LE MATEMATICHE
Vol. LXIII (2008) – Fasc. II, pp. 243–256

AN ALGORITHM TO COMPUTE THE STANLEY DEPTH OF MONOMIAL IDEALS

GIANCARLO RINALDO

In this article we describe an algorithm to compute the Stanley depth of I/J where I and J are monomial ideals. We describe also an implementation in CoCoA.

1. Introduction

Let K be a field, $S = K[x_1, \dots, x_n]$ be the polynomial ring in n variables with coefficient in K and M be a finitely generated $\mathbb{Z}^n$-graded S-module. Let $u \in M$ be a homogeneous element in M and Z a subset of the set of variables $\{x_1, \dots, x_n\}$. We denote by $uK[Z]$ the K-subspace of M generated by all elements uv where v is a monomial in $K[Z]$.

If $uK[Z]$ is a free $K[Z]$-module, the $\mathbb{Z}^n$-graded K-space $uK[Z] \subset M$ is called *a Stanley space of dimension* $|Z|$.

A *Stanley decomposition* of M is a presentation of the $\mathbb{Z}^n$-graded K-module M as a finite direct sum of Stanley spaces

$$\mathscr{D} : M = \bigoplus_{i=1}^{m} u_i K[Z_i]$$

Entrato in redazione: 2 gennaio 2009

AMS 2000 Subject Classification: 13P99,06A07
Keywords: Stanley depth, monomial ideal, Boolean lattice, partition
Supported by PRAGMATIC 08

in the category of $\mathbb{Z}^n$-graded K-spaces.

The number

$$\operatorname{sdepth} \mathscr{D} := \min\{|Z_i| : i = 1, \ldots, m\}$$

is called the *Stanley depth* of $\mathscr{D}$ and the number

$$\operatorname{sdepth} M := \max\{\operatorname{sdepth} \mathscr{D} : \mathscr{D} \text{ is a Stanley decomposition of } M\}$$

is called the *Stanley depth* of M.

The following widely open conjecture is due to Stanley [8]:

$$\operatorname{depth} M \leq \operatorname{sdepth} M \text{ for all } \mathbb{Z}^n\text{-graded } S\text{-modules } M.$$

Our goal is to give an algorithm (see Algorithm 1, Section 4) to compute the sdepth of I/J where I, J are monomial ideals of S based on [5, Theorem 2.1 and Corollary 2.5]. We give also some enhancements proved in Proposition 3.1 and in Theorem 3.5.

In the last section we describe an implementation in CoCoA (see [3]) and some examples of computation with remarks on the complexity of the algorithm.

2. Preliminaries

Let $S = K[x_1, \ldots, x_n]$, with K a field, and let $J \subset I$ be monomial ideals of S. In [5] the authors calculated the Stanley depth of I/J by considering partitionings of a poset induced by I/J. In this section we recall the construction given in [5] and we give an example.

A natural partial order on $\mathbb{N}^n$ is defined as follows:

$$a \leq b \text{ if and only if } a(i) \leq b(i) \text{ for } i = 1, \ldots, n,$$

with $a = (a(1), \ldots, a(n))$, $b = (b(1), \ldots, b(n))$.

Let ε_i be the ith canonical unit vector in $\mathbb{N}^n$, then $a = \sum_{i=1}^n a(i)\varepsilon_i$.

The set $\mathbb{N}^n$ with the partial order above defined is a distributive lattice with meet and join defined as follows:

$$(a \wedge b)(i) = \min\{a(i), b(i)\}, \ (a \vee b)(i) = \max\{a(i), b(i)\}, i = 1, \ldots, n.$$

For $a = (a(1), \ldots, a(n)) \in \mathbb{N}^n$ we set $x^a = x_1^{a(1)} \cdots x_n^{a(n)}$. Observe that $x^a \mid x^b$ if and only if $a \leq b$.

Let I and J be two monomial ideals of S. Suppose $I = (x^{a_1}, \ldots, x^{a_r})$ and $J = (x^{b_1}, \ldots, x^{b_s})$ and choose $g = (g(1), \ldots, g(n)) \in \mathbb{N}^n$ such that $a_i \leq g$ and $b_j \leq g$ for all i and j.

We define

$$P^g_{I/J} = \{c \in \mathbb{N}^n : c \leq g \text{ and } \exists i, a_i \leq c \text{ and } \forall j, c \ngeq b_j\}.$$

This finite poset is called the *characteristic poset* of I/J with respect to g.

There is a natural choice for $g = (g(1), \dots, g(n)) \in \mathbb{N}^n$, namely

$$g(t) = \max\{(a_i \vee b_j)(t), t = 1, \dots, n\},$$

for all i and j. We will call it *the canonical* g of I/J. In this case $P^g_{I/J}$ has the least number of elements and it is denoted by $P_{I/J}$.

Given a poset P and $a, b \in P$ we set

$$[a,b] := \{c \in P : a \leq c \leq b\}$$

and we call $[a,b]$ an *interval*. Note that $[a,b] \neq \emptyset$ if and only if $a \leq b$.

If P is a finite poset, a *partition* $\mathscr{P}$ of P is a disjoint union of intervals, namely

$$\mathscr{P} : P = \bigcup_{i=1}^{r} [a_i, b_i].$$

It is possible to associate to each partition of $P^g_{I/J}$ a Stanley decomposition of I/J. In order to describe this fact we need some notation and results from [5].

Definition 2.1. Let $c \in P^g_{I/J}$, we set

$$Z_c = \{x_j : c(j) = g(j)\},$$

and we define the function

$$\rho : P^g_{I/J} \to \mathbb{Z}_{\geq 0}, \qquad c \to \rho(c),$$

where $\rho(c) = |\{j : c(j) = g(j)\}| = |Z_c|$.

The following results holds (See [5], Theorem 2.1 and Corollary 2.5):

Theorem 2.2. *Let $\mathscr{P} : P^g_{I/J} = \bigcup_{i=1}^{r} [c_i, d_i]$ be a partition of $P^g_{I/J}$. Then*

$$\mathscr{D}(\mathscr{P}) : I/J = \bigoplus_{i=1}^{r} (\bigoplus_{c} x^c K[Z_{d_i}])$$

is a Stanley decomposition of I/J, where the inner direct sum is taken over all $c \in [c_i, d_i]$ for which $c(j) = c_i(j)$ for all j with $x_j \in Z_{d_i}$. Moreover

$$\operatorname{sdepth} \mathscr{D}(\mathscr{P}) = \min\{\rho(d_i) : i = 1, \dots, r\}$$

Theorem 2.3. *Let $J \subset I$ monomial ideals. Then*

$$\operatorname{sdepth} I/J = \max\{\operatorname{sdepth} \mathscr{D}(\mathscr{P}) : \mathscr{P} \text{ is a partition of } P^g_{I/J}\}.$$

In particular, there exists a partition $\mathscr{P} : P^g_{I/J} = \bigcup_{i=1}^r [c_i, d_i]$ of $P^g_{I/J}$ such that

$$\operatorname{sdepth} I/J = \min\{\rho(d_i) : i = 1, \dots, r\}.$$

We give an example to demonstrate how to compute the Stanley depth by using Theorem 2.2 and Theorem 2.3:

Example 2.4. *Let $S = K[x_1, x_2, x_3]$, $I = (x_1^2 x_2, x_1^3 x_3, x_2 x_3, x_3^3)$ and $J = (0)$. We may assume $g = (3,1,3)$. The following figure shows the poset $P_{I/J}$. Each point $p = (a_1, a_2, a_3)$ in this poset is represented as $a_1a_2a_3$: for example $g = (3,1,3)$ is represented by* 313.

011					
210	111	012	003		
310	301	211	112	103	013
311	302	212	203	113	
312	303	213			
313					

Since I is not a principal ideal it follows from[5] that $1 \leq \operatorname{sdepth} I \leq 2$. We first check whether there exists a partition $\mathscr{P}$ on the poset such that $\operatorname{sdepth} \mathscr{D}(\mathscr{P}) = 2$. It is useful to calculate for each point in the poset the ρ function (next to the colon):

011 : 1					
210 : 1	111 : 1	012 : 1	003 : 1		
310 : 2	301 : 1	211 : 1	112 : 1	103 : 1	013 : 2
311 : 2	302 : 1	212 : 1	203 : 1	113 : 2	
312 : 2	303 : 2	213 : 2			
313 : 3					

If we find a partition of the poset into intervals whose maximum elements have $\rho \geq 2$ we are done.

Such a partition exists. In fact, if we consider the intervals $I_1 = [011, 013]$, $I_2 = [210, 310]$, $I_3 = [111, 113]$, $I_4 = [003, 303]$, $I_5 = [301, 311]$, $I_6 = [211, 213]$, $I_7 = [302, 312]$ $I_8 = [313, 313]$, then

$$\mathscr{P} = \bigcup_{i=1}^{8} I_i$$

satisfies this condition.

3. Some improvements

In this section we give some improvements and remarks with respect to Theorem 2.2 and Theorem 2.3, necessary to clarify the algorithm in Section 4.

Let I and J be two monomial ideals of $S = K[x_1,\ldots,x_n]$,

$$I = (x^{a_1},\ldots,x^{a_r}) \text{ and } J = (x^{b_1},\ldots,x^{b_{r'}}).$$

The input of our algorithm is given by the vectors $a_1,\ldots,a_r,b_1,\ldots,b_{r'} \in \mathbb{N}^n$. For our purpose we choose $g = (g_1,\ldots,g_n)$ as the join of all the a_i and b_j. In this case the poset $P^g_{I/J}$ has the least number of points. From now on we denote the poset simply by $P_{I/J}$.

By Theorem 2.3 we have to consider all the partitions of the poset $P_{I/J}$. Since $\operatorname{sdepth} I/J = \max\{\operatorname{sdepth}\mathscr{D}(\mathscr{P})\}$, it is natural to check if for a "reasonable" high value s there exists a partition $\mathscr{P}$ such that $\operatorname{sdepth}\mathscr{P} = s$.

In [7, Section 3] A. Soleyman Jahan observed that if M is a $\mathbb{Z}^n$-graded S-module with $dim_K M_a \leq 1$ for all homogeneous components of M, then

$$\operatorname{sdepth} M \leq \min\{\dim S/P : P \in \operatorname{Ass}(M)\}.$$

This observation give us an upper bound for the value of s.

Proposition 3.1. *Let $J \subset I$ be ideals of S. Then*

$$\operatorname{sdepth} I/J \leq \min\{\dim S/P : P \in \operatorname{Ass}(S/J)\}.$$

Proof. Since $\operatorname{Ass}(I/J) \subset \operatorname{Ass}(S/I)$, the assertion follows from the above quoted result of Soleyman Jahan. □

We give the following

Definition 3.2. *Let P be a characteristic subposet of $\mathbb{N}^n$ and $0 \leq s \leq n$. We define two subsets of P,*

$$P_{<s} = \{p \in P : \rho(p) < s\},$$

and

$$P_{\geq s} = \{p \in P : \rho(p) \geq s\}$$

where $\rho()$ is the function defined in 2.1.

Once computed the poset P and the value $s = \min\{\dim S/P : P \in \operatorname{Ass}(S/J)\}$, we may start to test if the poset has a partition $\mathscr{P}$ whose sdepth is equal to s. To reach this goal we consider the poset P as a disjoint union of the two sets defined in 3.2

$$P = P_{<s} \cup P_{\geq s}.$$

It is easy to observe that if a partition $\mathscr{P}$ of the poset P with sdepth $\mathscr{P} = s$ exists, then

$$\mathscr{P} = A \cup B,$$

where

$$A = \bigcup_{p_i \in P_{<s}} [p_i, q_i], \qquad B = \bigcup_{p'_i \in P_{\geq s} \setminus A} [p'_i, q'_i]$$

and $\mathscr{P}$ can be refined by a new partition $\mathscr{P}'$

$$\mathscr{P}' = A \cup B'$$

with

$$B' = \bigcup_{p'_i \in P_{\geq s} \setminus A} [p'_i, p'_i].$$

Therefore in our algorithm we consider only the partitions A whose elements $[p,q]$ are $p \in P_{<s}$ and $q \in P_{\geq s}$.

To find such a partition A, we have to find for each element $p \in P_{<s}$ all possible "candidates" $q \in P_{\geq s}$. We give the following

Definition 3.3. *We define the function*

$$\widetilde{\rho} : P_{<s} \to 2^{P_{\geq s}}, \qquad \widetilde{\rho}(p) = S_p^{(s)}$$

where $S_p^{(s)} = \{q \in P_{\geq s} : q > p\}$. We call $S_p^{(s)}$ the ρ-shadow of p (with respect to s).

Lemma 3.4. *Let $[p,q]$ be an interval in $P_{I/J}$ such that $\rho(p) < s \leq \rho(q)$. Then for*

$$q_0 \in \min\{x \in [p,q] : \rho(x) \geq s\} \tag{3.1}$$

there exists a partition of the interval $[p,q]$

$$[p,q] = [p,q_0] \cup [p_1,q_1] \cup \cdots \cup [p_r,q_r] \text{ with } q_r = q$$

such that $\rho(q_i) \geq s$, $i = 0,\ldots,r$.

Proof. Applying the translation $x \to x - p$ we may assume without loss of generality $p = (0,\ldots,0) \in \mathbb{N}^n$.

Let $Z_{q_0} = \{i \in [n] : q_0(i) = g(i)\}$. Then

$$q_0 = \sum_{i=1}^{n} q_0(i)\varepsilon_i = \sum_{i \in Z_{q_0}} g(i)\varepsilon_i + \sum_{i \in [n] \setminus Z_{q_0}} q_0(i)\varepsilon_i.$$

Since q_0 is minimal in $S_p^{(s)}$, it follows that

$$q_0 = \sum_{i \in Z_{q_0}} g(i)\varepsilon_i.$$

After a suitable reordering of the indexes we may assume that

$$q_0 = (g(1), \ldots, g(s), 0, \ldots, 0), \quad q = (g(1), \ldots, g(s), q(s+1), \ldots, q(n)).$$

We claim that

$$\bigcup_{k=0}^{n-s} \bigcup_{s<i_1<i_2<\ldots<i_k\leq n} [\varepsilon_{i_1} + \ldots + \varepsilon_{i_k}, q_0 + q(i_1)\varepsilon_{i_1} + \ldots + q(i_k)\varepsilon_{i_k}] \tag{3.2}$$

is a partition of $[0, q]$.

Indeed for each point $u \in [0, q]$, we show that u belongs to exactly one of the intervals

$$[\varepsilon_{i_1} + \ldots + \varepsilon_{i_k}, q_0 + q(i_1)\varepsilon_{i_1} + \ldots + q(i_k)\varepsilon_{i_k}].$$

To show this we define the set

$$\operatorname{supp}_{>s}(u) = \{i \in [n] : u(i) \neq 0, i > s\}.$$

Let $k = |\operatorname{supp}_{>s}(u)|$. If $k = 0$, then

$$u = (u(1), \ldots, u(s), 0, \ldots, 0),$$

and hence $u \in [0, q_0]$.

If $k > 0$, let $\operatorname{supp}_{>s}(u) = \{i_1, \ldots, i_k\}$ with $s < i_1 < \ldots < i_k \leq n$. Therefore

$$u = \sum_{i=1}^{s} u(i)\varepsilon_i + \sum_{j=1}^{k} u(i_j)\varepsilon_{i_j},$$

where $\sum_{i=1}^{s} u(i)\varepsilon_i \in [0, q_0]$ and

$$\sum_{j=1}^{k} u(i_j)\varepsilon_{i_j} \in [\varepsilon_{i_1} + \ldots + \varepsilon_{i_k}, q(i_1)\varepsilon_{i_1} + \ldots + q(i_k)\varepsilon_{i_k}],$$

and since

$$\begin{aligned}[0, q_0] + [\varepsilon_{i_1} + \ldots + \varepsilon_{i_k}, q(i_1)\varepsilon_{i_1} + \ldots + q(i_k)\varepsilon_{i_k}] &= \\ &= [\varepsilon_{i_1} + \ldots + \varepsilon_{i_k}, q_0 + \varepsilon_{i_1} q(i_1) + \ldots + \varepsilon_{i_k} q(i_k)]\end{aligned}$$

the assertion follows.

We also observe that these intervals are disjoint. In fact if a point belongs to different intervals it means that the same point has distinct supports $\operatorname{supp}_{>s}$, contradiction. □

Theorem 3.5. *Let* $\operatorname{sdepth} I/J \geq s$. *Then there exists a Stanley decomposition* $\mathscr{D}(\mathscr{P})$ *with*

$$\mathscr{P} : P_{I/J} = \bigcup_{i=1}^{r} [p_i, q_i]$$

such that $q_i \in \min\{x \in [p_i, q] : \rho(x) \geq s\}$, *for* $i \in \{1, \ldots, r\}$.

Proof. Since $\operatorname{sdepth} I/J \geq s$, we have a partition on the poset $P_{I/J}$

$$\mathscr{P} : P_{I/J} = \bigcup_{i=1}^{r} [p_i, q_i]$$

with $q_i \geq s$. If there exists q_j that is not minimal in $S_{p_j}^{(s)}$ we apply Lemma 3.4 to the interval $[p_j, q_j]$. $\square$

By Theorem 3.5 and Lemma 3.4 in the calculation of the ρ-shadow of $p \in P_{<s}$ we consider only the minimal elements in the subposet $P_{\geq s}$ that are greater than p. This increases the speed of the algorithm especially when we have to check all the possible partitions. This is the case when we ask if the $\operatorname{sdepth} I/J$ is equal to t, but in reality $\operatorname{sdepth} I/J$ is $< t$.

We have a method for computing the minimal elements of the ρ-shadow that is given by the following

Corollary 3.6. *Let $p \in P_{<s}$ and $I = \{i \in [n] : g(i) = p(i)\}$. Then*

$$\min S_p^{(s)} = \{q \in P : q = p + \sum_{i \in \bar{I}} (g(i) - p(i))\varepsilon_i : I \subset \bar{I} \subset [n], |\bar{I}| = s\}.$$

Proof. Since we want to calculate $\min S_p^{(s)}$ we may restrict our attention to the subposet of P

$$P_{\geq p} = \{q \in P : q \geq p\}.$$

Applying the translation $x \to x - p$ we consider without loss of generality $p = (0, \ldots, 0) \in \mathbb{N}^n$ with

$$I = \{i \in [n] : g(i) = p(i)\} = \{i \in [n] : g(i) = 0\}.$$

For each maximal element $q \in P_{\geq p}$ we obtain an interval $[p, q]$. Let $q_0 \in \min S_p^{(s)}$ with $q_0 \in [p, q]$ and let $\bar{I} = \{i \in [n] : q_0(i) = g(i)\}$. By an observation in the proof of Lemma 3.4 we have that

$$q_0 = \sum_{i \in \bar{I}} g(i)\varepsilon_i, \text{ with } |\bar{I}| = s.$$

By the same argument we observe that if $q_0 \in [p, q]$ and

$$q_0 = \sum_{i \in \bar{I}} g(i)\varepsilon_i$$

with $|\bar{I}| = s$ then $q_0 \in \min S_p^{(s)}$. Applying this argument to each $q \in \max P_{\geq p}$ we have that

$$\min S_p^{(s)} = \{q \in P : q = \sum_{i \in \bar{I}} (g(i))\varepsilon_i : I \subset \bar{I} \subset [n], |\bar{I}| = s\}.$$

The assertion follows if we apply the translation $x \to x + p$ to the poset $P_{\geq p}$. $\square$

4. The algorithm

In this section we describe the main steps of the algorithm 1.

Algorithm 1: Stanley depth algorithm

Data: $a_1,\dots,a_r,b_1,\dots,b_{r'} \in \mathbb{N}^n$, $s \in \mathbb{N}$
Result: *true* if $\operatorname{sdepth} I/J = s$

1 P:=`Poset`$(a_1,\dots,a_r,b_1,\dots,b_{r'})$
2 **foreach** $p_i \in P_<$ **do**
    $S_i^{(s)} := \widetilde{\rho}(p_i)$
    $j_i := 0$
3 $i := 1$
4 **while** $1 \le i \le |S^{(s)}|$ **do**
5   **if** p_i *is covered* **then**
        isCovered:=true
6   **while** $\neg$ isCovered and $j_i < |S_i^{(s)}|$ **do**
        $j_i := j_i + 1$
        isCovered:=`tryInterval`$(p_i, S_i^{(s)}(j_i))$
7   **if** isCovered **then** $i := i+1$
    else
8         $j_i := 0$
        $i := i-1$
        goBack:=true
9   **while** $i > 0$ and goBack **do**
10   **if** $j_i = |S_i^{(s)}|$ **then**
                `uncoverInterval`$(p_i, S_i^{(s)}(j_i))$
                $j_i := 0$
                $i := i-1$
            else if $j_i \neq 0$ **then**
                `uncoverInterval`$(p_i, S_i^{(s)}(j_i))$
                goBack:=false
            else $i := i-1$
11 **if** $i = 0$ **then return** false **else return** true

- line 1. We define the poset. For simplicity we don't give any details on the inner structure of it.

- line 2. In this loop we calculate the ρ-shadow for each point $p_i \in P_<$ (see

Definition 3.3 and Theorem 3.5). In particular we have the family of sets

$$S^{(s)} = \{S_i^{(s)} = \widetilde{\rho}(p_i)\}$$

and for each set $S_i^{(s)}$ we define an index j_i that address an element of $S_i^{(s)}$. When j_i is "active" we have $1 \leq j_i \leq |S_i^{(s)}|$. In the beginning j_i is not active that is $j_i = 0$.

- line 3,4. We define the index i addressing the set $S_i^{(s)}$. In the main loop 4 we look for a partition and $1 \leq i \leq |S^{(s)}|$. This loop ends if either $i = 0$, and this means we do not find any partition, or $i = |S^{(s)}| + 1$ and we find a partition (see line 11).

- line 5. It could happen that the point p_i is already covered by another interval, this is the meaning of this condition. In the positive case we put the variable `isCovered` to true.

- line 6. In this loop we try to cover the uncovered point p_i that is we increment j_i and we observe if the interval

$$[p_i, S_i^{(s)}(j_i)],$$

  where $S_i^{(s)}(j_i)$ is the j_ith element of $S_i^{(s)}$, does not intersect the intervals already computed. The loop ends if either we find a covering interval starting from p_i or we fail. Even in this case this status is represented by the variable `isCovered`.

- line 7. If we find a covering for p_i we have to consider the next element that is p_{i+1}.

- line 8. If we fail to cover the point p_i it means that the previous computed intervals never will partition the poset P. Therefore we have to "backtrack" (see [10],Chapter 3.4). That is we have to go back to the last choice made and choose the next one (if there exists!). Before backtracking we have to reset j_i and put $i := i - 1$.

- line 9. We don't know "a priori" how much we have to decrement i. This loop calculates the right value of i restoring the old status at the ith loop (line 4). The loop go on if `goBack` is true and $i > 0$. We observe in fact that $i = 0$ means that there are no more backtracking possible that is no partition are allowed.

- line 10. Inside the loop we consider the three possible cases induced by the value of j_i.

  If $j_i = |S_i^{(s)}|$ then we do not have other elements in $S_i^{(s)}$, hence we have to recover the previous status (`uncoverInterval`), reset j_i and continue going back ($i := i-1$).

  If $0 < j_i < |S_i^{(s)}|$ then we have some other elements in $S_i^{(s)}$ to be checked therefore we recover and we stop going back (`goBack`:=false).

  If $j_i = 0$ then the point p_i is already covered by an other interval, therefore we have to continue going back.

- line 11. If the main loop exits with the value $i = 0$ means that we failed searching the partition. Otherwise we succeeded ($i = |S^{(s)}| + 1$).

5. An implementation in CoCoA

In this section we describe an implementation given by the author under GPL license and written in CoCoA (see [6]). We calculate in the meantime the Stanley depth of a Stanley-Reisner ring whose simplicial complex is Cohen-Macaulay but not shellable. These kind of examples are interesting because for shellable simplicial complex the Stanley's conjecture is proved (see [4], Theorem 6.5).

Let Δ be a triangulation of the 3 dimensional real projective space

```
 << "SdepthLib.coc";
N:=8;
Use S::=Q[x[1..N]];
I:=Ideal(1);

J:=Ideal(x[1]*x[2]*x[3]*x[4], x[2]*x[3]*x[4]*x[5],
 x[1]*x[3]*x[4]*x[6], x[1]*x[5]*x[6], x[2]*x[5]*x[6],
 x[1]*x[2]*x[4]*x[7], x[1]*x[5]*x[7], x[3]*x[5]*x[7],
 x[2]*x[6]*x[7], x[3]*x[6]*x[7], x[1]*x[2]*x[3]*x[8],
 x[1]*x[5]*x[8], x[4]*x[5]*x[8], x[2]*x[6]*x[8],
 x[4]*x[6]*x[8], x[3]*x[7]*x[8], x[4]*x[7]*x[8]);

Sd:=4;
IsSdepth(Sd,I,J);
```

We first load the libray `SdepthLib.coc` then we define the ring S. We already know that this Stanley-Reisner ring has dimension 4 and is Cohen-Macaulay. Therefore if $\operatorname{sdepth} K[\Delta] < 4$ we find a counterexample to Stanley's conjecture.

We define $I = (1)$ since we want to compute $K[\Delta] = S/J$. We define the ideal J, and we want to check if $\operatorname{sdepth} K[\Delta] = 4$. Therefore we call the function `IsSdepth(Sd,I,J)` with `Sd=4`. The output of the program is (7 seconds of computation): The poset $P_{K[\Delta]}$ that corresponds to the face poset of Δ and ...

```
0
1 2 3 4 5 6 7 8
12 13 14 15 16 17 18 23 24 25 26 27 28 34 35 36 37 38 45 46
47 48 56 57 58 67 68 78
123 124 125 126 127 128 134 135 136 137 138 145 146 147 148
167 168 178 234 235 236 237 238 245 246 247 248 257 258 278
345 346 347 348 356 358 368 456 457 467 567 568 578 678
1235 1236 1237 1245 1246 1248 1278 1345 1347 1348 1368 1467
1678 2346 2347 2348 2358 2457 2578 3456 3568 4567 5678
```

... the intervals covering $P_{K[\Delta]}$.

```
[0,1235]     [4,1245]     [6,1236]     [7,1237]
[8,1248]     [34,1345]    [38,1348]    [46,1246]
[47,1347]    [56,3456]    [57,2457]    [58,2578]
[67,1467]    [68,1368]    [78,1278]    [234,2348]
[238,2358]   [247,2347]   [346,2346]   [358,3568]
[567,4567]   [568,5678]   [678,1678]
```

With the suggestion of Prof. Naoki Terai we tested some and more difficult to compute simplicial complexes Cohen-Macaulay but not shellable (see also the site `http://www.math.tu-berlin.de/~lutz`, Frank H. Lutz). In some cases the computation took some days. But we did not find any counterexample.

Remark 5.1. *We observe that this algorithm is exponential in time and space.*

To show this fact we consider $\mathfrak{m}$ *the maximal ideal of the ring* $K[x_1,\ldots,x_n]$. *By the results proven in [1] we already know that* $\operatorname{sdepth}\mathfrak{m} = \lceil \frac{n}{2} \rceil$.

It is easy to see that the poset $P_{\mathfrak{m}}$ *has exactly* $2^n - 1$ *elements. By backtracking algorithm it is difficult to know "a priori" how many steps are needed to compute the partitions. Therefore we compute the speed by examples when* $n \in \{6,\ldots,12\}$. *The following table summarizes the results on a AMD Sempron 2.800 Mhz.*

n	*time (sec.)*
6	*0.38*
7	*1.60*
8	*5.21*
9	*25.69*
10	*89.39*
11	*464.18*
12	*1769.85*

The growing of the time of computation is obviously exponential.

Acknowledgements

We wish to thank all the staff of PRAGMATIC 2008 (Prof. Jürgen Herzog, Prof. Volkmar Welker, Dr. Marius Vladoiu, Dr. Xinxian Zheng) and also Dr. Mircea Cimpoeas for the useful discussion, suggestion and comments. We also wish to thank Prof. Naoki Terai for providing us interesting examples of nonshellable Cohen-Macaulay simplicial complexes.

REFERENCES

[1] C. Birò - D. M. Howard - M. T. Keller - W. T. Trotter - St. J. Young, *Interval partition and Stanley depth*, Preprint 2008.

[2] W. Bruns - J. Herzog, *Cohen-Macaulay rings*, Cambridge Univ. Press, Cambridge, 1997.

[3] CoCoATeam, CoCoA*: a system for doing Computations in Commutative Algebra*, Available at `http://cocoa.dima.unige.it`

[4] J. Herzog - D. Popescu. *Finite filtrations of modules and shellable multicomplexes*, Manuscripta Math. **126** (2006), 385–410.

[5] J. Herzog - M. Vladoiu - X.Zheng, *How to compute the Stanley depth of a monomial ideal*, arXiv:math.AC/0712.2308, 2007.

[6] G. Rinaldo, *Sdepth: Software to compute the Stanley depth for monomial ideals*, 2008, Available at `http://ww2.unime.it/rinaldo/sdepth`.

[7] A. S. Jahan, *Stanley decomposition of squarefree modules and Alexander duality*, arXiv:math.AC/0709.4145, 2007.

[8] R.Stanley. *Linear Diophantine equations and local cohomology*, Invent.Math. **68** (1982), 175–193.

[9] R.H. Villarreal. *Monomial Algebras*. Pure and applied mathematics. Marcel Dekker, New York/Basel, 2001.

[10] N. Wirth, *Algorithms + data structures = programs.* Prentice-Hall, Englewood Cliffs, N. J., 1976.

GIANCARLO RINALDO
Dipartimento di Matematica
Università di Messina
Contrada Papardo, salita Sperone, 31
98166 Messina, Italy
e-mail: `rinaldo@dipmat.unime.it`