Appendix for "Matrix representations for toric parametrizations": Implementation in Macaulay2

Nicolás Botbol

Departamento de Matemática
FCEN, Universidad de Buenos Aires, Argentina
& Institut de Mathématiques de Jussieu
Université de P. et M. Curie, Paris VI, France
E-mail address: nbotbol@dm.uba.ar

Alicia Dickenstein

Departamento de Matemática
FCEN, Universidad de Buenos Aires
Ciudad Universitaria, Pab.I
1428 Buenos Aires, Argentina
E-mail address: alidick@dm.uba.ar

Marc Dohm

Laboratoire J. A. Dieudonné
Université de Nice - Sophia Antipolis
Parc Valrose, 06108 Nice Cedex 2, France
E-mail address: dohm@unice.fr

Abstract

This is an appendix for the article [BDD]. Here we show how to compute a matrix representation with the method developed in this paper, using the computer algebra system Macaulay2 [M2].

In the paper we show that a surface in $\mathbb{P}^3$ parametrized over a 2-dimensional toric variety $\mathscr{T}$ can be represented by a matrix of linear syzygies if the base points are finite in number and form locally a complete intersection. This constitutes a direct generalization of the corresponding result over $\mathbb{P}^2$ established in [BJ03] and [BC05]. Exploiting the sparse structure of the parametrization, we obtain significantly smaller matrices than in the homogeneous case and the method becomes applicable to parametrizations for which it previously failed. We also treat the important case $\mathscr{T} = \mathbb{P}^1 \times \mathbb{P}^1$ in detail and give numerous examples.

Key words: matrix representation, rational surface, syzygy, approximation complex, implicitization, toric variety

Appendix: Implementation in Macaulay2

In this appendix we show how to compute a matrix representation with the method developed in this paper, using the computer algebra system Macaulay2 [M2]. As it is probably the most interesting case from a practical point of view, we restrict our computations to bi-homogeneous parametrizations of a certain bi-degree (e_1, e_2). However, the method is easily adaptable to the toric case, or more precisely to a given fixed Newton polytope N(f) and, where it is appropriate, we will give hints on what to change in the code. Moreover, we are not claiming that our implementation is optimized for efficiency; anyone trying to implement the method to solve computationally involved examples is well-advised to give more ample consideration to this issue. For example, in the toric case there are better suited software systems to compute the generators of the toric ideal J, see [4ti2].

Let us start by defining the parametrization f given by $(f_1, \dots, f_4)$.

```
S=QQ[s,u,t,v];
e1=4;
e2=2;
f1=s^4*t^2+2*s*u^3*v^2
f2=s^2*u^2*t*v-3*u^4*t*v
f3=s*u^3*t*v+5*s^4*t^2
f4=2*s*u^3*v^2+s^2*u^2*t*v
F=matrix{{f1,f2,f3,f4}}
```

The reader can experiment with the implementation simply by changing the definition of the polynomials and their degrees, the rest of the code being identical. We first set up the list st of monomials $s^i t^j$ of bidegree (e_1', e_2'). In the toric case, this list should only contain the monomials corresponding to points in the Newton polytope N$'(f)$.

```
st={};
l=-1;
d=gcd(e1,e2)
ee1=numerator(e1/d);
ee2=numerator(e2/d);

for i from 0 to ee1 do (
   for j from 0 to ee2 do (
   st=append(st,s^i*u^(ee1-i)*t^j*v^(ee2-j));
   l=l+1
   )
)
```

We compute the ideal J and the quotient ring A. This is done by a Gröbner basis computation which works well for examples of small degree, but which should be replaced by the matrix formula in [BDD, Formula (8)] for more complicated examples. In the toric case, there exist specialized software systems such as [4ti2] to compute the ideal J.

```
SX=QQ[s,u,t,v,w,x_0..x_l,MonomialOrder=>Eliminate 5]

X={};
st=matrix {st};
F=sub(F,SX)
st=sub(st,SX)

te=1;
for i from 0 to l do ( te=te*x_i )

J=ideal(1-w*te)
for i from 0 to l do (
    J=J+ideal (x_i - st_(0,i))
    )
J= selectInSubring(1,gens gb J)

R=QQ[x_0..x_l]
J=sub(J,R)
```

```
A=R/ideal(J)
```

Next, we set up the list ST of monomials $s^i t^j$ of bidegree (e_1, e_2) and the list X of the corresponding elements of the quotient ring A. In the toric case, this list should only contain the monomials corresponding to points in the Newton polytope $\mathrm{N}(f)$.

```
use SX
ST={};
for i from 0 to e1 do (
    for j from 0 to e2 do (
    ST=append(ST,s^i*u^(e1-i)*t^j*v^(e2-j));
    )
)

X={};
for z from 0 to length(ST)-1 do (
    f=ST_z;
    xx=1;
    is=degree substitute(f,{u=>1,v=>1,t=>1});
    is=is_0;
    it=degree substitute(f,{u=>1,v=>1,s=>1});
    it=it_0;
    iu=degree substitute(f,{t=>1,v=>1,s=>1});
    iu=iu_0;
    iv=degree substitute(f,{u=>1,t=>1,s=>1});
    iv=iv_0;
    ded=0;
  while ded < k do (
     for mm from 0 to 1 do (
        js=degree substitute(st_(0,mm),{u=>1,v=>1,t=>1});
        js=js_0;
        jt=degree substitute(st_(0,mm),{u=>1,v=>1,s=>1});
        jt=jt_0;
        ju=degree substitute(st_(0,mm),{t=>1,v=>1,s=>1});
        ju=ju_0;
        jv=degree substitute(st_(0,mm),{u=>1,t=>1,s=>1});
        jv=jv_0;
if is>=js and it>=jt and iu>=ju and iv>=jv then (
    xx=xx*x_mm;
    ded=ded+1;
    is=is-js;
    it=it-jt;
    iv=iv-jv;
    iu=iu-ju; )));
   X=append(X,xx); )
```

We can now define the new parametrization g by the polynomials $g_1, \ldots, g_4$.

```
X=matrix {X};
X=sub(X,SX)
(M,C)=coefficients(F,Variables=>
          {s_SX,u_SX,t_SX,v_SX},Monomials=>ST)
G=X*C
G=matrix{{G_(0,0),G_(0,1),G_(0,2),G_(0,3)}}
G=sub(G,A)
```

In the following, we construct the matrix representation M. For simplicity, we compute the whole module

$\mathscr{Z}_1$, which is not necessary as we only need the graded part $(\mathscr{Z}_1)_{\nu_0}$. In complicated examples, one should compute only this graded part by directly solving the linear system given by [BDD, Formula (1)] in degree ν_0. Remark that the best bound nu $= \nu_0$ depends on the parametrization.

```
use A
Z1=kernel koszul(1,G);
nu=2*d-1
S=A[T1,T2,T3,T4]
G=sub(G,S);
Z1nu=super basis(nu+d,Z1);
Tnu=matrix{{T1,T2,T3,T4}}*substitute(Z1nu,S);

lll=matrix {{x_0..x_l}}
lll=sub(lll,S)
ll={}
for i from 0 to l do { ll=append(ll,lll_(0,i)) }
(m,M)=coefficients(Tnu,Variables=>
          ll,Monomials=>substitute(basis(nu,A),S));
M;
```

The matrix M is the desired matrix representation of the surface $\mathscr{S}$.

Acknowledgements

We thank Laurent Busé and Marc Chardin for useful discussions.

References

[4ti2] 4ti2—A software package for algebraic, geometric and combinatorial problems on linear spaces, 4ti2 team, available at `http://www.4ti2.de`.

[BDD] Botbol, N., Dickenstein, A. and Dohm, M. Matrix representations for toric parametrizations. arXiv:0807.4802.

[BC05] Busé, L., Chardin, M., 2005. Implicitizing rational hypersurfaces using approximation complexes. J. Symbolic Comput. 40 (4-5), 1150–1168.

[BJ03] L. Busé and J.-P. Jouanolou. On the closed image of a rational map and the implicitization problem. *J. Algebra*, 265(1):312–357, 2003.

[M2] D. R. Grayson and M. E. Stillman. Macaulay 2, a software system for research in algebraic geometry. Available at `http://www.math.uiuc.edu/Macaulay2/`.

[Ma94] Manocha, D., Solving systems of polynomial equations, Computer Graphics and Applications 14, IEEE, 1994, pp. 46-55.