER DEL VALLE, VLADIK KREINOVICH, AND PIOTR J. WOJCIECHOWSKI

Abstract. In some practical situations (e.g., in econometrics), it is important to check whether a given linear subspace of a space $\mathbb{R}^m$ with component-wise order is a lattice – and if it is not, whether it is at least a directed ordered space. Because of the practical importance, it is desirable to have *feasible* algorithms for solving these problems – which in Computer Science is usually interpreted as algorithms whose computation time does not exceed a polynomial of the length of the input. No such algorithms were previously known. In this paper, we present feasible algorithms for solving both problems.

1. Introduction

In financial applications, it is important to check whether a vector space generated by non-negative vectors is lattice-ordered. Specifically, it was proven, in [3], that the existence of appropriate minimum-cost insured portfolios is equivalent to the fact that the linear space generated by the corresponding financial instruments is lattice-ordered.

A real vector space V is called an *ordered vector space* if it is equipped with a compatible partial order $\leq$, i.e., if for any vectors u, v and w from V, if $u \leq v$, then $u + w \leq v + w$, and for any positive $\alpha \in \mathbb{R}$, $\alpha u \leq \alpha v$. This order is a *lattice* if for any two elements u and v, there exist the least upper bound $u \vee v$ and the greatest lower bound $u \wedge v$.

Every lattice order is also *directed* in the sense that for every two elements u and v, there is an upper bound w for which $u \leq w$ and $v \leq w$, and similarly there is a lower bound. The set $V^+ = \{u \in V : u \geq 0\}$ is called a *positive cone* of V. It satisfies the three axioms of a cone:

- $K + K \subseteq K$
- $\mathbb{R}^+ K \subseteq K$
- $K \cap -K = \{0\}$.

Vice versa, any subset of V satisfying the three above conditions is a positive cone of a partial order on V. For a space to be directly ordered, is equivalent to the condition that $V = K - K$, i.e., the positive cone K is *generating*.

Throughout the paper by $\mathbb{R}^m$ we will understand the coordinate-wise ordered vector lattice $\bigoplus_{i=1}^m \mathbb{R}$. By a *subspace* W of $\mathbb{R}^m$ we understand any subspace ordered by the order of $\mathbb{R}^m$.

A vector subspace W of a vector lattice V is called a *lattice-subspace* if W equipped with the ordering from V is a vector lattice on its own, i.e., if the least

2000 *Mathematics Subject Classification.* Primary 06F25.
Key words and phrases. Ordered vector space, direct order, lattice-subspace.

upper bound of any two elements from W exists in W (and automatically does the greatest lower bound of the elements). In [1], Abramovich et al. studied the lattice-subspaces of $\mathbb{R}^n$ and gave equivalent conditions for a subspace to be a lattice-subspace. In [5], the authors gave equivalent conditions for a subspace to be directly ordered. In this article, we improve the efficiency of both algorithms. Since it is necessary to algorithmically compare two numbers, we will restrict our considerations to algebraic numbers (numbers which are solutions to polynomial equations with integer coefficients), for which we can use the Tarski-Seidenberg algorithm (e.g. [2, 7] and [8] for rational numbers).

2. Lattice Order

In their paper [1], Abramovich, Aliprantis and Polyrakis gave necessary and sufficient conditions for a subspace of $\mathbb{R}^m$ to be lattice-ordered. We assume that the partial orders considered are coordinate-wise and that the subspace is $\langle X \rangle$, a subspace generated by a set X of n linearly independent positive vectors. We put the vectors from X in a $n \times m$ matrix as columns and consider the *associated* set $Y = \{y_1, \ldots, y_m\}$ of the rows of the matrix. Their main Theorem 2.6 asserts that $\langle X \rangle$ is lattice-ordered if and only if the set X admits a *fundamental set of indices* I, which means that the subset $Y_I \subseteq Y$ of vectors indexed by I is linearly independent, and every vector from $Y \setminus Y_I$ is a nonnegative linear combination of vectors from Y_I. The authors also give a computer algorithm that, based on the above result, determines if a given subspace is lattice ordered. The algorithm requires $\binom{m}{n}$ steps, which grows exponentially with m.

Below we propose an alternative algorithm, which is of polynomial time. For the reasons mentioned in the introduction we limit our input to algebraic numbers. We begin with functions $\mathbf{INDEX}(Y)$ and $\mathbf{PREFUND}(Y)$ that output a subset I of indices and a subset Y_I of Y indexed by I, respectively. There are known (c.f. [4]) polynomial-time algorithms that solve linear programming problems. Therefore we can have a polynomial-time (boolean output) routine $\mathbf{NONNEGCOMB}(y_i|Z)$ that checks if a vector $y_i \in Y$ is a nonnegative linear combination of vectors from $Z \subseteq Y$.

input $m; Y$
$\mathbf{INDEX}(Y) := \{1, \ldots, m\}$; $Z := Y$
for($i = 1$; $i \leq m$; $i++$)
  if $\mathbf{NONNEGCOMB}(y_i|Z)$
    { $Z := Z \setminus \{y_i\}$;
    $\mathbf{INDEX}(Y) := \mathbf{INDEX}(Y) \setminus \{i\}$ }
$\mathbf{PREFUND}(Y) := Z$

Also, there are known polynomial-time algorithms checking linear independence of a set of vectors. Let $\mathbf{LININDEP}(Z)$ be such an algorithm checking linear independence of a set Z of vectors. Now our main algorithm $\mathbf{LATTICE}(X)$ can be written. Let $\mathbf{GETY}(X)$ be an algorithm returning the associated set Y given the input X.

input $m; X$
$Y := \mathbf{GETY}(X)$
if $\mathbf{LININDEP}(\mathbf{PREFUND}(Y))$
  $\mathbf{LATTICE}(X) = \texttt{yes}$
else

LATTICE$(X) = \texttt{no}$

In what follows we will prove that this algorithm is both correct and of polynomial time.

Definition 2.1. A subset $I \subseteq \{1, \ldots, m\}$ of indices of vectors from Y is called *pre-fundamental* if

$$\forall_{k\in\{1,\ldots,m\}}\, y_k = \sum_{i\in I} \alpha_i y_i, \text{ for some } \alpha_i \geq 0$$

and

$$k \in I \Leftrightarrow \alpha_i = \delta_{ik} \text{ in any representation of the above type.}$$

Here $\delta_{ik} = \begin{cases} 1 & \text{if} \quad i = k \\ 0 & \text{if} \quad i \neq k \end{cases}$. In other words all vectors from Y are nonnegative combinations of those from Y_I and for every $i \in I$ the only such combination yielding y_i is $y_i = 1 \cdot y_i$.

Lemma 2.2. *Let I and J be two pre-fundamental sets of indices. Then for every $j \in J$, $y_j = \alpha_l y_l$ for some $l \in I$ and $\alpha_l > 0$.*

Proof. If $j \in I$ then $l = j$ and $\alpha_l = 1$. If $j \notin I$, then

$$y_j = \sum_{i\in I} \alpha_i y_i, \alpha_i \geq 0 \tag{*}$$

and for some $i \in I \setminus J$, $\alpha_i > 0$. If $y_i = \alpha y_j$, for some $\alpha > 0$ we are done by putting $l = i$ and $\alpha_l = \frac{1}{\alpha}$. We show that the remaining case is impossible. If y_i is not a scalar multiple of y_j, then y_i is a nonnegative linear combination of vectors from Y_J and it has a positive coefficient by some $y_{j'} \neq y_j$, $j' \in J$. Since all coefficients are nonnegative, $y_{j'}$ will maintain a positive coefficient in (*) written in terms of vectors from Y_J. But since J is pre-fundamental, the vector equal to the right hand side of (*) is not in Y_J, which is a contradiction.

□

It follows, by reversing the roles of I and J in Lemma 2.2, that the sets Y_I and Y_J may only differ by positive scalar multiples of their elements. In particular we have

Theorem 2.3. *If the vectors from X admit a fundamental set of indices, then any pre-fundamental set I of indices is also fundamental.*

Proof. Let J be a fundamental set of indices. So J is also pre-fundamental, therefore the vectors from Y_J differ from those from Y_I by at most positive scalar multiples. Since Y_J is linearly independent, then so is Y_I and thus I is fundamental. □

Theorem 2.4. *Let Y be the set of m vectors associated with a set X of positive linearly independent vectors from $\mathbb{R}^m$, then* **INDEX**(Y) *is pre-fundamental.*

Proof. Let $I =$ **INDEX**(Y) and $Z = Y_I$. It is clear that given that the set Y has at least one nonzero vector, $Z \neq \emptyset$. Also is clear that for every index $k \in I$ the conditions from Definition 2.1 are satisfied. Let now $k \notin I$. If $k = m$ then y_k is a nonnegative linear combination of vectors from Z, so the claim holds true. If $k = m - 1$, then y_k is a nonnegative linear combination of vectors from $Z \cup \{y_m\}$ which, in case that $m \in I$, is equal to Z and we are done. In case that $m \notin I$, y_{m-1}

is a nonnegative linear combinations of vectors from $Z \cup \{y_m\}$ which may include a nonzero contribution from y_m. But y_m is a nonnegative linear combination of vectors from Z, so we conclude that y_{m-1} is also such a combination and thus satisfies the conditions from the Definition 2.1. Similarly we proceed backwards towards $k = 1$ to argue that all the indices $1 \leq i \leq m$ satisfy the conditions and thus that I is pre-fundamental. □

Now we can prove the correctness of our algorithm.

Theorem 2.5. *If $X \subseteq \mathbb{R}^m$ is a set of n positive linearly independent vectors, then* $\mathbf{LATTICE}(X) = \texttt{yes}$ *if and only if the subspace $\langle X \rangle$ is lattice ordered.*

Proof. ($\Leftarrow$) Assume that the subspace $\langle X \rangle$ is lattice-ordered. Then the set of vectors X admits a fundamental set of indices by Theorem 2.6 from [1]. Call it J. On the other hand $I = \mathbf{INDEX}(Y)$ is pre-fundamental by Theorem 2.4. Therefore by Theorem 2.3 the set I is linearly independent, which results in $\mathbf{LATTICE}(X) =$ `yes`.

($\Rightarrow$) If $\mathbf{LATTICE}(X) =$ `yes`, then the set $\mathbf{PREFUND}(Y)$ is linearly independent. Since by Theorem 2.4 $I = \mathbf{INDEX}(Y)$ is pre-fundamental we conclude that it is fundamental. Therefore the set X of vectors admits a fundamental set of indices, so by Theorem 2.6 from [1] the subspace $\langle X \rangle$ is lattice-ordered.

□

Theorem 2.6. *The algorithm* **LATTICE** *is polynomial-time.*

Proof. The algorithm only once calls the routines **GETY**, **LININDEP** and **PREFUND**. While the first two routines are polynomial-time on m, the third one calls m times the polynomial-time routine **NONNEGCOMB**. Therefore the total time is polynomial. □

3. Directed Order

In [5], the authors give equivalent conditions for determining when an n-dimensional subspace W of $\mathbb{R}^m$ is directly ordered. The associated algorithm runs in exponential time, $O(m^n)$. Here, we present an algorithm for checking if the order is directed which runs in polynomial time $\mathcal{O}\left(m^{4.5}\right)$. This algorithm is also restricted to vectors consisting of algebraic numbers.

In order to determine if W^+ is generating, we check that each element of the basis of W can be expressed as a difference of two nonnegative vectors.

Let $X = \{x_1, \ldots, x_n\}$ be a basis for W. Define by $x(i)$ the i^{th} component of x, where $1 \leq i \leq m$. There exist polynomial-time algorithms (e.g. [4]) that check the existence of algebraic numbers $c_1, \ldots, c_n$ which satisfy the inequalities

$$\sum_{k=1}^{n} c_k \, x_k \, (i) \geq 0 \tag{3.1}$$

and

$$\sum_{k=1}^{n} c_k \, x_k \, (i) + x_j \, (i) \geq 0 \tag{3.2}$$

for each $j = 1, \ldots, n$ and all i. Call **SYSLINEQ** an algorithm which checks for solutions to 3.1 and 3.2. We then define **DIRORDER** as follows:

input X, m
if **SYSLINEQ** (X)
  DIRORDER = `yes`
else
  DIRORDER = `no`

We claim that **DIRORDER** correctly identifies directly ordered subspaces, and that it runs in polynomial time $\mathcal{O}\left(m^{4.5}\right)$. First, let us verify the correctness of the algorithm:

Theorem 3.1. *Let $W \subseteq \mathbb{R}^m$ with $X = \{x_1, x_2, \ldots, x_n\}$ a basis for W and $n < m$. Then W is directly ordered if and only if **DIRORDER**$(X) =$ `yes`.*

Proof. If $W = W^+ - W^+$, then for each element $x_j \in X$ there exist u_j, $v_j \geq 0$ such that $x_j = u_j - v_j$. In turn, each u_j and v_j admit a unique linear combination of the basis vectors: $u_j = \sum_{k=1}^n \alpha_k x_k$ and $v_j = \sum_{k=1}^n \beta_k x_k$; either may be taken to satisfy 3.1. Note that

$$x_j = u_j - v_j = \sum_{k=1}^{n} \left(\alpha_k - \beta_k\right) x_k \tag{3.3}$$

By linear independence of X, we must have $\alpha_k = \beta_k$ when $k \neq j$, and $\alpha_k - \beta_k = 1$ when $k = j$. The α_k and β_k are fully described by one another, and so we focus on the β_k as a solution also to 3.2. Indeed, since $u_j = v_j + x_j$, we have

$$\sum_{k=1}^{n} \beta_k x_k \ + x_j = u_j \geq 0 \tag{3.4}$$

which proves that the β_k satisfy both inequalities, and so **DIRORDER**$(X) =$ `yes`. For the converse, if there exist $c_1, \ldots, c_n$ that satisfy 3.1 and 3.2, then set $u_j = \sum_{k=1}^n c_k x_k \ + x_j$ with $v_j = \sum_{k=1}^n c_k x_k$, and we have u_j, $v_j \geq 0$ where $x_j = u_j - v_j$, so that W is directly ordered. □

Theorem 3.2. *The algorithm **DIRORDER** is $\mathcal{O}\left(m^{4.5}\right)$.*

Proof. The algorithm **DIRORDER** calls **SYSLINEQ** m times. Since **SYSLINEQ** is known to be $\mathcal{O}(m^{3.5})$, we may conclude that **DIRORDER** is then $\mathcal{O}(m^{4.5})$. □

References

1. Y. A. Abramovich, C. D. Aliprantis, and I. A. Polyrakis, "Lattice-Subspaces and Positive Projections," *Proceedings of the Royal Irish Academy*, 1994, Vol. 94A, No. 2, pp. 237–253.
2. S. Basu, R. Pollack, and M.-F. Roy, *Algorithms in Real Algebraic Geometry*, Springer-Verlag, Berlin, 2006.
3. C. D. Aliprantis, D. J. Brown, and J. Werner, "Minimum-cost portfolio insurance", *Journal of Economic Dynamics & Control*, 2000, Vol. 24, pp. 1703–1719.
4. T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms* MIT Press, Cambridge, MA, 2009.
5. J.J. Del Valle, P.J. Wojciechowski *A Note on Directly Ordered Subspaces of* $\mathbb{R}^n$, Tatra Mt Math. Publ. 52 (2012), 101-113.

6. V. N. Katsikis, "Computational and mathematical methods in portfolio insurance – A MATLAB-based approach," In: E. P. Leite (Ed.), *MATLAB-Modeling, Programming, and Simulations*, Sciyo Publ., 2010.
7. A. Tarski, *A Decision Method for Elementary Algebra and Geometry*, 2nd ed., Berkeley and Los Angeles, 1951, 63 pp.
8. K. Weihrauch, *Computable Analysis*, Springer-Verlag, Berlin, 2000.

Department of Mathematical Sciences, Department of Computer Science, The University of Texas at El Paso, El Paso, TX 79968

E-mail address: jjhamlyn@utep.edu, vladik@utep.edu, piotrw@utep.edu